Université de Lille

Thèse de Doctorat

pour obtenir le grade de :

Docteur de l'Université de Lille

École doctorale : MADIS

Unité de recherche : INRIA - Institut national de recherche en informatique et en automatique Lille Nord Europe

dans la spécialité « Informatique et applications »

par

Gaurav Maheshwari

Measuring and Mitigating Allocation Unfairness Across the Machine Learning Pipeline

Mesurer et atténuer l'injustice d'allocation dans le processus d'apprentissage automatique

Thèse soutenue le 27 mars 2024 devant le jury composé de :

Rapporteurs :

Mme. Anne Lauscher	Maîtresse de conférences	University of Hamburg
M. Ivan Habernal	Professeur	Ruhr University Bochum

Examinateur et Président du jury :

M. Christophe Gravier	Professeur	Université Jean Monnet

Directeur de thèse :

M. Pascal Denis	Chargé de recherche HDR	INRIA

Co-encadrants de thèse :

M. Aurélien Bellet	Directeur de Recherche	INRIA
Mme. Mikaela Keller	Maîtresse de conférences	Université de Lille

Inria

Université de Lille

CRIStAL
Centre de Recherche en Informatique,
Signal et Automatique de Lille

Abstract

With the advent of machine learning, the government institutions and other bureaucracy are undergoing a paradigm shift, as algorithms increasingly assist in and even replace some of their functions. Consequently, just as early 20th-century philosophers scrutinized these institutional changes, it is crucial to analyze these algorithms through the lens of their societal impact.

In line with this general objective, this thesis aims to examine and propose ways to mitigate the harms associated with employing machine learning (ML). Specifically, we study the impact of ML algorithm in the settings where groups of population are unfairly assigned or withheld opportunities and resources. In response, we propose a series of algorithms designed to measure and counteract unfairness throughout the ML pipeline. We begin by proposing FairGrad, a gradient based algorithm which dynamically adjusts the influence of examples throughout the training process to ensure fairness. We then examine FairGrad, and various other fairness enforcing mechanism from the lens of intersectionality where multiple sensitive demographic attributes are considered together. Our experiments reveal that several approaches exhibit "leveling down" behavior, implying that they optimize for current fairness measures by harming the involved groups. We introduce a new fairness measure called α-Intersectional Fairness which helps uncover this phenomena.

Building upon these findings, our next step focuses on addressing the leveling down issue. To mitigate its effects, we introduce a data generation mechanism that exploits the hierarchial structure inherent to the intersectional setting, and augments data for groups by combining and transforming data from more general groups. Through our experiments we find that this approach not only produces realistic new examples but also enhances performance in worst-case scenarios. Finally, we explore the intersection of privacy, another societal concern, with fairness. We present FEDERATE, a novel method that combines adversarial learning with differential privacy to derive private representations that lead to fairer outcomes. Interestingly, our results suggest that in our experimental context privacy and fairness can coexist and frequently complement each other.

Résumé

Avec l'arrivée de l'apprentissage automatique, les institutions gouvernementales et autres bureaucraties connaissent un changement de paradigme, car les algorithmes les assistent de plus en plus, voire remplacent certaines de leurs fonctions. Par conséquent, tout comme les philosophes du début du XXe siècle ont examiné ces changements institutionnels, il est essentiel d'analyser ces algorithmes sous l'angle de leur impact sociétal.

Conformément à cet objectif général, cette thèse vise à examiner et à proposer des moyens d'atténuer les préjudices associés à l'utilisation de l'apprentissage machine. Plus précisément, nous étudions l'impact des algorithmes d'apprentissage automatique dans les contextes où des groupes de population se voient attribuer ou refuser des opportunités et des ressources de manière injuste. En réponse, nous proposons une série d'algorithmes conçus pour mesurer et contrecarrer l'injustice tout au long du processus d'apprentissage automatique. Nous commençons par proposer FairGrad, un algorithme fondé sur le gradient qui ajuste dynamiquement l'influence des exemples pendant le processus d'entraînement, afin de garantir l'équité. Ensuite, nous examinons FairGrad et divers autres mécanismes d'application d'équité sous l'angle de l'intersectionnalité, où de multiples attributs démographiques sensibles sont pris en compte simultanément. Nos expériences révèlent que plusieurs approches présentent un comportement de nivellement par le bas : elles optimisent les mesures d'équité actuelles en portant atteinte aux groupes concernés. Nous présentons une nouvelle mesure d'équité, α-Intersectional Fairness (α-Équité intersectionnelle), qui aide à mettre au jour ce phénomène.

Sur la base de ces résultats, notre étape suivante se concentre sur la résolution du problème de nivellement par le bas. Pour en atténuer les effets, nous introduisons un mécanisme de génération de données qui exploite la structure hiérarchique inhérente au cadre intersectionnel et augmente les données des groupes en combinant et en transformant les données de groupes plus généraux. À travers nos expériences, nous montrons que cette approche permet non seulement de produire de nouveaux exemples réalistes, mais aussi d'améliorer les performances dans les scénarios les plus défavorables. Enfin, nous explorons l'intersection entre protection de la vie privée, autre préoccupation sociétale, et équité. Nous présentons FEDERATE, une nouvelle méthode qui combine l'apprentissage antagoniste et la confidentialité différentielle pour dériver des représentations privées qui conduisent à des résultats plus équitables. Il est intéressant de noter que nos résultats suggèrent que, dans notre contexte expérimental, vie privée et équité peuvent coexister et se complètent fréquemment.

Acknowledgements

> Guru Govind dou khade, kaake laagoon paye;
> Balihari guru aapki, Govind diyo milaye.
>
> Kabir, a 15th century Indian poet

Roughly translating to: *God and my teachers are both standing. Whom should I bow to first? I will first bow to my teachers, because, they are the one who showed me the path to God.*

These words have never been more truer, than during my Ph.D. journey. Pascal, Mikaela, and Aurélien were steadfast guides, ensuring I navigated the path without stumbling yet allowing me the space to make mistakes and learn from them. They not only inspired me but also fostered my growth as an independent researcher, offering support and encouragement when needed. Their discussions and unwavering support have not only shaped me as a researcher but also as a better individual. Words fail to express the depth of my gratitude and respect for them.

I extend my sincere gratitude to all the members of my thesis advisory and examination committee. Their invaluable time and insightful discussions during the defense are deeply appreciated. I hope our paths will cross again in the future, fostering further enriching discussions. My heartfelt thanks also extend to my colleagues at Magnet. The camaraderie, banter, and shared coffee breaks infused my mundane days with vibrancy. A special nod goes to foosball. I am grateful to each one of you for making me feel less like a stranger.

Special thanks are owed to my friends in Lille and across the globe. You all served as my pit crew, without whom I would undoubtedly have run out of fuel. Your presence and support were instrumental, and I am deeply grateful for it. I extend profound thanks to my parents for their unwavering trust and support. Their safety net and constant support provided me with the confidence to take risks and pursue my dreams. Finally, a heartfelt thanks to Ritika, for being the cheerleader.

Contents

Chapter 1

Introduction

Decision-making is central to human society. The choices we make and the actions we take profoundly shape our lives. Nonetheless, outcomes and levels of success often hinge not just on personal decisions but also on choices made by others. For instance, decisions related to university admissions or loan approvals can significantly impact individuals, and these determinations are often made by external bodies. Given the impact of these decisions on individuals, it is imperative that they are made in a transparent and reliable manner, as well as for the right reasons.

Entrusting individuals with these crucial decisions introduces risks of *subjectivity, arbitrariness, and inconsistency* (Barocas, Hardt, and Narayanan, 2019). For instance, discrepancies in criminal sentencing can arise due to the personal beliefs of the judges (Albanese, 1984). A study from the 1919 (Everson, 1919) highlighted this issue by revealing that penalties for the same crime and similar income in New York City's Magistrate's court ranged from 17% to 80% of the offender's income. Another comprehensive analysis of 7,442 cases discovered that the imprisonment rates set by different judges varied from 34% to 58%. Similarly, healthcare studies (Chapman, Kaatz, and Carnes, 2013; Hood, 2001; Devine and Plant, 2012) have found biases in medical treatments that are influenced by the doctor's personal beliefs, which can correlate with the patient's race, ethnicity, or other factors. In education, Sprietsma (2013) found that essays associated with Turkish-sounding names were graded lower by German school teachers compared to those with German-sounding names. Likewise, Harber et al. (2012) observed that European American teachers provided more constructive feedback to essays they presumed were written by European American students than to those believed to be authored by African American students. Apart from these inconsistencies, individuals find articulating the reasoning behind their choices also challenging (Strandburg, 2019).

To counteract such biases, decision-making authority has increasingly shifted from individuals to collectives represented by bureaucratic systems and institutions, which operate based on well-defined processes and regulations (Weber, 2016). For instance, decisions concerning university admissions or loan approvals are typically shaped by the policies, rules, and protocols of established bureaucracies. They are taken collectively rather than by single individuals. Similarly, many countries have instituted medical boards and associations to standardize treatment procedures. To further enhance consistency, many jurisdictions have introduced sentencing guidelines for crimes, thus limiting individual discretion.

The development of these regulations often involves multiple experts and stakeholders and is typically subject to public scrutiny. While far from perfect, these institutions reduces inconsistency and subjectivity prevalent in individual decision-making. Furthermore, in addition to clear rules and regulations, bureaucracies usually offer mechanisms to challenge and correct decisions.

1.1 Machine assisted Bureaucracies

In recent years, to enhance the efficiency of bureaucratic systems and further reduce biases, there has been a notable increase in the automation of various institutional aspects. Barocas, Hardt, and Narayanan (2019) outline several methods by which software systems have started to assist bureaucracies. We can broadly categorize these methods into two main groups:

- **Software assisted rule based automation:** These involve the use of software engineering to automate decisions by translating explicit rules and regulations which were set down by hand into the software. For instance, automatically determine eligibility and enrolment in the government program such as Medicaid[1], or to screen resumes based on keywords (Sinha, Amir Khusru Akhtar, and Kumar, 2021).
- **Machine Learning assisted automation:** These involve use of machine learning to either replicate the informal judgment of bureaucrats or uncovering patterns in data to assist policy making. Example include grading essays automatically (Ramesh and Sanampudi, 2022) or guiding policing strategies, [2] such as determining which areas to prioritize during patrolling.

Rule-based automation, often referred to as robotic process automation (RPA)[3], streamlines bureaucracies by eliminating repetitive tasks traditionally performed by humans. This results in enhanced efficiency and speed. According to a recent survey[4], 65% of federal agencies in the United States have adopted some form of RPA. Despite its benefits, RPA introduces several concerns. It can make the process more brittle and error-prone, as software developers could incorrectly implement or interpret rules. Additionally, it can further exacerbate the dehumanizing effect of bureaucracies (Nissenbaum, 1996). However, it is important to note that the concerns raised due to this automation are more inherent to software development.

Automating the informal judgment of humans via machine learning can help address the concerns of arbitrariness and inconsistency in human decision making. For instance, automatic grading systems have been shown to be more consistent than teachers (Wang, Chang, and Li, 2008). Similar examples can be found in health care, such as more consistent and earlier detection of diabetic retinopathy (Alyoubi, Shalash, and Abulkhair, 2020) or better patient triage in emergency department (Raita et al., 2019). However, learning from human decisions risks replicating and exacerbating the biases of those who made these decisions before (Caliskan, Bryson, and Narayanan, 2017; Chang, Prabhakaran, and Ordonez, 2019). Additionally, machine learning systems might achieve similar performance as humans but could have

[1] https://www.medicaid.gov/medicaid/eligibility/index.html

[2] https://www.predpol.com/

[3] https://www2.deloitte.com/content/dam/Deloitte/nl/Documents/public-sector/deloitte-nl-Robotic-process-automation-in-the-public-sector.pdf

[4] https://cdn.fedscoop.com/robotic-process-automation-in-government-report.pdf

completely different error patterns (Ribeiro, Singh, and Guestrin, 2016), failing in unexpected ways. In other words, they could be right for the wrong reason.

In addition to replicating human judgment, machine learning is increasingly utilized as a tool for policy creation. Here, experts define the target objective, and then, instead of relying on experts' intuition and normative reasoning, algorithms are employed to discover the features and patterns from the data to achieve the objective. This form of assistance has found several use cases, ranging from optimizing costs in industries (Evans and Gao, 2016; Li et al., 2019) to the discovery of new drugs (Patel and Shah, 2022). However, this methodology is not without risks, such as a potential mismatch between the objective and the target (Dressel and Farid, 2018), or the lack of diverse examples in the dataset (Suresh and Guttag, 2021). O'Neil (2016) highlights issues in various domains, from justice to finance, where machine learning has adversely affected specific subgroups.

Furthermore, machine learning's influence is not confined to merely supplanting traditional systems; it is progressively permeating myriad facets of human society. Consequently, understanding the potential ramifications of deploying such systems becomes imperative. In the seminal keynote talk at NeurIPS 2017[5], Kate Crawford identified two predominant types of harm[6] associated with machine learning:

- **Allocation harm** occurs when certain subgroups are unfairly assigned or withheld opportunities and resources due to algorithmic intervention. An infamous example of this kind of harm is COMPAS[7] software, which systematically accused the African-American defendants of reoffending more than the European-American defendant.

- **Representational harm** emerges when algorithmic systems perpetuate and amplify stereotypes of certain groups. These include examples such as stereotypical representation of race in large language models (Weidinger et al., 2021; Bender et al., 2021), or skewed portrayal of woman in image searches (Otterbacher, Bates, and Clough, 2017; Kay, Matuszek, and Munson, 2015).

While allocation harms are typically direct, immediate, and measurable, representational harms tend to be more long-term, diffused, and challenging to measure. Addressing them often requires moving beyond mathematical models and thinking from broader social context involving multiple stakeholders[8] which is beyond the scope of this thesis.

In this thesis, we propose various methods to measure and mitigate allocation harm when machine learning systems are used to either replicate informal judgment or uncover patterns in the data. Our proposed solutions attack these problem at various stages of machine learning pipelines including data generation, training mechanism, and evaluation metrics.

[5] The Trouble with Bias: https://www.youtube.com/watch?v=fMym_BKWQzk

[6] Overtime researchers have further subdivided and added different kinds of harm. For a more complete list, please refer to Shelby et al. (2022).

[7] https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

[8] https://machinesgonewrong.com/

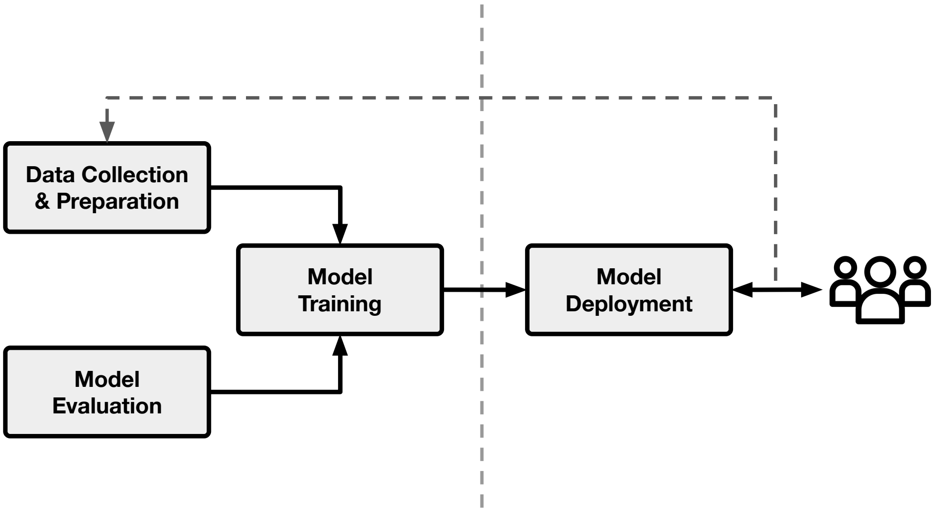


FIGURE 1.1: A typical machine learning pipeline consists of a data collection phase, followed by a training module that interacts with an evaluation module. Once the model is trained, it is deployed in real-world scenarios where users interact with the model. These interactions are subsequently gathered as training data, serving to continually improve the model.

1.2 Source of Unfairness in ML pipeline

To discuss the different ways unfairness can infiltrate and propagate into machine learning systems, we begin by introducing a generic machine learning pipeline. Presented in Figure 1.1, this simplified pipeline consists of the following phases: data collection and preparation, followed by model training and evaluation, and finally deployment. In the following, we elaborate on the issues that can arise in each of these stages and highlight how our contributions aim to address them.

1.2.1 Data collection and Preparation

Data serve as the foundation of any machine learning system. It operationalizes and delineates the practitioner's goal, subsequently guiding later stages to achieve the task at hand. This step typically involves defining a target population and recording features and labels considered relevant by the practitioner. After data collection, pre-processing steps such as encoding and standardizing transform the data making it easy to process.

Mehrabi et al. (2022) and Suresh and Guttag (2021) outline multiple types of bias that can emerge during the data collection phase. These include **representational bias** (Suresh and Guttag, 2019), where parts of the population are underrepresented, resulting in models to poorly generalize over these subgroups. For instance, the widely-utilized Twitter Hate Speech dataset (Huang et al., 2020) for training hate speech detection models contains over 50,000 tweets. However, target demographics are skewed, with young European American males having over 10 times more samples than elderly African American females. Consequently, models trained on this data show much lower error rates for young European American males than elderly African American females. Another prevalent form of bias is **historical bias**, where the collected data reflects the biases and prejudices of the real world (Ahmed, Granberg, and Khanna, 2021; Quillian et al., 2017). **Measurement bias** is also common, where the label imprecisely captures the true goal or exhibits variable accuracy due to measurement error across different demographic groups of the population (Hoffmann and Tarzian, 2001; Phelan et al., 2015).

In Chapter 6 of this thesis, we address the problem of representational bias by proposing a novel data generation mechanism. Specifically, we propose a Maximum Mean Discrepancy-based (Gretton et al., 2012) approach that generates data for a group such as elderly African American females by combining and transforming data from related parent groups like elderly African American, elderly females, and African American females. This strategy capitalizes on the fact that these parent groups by design have more examples than the targeted group itself. Through experiments over various datasets, we find that a classifier trained with our proposed data augmentation mechanism improves its performance over these underrepresented groups.

1.2.2 Model Evaluation

The evaluation module works in conjunction with the training mechanism to assess model performance across different settings. It typically involves defining a target metric that best captures the model's effectiveness for the given task and data. For instance, in predictive justice scenarios with the stance that *it is better to let a guilty person go free than to condemn the innocent*[9], metric such as False Positive rate can potentially captures the task objective. It measures predictions where the model incorrectly rejects the null hypothesis – that a defendant is innocent, thus minimizing it aligns with the objective of avoiding harm to the innocent.

Conventional evaluation metrics such as Accuracy and True Positive Rate provide aggregated insights over the entire population. However, these broad measures can obscure performance declines and disparities amongst different demographic groups. For example, while a hate speech detection model might demonstrate high overall accuracy, it could still exhibit performance gaps across different demographics like African Americans versus European American. Additionally, focusing solely on one metric type, such as False Positive Rate, could hide disparities in other error types like True Positive Rate. In response to such concerns, the fairness research community has introduced various fairness definitions (Hardt, Price, and Srebro, 2016; Calders, Kamiran, and Pechenizkiy, 2009; Zafar et al., 2017a). These definitions typically strive to mathematically capture the unfairness caused due to the performance gaps across different demographics, while also capturing stances like the predictive justice example described above.

Much of the discourse in this field centers on singular demographic identities, for example, gender or race. However, capturing unfairness at the level of a single identity does not ensure fairness when multiple sensitive axes are considered together, such as those defined by both gender and race. These observations also resonate with the analytical framework of *intersectionality* (Crenshaw, 1989), which argues that systems of inequality based on various demographic attributes (like gender and race) may "intersect" to create unique effects.

[9] https://www.law.cornell.edu/supremecourt/text/156/432

In Chapter 5, we benchmark various fairness inducing approaches in intersectional fairness settings. We find that several methods improve over existing fairness metrics by "leveling down", that is, by harming the groups involved. In response, we propose a new measure called the α-Intersectional Fairness, which is robust to leveling down. More specifically, α-Intersectional Fairness combines the absolute and the relative performance across different demographic groups and can be seen as a generalization of existing fairness measures. We also highlight several desirable properties of the proposed measure and analyze its relation to other fairness measures.

1.2.3 Model Training

The model training phase typically involves defining a loss function, and a parameterized model architecture. The model's parameters are then optimized by minimizing the loss function over the training data using specialized optimizers. Generally, multiple loss functions, optimizers, and model architectures are tested to achieve optimal performance, as measured by the evaluation module. For classification tasks using deep neural networks, the model architecture commonly consists of two main components: (i) an encoder that transforms the raw input into a representative embedding, which is then passed to (ii) a classifier that classifies the embedding. A key advantage of this architecture is that encoders can be pre-trained on large volumes of possibly unlabelled data which reduces reliance on task-specific in-domain data and annotation.

Modern machine learning techniques are surprisingly good at modeling the objective based on the training data. In other words, the models can faithfully reflect the characteristics of the underlying data. However, this implies that without any specific intervention, any bias in the training data might not only get reflected in the output, but can get amplified (Hall et al., 2022). As highlighted by Barocas, Hardt, and Narayanan, 2019, a part of the training data represents the signal we wish to mine, but might also consists of stereotypical pattern that we might want to avoid. As a practitioner, it is difficult to control what the model focuses on.

For example, De-Arteaga et al. (2019) investigated predicting occupations from biographical descriptions to improve job recommendations and hiring decisions. Their analysis revealed that the classifier tends to be more correct when the occupation aligns with the stereotypical gender. For instance, the biographies authored by male doctors were more likely to be classified as doctors than those written by female doctors. Analogous gender-based stereotypical tendencies were observed for other professions, including professor, model, and accountant. Addressing these inherent biases within the training data is a nontrivial, owing to the myriad of latent and confounding variables influencing these associations. For instance, even after removing explicit gender indicators such as pronouns and names and balancing the training data, they found that the classifier still exhibited gender bias.

Thus, eliminating bias from the training data is not only challenging, but also often inadequate. Furthermore, it is unclear how to manipulate training data to enforce fairness definitions described above. Moreover, certain design decisions during the training phase, like the choice of optimization function or regularization techniques, might inadvertently introduce unfairness, even when the input data is unbiased (Baeza-Yates, 2018; Danks and London, 2017).

To address these issues, various in-processing approaches have been developed that augments existing training mechanisms to promote fairness. These methods range from introducing additional constraints during the training phase to improving data sampling. In this context, we also propose two distinct mechanisms (Chapter 7 and Chapter 4 respectively), each targeting different components of the model architecture:

- FEDERATE, which approaches the fairness problem from the lens of removing stereotypical associations at the level of encoder. It combines the ideas from differential privacy and adversarial training to create representations devoid of demographic information resulting in fairer models.
- FairGrad, where instead of removing sensitive information, we directly optimize the fairness definition at hand by adding it as an additional loss function. This results in a simple-to-use fairness-enforcing mechanism that requires minimal changes to the existing machine-learning pipeline while supporting various fairness definitions.

1.2.4 Model Deployment

Once trained, models are generally deployed in real-world settings for people to use. Deployment involves steps like packaging the model into a production environment, monitoring performance over time, and logging user interactions that may be then used to further refine training.

One potential fairness issue arises when user feedback is misinterpreted. For instance, in a recommendation system, a user clicking the first link could reflect relevance or just placement (Lerman and Hogg, 2014). Similarly, in predictive policing, targeting areas of predicted high risk can increase police presence and arrests there. This amplifies a feedback loop, as more arrests further raise the assessed risk (Lum and Isaac, 2016; Ensign et al., 2018). Additionally, population shifts over time may alter real-world error rates, even if the model was fair on training data. Addressing these challenges requires examining the broader techno-social systems encompassing machine learning, an important direction for future work, but one that is beyond this thesis's scope.

1.3 Thesis Outline

The remainder of this thesis is organized into three main parts. First, we introduce relevant background and related work. We then present our various contributions over the next four chapters. Finally, in the concluding chapter, we summarize the work and discuss future research directions.

Chapter 2 introduces various concepts related to supervised machine learning, specifically focusing on tools and techniques relevant to the thesis. We then provide background on fair machine learning in Chapter 3, covering the historical framework, metrics, and methods proposed in this field.

In Chapter 4, we address the problem of fairness in classification. More specifically, we propose FairGrad, a method to enforce fairness based on a re-weighting scheme that iteratively learns group-specific weights based on whether they are advantaged or not during training. Our experiments reveal that FairGrad is competitive with standard baselines over various datasets, including ones used in natural language processing and computer vision.

In Chapter 5, we analyze various fairness-inducing techniques, including FairGrad, from the lens of intersectionality. We find that many of these methods optimize for existing intersectional fairness measures by harming the subgroups, also called "leveling down" (Mittelstadt, Wachter, and Russell, 2023). To counter these problems, we propose α-Intersectional Fairness, which combines the performance of a classifier over the worst-off subgroup and the relative performance between subgroups. Through various experiments, we show that our proposed metric is more robust and generalizes over existing fairness measures. In Chapter 6, we propose a novel MMD-based data generation mechanism to counter the above leveling-down phenomena. We validate our approach over various datasets and find it consistently improves performance of a classifier over both best and worst-off groups.

Chapter 7 shifts the focus to privacy, another critical ethical concern in modern machine learning systems. Our empirical analysis investigates the relationship between privacy and fairness, revealing that they can mutually enhance each other in certain scenarios. Specifically, we introduce an approach that integrates differential privacy with adversarial learning to learn privatized text representations that also leads to fairer models. Our extensive experiments demonstrate that our proposed mechanism can achieves simultaneous fairness and privacy with minimal impact on accuracy. Finally, Chapter 8 summarizes our contributions and outlines several promising avenues for future research.

1.4 List of Publications

- Gaurav Maheshwari, Aurélien Bellet, Pascal Denis, and Mikaela Keller. "Fair NLP Models with Differentially Private Text Encoders." In Findings of the Association for Computational Linguistics: Empirical Methods in Natural Language Processing 2022.
- Gaurav Maheshwari, and Michael Perrot. "FairGrad: Fairness Aware Gradient Descent". Transactions on Machine Learning Research (2023).
- Gaurav Maheshwari, Pascal Denis, Mikaela Keller, and Aurélien Bellet. "Fair NLP Models with Differentially Private Text Encoders." In Empirical Methods in Natural Language Processing 2023.

Chapter 2

Neural Networks

This thesis builds upon several key concepts and techniques developed in machine learning, which we introduce in this chapter. We begin by an overview of learning framework, focusing on empirical risk minimization and its component in supervised setting. We then discuss model architectures commonly used in this thesis.

2.1 Learning Framework

In their seminal work, Michalski, Carbonell, and Mitchell (2013) define a machine learning algorithm as:

> A computer program that learns from examples E with respect to a specific class of tasks T and performance measure L, improving its performance on tasks in T, as measured by L, through examples E.

In other words, the central objective is to devise algorithms that effectively *generalize* over task T. Here generalization refers to the algorithm's ability to perform well on new, previously unseen examples. This concept of generalization is captured by the notion of risk. However, before delving into risk, we first describe the setup for supervised learning.

In a typical supervised learning setup, we assume an input space $\mathcal{X}$ and a label space $\mathcal{Y}$. We further assume that there exists a unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$. The objective is then to find parameters $\theta \in \Theta$ for an hypothesis h, such that $h_\theta : \mathcal{X} \to \mathcal{Y}$ the output of h_θ should correctly represent the relationship between $\mathcal{X}$ and $\mathcal{Y}$ for the points drawn from $\mathcal{D}$. However, $\mathcal{D}$ is generally unknown, and instead we have access to finite dataset $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$ consisting of n examples drawn i.i.d. from $\mathcal{D}$. The primary objective of a machine learning algorithm is to learn h_θ using $\mathcal{T}$, so that it generalizes well over new examples drawn from $\mathcal{D}$.

A crucial component in the aforementioned setup is the formulation of a *loss* function, which quantifies the model's performance for a given problem. The design of the loss function can vary considerably based on the problem at hand. For instance, in classification task where $\mathcal{Y}$ is finite and discrete, with the aim to predict the precise class, an appropriate loss function $l(\cdot, \cdot)$ can be formulated as:

$$l(h_\theta, z) = \begin{cases} 0 & \text{if } h_\theta(x) = y \\ 1 & \text{otherwise} \end{cases} \tag{2.1}$$

where z is an example of the form $(x, y) \in \mathcal{D}$ and h_θ is the hypothesis function under consideration. In the above defined loss function, the value is zero if, and only if, the prediction matches the true label exactly, disregarding any closeness in prediction. Conversely, for regression problems where y is continuous, the objective is to get predictions as close to the true value as feasible. In other words, given two predictions, the one closer to the true label is a better prediction than the one further away. However, in classification problems, both predictions are equally incorrect. Thus a more appropriate loss function for regression would emphasize this notion of closeness. Below is an example of such a loss function:

$$l(h_\theta, z) = |h_\theta(x) - y| \tag{2.2}$$

A loss function $l : \Theta \times \mathcal{Z} \to \mathbb{R}_+$, given a hypothesis h_θ parametrized with $\theta \in \Theta$ and example $z \in \mathcal{D}$, returns a positive real value in $\mathbb{R}_+$. In Section 2.2, we provide examples of commonly used loss functions. Equipped with loss, we define risk as.

Definition 1. True Risk: Given a loss function $l : \Theta \times \mathcal{Z} \to \mathbb{R}_+$, and a distribution $\mathcal{D}$, the true risk of an hypothesis h_θ is:

$$R(h_\theta) = \underset{z \sim \mathcal{D}}{E}[l(h_\theta, z)] \tag{2.3}$$

Recall that our ultimate goal is to find optimal parameters of the hypothesis that best describes the distribution $\mathcal{D}$. This goal can be casted as an optimization problem where the best parameters h_{θ^*} minimizes the above risk:

$$h_{\theta^*} = \arg\min_{\theta \in \Theta} R(h_\theta) \tag{2.4}$$

Although appealing, the risk $R(h_\theta)$ cannot be generally calculated as $\mathcal{D}$ is usually unknown. Instead, it is estimated on the dataset $\mathcal{T}$ and is termed the empirical risk.

Definition 2. Empirical Risk: Given a loss function $l : \Theta \times \mathcal{Z} \to \mathbb{R}_+$ and a distribution $\mathcal{D}$, the empirical risk of an hypothesis h_θ is:

$$\hat{R}(h_\theta) = \frac{1}{n} \sum_{z \in T} l(h_\theta, z) \tag{2.5}$$

Consequently, **empirical risk minimization** can be defined as:

$$h_{\theta^*} = \arg\min_{\theta \in \Theta} \hat{R}(h_\theta) \tag{2.6}$$

The idea behind empirical risk minimization is that we hope that minimizing it leads to minimizing the true risk. In other words:

$$\arg\min_{\theta \in \Theta} R(h_\theta) \approx \arg\min_{\theta \in \Theta} \hat{R}(h_\theta) \tag{2.7}$$

Several factors influence the approximation of the true risk by the empirical risk. Key among them are:

- The amount of training data available. More data typically narrows the gap between the two terms.
- The loss function l. A well-designed loss function with various properties such as strong convexity, Lipschitz continuity, and smoothness enables better generalization.
- The complexity of the set of hypotheses $\{h_\theta; \theta \in \Theta\}$, and the training mechanism to pick the appropriate parameters θ .

An important tradeoff for machine learning practitioners is balancing model complexity. Complex hypotheses, i.e. hypothesis with large number of parameters, can overfit training data, leading to lower empirical risk but higher true risk. In contrast, simpler models might not capture the underlying patterns of the task sufficiently. This tradeoff between different complexity model is known as bias-variance tradeoff. We refer the interested readers to (Mohri, Rostamizadeh, and Talwalkar, 2018; Shalev-Shwartz and Ben-David, 2014; Neal et al., 2018; Yang et al., 2020) for in-depth discussion of this tradeoff. In the remainder of this chapter, we will discuss various loss functions, common hypothesis classes, and training mechanisms. We will also discuss mechanisms to control model complexity and a common technique used to reduce the volume of training data required to effectively train the model.

2.2 Loss Functions

At its core, the objective of the loss function is to evaluate the quality of a hypothesis for a given example and assign it a numerical score.

Definition 3. Loss function Given an hypothesis h_θ parametrized by $\theta \in \Theta$, and data space $\mathcal{Z}$, a loss function $l : \Theta \times \mathcal{Z} \to \mathbb{R}_+$ is any function such that.

- $\forall \theta \in \Theta$ and $z \in \mathcal{Z}$, $l(h_\theta, z) \geq 0$
- $h_{\theta_1}, h_{\theta_2} \in \Theta$ and $z \in \mathcal{Z}$, $l(h_{\theta_1}, z) \leq l(h_{\theta_2}, z)$ indicates that hypothesis h_{θ_1} performs better than h_{θ_2} on example z.

We now discuss several standard loss functions for classification tasks.

Zero-One loss: As described in introduction, it is a simple loss function which counts the misclassification by an hypothesis. It returns 1 for a misclassification, and returns 0 otherwise:

$$\forall \theta \in \Theta,\, z \in \mathcal{Z},\, l(h_\theta, z) = \begin{cases} 0 & \text{if } h_\theta(x) = y \\ 1 & \text{otherwise} \end{cases} \tag{2.8}$$

While appealing the loss is typically not used in practice as it is neither convex nor continuous. Lack of these properties makes the optimization problem relying on these loss functions difficult to solve. Instead, various surrogates have been proposed that approximate the zero-one loss by either relaxing or upper-bounding this loss.

Hinge Loss: It approximates the zero-one loss by linearly penalizing every prediction proportional to disagreement. The Hinge loss (Gentile and Warmuth, 1998) is defined as:

$$\forall \theta \in \Theta,\, z \in \mathcal{Z},\, l(h_\theta, z) = \max(0, 1 - h_\theta(x) \cdot y) \tag{2.9}$$

Although not strictly convex, this loss is continuous and almost differential everywhere (except at $h_\theta(x) \cdot y = 1$), leading to several applications. Notably, it is frequently used to optimize the Support Vector Machines (Boser, Guyon, and Vapnik, 1992; Mathur and Foody, 2008). The primary limitation of the hinge loss is its sensitivity to outliers. Additionally, its non-differentiability at $h_\theta(x) \cdot y = 1$ can sometimes make optimization unstable. To address this, variants like smooth hinge loss (Rennie, 2005) and quadratically smooth hinge loss (Zhang, 2004) have been introduced. Several other loss functions, such as Huber loss (Huber, 1965) and square loss (Tibshirani, 1996), also extend the idea of the zero-one loss.

The losses discussed so far are based on the concept of margin. They typically compare the final prediction with the target label. In contrast, probability-based loss functions take the prediction's probability into account. One of the most commonly used one is:

CrossEntropyLoss: The cross entropy loss is defined as:

$$\forall \theta \in \Theta \text{ and } z \in \mathcal{Z},\, l(h_\theta, z) = \sum_{y' \in \mathcal{Y}} p(y') \cdot log(p(h_\theta(x) = y')) \tag{2.10}$$

where $p(h_\theta(x) = y')$ is the probability of prediction y' by hypothesis h for input x, and $p(y')$ is the probability of the true label to be y'. For a given example (x, y):

$$\forall (x, y) \in \mathcal{Z} \;\; p(y') = \begin{cases} 1 & \text{if y' = y} \\ 0 & \text{otherwise} \end{cases} \tag{2.11}$$

Intuitively, the cross-entropy loss measures the average number of bits needed to identify an example if a coding scheme is based on the estimated distribution instead of the true distribution $\mathcal{D}$. This concept of entropy has also found applications in Negative Log-Likelihood (NLL) and Kullback-Leibler (KL) divergence loss. We refer the more interested readers to Ciampiconi et al. (2023), which provides an extensive overview of various loss functions.

2.3 Regularization

Loss functions are commonly evaluated over training data. A potential pitfall of minimizing these loss functions solely based on training data is the risk of overfitting. In essence, the hypothesis could excel on the training data but fail to generalize. To counteract overfitting and promote model simplicity (biasvariance tradeoff), an additional term known as regularization is incorporated alongside the loss function either implicitly or explicitly. An example of a regularization term is:

l_p **norms:** Parameterized by $p \geq 0$, the l_p norm can be defined as:

$$\forall\theta \in \Theta \quad \|\theta\|_p = (\sum_{i=1}^{d} \|\theta_i\|^p)^{\frac{1}{p}} \tag{2.12}$$

where Θ is a $\mathbb{R}^d$ space. In practice it is typical to set the value of p to 1 or 2 corresponding to l_1 (Tibshirani, 1996) and l_2 norm (Cortes and Vapnik, 1995).

In the fairness literature, various mechanisms, just like regularization, introduce an additional term to the loss function to promote fairness. This term penalizes the model for exhibiting unfair behavior towards specific subsets of the population. In Section 3.5.2, we list several such approaches.

2.4 Hypothesis Functions

In this section, we explore the hypothesis function, a crucial component of empirical risk minimization. Although a myriad of hypothesis functions exist, we will narrow our focus to Neural Networks in the classification context. We begin our discussion with simple linear neural networks, then build toward nonlinear architectures, including multi-layer perceptrons, convolutional neural networks, and transformers.

there is a confusion here: logistic regression is not a hypothesis function, it is the combination of choosing a linear hypothesis with softmax and the cross-entropy loss.

I would just call this linear model and note that combined with the cross-entropy loss, this leads to (multinomial) logistic regression

Linear Model: A simple yet prevalent hypothesis function is linear model which learns a linear map between the input space $\mathcal{X}$ and the discrete label space consisting of c labels $\{0, \cdots, c\}$. More specifically, linear model consists of transformation function, followed by a softmax funtion, and then an argmax function. The transformation function can be expressed as:

$$h_\theta(x) = \mathbf{W}x + \mathbf{b}$$

here, $\mathbf{W} \in \mathbb{R}^{c\times d}$, $\mathbf{b} \in \mathbb{R}^c$, $x \in \mathbb{R}^d$. The variables d represent the dimension of the input example. Typically, **W** is referred to as weight matrix and **b** is the bias vector. Together, they constitute the model's parameters, denoted by θ which includes $\mathbf{W}, \mathbf{b}$. In order to transform the output of $h_\theta(x)$ from a vector of numbers to vector of probability, it is typical combined with a softmax function and can be expressed as.

$$\begin{aligned}\mathbf{l} &= h_\theta(x)\\ \hat{y} &= \text{softmax}(\mathbf{l})\end{aligned}$$

where $\mathbf{l} \in [0,1]^c$ and $\hat{y} \in [0,1]^c$. Here, softmax is defined as:

$$\text{softmax}(\mathbf{l}) = [\frac{e^{l_0}}{\sum_{j=0}^{c} e^{l_j}}, \cdots, \frac{e^{l_i}}{\sum_{j=0}^{c} e^{l_j}}, \cdots, \frac{e^{l_c}}{\sum_{j=0}^{c} e^{l_j}}]$$

where l_i and l_j are the i-th and j-th coordinate of vector $\mathbf{l}$. In pratice, it is common to optimize the above function with cross entropy loss (See Section 2.2) and is called logistic regression. To get the final predictions, we take an argmax of $\hat{y}$. Thus the final linear hypothesis function which maps input x to output classes is:

$$\hat{final\_y} = \arg\max(\text{softmax}(h_\theta(x)))$$

where $\hat{final\_y} \in \{0, \cdots, c\}$.

Feed Forward Neural Network: While linear models are effective, it captures only linear relationships. To address this limitation, non-linearity is incorporated. A simple feed forward neural network also called two-layer multilayer perceptron can be expressed as:

$$\begin{aligned} \mathbf{x}_{transformed} &= \sigma(\mathbf{W_1}x + \mathbf{b_1}) \\ output &= \mathbf{W_2}\mathbf{x}_{transformed} + \mathbf{b_2} \\ \hat{y} &= \text{softmax}(output) \end{aligned}$$

In this equation, $\mathbf{W_1}, \mathbf{W_2}, \mathbf{b_1}, \mathbf{b_2}$ are the parameters of the hypothesis function, and σ is the activation function which introduces the non-linearity. Some of the most widely used activation functions include:

- **Sigmoid:** This activation function maps real numbers to the interval between 0 and 1. It is represented as:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.13}$$

- **Hyperbolic Tangent (tanh):** Unlike sigmoid, this function maps real numbers to the interval between -1 and 1. Its expression is:

$$\sigma(x) = tanh(x) \tag{2.14}$$

- **Rectified Linear Unit (ReLU) (Agarap, 2018):** It is a combination of a threshold and a linear function, which can be represented as:

$$\sigma(x) = \max(0, x) \tag{2.15}$$

Neural networks are compositions of multiple functions. For instance, linear model can be expressed as $e(f(g(x)))$ where $e()$ is the arg max function, $f()$ is the softmax function, and $g()$ is the linear model $\mathbf{W}x + \mathbf{b}$. In neural network terminology, each of these functions is a *layer* and their composition is akin to stacking these layers atop one another. Generally, all layers apart from the output layer are called hidden layers. For instance, in the multilayer perceptron architecture described above, the first hidden layer is $\sigma(\mathbf{W_1}x + \mathbf{b_1})$, the second is $\mathbf{W_2}\vec{x}_{transformed} + \mathbf{b_2}$, and the output

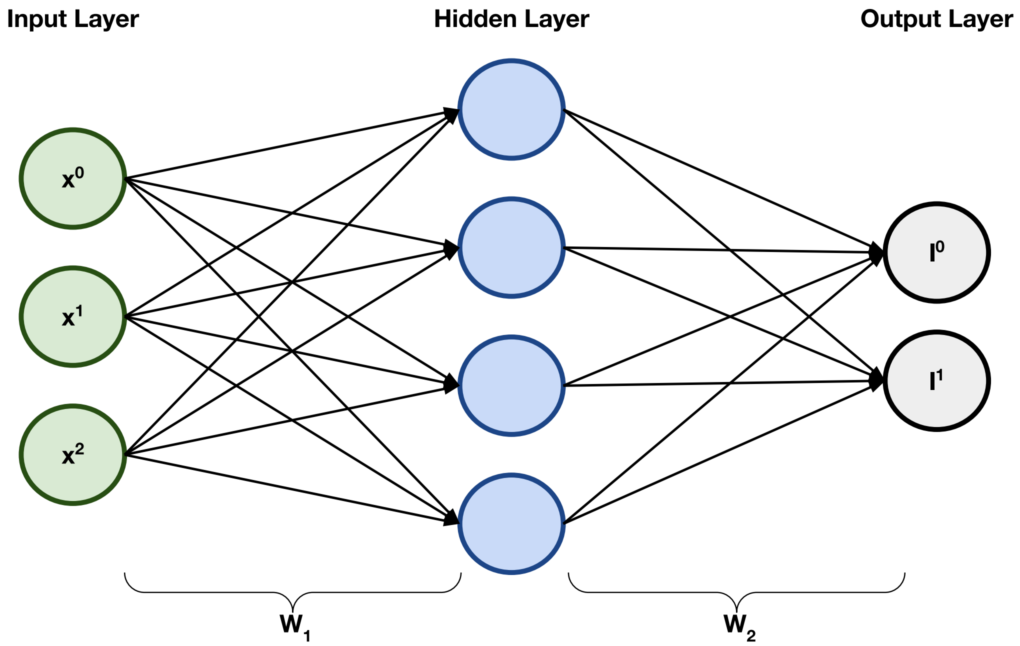


FIGURE 2.1: Neural Network with two hidden layers with input $\mathbf{x} \in \mathbb{R}^3$ and $\mathbf{l} \in [0,1]^2$.

layer is $\hat{y} = softmax(output)$. Given its two hidden layers, it is also called a two-layer feed forward network. A m layer feed forward network can be characterized as:

$$\begin{aligned}
\mathbf{l}_0 &= \sigma(\mathbf{W_1}x + \mathbf{b_1}) \\
&\vdots \\
\mathbf{l}_k &= \sigma(\mathbf{W_k}\mathbf{l}_{k-1} + \mathbf{b_k}) \\
&\vdots \\
\mathbf{l}_m &= \sigma(\mathbf{W_m}\mathbf{l}_{m-1} + \mathbf{b_m}) \\
\hat{y} &= \text{softmax}(\mathbf{l}_m)
\end{aligned}$$

Here $\mathbf{W_k}$ and $\mathbf{b_k}$ are the parameters of the layer k where $k \in \{0, \cdots, m\}$.

Figure 2.1, visualizes a two-layer neural network, alongside weight matrix and bias parameters. Different layers have been devised to cater to specific modalities and tasks:

- **Feed Forward Layer:** Comprises of a weight matrix that linearly transforms its input. Multi layer perceptron is built by stacking multiple feed forward layers with some non-linearities.
- **Convolutional Layer:** Typically used for image analysis, this layer performs convolutional operations on grid-like data using parameterized multi dimensional kernels.
- **Recurrent Layer:** Applied for sequential tasks, such as text processing, recurrent layers retain a "memory" of past inputs. This memory is updated according to

the current input. In essence, this layer functions similarly to a feed forward layer but shares parameters across inputs.

- **Attention Layer:** Drawing inspiration from the human concept of attention, this layer allows the neural network to focus on specific parts of the input. Variants include self-attention (Bahdanau, Cho, and Bengio, 2015), cross-attention, and multi-head attention (Vaswani et al., 2017).

A typical neural network combines multiple such layers. For example, ResNet-152 (He et al., 2016a) consists of 152 layers. Similarly, BERT (Devlin et al., 2019), a large neural network typically used for text processing, is composed of 12 transformer blocks. Each transformer block is further composed of multiple attention and feed forward layers.

2.5 Optimization

Recall, our objective is to find optimal parameters θ^* which minimizes the following empirical risk:

$$h_{\theta^*} = \arg\min_{\theta \in \Theta} \hat{R}(h_\theta) \tag{2.16}$$

In this section, we discuss common methods for selecting parameters that effectively capture the relationship between inputs and outputs. A direct method involves searching for a closed-form solution to the optimization problems outlined earlier. However, the hypothesis functions we explore in this study have several thousand parameters, rendering analytical solutions impractical. Furthermore, the typical loss functions used for classification often lack closed-form solutions. As a result, we turn our attention to gradient-based optimization techniques. Although these techniques do not always guarantee optimal parameter selection, they are generally effective in practical applications. Most of the gradient-based optimization techniques assume that the loss function is differentiable, a property exhibited by most of the previously discussed loss functions. We begin our discussion with gradient descent and then move on to various extensions commonly employed in machine learning.

Batch Stochastic Gradient Descent (SGD): It is a first-order optimization algorithm (Robbins, 1951; Kiefer and Wolfowitz, 1952) that iteratively refines its initial guess to find a minimum of the empirical risk. The core idea behind gradient descent is to take repeated steps in the opposite direction of the gradient of the loss function at the current step. Algorithm 1 illustrates Batch SGD:

Based on the batch size, the algorithm mentioned has three primary variants:

- Gradient Descent (GD): Here, the batch size is set to n, representing the total number of examples in the dataset. Consequently, the gradient at time t is the average of all the examples in the datasets. This variant is rarely used in practice due to its memory and computation intensive nature. Moreover the convergence is generally slow and suffers from overfitting when training deep neural network.

- Stochastic Gradient Descent (SGD): For this variant, the batch size is one. In other words, the parameters of the model are updated after each example in the dataset. While appealing, the convergence path of SGD is much noisier than GD.

Algorithm 1 Batch Stochastic Gradient Descent

Input: Dataset $\mathcal{T} = \{(x_i, y_i)\}_{i=1}^{n}$, hypothesis h_{θ_0} parameterized by $\theta^0 \in \mathbb{R}^d$ as our initial guess, $l(;)$ as the loss function, T as the number of iterations, learning rate α, and b is the batch size.

Output: θ^T

```
for $t = 0$ to $T - 1$ do
    $g_t = 0$
    Randomly sample $b$ number of examples from $\mathcal{T}$ and set it to $\mathcal{T}^b$
    $g_t = \frac{1}{b} \cdot \sum_{(\text{x,y})\ in\ \mathcal{T}^b} \nabla l(h_{\theta^t}(x), y)$
    $\theta^{t+1} = \theta^t - \alpha \cdot g_t$
end for
return $\theta^T$
```

- Batch Stochastic Gradient Descent (Batch SGD): This merges the benefits of the previous two variants by using a small batch size, encompassing a small subset of examples.

While Batch SGD has been employed in various practical applications, it tends to converge slowly due to the noise in individual gradients (Sutton, 1986). Additionally, it can get trapped in bad local minima where gradients become zero, resulting in no parameter update. We discuss a few optimizers designed to address these issues. Instead of detailing the entire optimization algorithm as in Algorithm 1, we focus on the core update step. For instance, the update step for Batch SGD is:

$$\theta^{t+1} = \theta^t - \alpha \cdot g_t \tag{2.17}$$

Stochastic Gradient Descent with Momentum: The core concept of momentum (Qian, 1999) involves considering past gradients. Specifically, it computes an exponentially weighted average of gradients to update model parameters. By accounting for prior gradients, momentum can build inertia, akin to a ball rolling downhill, to overcome local minima and oscillations arising from noisy gradients. The update step is as follows:

$$\begin{aligned} v^t &= \gamma \cdot v^{t-1} + \alpha \cdot g_t \\ \theta^{t+1} &= \theta^t - v^t \end{aligned}$$

Here, α is the learning rate, γ is the momentum term, v_{t-1} represents weighted sum of gradients until time $t - 1$, and g_t are the gradients at time t.

Adagrad: In addition to noisy gradients, a challenge with batch SGD is the necessity for practitioners to carefully adjust the learning rate. Moreover, in sparse data scenarios, certain parameters undergo updates more often than others. Adagrad (Duchi, Hazan, and Singer, 2011) addresses these issues by modifying the learning rate for each parameter independently. It applies larger updates for infrequently updated parameters and smaller updates for those adjusted more regularly. Before delving into a vectorized approach, we first demonstrate a per-parameter update. Let $g_{t,i}$ be the gradients of the parameter θ_i at time t for example (x, y):

$$g_{t,i} = \nabla l(h_{\theta^{t,i}}(x), y) \tag{2.18}$$

The update step of Adagrad for parameter $\theta^{t,i}$ becomes:

$$\theta^{t+1,i} = \theta^{t,i} - \frac{\alpha}{\sqrt{G_{t,ii} + \epsilon}} \cdot g^{t,i} \tag{2.19}$$

$G_t \in \mathbb{R}^{d*d}$ is a diagonal matrix, with ii representing the sum of squares of gradients with respect to θ_i until time t. In other words, G_t represents outer product of all previous gradients, i.e. $G_t = \sum_{\tau=0}^{t} g_\tau \cdot g_\tau^T$ Here ϵ is the smoothening factor to avoid zero divisibility. The overall vectorized updates step becomes:

$$\theta_{t+1} = \theta_t - \frac{\alpha}{\sqrt{G_t + \epsilon}} \odot g_t \tag{2.20}$$

Here $\odot$ represents the element wise matrix multiplication between two diagonal matrices, and (i, i) element in g_t represents gradients of parameter $\theta^{t,i}$.

Adam: Introduced by Kingma and Ba (2015), it combines the idea of momentum with Adagrad alongside bias correction terms. However, unlike Adagrad which normalizes the current gradients with the sum of squares of all previous gradients, Adam uses an exponential moving average strategy. The update step is as follows:

$$\begin{aligned}
m_t &= \beta_1 m_{t-1} + (1 - \beta_2) g_t \\
v_t &= \beta_1 v_{t-1} + (1 - \beta_2) g_t^2 \\
\hat{m}_t &= \frac{m_t}{1 - \beta_1} \\
\hat{v}_t &= \frac{v_t}{1 - \beta_2} \\
\theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{v_t} + \epsilon} \hat{m}_t
\end{aligned}$$

Here β_1 and β_2 are the initial decay rates, and $\hat{m}$ and $\hat{v}$ are bias correction terms. Throughout this thesis, we adopt Adam as our default optimizer, unless specified differently. For a comprehensive overview of other loss functions, we direct interested readers to Ruder (2016).

2.6 Training Neural Networks

In this section, we combine the ingredients discussed thus far to perform empirical risk minimization with neural networks. First, we introduce two key concepts - forward propagation and backpropagation.

Forward Propagation: As highlighted in Section 2.4, neural networks are essentially layers stacked upon one another. These layers progressively transform the input into the final output. This mechanism wherein the output of a preceding layer serves as the input for the subsequent layer is termed forward propagation.

Backpropagation: Recall, in Section 2.2 we discussed several loss functions which are then optimized using optimizers discussed in Section 2.5 to pick optimal parameters of the hypothesis function. These optimizers typically rely on calculating the gradient of each parameter with respect to the loss function. In order to efficiently compute these gradients, we commonly use the backpropagation algorithm (LeCun et al., 1989) based on dynamic programming and chain rule.

The central idea of backpropagation, a cornerstone algorithm in neural network training, is to compute these gradients through a systematic application of the chain rule, propagating the gradient of the loss backward through the network's layers. More specifically, the calculation of gradients of parameters in layer L^k with respect to the loss function l, only relies on the gradients of layer L^{k+1} with respect to loss function, the input to the layer L^k and the gradients of the output with respect to layer L^k. To illustrate backpropogation, consider a simple m layer feed forward network (as defined in Section 2.4) where a layer L^k is characterized as follows:

$$\vec{l}_k = \sigma(W_k \cdot \vec{l}_{k-1}) \tag{2.21}$$

Here, W_k is the weight matrix of layer L^k, and σ represents sigmoid activation function. $\vec{l}_{k-1}$ is the output of layer L^{k-1}. For simplicity, we exclude bias parameters. Using the chain rule, gradients of W_k with respect to loss l can be reformulated as:

$$\frac{\partial l}{\partial W_k} = \frac{\partial l}{\partial W_{k+1}} \frac{\partial W_{k+1}}{\partial W_k} \tag{2.22}$$

Given the derivative of $\sigma(x)$ with respect to x is $\sigma(x) \cdot (1 - \sigma(x))$, we can simplify the equation as:

$$\frac{\partial l}{\partial W_k} = \frac{\partial l}{\partial W_{k+1}} \cdot \sigma(W_k \cdot \vec{l}_{k-1}) \cdot (1 - \sigma(W_k \cdot \vec{l}_{k-1})) \cdot \vec{l}_{k-1} \tag{2.23}$$

This demonstrates that gradients $\frac{\partial l}{\partial W_k}$ only depend on the gradients of the next layer, gradient of the output of the current layer with respect to its parameters, and input to the current layer. Thus to compute the gradients for each layer, backpropogation starts with the last layer and then *propagates* the gradients backwards, giving the technique its name. Modern neural networks often utilize specialized autograd libraries like PyTorch (Paszke et al., 2019), TensorFlow (Abadi et al., 2015), and Jax (Bradbury et al., 2018) for computing these gradients.

With the necessary components in place, Algorithm 2 presents a conventional training loop. This process typically involves sampling training examples, passing them through the model (forward propagation), computing the loss, determining the gradients (backward propagation), and finally employing the optimizer to update the model parameters. In the fairness literature, various methods enhance this training procedure by incorporating instance reweighting (Iosifidis and Ntoutsi, 2019; Jiang and Nachum, 2020), refining the sampling mechanism (Chakraborty et al., 2020; Roh et al., 2021), and applying bi-level training strategies (Ozdayi, Kantarcioglu, and Iyer, 2021). We provide an overview of these techniques in Section 3.5.

Algorithm 2 Typical Training Procedure

```
Input: Input dataset $\mathcal{T}$
Output: Trained hypothesis.
Define and initialize hypothesis $h$.
Define and set hyperparameters of loss function $l$.
Define and set hyperparameters of optimizer $o$.
for $\mathcal{T}^b$ to $\mathcal{T}$ do
    Forward pass $\mathcal{T}^b$ through hypothesis $h$, and save its output.
    Calculate loss over the output.
    Calculate gradients with respect to the loss.
    Use optimizer to update the parameters of the hypothesis based on the gradients.
end for
return Trained hypothesis $h$.
```

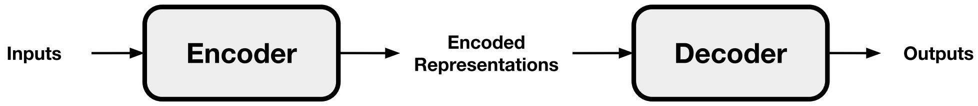


FIGURE 2.2: Encoder Decoder Networks which embeds the input to an intermediary representation which gets decoded by the decoder to final representation.

2.7 Common Neural Network Architectures

In this section we discuss two neural network architectures which we use throughout the thesis.

2.7.1 Encoder Decoder Networks

Neural networks have found applications across a diverse range of tasks, encompassing multiple input modalities like images (Li et al., 2014; Guo et al., 2017), text (Wang, Jiang, and Luo, 2016; Wang et al., 2018), videos (Karpathy et al., 2014; Kappeler et al., 2016), and music (Choi et al., 2017; Singh and Bohat, 2021). Correspondingly, their outputs can range from text generation (Floridi and Chiriatti, 2020; Su et al., 2021) and classification (Wang, Jiang, and Luo, 2016; Karpathy et al., 2014) to image generation (Ramesh et al., 2022; Wu, Lischinski, and Shechtman, 2021). For instance, in image captioning, the input is an image and the output is a text caption. In sentiment classification, the input is a textual passage and the outputs are its corresponding sentiment classes. To address this wide spectrum of modalities, a standard strategy in neural networks is the encoder-decoder architecture, which comprises two primary components:

- **Encoder:** This component of the model takes an example from the input space $\mathcal{X}$ and *encode* it to latent feature space $\mathcal{X}_{enc}$. Ideally, this mapping preserves all the information pertinent to the task within the encoded feature space.
- **Decoder:** This component of the model *decodes* the representation in $\mathcal{X}_{enc}$ to produce the output in space $\mathcal{Y}$.

With the use of the intermediary representation ($\mathcal{X}_{enc}$), the encoder-decoder architecture effectively decouples input and output modality. This flexibility means that different layers can be interchanged with minimal engineering overhead. Figure 2.2, visualizes this setup and we represent this architecture as follows:

$$\begin{aligned} enc &= E(x) \\ output &= C(enc) \end{aligned}$$

Here E is the encoder, C is the decoder, and x is the input example.

One of the disadvantage of large neural networks is that they require enormous amount of training data which is often not available. The encoder-decoder framework addresses this issue: the encoder can be pre-trained on a vast array of related data, often using unsupervised methods. This pre-training ensures that the encoder can learn overarching patterns within the modality, like edge detection in images or semantic nuances in text. Subsequently, the combined encoder-decoder structure, or occasionally just the decoder, is fine-tuned on the specific dataset in question. This approach will be especially useful in our context, as fairness datasets tend to be limited in volume and span various modalities.

2.7.2 Adversarial Networks

Adversarial Networks augment the encoder-decoder architecture by equipping it with the capability to selectively remove specific types of information during the encoding process. As we will see in Chapter 7 the architecture can be used to enforce fairness by removing sensitive demographic information from the encoded representation. This is typically achieved through the addition of an adversarial classifier designed specifically to predict this sensitive attribute. The encoder, in this setup, has a twofold role: (i) generating a representation that prevents the adversarial classifier from discerning sensitive information and (ii) ensuring that the primary task-relevant information remains intact in the encoded representation. Formally, the adversarial neural networks can be characterized as:

$$\begin{aligned} enc &= E(x) \\ output &= C(enc) \\ adv\_output &= A(enc) \end{aligned}$$

Here E is the encoder, C is the task classifier, A is the adversarial classifier, and x is the input example. A simple way to train this network would be to add the adversarial loss to the task loss. However, adding these two loss functions would result in the encoder striving to improve both the adversarial and task classifiers. Instead, we reverse the sign of gradients flowing from the adversarial classifier to the encoder. This results in the encoder aiming to make the adversarial classifier worse while the adversarial classifier tries to improve itself based on the encoder representation. In other words, the adversarial classifier and the encoder play a minmax game. Thus, the final optimization equation can be represented as:

$$\min_{\theta_E,\theta_C}\max_{\theta_A} l_{class}(\theta_E,\theta_C) - \lambda l_{adv}(\theta_E,\theta_A), \tag{2.24}$$

where $l_{class}(\theta_E,\theta_C)$ is the loss function for the encoder and classifier branch, while $l_{adv}(\theta_E,\theta_A)$ is the loss function for adversarial brach. θ_E, θ_C, θ_A are the parameters of the encoder, task classifier, and adversarial classifier respectively. Hyperparameter $\lambda \geq 0$ here represents the tradeoff between adversarial and task loss.

2.8 Conclusion

In this chapter, we presented an overview of the learning framework, emphasizing empirical risk minimization. We outlined its fundamental components, which include: (i) the hypothesis function, with a particular focus on neural networks; (ii) the loss function that quantifies the performance of the hypothesis function; (iii) optimization that selects the optimal parameters for a given hypothesis class; and (iv) the training procedure that ties all these components together. Additionally, we discussed a few prevalent neural network architectures. In subsequent chapters, we will build upon these foundations to enhance fairness in machine learning systems. In the next chapter, we delve deeper into fairness in machine learning.

Chapter 3

Fairness in Machine Learning

In Chapter 1, we introduced various sources of unfairness in machine learning in the context of a generic pipeline. In this chapter, we delve deeper into fairness in machine learning, covering historical context, common metrics, and prevalent methods. We also provide a general introduction to the problem and define key terminology used throughout the thesis. Note that while this chapter gives a comprehensive background necessary to understand our contributions, we reserve the detailed examination of methods and metrics closely related to our contributions to their corresponding chapters.

3.1 History of Studies on Fairness

While fairness in machine learning is a relatively new field, with early roots tracing back to seminal works by Pedreschi, Ruggieri, and Turini (2008), Dwork et al. (2012), and Calders, Kamiran, and Pechenizkiy (2009), concerns about fairness in broader social systems are much older. Hutchinson and Mitchell (2019) trace the history of fairness to the United States Civil Rights Act of 1964,[1] which effectively outlawed discrimination based on identities such as gender, color, or race in government and employment sectors.

This act shaped public opinion on unfairness and spurred research efforts in the field of social science. Early focus areas included employment sectors (Guion, 1966; Williams et al., 1980) and standardized testing in higher education (Cleary, 1966), where various concepts were proposed to define and measure unfairness (Petersen and Novick, 1976; Darlington, 1971) across social and cultural contexts. Interestingly, several contemporary concepts currently emerging in fair machine learning have parallels to these older notions. Yet, as highlighted by Hutchinson and Mitchell (2019), this research field faded in the 1970s as multiple, often conflicting fairness concepts left practitioners confused on the applicability and validity of these notions.

With machine learning rapidly automating critical bureaucratic functions, as discussed in the Introduction (see Chapter 1), the potential for harm has sparked calls for greater accountability and transparency by researchers (Weidinger et al., 2021; Burrell, 2016; Metcalf and Crawford, 2016), government agencies (Commission, 2018;

[1] https://www.dol.gov/agencies/oasam/civil-rights-center/statutes/civil-rights-act-of-1964

Barocas et al., 2017) and NGOs (Buchanan, 2012). This has reinvigorated interest in fairness, with researchers responding in two primary ways:

- **Define and Detect Unfairness:** By formulating metrics (Zafar et al., 2017a; Berk et al., 2021), testing framework (Jentzsch et al., 2019; May et al., 2019), and datasets (Nadeem, Bethke, and Reddy, 2021; Nangia et al., 2020) that encapsulate and define various facets of harm. These can be divided into two categories based on the type of harm uncovered: allocation harm and representational harm (see Chapter 1).
- **Fairness Promoting Mechanisms:** By developing mechanisms to mitigate the aforementioned formalized harms through data manipulation (Kamishima et al., 2012; Calders and Verwer, 2010), training modifications (Cotter, Jiang, and Sridharan, 2019; Kearns et al., 2018), and post-processing of the machine learning pipeline output (Hardt, Price, and Srebro, 2016; Kleinberg, Mullainathan, and Raghavan, 2017).

In this context, our contribution in Chapter 5 belongs to the category of designing metrics, while Chapters 4, 6, and 7 fall under the category of intervention strategies. In Section 3.3 we provide an in-depth overview of allocation harm based metrics, which is the focus of thesis. We then provide a brief overview of metrics related to allocation harm in Section 3.4. Finally, in Section 3.5, we delve into fairness interventions mechanisms. However, we first begin by describing a simple case study which we will use to illustrate key concepts. We also formalize the notion of sensitive groups and introduce notation that will be used across the thesis.

3.2 Groups and Performance Measures

In this section, we introduce key terminology used throughout the thesis using a case study as an illustrative tool. The case study involves a simplified binary task of determining creditworthiness (accept or reject) of an applicant while ensuring fairness across gender (binary) and race (binary). In this setup, the favorable label benefiting applicants is "accept". To formalize the case study, we first discuss the concepts of sensitive attributes and corresponding sensitive groups. Finally, we also introduce the concept of group wise performance measure which forms the basis of several fairness definitions introduced later.

3.2.1 Groups

Sensitive Axes Most fairness interventions in machine learning aim to achieve *equality* among various sensitive or protected groups, terms we use interchangeably throughout this thesis (Verma and Rubin, 2018). These groups are in turn defined using demographic attributes of the population such as gender, ethnicity, and age. In this thesis, we refer to these socio-demographic features as *sensitive axes*. Although the specific set of sensitive axes may vary depending on the application, several frameworks legally categorize certain axes as sensitive (Yeung, 2018; Lee, 2018).

It is imperative to carefully select these sensitive axis as they underpin most fairness interventions. Omitting a relevant axes implies fairness approaches are unlikely to positively impact the excluded groups. Additionally, some attributes, while not explicitly sensitive, can act as strong proxies for protected characteristics, e.g., zip codes serving as indicators of race (Datta et al., 2017; Chen and Krieger, 2021).

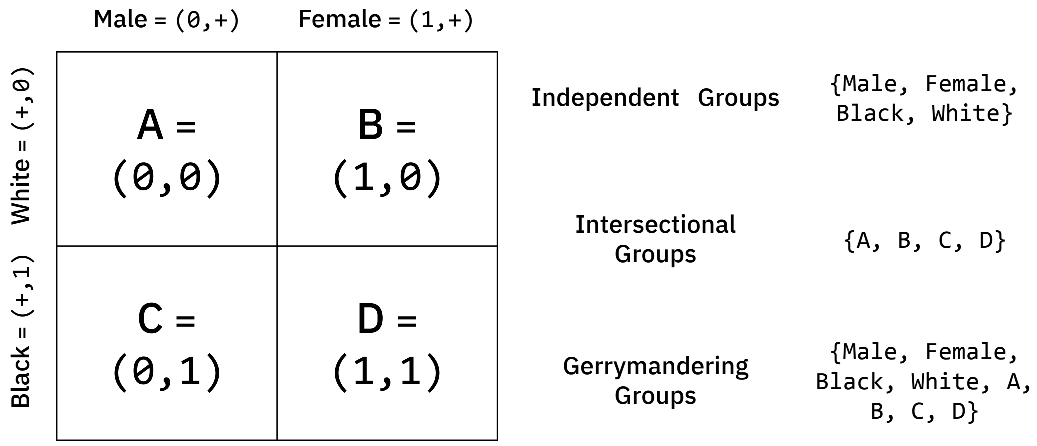


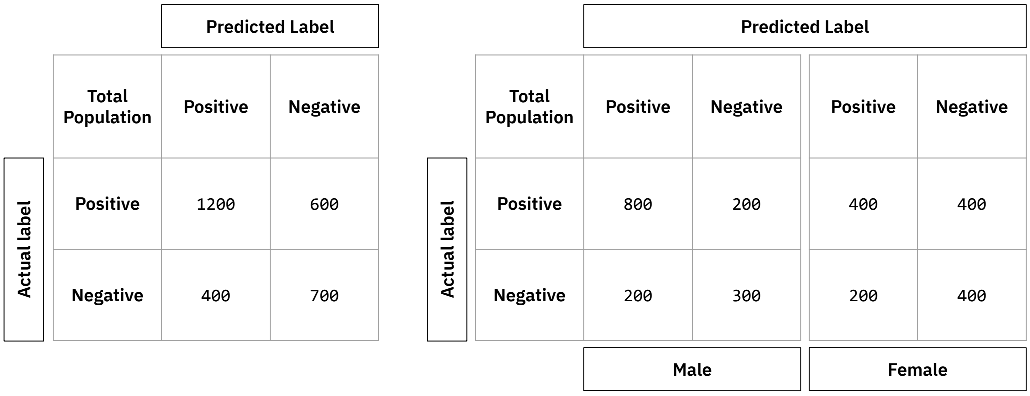
FIGURE 3.1: Various ways of partitioning groups and their corresponding encoding mechanism.

Therefore, practitioners must diligently identify and define sensitive axes for their applications. The chosen axes largely influence the scope and efficacy of fairness-enhancing solutions (Caton and Haas, 2020).

Notations for Sensitive Axes Let p represent the number of distinct *sensitive axes* of interest. We denote these axes as $A_1, \ldots, A_p$, where each A_i is a set of discrete-valued sensitive attributes. For example, in our case study the sensitive axes are gender and race, where gender= {male, female} and race= {European American, African American}. Furthermore, we add another sensitive attribute $+$ to each A_i, representing the union of all sensitive attributes for that axis. We illustrate and clarify the use of $+$ in the subsequent paragraph. Finally, for simplicity we encode these categorical values as $\{0,1\}$. Thus the final representation is $A_1 = \{0,1,+\}, A_2 = \{0,1,+\}$, where A_1 and A_2 correspond to gender and race respectively.

Sensitive Groups After determining the sensitive axes, we partition the population into sensitive groups. A *sensitive group* $\mathbf{g}$ is any k-dimensional vector in the Cartesian product set $\mathcal{G} = A_1 \times \cdots \times A_p$ of these axes. A group $\mathbf{g} \in \mathcal{G}$ can be expressed as $(a_1, \ldots, a_p)$, where $a_j \in A_j$. For instance, in our case study with two sensitive axes, the group of all males which are European American is $(0,0)$. As previously mentioned, if one or more axes are not considered, their value defaults to $+$, representing the union of all the attributes for the said axis. For instance, the group of all females can be represented as $(1,+)$, which effectively entails the union of the groups $(1,0)$ and $(1,1)$. These sets of sensitive groups can be further divided into three major categories as identified by Yang, Cisse, and Koyejo (2020):

- **Independent Groups:** Comprised of groups formed by creating a separate group for each attribute value within each sensitive axis. These groups overlap, meaning a member can belong to multiple groups. In our notation, groups with "$+$" for all dimensions except one constitute this set. For p binary sensitive attributes, this set contains $2p$ groups.
- **Intersectional Groups:** Consist of groups derived from all possible combinations of all the sensitive axes. These groups are non-overlapping, ensuring members belong to only a single group. In our notation, all groups having



		Predicted Label	
	Total Population	Positive	Negative
Actual label	Positive	1200	600
	Negative	400	700

		Predicted Label			
	Total Population	Positive	Negative	Positive	Negative
Actual label	Positive	800	200	400	400
	Negative	200	300	200	400
		Male		Female	

FIGURE 3.2: Overall performance of a classifier and the performance split on the basis of gender.

no $+$ in any dimension constitute this set. For p binary sensitive axes, this set comprises of 2^p groups.

- **Gerrymandering Groups:** Encompass groups formed using any combination of sensitive axes. Like independent groups, these groups overlap. In our notation, all possible group representations form this set. Given p binary sensitive axes, the total number of groups in this set is $3^p - 1$.

Formalizing the Case Study Let the input be represented as $x \in \mathcal{X}$. In our case study, x could include features such as an applicant's credit history, income, and address. Each input is associated with a binary response variable, $y \in \{0, 1\}$, indicating "accept" or "reject". Moreover, each example has two binary sensitive axes: A_1 for gender and A_2 for race. Figure 3.1 illustrates all potential groups and their associated categories in this context. We also consider a model h_θ parameterized by θ, trained on this dataset. This model produces predictions as in the form of $\hat{y} \in \{0, 1\}$ where 0 stands for "reject" and 1 stands for "accept".

3.2.2 Performance Metrics

Most of the fairness evaluation mechanisms rely on calculating group wise performance measure and then comparing them. In this subsection, we illustrate the commonly performance metrics by using the case study described above. These performance metrics can be explained through a confusion matrix, which is presented in Figure 3.2. The left side of Figure 3.2 shows a hypothetical confusion matrix for model h_θ applied to the case study dataset. The right side splits this by gender. For simplicity, we omit race in these examples.

- **True Positive (TP):** Refers to instances where the model correctly predicts the positive class. In other words, the predicted label matches the actual label, and both belong to the positive class. In our case study, there are 1200 true positives; 800 for males and 400 for females when broken down by gender.

- **False Positive (FP):** Denotes instances where the model incorrectly predicts the case to be positive class when the actual label is negative. In our study, there are 400 false positives; 200 for males and 200 for females.

- **True Negative (TN):** Akin to TP, it refers to cases where the model correctly predicts the label of examples belonging to negative class. In our case study, 700 are true negatives; 300 for males and 400 for females.
- **False Negative (FN):** Analogous to FP, this represents situations where the model incorrectly labels negative instances. In our study, there are 600 false negatives; 200 for males and 400 for females.

Based on these quantities, we compute the following performance metrics:

- **True Positive Rate (TPR):** Also known as sensitivity or recall, this metric measures the proportion of actual positive cases that are correctly identified as positive ($\frac{TP}{TP+FN}$). In our study, the overall TPR is $\frac{2}{3}$, with $\frac{4}{5}$ for males and $\frac{1}{2}$ for females. From a probabilistic perspective, it refers to the probability of the positive examples to be classified as positive ($P\left(\hat{y}=1|y=1\right)$).
- **False Positive Rate (FPR):** Also called specificity, this metric quantifies the fraction of negative examples incorrectly predicted as positive class ($\frac{FP}{FP+TN}$). In our context, the FPR is $\frac{4}{11}$ overall, with $\frac{2}{5}$ for males and $\frac{1}{3}$ for females. From a probabilistic perspective, it refers to the probability of the negative examples to be classified as positive ($P\left(\hat{y}=1|y=0\right)$).
- **Accuracy:** Refers to the fraction of correctly predicated examples (positive or negative) out of all the examples ($\frac{TP+TN}{TP+TN+FP+FN}$). In our study, the overall accuracy is $\frac{19}{29}$; $\frac{11}{15}$ for males and $\frac{4}{7}$ for females. From a probability standpoint, it assesses the likelihood of an instance being correctly classified, irrespective of label ($P\left(\hat{y}=a|y=a\right) \forall a \in \{0,1\}$).

These metrics are inherently in range between $[0,1]$. It is important to note that for most of the performance metrics mentioned above, higher values signify superior performance, except for FPR where lower values are preferable. We now begin our discussion on various fairness definitions.

3.3 Metrics for Allocation Harm

Allocation harm occurs when a machine learning system unevenly allocates resources and opportunities to various population subgroups. As defined by Mehrabi et al. (2022), *fairness denotes the absence of prejudice or favoritism towards an individual or a group based on their inherent or acquired traits during decision-making*. However, translating this broad perspective into a precise definition is challenging (Chan, 2011), as terms like "absence","prejudice", and "group" can shift in meaning across social contexts and applications. Given the absence of a universal definition of fairness, it is unsurprising that the research community has proposed myriad, and at times conflicting (Kleinberg, Mullainathan, and Raghavan, 2017; Chouldechova, 2017), concepts of fairness. In this section, we delve into several prevalent definitions that aim to encapsulate and quantify unfairness in machine learning systems. We categorize these definitions into three primary categories:

- **Independent Group Fairness:** These definitions typically consider fairness along single sensitive axis, like gender or race. For datasets featuring multiple sensitive axes, the definitions treat them independently, akin to the Independent groups mentioned in Section 3.2.1. For example, if a dataset contains both gender and race, and a model satisfies these fairness notions, it indicates fairness with respect to race and gender individually, not jointly.

- **Intersectional Group Fairness:** These definitions are usually applied when multiple sensitive axes are present in the dataset. Unlike previous definitions, they consider all axes jointly rather than independently, akin to the Intersectional groups mentioned in Section 3.2.1. Thus, if a model satisfies intersectional fairness for a dataset with gender and race, it indicates fairness with respect to the combination of gender and race simultaneously, not just each attribute individually.
- **Individual Fairness:** Distinct from the group-centric definitions mentioned above, these definitions emphasize fairness at the individual instance level, advocating that similar individuals receive similar treatment.

This section is geared towards classification problems in independent and intersectional group fairness setting, which is the focus of this thesis. We provide a very brief overview of individual fairness and refer the interested readers to Mehrabi et al. (2022). Moreover, we assume sensitive axes are pre-defined and all sensitive attributes for each example are present.

3.3.1 Independent Group Fairness

Independent group fairness forms the most widely used set of fairness definitions, with over 10 specific definitions identified by researchers (Verma and Rubin, 2018; Mehrabi et al., 2022). The core idea behind these definitions is to compare performance metrics (as outlined in the preceding section) across various independent groups. In this subsection, given that sensitive attributes are evaluated independently, we narrow our case study to exclusively focus on binary gender. As explained above, we represent the two gender groups, male and female, as $\mathbf{g} = (0, +)$ and $\mathbf{g}' = (1, +)$, respectively.

Demographic Parity (also called Statistical Parity by Dwork et al. 2012, Equal Acceptance Rate by Zliobaite 2015): A model h_θ satisfies demographic parity (Dwork et al., 2012) with respect to all group $\mathbf{g} \in \mathcal{G}$, where $y = 1$ is the preferred label if:

$$P\left(h_\theta(x) = 1\right) = P\left(h_\theta(x) = 1|\mathbf{g}\right)$$

In other words, the probability of being classified as credit worthy ($y = 1$) should be equal between for group $\mathbf{g}$ and the overall population, regardless of the true label. Demographic parity is typically employed in scenarios where the underlying representations and the label cannot be trusted as one of the groups has been historically prejudiced. In other words, demographic parity is generally applied in settings where bias such as representational bias, or historical bias can emerge during the data collection phase (see Section 1.2.1). For example, an automated loan allocation system might disproportionately reject minority applicants as it has been trained on data that shows strong historical stigma against them. However, beyond the specific contexts such as the one described before, applying demographic parity might lead to unintended consequences, as it overlooks the inherent differences between the groups.

Equal Opportunity (also known as Predictive Parity by Chouldechova 2017): A model h_θ satisfies equal opportunity (Hardt, Price, and Srebro, 2016) with respect to group $\mathbf{g} \in \mathcal{G}$, where $y = 1$ is the preferred label if:

$$P\left(h_\theta(x) = 1 | y = 1\right) = P\left(h_\theta(x) = 1 | \mathbf{g}, y = 1\right)$$

In other words, the probability that a credit worthy person is classified as credit worthy should be equal for any group and the overall population. In contrast to demographic parity, equal opportunity focuses on fairness specifically among qualified candidates who "deserve" the positive classification. This implies, while demographic parity aligns positive prediction rates across all groups regardless of qualifications, equal opportunity aims to equalize the rate only amongst qualified people.

Accuracy Parity: A model h_θ satisfies accuracy parity (Berk et al., 2021) with respect to all group $\mathbf{g} \in \mathcal{G}$ if:

$$P\left(h_\theta(x) = a | y = a\right) = P\left(h_\theta(x) = a | \mathbf{g}, y = a\right) \forall a \in \{0, 1\}$$

In other words, if the accuracy between any group and overall population is similar then it satisfies accuracy parity. However, one shortcoming of accuracy parity is its inability to differentiate between distinct error types, like false positives and false negatives, which may impact different subgroups differently.

Equalized Odds (also called Disparate Mistreatment by Zafar et al. 2017a): A model h_θ satisfies equalized odds (Hardt, Price, and Srebro, 2016) with respect to all group $\mathbf{g} \in \mathcal{G}$ if:

$$P\left(h_\theta(x) = 1 | y = a\right) = P\left(h_\theta(x) = 1 | \mathbf{g}, y = a\right) \forall a \in \{0, 1\}$$

This means that a model achieves equalized odds when its predictions are independent of group membership, given the true label y. It can also be interpreted as requiring the same false positive and true positive rates across all groups. In contrast, equal opportunity only considers true positive rates. While equal opportunity focuses solely on true positive rates, equalized odds, by considering both false and true positive rates, recognizes that misclassifications can disproportionately affect disadvantaged groups (Weerts et al., 2023). For instance, while modelling negative outcomes, such as recidivating, which already disproportionately affects minority, false positives reflect the pre-existing disparities in outcomes between groups. Furthermore, in contrast to accuracy parity, equalized odds specifically considers distinct error rates, requiring similar FPR and TPR across all groups.

In the next two chapters, we will extensively utilize these definitions to quantify unfairness. Some other independent group fairness metrics include equalizing disincentives (Jung et al., 2020), false positive error rate balance (Chouldechova, 2017), and treatment equality (Berk et al., 2021). The definitions we have discussed until now focus solely on the predicted outcome and the true label. However, some fairness definitions, termed as calibration-based fairness definitions, consider predicted probabilities instead of just the final outcome. Examples include test fairness (Chouldechova, 2017) and well-calibration (Kleinberg, Mullainathan, and Raghavan, 2017). For an extensive list of both independent and calibration fairness definitions, we refer the interested readers to Verma and Rubin (2018).

3.3.2 Quantifying Unfairness in Independent Group Fairness

In the literature, unfairness is typically quantified either as the absolute difference or as the ratio of the metric being studied. According to demographic parity, the unfairness of the model, defined based on the difference is, ϵ if:

$$-\epsilon \leq P\left(h_\theta(x) = 1\right) - P\left(h_\theta(x) = 1|\mathbf{g}\right) \leq \epsilon$$

while the unfairness of the model, defined based on the ratio, is ϵ if:

$$\frac{1}{\epsilon} \leq \frac{P\left(h_\theta(x) = 1\right)}{P\left(h_\theta(x) = 1|\mathbf{g}\right)} \leq \epsilon$$

A classifier is said to be strictly fair if $\epsilon = 0$. In our case study (see Figure 3.2), ϵ for gender is 0.34 and 1.62 for difference and ratio respectively. In many cases, instead of aiming for strict equality, models are evaluated based on approximate fairness. Here, the goal is to achieve comparable performance within a specific threshold. The idea originates from a widely-accepted guideline suggesting the selection ratio for minorities should be within 80% of the majority selection.[2] This corresponds to setting ϵ as 0.8 in the above equation. It is crucial to emphasize that these guidelines were formulated specifically within the employment context. Applying them elsewhere could be problematic and potentially harmful.[3]

3.3.3 Intersectional Group Fairness

These definitions are based on the idea that discrimination cannot be captured by looking at a single identity alone (Crenshaw, 1989). A model might exhibit fairness along a single sensitive dimension, yet display bias towards intersectional groups. For instance, Buolamwini and Gebru (2018) found that multiple commercial classifiers exhibited higher error rates for darker-skinned females compared to their lighter-skinned male counterparts. In contrast to independent group fairness, the emphasis here is on achieving fairness for both intersectional and gerrymandered groups. This subfield has attracted relatively limited attention, characterized by substantially fewer definitions compared to independent group fairness. Among these, Kearns et al. (2018) pioneered the concept of subgroup fairness, marking one of the earliest attempts to quantify intersectional unfairness.

Subgroup Fairness: A model h_θ is γ-SG fair , if $\forall \mathbf{g} \in \mathcal{G}$:

$$\left|P\left(h_\theta(x) = 1\right) - P\left(h_\theta(x) = 1|\mathbf{g}\right)\right| \cdot P\left(\mathbf{g}\right) \leq \gamma$$

Subgroup fairness consists of the product of two terms, namely: (i) The difference between the overall TPR and the TPR for the group in question, and (ii) the size of the group. The aforementioned equation is similar in spirit to equal opportunity

[2] https://www.law.cornell.edu/cfr/text/29/1607.4

[3] https://fairlearn.org/v0.9/user_guide/fairness_in_machine_learning.html#the-portability-trap

described in the previous section, as both take TPR into account. Kearns et al. (2018) also proposed a similar definition for FPR where they replaced the TPR term in the equation with FPR.

Definitions of subgroup fairness have also been adapted to other contexts. Notably, Hébert-Johnson et al. (2018) and Gopalan et al. (2022) generalized the definition to calibration-based fairness notions that consider predicted probabilities, while Yona and Rothblum (2018) addressed the problem by considering the distance between groups. For a comprehensive overview of various intersectional fairness measures, we direct interested readers to the work by Gohar and Cheng (2023).

One potential drawback of subgroup fairness is its inclusion of the second term. By reweighing the outcome by the size of the subgroup, the definition reduces the impact of small subgroups. Consequently, smaller subgroups, which are often the most vulnerable, may remain unprotected.

Differential Fairness: To circumvent the aforementioned issue, Foulds et al. (2020) proposed Differential Fairness (*DF*), which puts a constraint on the relative performance between all pairs of groups. A model h_θ is ϵ-Differentially Fair, if $\forall \mathbf{g}, \mathbf{g}' \in \mathcal{G}$:

$$\frac{P\left(h_\theta(x) = 1|\mathbf{g}'\right)}{P\left(h_\theta(x) = 1|\mathbf{g}\right)} \leq \gamma$$

In contrast to prior definitions, DF safeguards all groups, not just the larger ones. Furthermore, it also has other useful properties such as by evaluating *DF* on only intersectional groups ensures fairness over gerrymandered groups. Morina et al. (2019) extended *DF* to other group fairness notions as well, including false positive rate equality and equalized odds.

> In Chapter 5, we highlight several limitations of *DF*. Our analysis reveals that *DF* can be trivially satisfied by harming all involved groups, potentially hiding the leveling down phenomena. It is primarily due to its strict egalitarian view that considers only relative, not individual, group performance. We also introduce new definitions that not only generalize over *DF* but also mitigate these shortcomings.

3.3.4 Individual Fairness

In contrast to group-based fairness measures, which assign individuals to a sensitive group and then compare and contrast group wise performance, individual fairness definitions make direct comparisons between individuals in the dataset. In this subsection, we delve into some of these definitions. For a more comprehensive treatment, we refer the interested readers to Mehrabi et al. (2022).

Counterfactual Fairness: A model satisfies counterfactual fairness (Kusner et al., 2017) if its prediction for an individual is the same in the actual world and in a counterfactual world where the individual belongs to a different sensitive group. Such definitions are commonly applied when explicit demographic attributes are available. For instance, in text-based problems where gender is the sensitive attribute, inputs can be transformed by altering gendered words, such as names and pronouns in English language. Several works (Kilbertus et al., 2017; Chiappa, 2019) have also approached this fairness definition from a causal perspective, examining the influence of the sensitive attributes on the final predictions. One of the difficulties in applying this definition is that implicit indicators might predict sensitive attributes, and thus must be carefully considered when creating counterfactuals.

Fairness Through Awareness: A model satisfies fairness through unawareness if it gives similar predictions for similar individuals (Dwork et al., 2018). The central concept of this definition involves proposing a similarity metric between individuals and then evaluating the model's behavior over similar individuals. A significant advantage of these definitions is their acknowledgment of plurality of individuals amongst the same groups. However, defining the right similarity metric to capture relevant similarities is often challenging in practice, limiting their applicability.

3.4 Metrics for Representational Harm

Representational harm arises when a machine learning system misrepresents some sensitive groups, often reinforcing their subordination (Shelby et al., 2022). This includes stereotyping (Weidinger et al., 2022), erasing (Katzman et al., 2023), alienating (Wang, Ramaswamy, and Russakovsky, 2022; DeVos et al., 2022), and demeaning groups (Sweeney, 2013). Unlike allocation harms, representational harms are typically indirect with long-term implications, making them harder to quantify. Moreover, allocation harm focuses on the final classification outputs, which are easier to measure. In contrast, representational harm involves the model's internal representations and requires deeper social/cultural understanding. This is similar in spirit to individual fairness discussed in the previous section. To capture such unfairness, researchers typically construct challenge sets, featuring both stereotypical and anti-stereotypical examples, and then compare model representations and outputs against them. These challenge sets are generally domain, language, and application specific. In this section, we briefly outline some of these methods.

Word Embedding Association Test (WEAT): The test (Jentzsch et al., 2019) quantifies bias in word embeddings using the cosine similarity between two sets of target words and two sets of attribute words. It involves constructing attribute word sets representing two sensitive groups, and target word sets with stereotypical relations to the attribute sets. Embeddings are considered unbiased if the relative similarity between target and attribute sets is equal. In other words, WEAT evaluates how related the target word sets are to each attribute set. It was originally designed for binary gender-occupations and race-pleasant/unpleasant word biases in English. Several extensions have been proposed in the literature, including new languages (Sabbaghi and Caliskan, 2022; Mulsa and Spanakis, 2020) and improvements in the underlying mechanism (Schröder et al., 2021; Ethayarajh, Duvenaud, and Hirst, 2019). Most notably, May et al. (2019) proposed an extension to evaluate contextualized embedding models. Here, instead of a set of words, sentence templates such as "[Male Name/Female Name] is a/an [Occupation]" are used for evaluating bias.

Context Association Test (CAT): Proposed by Nadeem, Bethke, and Reddy (2021), the test measures the language model affinity towards stereotypical outputs compared to neutral and anti-stereotypical outputs. This is operationalized as a sentence completion task with three completion options, namely: (i) Anti-stereotypical, (ii) Stereotypical, and (iii) Meaningless. The fairness score is based on the number of times the model choose a stereotypical setting in comparison to the other options. Similarly, Nangia et al. (2020) proposed the Crowdsourced Stereotype Pairs benchmark, where a model is presented with a more and less stereotyping sentence. In this case, they used a pseudo-log-likelihood fairness metric based on the perplexity score of all tokens conditioned on the stereotypical tokens in the sentence.

In Natural language Processing, several similar challenge tasks, such as Discovery of correlations (Webster et al., 2020), Direct Bias (Bolukbasi et al., 2016), have been proposed. A primary limitation with these challenge sets is their limited applicability with monolingual focus. Most of these solutions need to be adopted for different languages and cultures. Additionally, a model devoid of representational harm does not guarantee fairness against allocation harm. Overall, there is a need for *comprehensive formal testing alongside alongside multi-faceted bias measures* (Stanczak and Augenstein, 2021). In conclusion, akin to the concerns with allocation harms, practitioners should exercise caution when utilizing representational harm metrics. Performance on these limited benchmarks does not necessarily indicate fairness in deployment.

3.5 Fairness Promoting Mechanisms

In this section, we provide a brief overview of various fairness intervention approaches in machine learning. Following d'Alessandro, O'Neil, and LaGatta (2019) and Mehrabi et al. (2022), we categorize these algorithms according to their point of intervention in the ML pipeline to mitigate harm:

- **Pre-Processing:** These methods transform the training data with the goal of making models trained on transformed data fairer
- **In-Processing**: These approaches manipulate the training mechanism and the model itself to promote fairness.
- **Post-Processing**: These strategies adjust the outputs of a previously trained, yet potentially unfair, model to enhance fairness.

We will now provide a brief overview of all the three categories. Note that this section is heavily derived from various survey works on fairness in machine learning including d'Alessandro, O'Neil, and LaGatta (2019), Mehrabi et al. (2022), Caton and Haas (2020), Parraga et al. (2022), and Gohar and Cheng (2023). Moreover, with myriads of fairness interventions approaches proposed in the literature, not every approach neatly falls into the above three categories (Caton and Haas, 2020). Several of them are hybrid falling into multiple categories, or into neither of them. Lastly, similar to the prior section, the framework in which these fairness approaches are defined is classification settings and assume access to sensitive axes.

3.5.1 Pre-Processing Methods

These methods address the problem of fairness in machine learning at the data collection and preparation stage (see Section 1.2.1). The central idea is that models trained on more representative and less biased data will be fairer (Parraga et al., 2022). In the following discussion, we will highlight several techniques commonly used to modify data.

Blinding Methods: A direct strategy to address fairness involves removing sensitive attributes from the data (Kamishima et al., 2012). For instance, in our case study, we might decide against collecting gender information during the data gathering process. However, various studies have indicated that such *blinding* of models to sensitive variables often leads to a decrease in accuracy while still perpetuating unfairness (Calders and Verwer, 2010; Kamishima et al., 2012; Pedreschi, Ruggieri, and Turini, 2008). Another major concern is that other variables can serve as proxy indicators, potentially predicting the sensitive attribute and thus introducing overt (Kleinberg et al., 2018) or latent biases (Pin Calmon et al., 2018). For instance, name can be a proxy indicator for age and location. Moreover, many problems, especially those involving text and images, do not have clear-cut sensitive attributes that can be readily excluded.

Causal Methods: These mechanisms (Galhotra, Brun, and Meliou, 2017; Kusner et al., 2018; Salimi, Howe, and Suciu, 2019) focus on identifying causal relationships between the model output and sensitive attributes, and leveraging this understanding to modify the training data. For instance, Capuchin (Salimi et al., 2019) excludes certain data points and adjusts the empirical distribution to minimize the influence of the sensitive attribute on the model's predictions. Other works, such as those of Adler et al. (2018) and Chiappa and Isaac (2018), have employed analogous mechanisms to model the relationships between proxy indicators and sensitive attributes, further diminishing their impact. However, these methods require extensive background context to model these dependencies accurately. This often proves to be impractical, consequently narrowing the scope of their application (Salimi et al., 2019).

Sampling and Reweighing Methods: These methods alter the training data distributions either by modifying the number of examples in each sensitive group or by reweighing the group itself. One of the first reweighing approach was introduced by Calders, Kamiran, and Pechenizkiy (2009), in which they proposed reweighing individual instances based on both group membership and labels. The core principle is that by increasing or decreasing the weights of instances within a particular group, one can effectively alter that group's influence during the model's training phase. This was further extended by Li and Liu (2022), who proposed assigning weights to individual instances instead of using broader group-level weights.

Reweighing can also be achieved by altering the data distribution through downsampling (Chakraborty et al., 2020; Roh et al., 2021; Iofinova, Konstantinov, and Lampert, 2022) or upsampling specific groups. While downsampling typically involves removing data points to reduce representation, upsampling can be achieved either by duplicating existing data points (Iosifidis and Ntoutsi, 2018; Roh et al., 2021) or synthesizing new ones (Yan, Kao, and Ferrara, 2020; Singh et al., 2022; Dablain, Krawczyk, and Chawla, 2022). Generative techniques, such as SMOTE (Chawla et al., 2002), MixUp (Zhang et al., 2018), and generative adversarial networks (Goodfellow et al., 2014b), have been utilized for this purpose. Additionally, counterfactual (Sharma et al., 2020) and causal mechanisms (Zhang, Wu, and Wu, 2017) have also been investigated by researchers to create new data instances.

Transformation Methods: These methods seek to transform data by projecting or mapping it into a space with reduced bias while preserving as much relevant information as possible. These methods are especially useful in case of historical bias where the collected data reflects existing prejudice. Most of these techniques frame the challenge as an optimization problem (Zemel et al., 2013; Zehlike, Hacker, and Wiedemann, 2020; Lahoti, Gummadi, and Weikum, 2019), having two primary objectives: to eliminate sensitive information and to minimize the loss of other essential signal. For example, Bolukbasi et al. (2016) proposed a two-step strategy to remove sensitive information, like gender from word representation. First, they identify the direction of the gender subspace using representative words. Then, they remove the gender subspace from all gender-neutral word representations while ensuring these remain equidistant from gender-specific terms. Other researchers have explored using neural style transfer (Quadrianto, Sharmanska, and Thomas, 2018), auto-encoders (Wu et al., 2022; Oh et al., 2022), and dimensionality reduction (Calders, Kamiran, and Pechenizkiy, 2009) to transform data effectively. However, Caton and Haas (2020) enumerate several challenges with these transformations, including (i) the lack of guarantees that the transformed data is bias-free, (ii) the computational expense of optimization in high-dimensional settings, and (iii) the persistent issue of proxy variables—a challenge also present in blinding and causal methods.

Relabelling and Perturbation Methods: These methods represent a subset of transformation approaches where they perturb and/or relabel the dataset to improve the representation of underlying groups. Relabelling techniques such as those presented in Calders, Kamiran, and Pechenizkiy (2009) and Žliobaite, Kamiran, and Calders (2011), are termed "Data Massaging" by Kamiran and Calders (2009). They typically involve identifying candidates using decision boundary or neighborhood information (Thanh, Ruggieri, and Turini, 2011)—and then change their labels. In contrast to label alteration, perturbation techniques (Feldman et al., 2015; Lum and Johndrow, 2016; Li et al., 2022a) adjust non-sensitive attributes to make the representations of different sensitive groups more alike. Caton and Haas (2020) highlights that perturbation techniques are especially common in discrimination-aware data mining and are often employed for privacy preservation.

Generative Methods: These methods aim to enhance the fairness of classifiers by training them on augmented and modified datasets. Most research in this category has proposed GAN-based data augmentation, predominantly focusing on images. For example, GANSan (Aïvodji et al., 2021) creates new instances from the original data, making it more challenging to infer sensitive information. Similarly, FairGan (Xu et al., 2018) and FairGan+ (Xu et al., 2019b) generate entirely new distributions designed to protect sensitive attributes. In the realm of NLP, Qian et al. (2022) introduced PANDA, a seq-2-seq model that perturbs the original data, resulting in fairer language models. However, a notable limitation of these approaches is their modality specificity, which poses challenges in adapting them to broader contexts.

In Chapter 6 of this thesis, we present a pre-processing technique designed to enhance the performance of the classifier for the most disadvantaged sensitive group in intersectional setting. Our approach exploits the hierarchical structure of intersectional groups to generate data for these disadvantaged groups. More specifically, we propose to transform examples from more abstract groups to more specific groups using statistical distance based measures and then augment the original examples with the generated ones. Further, in Chapter 7, drawing inspiration from the domain of differential privacy, we explore the use of randomly perturbing data representations to improve the fairness of the model

3.5.2 In-Processing Methods

Although pre-processing methods are appealing and useful, they often fall short because data is not the only source of bias. As discussed in Section 1.2.3, training procedures can introduce and amplify biases absent in the training data (Wang et al., 2019; Wang and Russakovsky, 2021). Moreover, with modern neural networks, it is common to use pre-trained models, and practitioners may lack access or resources to debias data and retrain. In-processing methods address these issues by directly incorporating fairness into model training, modifying models and augmenting training procedures. Additionally, they can fine tune and debias pre-trained models without huge retraining efforts. In this section, we provide a brief overview of various in-processing techniques used to induce fairness.

Regularization and Constraint Optimization Methods In the context of fairness, these methods often incorporate constraints into the loss function to represent the specific notion of fairness being considered. A primary challenge they confront is that fairness constraints are non-convex and non-differentiable. This often leads to an unstable training process, exhibiting significant variations with minor changes in the dataset (Cotter et al., 2019). To tackle these challenges, fairness procedures typically relax the fairness constraints (Zafar et al., 2017a; Donini et al., 2018; Wu, Zhang, and Wu, 2019) and devise specialized training mechanisms (Cotter, Jiang, and Sridharan, 2019; Agarwal et al., 2018). For instance, Agarwal et al. (2018) proposed to relax the problem by searching for a distribution rather than a single model and then proposed a training procedure based on cost-sensitive learning. Similarly, Cotter, Jiang, and Sridharan (2019) proposed a projected gradient descent-based approach after replacing the fairness error term with the corresponding loss. Several works (Kamiran, Calders, and Pechenizkiy, 2010; Wang, Li, and Wang, 2022) have modified the splitting criteria in decision tree-based models to incorporate fairness. Although these methods were initially introduced for independent group fairness definitions, many works have now extended them to individual (Gillen et al., 2018; Kim, Reingold, and Rothblum, 2018) and intersectional fairness (Padh et al., 2021; Kearns et al., 2018).

Reweighing Methods: Akin to the sampling methods discussed in the previous subsection, these techniques (Ozdayi, Kantarcioglu, and Iyer, 2021; Roh et al., 2020; Iosifidis and Ntoutsi, 2019; Jiang and Nachum, 2020) assign weights to either groups or individual instances. However, in contrast to pre-processing—which typically allocates static weights prior to training—these methods adjust instance weights dynamically throughout the training process. The general idea is to allocate lower weights to the advantaged groups and higher weights to the disadvantaged ones, thereby modulating their influence. For example, Ozdayi, Kantarcioglu, and Iyer (2021) introduce a bilevel optimization approach for fairness, learning weights in the outer loop while optimizing accuracy in the inner loop. Likewise, Iosifidis and Ntoutsi (2019) propose a boosting-based framework where instance weights are determined by both the performance of the current strong classifier and group membership. Instead of modifying instance weights, FairBatch (Roh et al., 2020) suggests adjusting the sampling distributions for each batch based on the model's current fairness level.

Adversarial Learning: These techniques introduce an adversary in the form of a classifier and train both the model and the adversary simultaneously (Dalvi et al., 2004; Ganin et al., 2016). The adversary typically serves as a feedback mechanism to fine-tune the model for fairness. For instance, (Raff and Sylvester, 2018; Beutel et al., 2017; Sadeghi, Yu, and Boddeti, 2019) employ adversarial learning to generate representations such that the adversary cannot predict the sensitive attributes, while preserving sufficient information for the model to predict class label. Several methods, such as (Lahoti et al., 2020; Petrovic et al., 2022), explore the use of adversarial feedback to reweigh instances, akin to the approaches mentioned above. Researchers have also investigated the application of adversarial learning in the context of individual (Yurochkin, Bower, and Sun, 2020) and causal fairness (Xu et al., 2019a).

Multiple Model Methods: These approaches (Oneto et al., 2019; Monteiro and Reynoso-Meza, 2021) typically train multiple models and use a separate model for each sensitive group (Boulitsakis-Logothetis, 2022) or an ensemble-based mechanism for prediction (Chen et al., 2022; Kobayashi and Nakao, 2022). For instance, Dwork et al. (2018) propose learning a separate model for each sensitive group. However, to combat data scarcity, they combine it with transfer learning. Training multiple classification models also enables researchers to explore multiple fairness-accuracy tradeoffs (Blanzeisky and Cunningham, 2022; Roy, Iosifidis, and Ntoutsi, 2021) and as well as fulfill various fairness definitions simultaneously (Mishler and Kennedy, 2022).

> In Chapter 4, we present FairGrad, a simple-to-implement reweighing mechanism which supports multiple fairness definitions. More specifically, we propose a training procedure which iteratively learns group specific weights thereby increasing or decreasing their influence in the final loss function. Additionally, within this chapter, we illustrate the connection between constraint optimization and reweighing techniques. Finally, in Chapter 7 of this thesis, we introduce FEDERATE, a method that augments adversarial learning with differential privacy to learn private representations devoid of sensitive information which also induces fairer downstream model.

3.5.3 Post-Processing Methods

In this subsection, we discuss fairness intervention techniques applied after the model has been trained. A distinct advantage of some of these methods is that they can treat the model and the training data as black boxes, as they typically rely on manipulating the input data (Adler et al., 2018; Li et al., 2022b) (at inference time) and the model output.

Calibration Methods: The primary idea behind these approaches is to change the response of the classifier to ensure fairness with respect to the fairness definitions at hand. For instance, Hardt, Price, and Srebro (2016) propose to randomly flip the output of the classifier to ensure fairness. However, this approach negatively affects accuracy, and the instances for which the decision was flipped are not necessarily positively affected (Kleinberg, Mullainathan, and Raghavan, 2017; Pleiss et al., 2017). Noriega-Campero et al. (2019) tackle these problems by proposing an approach that balances the error parity and calibration. Researchers have also explored the idea in the context of decision trees by relabeling leaf nodes (Kamiran, Calders, and Pechenizkiy, 2010) as well as changing the branching threshold (Kanamori and Arimura, 2021).

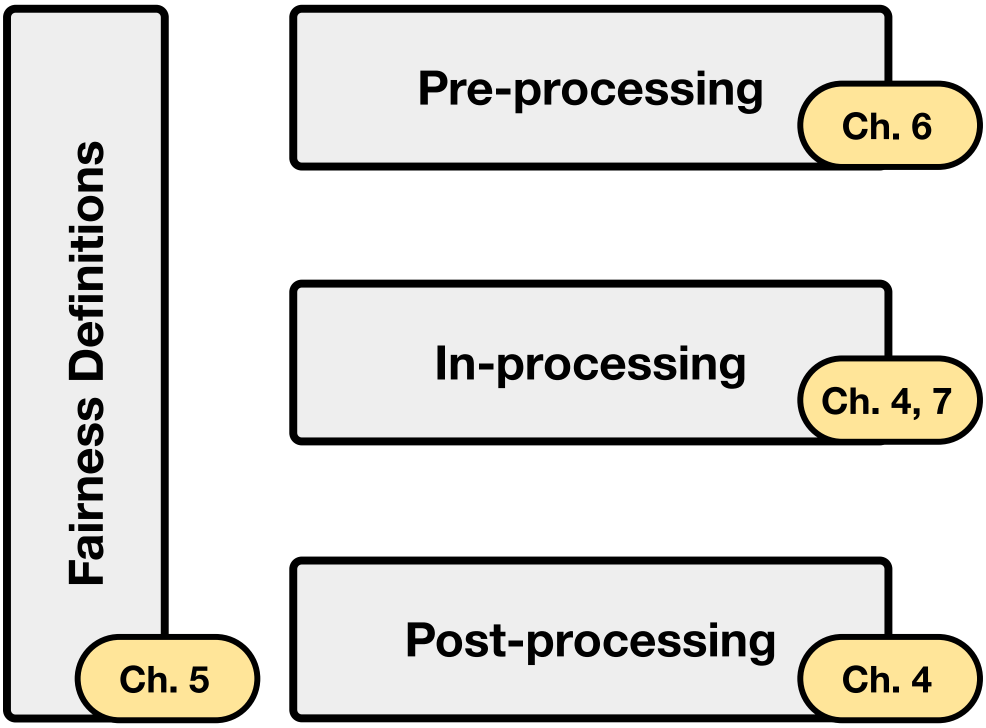


FIGURE 3.3: Outline of the key components of fairness literature and our contributions within these components.

Thresholding Methods: These methods rely on the idea that most unfairness stems from instances close to the classifier's decision boundary (Caton and Haas, 2020). To mitigate this, researchers have proposed distinct thresholds for each sensitive group (Menon and Williamson, 2018) or advocated for post-processing regularization (Fish, Kun, and Lelkes, 2016). In a similar vein, Valera, Singla, and Rodriguez (2018) employ a posterior sampling-based approach, while Iosifidis, Fetahu, and Ntoutsi (2019) introduce an ensemble comprising of multiple classifiers.

3.6 Datasets

As discussed in Section 1.2.1, datasets serve as the cornerstone of all machine learning systems by operationalizing the objective and forming the foundation for subsequent steps. Unsurpisingly, a variety of datasets have been proposed in the field of fairness, differing in (i) size, with instances ranging from 1,000 to 300,000, (ii) domain, spanning finance to social media, and (iii) modality, incorporating both images and text. Table 3.1 provides a compilation of datasets utilized to evaluate our proposed methods and hypotheses.

3.7 Summary

In this section, we explored various fairness definitions designed to capture the potential biases in a machine learning system. We also discussed a range of fairness promoting mechanisms, each optimized for distinct stages of the machine-learning pipeline. Figure 3.3 offers a visual summary of this section and places the contributions of this thesis in this context.

It is important to note that many of these definitions have assumptions built into them. For instance, Demographic Parity implicitly assumes that either the task or the data is biased. In contrast, Equal Opportunity assumes a similar cost of misclassification across different sensitive groups. Similarly, Accuracy Parity posits that different types of error rates do not disproportionately affect various groups. It is also worth noting that many fairness definitions can be at odds with one another. Therefore, practitioners must assess their relevance carefully, as violating these assumptions can result in a misleading sense of fairness. Additionally, they must rigorously evaluate demographic factors as any group not initially included typically would not benefit from fairness measures.

Just as fairness definitions vary, fairness promoting mechanisms each have their strengths and weaknesses (Pessach and Shmueli, 2023). Pre-processing mechanisms might be favored for their model-agnostic nature and ease of integration into machine learning pipelines. However, they are generally difficult to tailor to a specific fairness definition and might lead to reduced model accuracy (Woodworth et al., 2017). On the other hand, post-processing mechanisms are valuable since they can function without needing access to the model or its training data. They are especially relevant when fairness interventions are employed after a model is deployed. Yet, similar to pre-processing, they may result in decreased accuracy and support a limited number of fairness definitions. In-processing approaches are particularly suited for fairness as they can explicitly impose fairness measures. This is also reflected in the literature, where a large majority of the fairness promoting mechanisms belong to this category. However, these techniques are tightly coupled with the model definition and training.

A central challenge inherent to these approaches is navigating the balance between accuracy and fairness. While numerous studies, such as (Menon and Williamson, 2018; Chen, Johansson, and Sontag, 2018), have emphasized this tradeoff by framing it through various analytical perspectives, recent advancements challenge its inevitability. For instance, work by Dutta et al. (2020) has conceptualized theoretical datasets in which methods can simultaneously achieve optimal accuracy and fairness. They argue that methods trained over real world datasets shows this tradeoff because of *noisier mappings for the unprivileged group due to historical differences in opportunity, representation, etc., make their positive and negative labels less separable.*

Efforts have been dedicated to discerning the most effective fairness mechanisms. Yet, results often lack consensus, with different mechanisms excelling in different scenarios (Roth, 2018; Friedler et al., 2019; Jones et al., 2020). Therefore, practitioners must be proactive in experimenting with a myriad of strategies. It is also vital to remain aware of the underlying assumptions of fairness definitions and the ever-present accuracy-fairness tradeoff.

Dataset	Description	Size	Type	Sensitive Axes
UCI Adult	Derived from the 1994 Current Population Survey conducted by the US Census Bureau. Consists of various attributes such as income, location, and age (Kohavi, 1996).	45,222	Tabular	Gender
Folktables	An updated version (Ding et al., 2021) of the Adult dataset derived from newer data released by the US Census Bureau.	1,664,500	Tabular	Gender
CelebA	Comprises images of human faces alongside 40 binary attributes such as gender, hair color, and presence of eyeglasses (Liu et al., 2015).	202,599	Images + Tabular	Can vary: any of the 40 attributes
Dutch	Derived from the Dutch Census, it (Žliobaite, Kamiran, and Calders, 2011) consists of 12 attributes such as income and location, similar to the Adult dataset.	60,420	Tabular	Gender
Compas	Includes 53 features and is aimed at predicting recidivism (Larson et al., 2016).	6,172	Tabular	Race
Crime	Consists of 128 features with the aim of predicting violent crimes in the community (Redmond and Baveja, 2002).	1,994	Tabular	Race
German Credit	Comprises 20 features with the objective of predicting a person's creditworthiness (Dua, Graff, et al., 2017).	1,000	Tabular	Age
Twitter Sentiment	Consists of tweets annotated with sentiment labels and race (Blodgett, Green, and O'Connor, 2016).	200k	Text	Race
Bias in Bios	Comprises biographies annotated with gender and occupation labels (De-Arteaga et al., 2019).	393,424	Text	Gender
Twitter Hate Speech	Derived from a multilingual Twitter hate speech corpus (Huang et al., 2020), consisting of tweets annotated with various demographic attributes of the author.	8,502	Text	Age, Race, Gender, Country
Numeracy	Consists of free-text responses alongside numerical scores, reflecting the individual's numerical comprehension capability (Abbasi et al., 2021).	1,000	Text	Gender, Race, Age, Income

TABLE 3.1: Summary of the datasets used in this thesis.

Chapter 4

FairGrad: Fairness Aware Gradient Descent

Abstract

In this chapter, we introduce FairGrad, an in-processing approach to enforcing fairness in classification. FairGrad is based on a re-weighting scheme that iteratively learns group-specific weights based on whether they are advantaged or not. It is easy to implement, accommodates various standard independent group fairness definitions, and has minimal overhead. Furthermore, through our experiments, we show that it is competitive with standard baselines over various datasets, including ones used in natural language processing and computer vision.

This chapter is based on the article - Maheshwari, Gaurav, and Michaël Perrot. "FairGrad: Fairness Aware Gradient Descent." Transactions on Machine Learning Research, 2023. It is also available as a PyPI package at - `https://pypi.org/project/fairgrad`

4.1 Introduction

In the Introduction Chapter, we outlined a simplified machine learning pipeline (see Figure 4.1) consisting of four main components: (i) data collection and preparation, (ii) evaluation, (iii) training, and (iv) deployment. In our first contribution we focus on the model training aspect of the pipeline, specifically, on in-processing fairness promoting methods (see Section 3.5.2). A key challenge of employing these methods is that they present significant adaptation challenges when integrating them into existing training pipeline. They often require specialized training procedures or modifications to the original model. For instance, constraint optimization methods, discussed in Section 3.5.2, typically require specialized training mechanisms such as casting the problem as cost-sensitive learning in Agarwal et al. (2018) or altering the underlying solver in Cotter et al. (2019). Adversarial methods, in contrast, requires the use of nonlinear models and changes to it in the form of adversarial branch.

Moreover, in-processing approaches are also limited in the range of problems to which they can be applied. For example, the work of Agarwal et al. (2018) can only be applied in a binary classification setting, while the work of Ozdayi, Kantarcioglu, and Iyer (2021) is limited to two sensitive groups. Furthermore, they may come

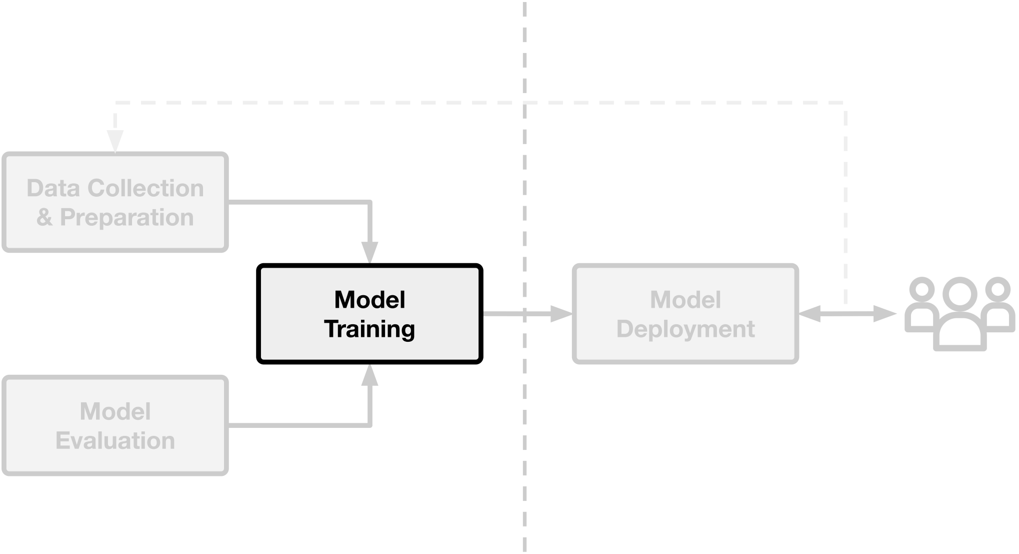


FIGURE 4.1: Fairgrad targets the model training aspect of the machine learning pipeline.

with several hyper-parameters that need to be carefully tuned to obtain fair models. For instance, the scaling parameter in adversarial learning (Raff and Sylvester, 2018; Li, Baldwin, and Cohn, 2018) or the number of iterations in inner optimization for bi-level optimization based mechanisms (Ozdayi, Kantarcioglu, and Iyer, 2021). The complexity of the existing methods might hinder their deployment in practical settings. Hence, there is a need for simpler methods that are straightforward to integrate into existing systems.

Contributions: To circumvent above challenges we present FairGrad, a general purpose approach to enforce fairness in empirical risk minimization solved using gradient descent. We propose to dynamically update the influence of the examples after each gradient descent update to precisely reflect the fairness level of the models obtained at each iteration and guide the optimization process in a relevant direction. Hence, the underlying idea is to use lower weights for examples from advantaged groups than those from disadvantaged groups. Our method is inspired by recent re-weighting approaches, as discussed in Section 3.5.2, that also propose to change the importance of each group while learning a model (Iosifidis and Ntoutsi, 2019; Krasanakis et al., 2018; Jiang and Nachum, 2020; Roh et al., 2020; Ozdayi, Kantarcioglu, and Iyer, 2021). Interestingly, we also find that FairGrad can be seen as solving a kind of constrained optimization problem. In Section 4.3, we expand upon this link and show how FairGrad can be seen as a solution that connects these two kinds of methods.

```python
# The library is available at https://pypi.org/project/fairgrad.
from fairgrad.torch import CrossEntropyLoss

# Same as PyTorch's loss with some additional meta data.
# A fairness rate of 0.01 is a good rule of thumb for standardized data.
criterion = CrossEntropyLoss(y_train, s_train, fairness_measure, fairness_rate=0.01)

# The dataloader and model are defined and used in the standard way.
for x, y, s in data_loader:
  optimizer.zero_grad()
  loss = criterion(model(x), y, s)
  loss.backward()
  optimizer.step()
```

LISTING 4.1: A standard training loop where the PyTorch's loss is replaced by FairGrad's loss.

A key advantage of FairGrad is that it is straightforward to incorporate into standard gradient based solvers that support examples re-weighting like Stochastic Gradient Descent. Hence, we developed a Python library where we augmented standard PyTorch losses to accommodate our approach. From a practitioner point of view, it means that using FairGrad is as simple as replacing their existing loss from PyTorch with our custom loss and passing along some meta data, while the rest of the training loop remains identical. This is illustrated in Figure 4.1. It is interesting to note that besides the usual optimization hyper-parameters (learning rates, batch size, ...), FairGrad only brings one extra hyper-parameter, the fairness rate. Moreover, FairGrad incurs minimal computational overhead during training as it relies on objects that are already computed for standard gradient descent, namely the predictions on the current batch and the loss incurred by the model for each example. In particular, the overhead is independent of the number of parameters of the model. Furthermore, as many in-processing approaches in fairness (Cotter, Jiang, and Sridharan, 2019; Roh et al., 2020), FairGrad does not introduce any overhead at test time.

Overall, FairGrad is a lightweight solution that is compatible with various group fairness notions, including exact and approximate fairness, can handle both multiple sensitive groups and multiclass problems, and can fine tune existing unfair models. Through extensive experiments, we also show that, in addition to its versatility, FairGrad is competitive with several standard baselines in fairness on both standard datasets as well as complex NLP and CV tasks.

4.2 Problem Setting, Notations

In the remainder of this chapter, we assume that we have access to a feature space $\mathcal{X}$, a finite discrete label space $\mathcal{Y}$, and a set $\mathcal{G}$ of all sensitive groups (see Section 3.2.1). We further assume that there exists a distribution $\mathcal{D} \in \mathcal{D}_{\mathcal{Z}}$ where $\mathcal{D}_{\mathcal{Z}}$ is the set of all distributions over $\mathcal{Z} = \mathcal{X} \times \mathcal{Y} \times \mathcal{G}$. Our goal is then to learn an accurate model $h_\theta \in \mathcal{H}$, with learnable parameters $\theta \in \mathbb{R}^d$, such that $h_\theta : \mathcal{X} \to \mathcal{Y}$ is fair with respect to a given fairness definition that depends on the sensitive groups. In Section 4.2.1, we formally define the family of fairness measures that are compatible with our approach and provide several examples of popular notions encompassed by our fairness definition.

As usual in machine learning, we will assume that $\mathcal{D}$ is unknown and that we only get to observe a finite dataset $\mathcal{T} = \{(x_i, y_i, \mathbf{g}_i)\}_{i=1}^n$ of n examples drawn i.i.d. from $\mathcal{D}$. Let $\mathbb{P}(E(X,Y,G))$ represent the probability that an event E happens with respect to $(X,Y,G) \sim \mathcal{D}$ while $\widehat{\mathbb{P}}(E(x,y,g)) = \frac{1}{n}\sum_{i=1}^n \mathbb{I}_{E(x_i,y_i,\mathbf{g}_i)}$ is an empirical estimate with respect to $\mathcal{T}$ where $\mathbb{I}_P$ is the indicator function which is 1 when the property P is verified and 0 otherwise. In the remainder of this chapter, all our derivations will be considered in the finite sample setting and we will assume that what was measured on our finite sample is sufficiently close to what would be obtained if one had access to the overall distribution. This seems reasonable in light of the previous work on generalization in standard machine learning (Shalev-Shwartz and Ben-David, 2014) and the recent work of Woodworth et al. (2017) or Mangold et al. (2022) which show that the kind of fairness measures we consider in this chapter tend to generalize well when the hypothesis space is not too complex, as measured respectively by the VC or the Natarajan Dimension (Shalev-Shwartz and Ben-David, 2014). Since these generalization results only rely on a capacity measure of the hypothesis space and are otherwise algorithm agnostic, they are applicable to the models returned by FairGrad when they have finite VC or Natarajan dimensions. This is for example the case for linear models.

4.2.1 Fairness Definition

In this chapter, we assume that the data may be partitioned into K disjoint groups denoted $\mathcal{T}_1, \ldots, \mathcal{T}_k, \ldots, \mathcal{T}_K$ such that $\bigcup_{k=1}^K \mathcal{T}_k = \mathcal{T}$ and $\bigcap_{k=1}^K \mathcal{T}_k = \emptyset$. These groups highly depend on the fairness notion under consideration. They might correspond to the usual intersectional sensitive groups (defined in Section 3.2.1) as is the case for Accuracy Parity (see Example 1), or might be subgroups of the intersectional sensitive groups, as in Equalized Odds where the subgroups are defined with respect to the true labels (see Example 2 in Appendix A.1). For each group, we assume that we have access to a function $\widehat{F}_k : \mathcal{D}^n \times \mathcal{H} \to \mathbb{R}$ such that $\widehat{F}_k > 0$ when the group k is advantaged by the given classifier and $\widehat{F}_k < 0$ when the group k is disadvantaged. Furthermore, we assume that the magnitude of $\widehat{F}_k$ represents the degree to which the group is (dis)advantaged. Finally, we assume that each $\widehat{F}_k$ can be rewritten as:

$$\widehat{F}_k(\mathcal{T}, h_\theta) = C_k^0 + \sum_{k'=1}^{K} C_k^{k'} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_{k'}\right) \tag{4.1}$$

where the constants C are group specific and independent of h_θ. The probabilities $\widehat{\mathbb{P}}(h_\theta(x) \neq y|\mathcal{T}_{k'})$ represent the error rates of h_θ over each group $\mathcal{T}_{k'}$ with a slight abuse of notation. Below, we show that Accuracy Parity (Zafar et al., 2017a) respects this definition. In Appendix A.1, we show that Equality of Opportunity (Hardt, Price, and Srebro, 2016), Equalized Odds (Hardt, Price, and Srebro, 2016), and Demographic Parity (Calders, Kamiran, and Pechenizkiy, 2009) also respect this definition. It means that using this generic formulation allows us to simultaneously reason about multiple fairness notions.

Example 1 (Accuracy Parity (AP) (Zafar et al., 2017a)). A model h_θ is fair for Accuracy Parity when the probability of being correct is independent of the sensitive group, that is, $\forall \mathbf{g} \in \mathcal{G}$

$$\widehat{\mathbb{P}}\left(h_\theta(x) = y \,|\, \mathbf{g}\right) = \widehat{\mathbb{P}}\left(h_\theta(x) = y\right).$$

It means that we need to partition the space corresponding to intersectional groups, $\forall \mathbf{g} \in \mathcal{G}$, we define $\widehat{F}_{(r)}$ as the fairness level of group $\mathbf{g}$

$$\begin{aligned}\widehat{F}_{(r)}(\mathcal{T}, h_\theta) &= \widehat{\mathbb{P}}\left(h_\theta(x) \neq y\right) - \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}\right) \\ &= (\widehat{\mathbb{P}}\left(\mathbf{g}\right) - 1)\widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}\right) + \sum_{(\mathbf{g}') \neq (\mathbf{g})} \widehat{\mathbb{P}}\left(\mathbf{g}'\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}'\right)\end{aligned}$$

where the law of total probability was used to obtain the last equality. Thus, Accuracy Parity satisfies all our assumptions with $C_{\mathbf{g}}^{\mathbf{g}} = \widehat{\mathbb{P}}\left(\mathbf{g}\right) - 1$, $C_{\mathbf{g}}^{\mathbf{g}'} = \widehat{\mathbb{P}}\left(\mathbf{g}'\right)$ with $\mathbf{g}' \neq \mathbf{g}$, and $C_{\mathbf{g}}^0 = 0$.

4.3 General Formulation

In Section 3.5.2, we list various in-processing approaches. Primary amongst them are methods relying on formulating the problem as either a constrained optimization, which is later relaxed to an unconstrained case or using re-weighting techniques where examples are dynamically re-weighed based on the fairness levels of the model (Caton and Haas, 2020). In this section, we provide a general formulation of these mechanisms and list the similarities and differences between FairGrad and the corresponding approaches. Additionally, we will also demonstrate how FairGrad can be seen as a solution that connects these two streams of work. In the next subsection, we list very closely related works.

Constrained Optimization The problem of fair machine learning can be seen as the following constrained optimization problem (Cotter, Jiang, and Sridharan, 2019; Agarwal et al., 2018):

$$\begin{aligned}&\arg\min_{h_\theta \in \mathcal{H}} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y\right) \\ &\text{s.t. } \forall k \in [K], \widehat{F}_k(\mathcal{T}, h_\theta) = 0. \end{aligned} \tag{4.2}$$

This problem can then be reformulated as an unconstrained optimization problem using Lagrange multipliers. More specifically, with multipliers denoted by $\lambda_1, \ldots, \lambda_K$, the unconstrained objective that should be minimized for $h_\theta \in \mathcal{H}$ and maximized for $\lambda_1, \ldots, \lambda_K \in \mathbb{R}$ is:

$$\mathcal{L}\left(h_\theta, \lambda_1, \ldots, \lambda_K\right) = \widehat{\mathbb{P}}\left(h_\theta(x) \neq y\right) + \sum_{k=1}^{K} \lambda_k \widehat{F}_k(\mathcal{T}, h_\theta)\,. \tag{4.3}$$

Several strategies may then be employed to find a saddle point for the aforementioned objective[1]. Agarwal et al. (2018) first relax the problem by searching for a distribution over the models rather than a single optimal hypothesis. Then, they alternate between using an exponentiated gradient step to find $\lambda_1, \ldots, \lambda_K \in \mathbb{R}$ and a procedure based on cost sensitive learning to find the next h_θ to add to their distribution. Similarly, Cotter, Jiang, and Sridharan (2019) also search for a distribution over the models using an alternating approach based on Lagrange multipliers where they relax objective (4.3) by replacing the error rate with a loss term. To update the λ

[1] These min-max formulations are not new in the literature and was already used in the 1940's (Wald, 1945). More recently, Madry et al. (2018) employed the formulation to make deep neural networks more robust against adversarial attacks. Similarly, Ben-Tal et al. (2012) modeled uncertainty in input via this formulation.

multipliers, unlike Agarwal et al. (2018), they use projected gradient descent based on the original fairness terms. To search the next h_θ to add to their distribution of models they use a projected gradient descent update over a relaxed overall objective function where the fairness measures are replaced with smooth upper bounds.

In this work, we also use an alternating approach based on objective (4.3). However, we look for a single model rather than a distribution of models. To this end, at each iteration, we update λ using a projected gradient descent step similar to Cotter, Jiang, and Sridharan (2019), that is using the original fairness measures. To solve for h_θ, contrary to Cotter, Jiang, and Sridharan (2019), we first show that Objective (4.3), with fixed λ, may be rewritten as a weighted sum of group-wise error rates. This is similar in spirit to the cost-sensitive learning method of Agarwal et al. (2018) but can be applied beyond binary classification. We then follow Cotter, Jiang, and Sridharan (2019) and replace in our new objective the error rate terms with a loss function, albeit not necessarily an upper bound, to obtain meaningful gradient directions.

Re-weighting Another way to learn fair models is to use a re-weighting approach (see Section 3.5.2) where each example x is associated with a weight $w_x \in \mathbb{R}$ so that minimizing the following objective for h_θ outputs a fair model:

$$\mathcal{W}\left(h_\theta\right) = \widehat{\mathbb{E}}\left(w_x \mathbb{I}_{\{h_\theta(x) \neq y\}}\right).$$

The underlying idea for the methods which posit the problem as above is to propose a cost function that outputs weights for each example. Recall that on the one hand, the weights can be determined in a pre-processing step (Kamiran and Calders, 2012), based on the statistics of the data under consideration. On the other hand, the weights may evolve with h_θ, that is they are dynamically updated each time the model changes during the training process (Roh et al., 2020).

In this work, to find h_θ, we also use a dynamic re-weighting approach where the weights change at each iteration. To choose the weights, we initially give the same importance to each example. Then, we increase the weights of disadvantaged examples and decrease the weights of advantaged examples proportionally to the fairness level of the current model for their group. An important feature of our approach, unlike other re-weighting approaches, is that we do not constrain ourselves to positive weights but rather allow the use of negative weights. Indeed, we show in Lemma 1 that the latter are sometimes necessary to learn fair models.

To summarize, we first frame the task as a constrained optimization problem, similar to Cotter, Jiang, and Sridharan (2019) and Agarwal et al. (2018). We then propose an alternating approach, where we update λ at each iteration using a projected gradient descent step similar to Cotter, Jiang, and Sridharan (2019). However, in order to learn the model h_θ, we show that Objective (4.3), with fixed λ, can be rewritten as a weighted sum of group-wise error rates. This step can be interpreted as an instance of re-weighting where the weights change at each iteration. Thus our method can be seen as a connection between constrained optimization and re-weighting.

4.4 FairGrad

In the above section, we argued that FairGrad is connected to both constrained optimization and re-weighting approaches. In this section, we provide details on our method and we present it starting from the constrained optimization point of view as we believe it makes it easier to understand how the weights are selected and updated. We begin by discussing FairGrad for exact fairness and then extend it to the approximate fairness also referred as ϵ-fairness.

4.4.1 FairGrad for Exact Fairness

To solve the problem described in equation 4.3, we propose to use an alternating approach where the hypothesis and the multipliers are updated one after the other[2]. We begin by describing our method to update the multipliers and then the model.

Updating the Multipliers. To update $\lambda_1, \ldots, \lambda_K$, we will use a standard gradient ascent procedure. Hence, given that the gradient of Problem (4.3) is

$$\nabla_{\lambda_1,\ldots,\lambda_K} \mathcal{L}\left(h_\theta, \lambda_1, \ldots, \lambda_K\right) = \begin{pmatrix} \widehat{F}_1(\mathcal{T}, h_\theta) \\ \vdots \\ \widehat{F}_K(\mathcal{T}, h_\theta) \end{pmatrix}$$

we have the following update rule $\forall k \in [K]$:

$$\lambda_k^{T+1} = \lambda_k^T + \eta_\lambda \widehat{F}_k\left(\mathcal{T}, h_\theta^T\right)$$

where η_λ is a rate that controls the importance of each update. In the experiments, we use a constant rate of 0.01 as our initial tests showed that it is a good rule of thumb when the data is properly standardized.

Updating the Model. To update the parameters $\theta \in \mathbb{R}^D$ of the model h_θ, we use a standard gradient descent. However, first, we notice that, given our fairness definition, Equation (4.3) can be written as

$$\mathcal{L}\left(h_\theta, \lambda_1, \ldots, \lambda_K\right) = \sum_{k=1}^{K} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) \left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'}\right] + \sum_{k=1}^{K} \lambda_k C_k^0. \quad (4.4)$$

where $\sum_{k=1}^{K} \lambda_k C_k^0$ is independent of h_θ by definition. Hence, at iteration t, the update rule becomes

$$\theta^{T+1} = \theta^T - \eta_\theta \sum_{k=1}^{K} \left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'}\right] \nabla_\theta \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right)$$

where η_θ is the usual learning rate that controls the importance of each parameter update. Here, we obtain our group specific weights $\forall_k, w_k = \left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'}\right]$, that depend on the current fairness level of the model through $\lambda_1, \ldots, \lambda_K$, the relative size of each group through $\widehat{\mathbb{P}}\left(\mathcal{T}_k\right)$, and the fairness notion under consideration through the constants C. The exact values of these constants are given in Section 4.2.1 and Appendix A.1 for various group fairness notions. Overall, they are such that, at each iteration, the weights of the advantaged groups are reduced and the weights of the disadvantaged groups are increased.

[2] It is worth noting that, here, we do not have formal duality guarantees and that the problem is not even guaranteed to have a fair solution. Nevertheless, the approach seems to work well in practice as can be seen in the experiments.

Algorithm 3 FairGrad for Exact Fairness

Input: Groups $\mathcal{T}_1, \ldots, \mathcal{T}_K$, Functions $\widehat{F}_1, \ldots, \widehat{F}_K$, Function class $\mathcal{H}$ of models h_θ with parameters $\theta \in \mathbb{R}^D$, Learning rates $\eta_\lambda, \eta_\theta$, and Iterator *iter* that returns batches of examples.

Output: A fair model h_θ^*.

```
Initialize the group specific weights and the model.
for B in iter do
    Compute the predictions of the current model on the batch B.
    Compute the group-wise losses using the predictions.
    Compute the current fairness level using the predictions and update the group-wise weights.
    Compute the overall weighted loss using the group-wise weights.
    Compute the gradients based on the loss and update the model.
end for
return the trained model $h_\theta^*$
```

The main limitation of the above update rule is that one needs to compute the gradient of $0-1$-losses since $\nabla_\theta \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) = \frac{1}{n_k} \sum_{(x,y)\in\mathcal{T}_k} \nabla_\theta \mathbb{I}_{\{h_\theta(x)\neq y\}}$. Unfortunately, this usually does not provide meaningful optimization directions. To address this issue, we follow the usual trend in machine learning and replace the $0-1$-loss with one of its continuous and differentiable surrogates that provides meaningful gradients. For instance, in our experiments, we use the cross entropy loss.

4.4.2 Computational Overhead of FairGrad.

We summarize our approach in Algorithm 3, where we have used italic font to highlight the steps inherent to FairGrad that do not appear in classic gradient descent. We consider batch gradient descent rather than full gradient descent as it is a popular scheme. We empirically investigate the impact of the batch size in Section 4.6.7. The main difference is Step 5, that is the computation of the group-wise fairness levels. However, these can be cheaply obtained from the predictions of $h_\theta^{(t)}$ on the current batch which are always available since they are also needed to compute the gradient. Hence, the computational overhead of FairGrad is very limited.

4.4.3 Importance of Negative Weights.

A key property of FairGrad is that we allow the use of negative weights, that is $\left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'}\right]$ may become negative, while existing methods (Roh et al., 2020; Iosifidis and Ntoutsi, 2019; Jiang and Nachum, 2020) restrict themselves to positive weights. In this section, we show that these negative weights are important as they are sometimes necessary to learn fair models. Hence, in the next lemma, we provide sufficient conditions so that negative weights are mandatory if one wants to enforce Accuracy Parity.

Lemma 1 (Negative weights are necessary.)**.** Let the fairness notion be Accuracy Parity (Example 1). Let h_θ^* be the most accurate and fair model. Then using negative weights is necessary as long as

$$\min_{\substack{h_\theta\in\mathcal{H} \\ h_\theta \text{unfair}}} \max_{\mathcal{T}_k} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) < \widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y\right).$$

Proof. The proof is provided in Appendix A.2. □

The previous condition can sometimes be verified in practice. As a motivating example, assume a binary setting with only two sensitive groups $\mathcal{T}_1$ and $\mathcal{T}_{-1}$. Let h_θ^{-1} be the model minimizing $\widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_{-1}\right)$ and assume that $\widehat{\mathbb{P}}\left(h_\theta^{-1}(x) \neq y\right) < \widehat{\mathbb{P}}\left(h_\theta^{-1}(x) \neq y | \mathcal{T}_{-1}\right)$, that is group $\mathcal{T}_{-1}$ is disadvantaged for accuracy parity. Given h_θ^* the most accurate and fair model, we have

$$\min_{\substack{h_\theta \in \mathcal{H} \\ h_\theta \text{unfair}}} \max_{\mathcal{T}_k} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) = \widehat{\mathbb{P}}\left(h_\theta^{-1}(x) \neq y | \mathcal{T}_{-1}\right) < \widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y\right)$$

as otherwise we would have a contradiction since the fair model would also be the most accurate model for group $\mathcal{T}_{-1}$ since $\widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y\right) = \widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y | \mathcal{T}_{-1}\right)$ by definition of Accuracy Parity. In other words, a dataset where the most accurate model for a given group still disadvantages it requires negative weights. This might be connected to the notion of "leveling down" (Zietlow et al., 2022; Mittelstadt, Wachter, and Russell, 2023), where fairness can only be achieved by harming all the groups or bringing advantaged groups closer to disadvantaged groups by harming them. It is generally an artifact of strictly egalitarian fairness measures. We investigate this leveling down phenomena in more depth in the next chapter.

4.4.4 FairGrad for ϵ-fairness

In the previous section, we considered exact fairness and we showed that this could be achieved by using a re-weighting approach. Here, we extend this procedure to ϵ-fairness where the fairness constraints are relaxed and a controlled amount of violations is allowed. Usually, ϵ is a user defined parameter but it can also be set by the law, as it is the case with the 80% rule in the US (Biddle, 2006). The main difference with exact fairness is that each equality constraint in Problem (4.2) is replaced with two inequalities of the form

$$\begin{aligned} &\forall k \in [K], \widehat{F}_k(\mathcal{T}, h_\theta) \leq \epsilon \\ &\forall k \in [K], \widehat{F}_k(\mathcal{T}, h_\theta) \geq -\epsilon. \end{aligned}$$

The main consequence is that we need to maintain twice as many Lagrange multipliers and that the group-wise weights are slightly different. Since the two procedures are similar, we omit the details here but provide them in Appendix A.3 for the sake of completeness.

4.5 Related Work

For an extensive discussion of fairness interventions approaches, refer to Section 3.5. Here, we focus on recent works that are more closely related to our approach.

BiFair (Ozdayi, Kantarcioglu, and Iyer, 2021). This paper proposes a bilevel optimization scheme for fairness. The idea is to use an outer optimization scheme that learns weights for each example so that the trade-off between fairness and accuracy is as favorable as possible while an inner optimization scheme learns a model that is as accurate as possible. One limitation of this approach is that it does not directly optimize the fairness level of the model but rather a relaxation that does not provide any guarantees on the goodness of the learned predictor. Furthermore, it is limited to binary classification with a binary sensitive attribute. In this chapter, we also learn weights for the examples in an iterative way. However, we use a different update rule. Furthermore, we focus on exact fairness definitions rather than relaxations and our objective is to learn accurate models with given levels of fairness rather than a trade-off between the two. Finally, our approach is not limited to the binary setting.

FairBatch (Roh et al., 2020). This paper proposes a batch gradient descent approach to learn fair models. More precisely, the idea is to draw a batch of examples from a skewed distribution that favors the disadvantaged groups by oversampling them. In FairGrad, we propose to use a re-weighting approach which could also be interpreted as altering the distribution of the examples based on th eir fairness level if all the weights were positive. However, we allow the use of negative weights, and we prove that they are sometimes necessary to achieve fairness. Furthermore, we employ a different update rule for the weights.

AdaFair (Iosifidis and Ntoutsi, 2019). This paper proposes a boosting based framework to learn fair models. The underlying idea is to modify the weights of the examples depending on both the performances of the current strong classifier and the group memberships. Hence, examples that belong to the disadvantaged group and are incorrectly classified receive higher weights than the examples that belong to the advantaged group and are correctly classified. In this chapter, we use a similar high level idea but we use different weights that do not depend on the accuracy of the model but solely on its fairness. Furthermore, rather than a boosting based approach, we consider problems that can be solved using gradient descent. Finally, while AdaFair only focuses on Equalized Odds, we show that our approach works with several fairness notions.

Identifying and Correcting Label Bias in Machine Learning (Jiang and Nachum, 2020). This paper tackles the fairness problem by assuming that the observed labels are biased compared to the true labels. The goal is then to learn a model with respect to the true labels using only the observed labels. To this end, it proposes to use an iterative re-weighting procedure where positive example-wise weights and the model are alternatively updated. In FairGrad, we also propose a re-weighting approach. However, we use different weighing mechanism which is not restricted to positive weights. Furthermore, our approach is not limited to binary labels and can handle multiclass problems.

4.6 Experiments

In this section, we present several experiments that demonstrate the competitiveness of FairGrad as a procedure to learn fair models for classification. We begin by presenting results over standard fairness datasets and a Natural language Processing dataset in Section 4.6.4. We then study the behaviour of the ϵ-fairness variant of FairGrad in Section 4.6.5. Next, we showcase the fine-tuning ability of FairGrad on a Computer Vision dataset in Section 4.6.6. Finally, we investigate the impact of batch size on the learned model in Section 4.6.7 and present results related to the computational overhead incurred by FairGrad in Section 4.6.8.

4.6.1 Datasets

In the main chapter, we consider 4 different datasets and postpone the results on another 6 datasets to Appendix A.4.3 as they follow similar trends. We also postpone the detailed descriptions of these datasets as well as the pre-processing steps to Appendix A.4.2.

We consider commonly used fairness datasets, namely **Adult Income** (Kohavi, 1996) and **CelebA** (Liu et al., 2015). Both are binary classification datasets with binary sensitive attributes (gender). We also consider a variant of the Adult Income dataset where we add a second binary sensitive attribute (race) to obtain a dataset with 4 disjoint sensitive groups. For both datasets, we use 20% of the data as a test set and the remaining 80% as a train set. We further divide the train set into two and keep 25% of the training examples as a validation set. For each repetition, we randomly shuffle the data before splitting it, and thus we have unique splits for each random seed. Lastly, we standardize each features independently by subtracting the mean and scaling to unit variance which were estimated on the training set.

To showcase the wide applicability of FairGrad, we consider the **Twitter Sentiment**[3] (Blodgett, Green, and O'Connor, 2016) dataset from the Natural Language Processing community. It consists of $200k$ tweets with binary sensitive attribute (race) and binary sentiment score. We employ the same setup, splits, and the pre-processing as proposed by Han, Baldwin, and Cohn (2021) and Elazar and Goldberg (2018) and create bias in the dataset by changing the proportion of each subgroup (race-sentiment) in the training set. Following the footsteps of Elazar and Goldberg (2018) we encode the tweets using the DeepMoji (Felbo et al., 2017) encoder with no fine-tuning, which has been pre-trained over millions of tweets to predict their emoji, thereby predicting the sentiment. We also employ the **UTKFace** dataset[4] (Zhang, Song, and Qi, 2017) from the Computer Vision community. It consists of $23,708$ images tagged with race, age, and gender with pre-defined splits.

4.6.2 Performance Measures

For fairness, we consider the four measures introduced in Section 4.2.1 and Appendix A.1, namely Equalized Odds (EOdds), Equality of Opportunity (EOpp), Accuracy Parity (AP), and Demographic Parity (DP). For each specific fairness notion, we report the average absolute fairness level of the different groups over the test set, that is $\frac{1}{K}\sum_{k=1}^{K}\left|\widehat{F}_k(\mathcal{T}, h_\theta)\right|$ (lower is better). To assess the utility of the learned models, we use their accuracy levels over the test set, that is $\frac{1}{n}\sum_{i=1}^{n}\mathbb{I}_{h_\theta(x_i)=y_i}$ (higher is better). All the results reported are averaged over 5 independent runs and standard deviations are provided. Note that, in the main chapter, we graphically report a subset of the results over the aforementioned datasets. We provide detailed results in Appendix A.4.3, including the missing pictures as well as complete tables with accuracy levels, fairness levels, and fairness level of the most well-off and worst-off groups for all the relevant methods.

4.6.3 Methods

We compare FairGrad to a wide variety of baselines, namely:

[3] http://slanglab.cs.umass.edu/TwitterAAE/

[4] https://susanqq.github.io/UTKFace/

- **Unconstrained**, which is oblivious to any fairness measure and is trained using a standard batch gradient descent method.
- **Adversarial** learning based method where we employ adversarial mechanism (Goodfellow et al., 2014a) using a gradient reversal layer (Ganin and Lempitsky, 2015), similar to GRAD-Pred (Raff and Sylvester, 2018), where an adversary, with an objective to predict the sensitive attribute, is added to the unconstrained model
- Bi-level optimization based method implemented in the form of **BiFair** (Ozdayi, Kantarcioglu, and Iyer, 2021)
- Re-weighting based methods in the form of **FairBatch** (Roh et al., 2020). We also compare against a simpler baseline called **Weighted ERM** where each example is reweighed based on the size of the sensitive group the example belongs to in the beginning. Unlike FairBatch these weights are not updated during training.
- Constrained optimization based method as proposed by Cotter, Jiang, and Sridharan (2019). We refer to this method as **Constraints** in this article.
- **Reduction** implements the exponentiated gradient based fair classification approach as proposed by Agarwal et al. (2018).

In all our experiments, we consider two different hypothesis classes. On the one hand, we use linear models implemented in the form of neural networks with no hidden layers. On the other hand, we use a more complex, non-linear architecture with three fully-connected hidden layers of respective sizes 128, 64, and 32. We use ReLU as our activation function with batch normalization and dropout. In both cases, we optimize the cross-entropy loss.

In several experiments, we only consider subsets of the baselines due to the limitations of the methods. For instance, BiFair was designed to handle binary labels and binary sensitive attributes and thus is not considered for the datasets with more than two sensitive groups or two labels. Furthermore, we implemented it using the authors code that is freely available online but does not include AP as a fairness measure, thus we do not report results related to this measure for BiFair. Similarly, we also implemented FairBatch from the authors code which does not support AP as a fairness measure, thus we also exclude it from the comparison for this measure. For Constraints, we based our implementation on the publicly available authors library but were only able to reliably handle linear models and thus we do not consider this baseline for non-linear models. Finally, for Adversarial, we used our custom made implementation. However, it is only applicable when learning non-linear models since it requires at least one hidden layer to propagate its reversed gradient.

Apart from the common hyper-parameters such as dropout, several baselines come with their own set of hyper-parameters. For instance, BiFair has the *inner loop length*, which controls the number of iterations in its inner loop, while Adversarial has the *scaling*, which re-weights the adversarial branch loss and the task loss. We provide details of common and approach specific hyper-parameters in Appendix A.4.1.

With several hyper-parameters for each approach, selecting the best combination is often crucial to avoid undesirable behaviors such as over-fitting (Maheshwari et al., 2022). In this chapter, we opt for the following procedure. First, for each method, we consider all the X possible hyper-parameter combinations and we run the training procedure for 50 epochs for each combination. Then, we retain all the models returned by the last 5 epochs, that is, for a given method, we have $5X$ models and the goal is to select the best one among them. Since we have access to two performance measures, we can select either the most accurate model, the most fair, or a trade-off between the two depending on the end goal. Here, we chose to focus on the third option and select the model with the lowest fairness score between certain accuracy intervals. More specifically, let α^* be the highest validation accuracy among the $5X$ models. We choose the model with the lowest validation fairness score amongst all models with a validation accuracy in the interval $[\alpha^* - k, \alpha^*]$. In this work, we fix k to 0.03. We provide more details around this selection strategy in Chapter 7.

4.6.4 Results for Exact Fairness

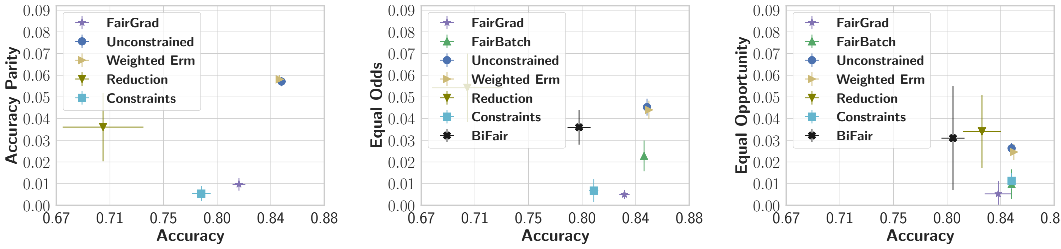
We report the results over the Adult dataset using a linear model, the Adult dataset with multiple groups with a non-linear model, and the Twitter sentiment dataset using both linear and nonlinear models in Figures 4.2, 4.3, and 4.4 respectively. In these figures, the best methods are closer to the bottom right corner. If a method is closer to the bottom left corner, it has good fairness but reduced accuracy. Similarly, if the method is closer to the top right corner it has good accuracy but poor fairness.

The main take-away from these experiments is that there is no fairness enforcing method that is consistently better than the others in terms of both accuracy and fairness. All of them have strengths, that is datasets and fairness measures where they obtain good results, and weaknesses, that is datasets and fairness measures for which they are sub-optimal. FairBatch induces better accuracy than the other approaches over Adult with linear model and EOdds and only pays a small price in fairness. However, it is significantly worse in terms of fairness over the Adult Multigroup dataset with a non-linear model. Similarly, BiFair is sub-optimal on Adult with EOpp, while being comparable to the other approaches on the Twitter Sentiment dataset. We observed similar trends on the other datasets, available in Appendix A.4.3, with different methods coming out on top for different datasets and fairness measures.

Interestingly, FairGrad generally outperforms other approaches in terms of fairness, albeit with a slight loss in accuracy. These observations are even more amplified in the Accuracy Parity and Equalized Odds settings. Moreover, it is generally more robust and tends to show a lower standard deviation in accuracy and fairness than the other approaches. Even in terms of accuracy, the largest difference is over the Crime dataset, where the difference between FairGrad and Unconstrained is 0.04. However, in most cases, the difference is within 0.02. In terms of the multi-group setup, we find similar observations, that is FairGrad outperforms other approaches in fairness, albeit with a drop in accuracy. In fact, for Equality of Opportunity FairGrad almost outperforms all approaches in terms of fairness and accuracy. Overall, FairGrad performs reasonably well in all the settings we considered with no obvious weaknesses, that is no datasets with the lowest accuracy and fairness compared to the baselines.

4.6.5 Accuracy Fairness Trade-off

In this second set of experiments, we demonstrate the capability of FairGrad to support approximate fairness (see Section 4.4.4). In Figure 4.5, we show the performance, as accuracy-fairness pairs, of several models learned on the CelebA dataset by varying the fairness level parameter ϵ. These results suggest that FairGrad respects the constraints well. Indeed, the average absolute fairness level (across all the groups, see Section 4.6.2) achieved by FairGrad is either the same or less than the given

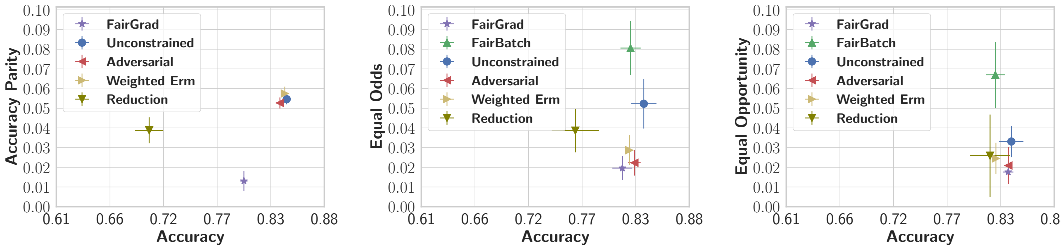


(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

FIGURE 4.2: Results for the Adult dataset using Linear Models.



(A) Non Linear - AP

(B) Non Linear - EOdds

(C) Non Linear - EOpp

FIGURE 4.3: Results for the Adult Multigroup dataset using Non Linear models.

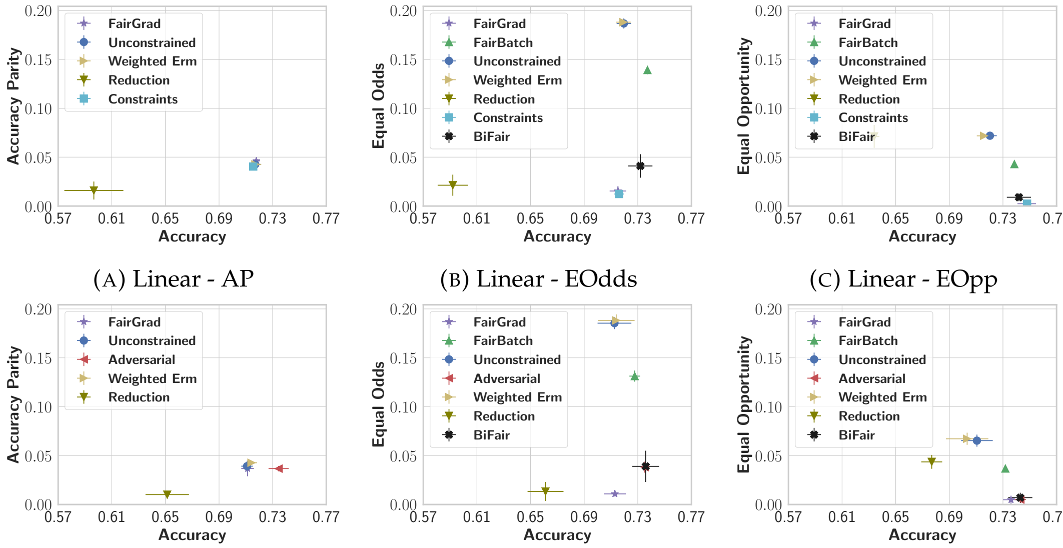
threshold. It is worth mentioning that FairGrad is designed to enforce ϵ-fairness for each constraint individually which is slightly different from the summarized quantity displayed here. Finally, as the fairness constraint is relaxed, the accuracy of the model increases, reaching the same performance as Unconstrained when the fairness level of the latter is below ϵ.

4.6.6 FairGrad as a Fine-Tuning Procedure

While FairGrad has primarily been designed to learn fair classifiers from scratch, it can also be used to fine-tune an existing classifier to achieve better fairness. To showcase this, we fine-tune the ResNet18 (He et al., 2016b) model, developed for image recognition, over the UTKFace dataset (Zhang, Song, and Qi, 2017), consisting of human face images tagged with Gender, Age, and Race information. Following the same process as Roh et al. (2020), we use Race as the sensitive attribute and consider two scenarios. Either we consider Demographic Parity as the fairness measure and use the gender (binary) as the target label or we consider Equalized Odds and predict the age (multi-valued). The results are displayed in Table 4.1. In both settings, FairGrad learns models that are more fair than an Unconstrained fine-tuning procedure, albeit at the expense of accuracy.

Method	s=Race ; y=Gender		s=Race ; y=Age	
	Accuracy	DP	Accuracy	EOdds
Unconstrained	0.8691 ± 0.0075	0.0448 ± 0.0066	0.6874 ± 0.0080	0.0843 ± 0.0089
FairGrad	0.8397 ± 0.0085	0.0111 ± 0.0064	0.6491 ± 0.0082	0.0506 ± 0.0059

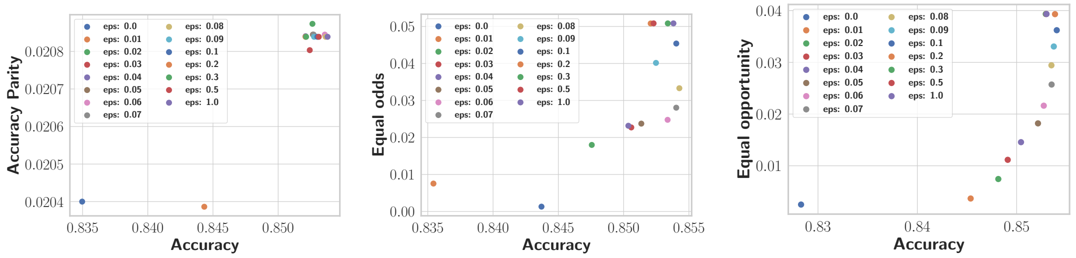
TABLE 4.1: Results for the UTKFace dataset where a ResNet18 is fine-tuned using different strategies.



(A) Linear - AP (B) Linear - EOdds (C) Linear - EOpp

(D) Non Linear - AP (E) Non Linear - EOdds (F) Non Linear - EOpp

FIGURE 4.4: Results for the Twitter Sentiment dataset for Linear and Non Linear Models.



(A) Celeb - AP (B) Celeb - EOdds (C) Celeb - EOpp

FIGURE 4.5: Results for CelebA using Linear models. The Unconstrained Linear model achieves a test accuracy of 0.8532 with fairness level of 0.0499 for EOdds, 0.0204 for AP, and 0.0387 for EOpp.

4.6.7 Impact of the Batch-size

In this section, we evaluate the impact of batch size on the fairness and accuracy level of the learned model. Indeed, at each iteration, in order to minimize the overhead associated with FairGrad (see Section 4.4.1), we update the weights using the fairness level of the model estimated solely on the current batch. When these batches are small, these estimates are unreliable and might lead the model astray. In Table 4.2 we present the performances of several linear models learned with different batch sizes on the CelebA dataset. Over this dataset, we observe that FairGrad consistently learns a fair model across all batch sizes and obtains reasonable accuracy since Unconstrained has an accuracy of 0.8532 for this problem. Nevertheless, we still recommend the practitioners to use a larger batch size whenever possible as we observe a slight reduction in terms of fairness standard deviations.

4.6.8 Computational Overhead

In this last experiment, we evaluate the overhead of FairGrad, by reporting the wall clock time in seconds to train for an epoch with the Unconstrained approach and our

Batch Size	8	16	32	64	128	256	512	1024	2048
Accuracy	0.8186	0.8234	0.8215	0.8268	0.8273	0.8286	0.8292	0.8289	0.8303
Accuracy Std	0.0013	0.006	0.0028	0.0025	0.0031	0.0008	0.0027	0.0017	0.0031
Fairness	0.0031	0.0091	0.0045	0.0036	0.0051	0.0046	0.004	0.0038	0.0057
Fairness Std	0.0042	0.0062	0.0012	0.0014	0.0025	0.0032	0.0026	0.0019	0.0018

TABLE 4.2: Batch size effect on the CelebA dataset with Linear Models and EOdds as the fairness measure.

Setting	Parameters	BS	Unconstrained	FairGrad	Delta
Linear model - Adult Dataset -CPU	106	512	0.277 ± 0.031	0.307 ± 0.01	0.03
2 layers -Adult Dataset -CPU	1762	512	0.315 ± 0.036	0.316 ± 0.029	0.01
5 layers -Adult Dataset -CPU	21346	512	0.370 ± 0.042	0.394 ± 0.025	0.02
10 layers -Adult Dataset -CPU	39042	512	0.483 ± 0.021	0.499 ± 0.034	0.02
20 layers -Adult Dataset -CPU	80642	512	0.672 ± 0.034	0.689 ± 0.026	0.02
ResNet18 trained -UTKFace -GPU	11177538	64	31.173 ± 0.085	31.588 ± 0.055	0.42
Bert Twitter Sentiment -GPU	109505310	32	2246.342 ± 3.20	2294.382 ± 4.01	48.04

TABLE 4.3: The computational overhead of FairGrad in various settings. BS here refers to Batch Size, and the Unconstrained and FairGrad columns refers to the average time in seconds taken by these approaches for an epoch, respectively. Delta refers to the difference in time between these two approaches.

method in various settings.

- We show the effect of model size by varying the number of hidden layers of the model over the Adult Income dataset, which consists of $45,222$ records. We used an Intel Xeon E5-2680 CPU to train.
- We consider a large convolutional neural network (ResNet18 (He et al., 2016b)) fine tuned over the UTK-Face dataset consisting of $23,708$ images. We trained the model using a Tesla P100 GPU.
- We experiment with a large transformer (bert-base-uncased (Devlin et al., 2019)) fine tuned over the Twitter Sentiment Dataset consisting of $200k$ tweets. We trained it using a Tesla P100 GPU.

We present results of the computation overhead of FairGrad in Table 4.3. We find that the overhead is limited and should not be critical in most applications as it does not depend on the complexity of the model but, instead, on the number of examples and the batch size. Overall, these observations are in line with the arguments presented in Section 4.4.2.

4.7 Conclusion

In this chapter, we proposed FairGrad, a fairness aware gradient descent approach based on a re-weighting scheme. We showed that it can be used to learn fair models for various group fairness definitions and is able to handle multiclass problems as well as settings where there is multiple sensitive groups. We empirically showed the competitiveness of our approach against several baselines on standard fairness datasets and on a Natural Language Processing task. We also showed that it can be used to fine-tune an existing model on a Computer Vision task. Finally, since it is based on gradient descent and has a small overhead, we believe that FairGrad could be used for a wide range of applications, even beyond classification.

Limitations and Societal Impact

While appealing, FairGrad also has limitations. It implicitly assumes that a set of weights that would lead to a fair model exists but this might be difficult to verify in practice. Thus, even if in our experiments FairGrad seems to behave quite well, a practitioner using this approach should not trust it blindly. It remains important to always check the actual fairness level of the learned model. On the other hand, we believe that, due to its simplicity and its versatility, FairGrad could be easily deployed in various practical contexts and, thus, could contribute to the dissemination of fair models.

Chapter 5

Fair Without Leveling Down: A New Intersectional Fairness Definition

Abstract

In this work, we consider the problem of intersectional group fairness in the classification setting, where the objective is to learn discrimination-free models in the presence of several intersecting sensitive groups. First, we illustrate various shortcomings of existing fairness measures commonly used to capture intersectional fairness. Then, we propose a new definition called the α-Intersectional Fairness, which combines the absolute and the relative performance across sensitive groups and can be seen as a generalization of the notion of differential fairness. We highlight several desirable properties of the proposed definition and analyze its relation to other fairness measures. Finally, we benchmark multiple popular fair machine learning approaches using our new fairness definition and show that they do not achieve any improvement over a simple baseline. Our results reveal that the increase in fairness measured by previous definitions hides a "leveling down" effect, i.e., degrading the best performance over groups rather than improving the worst one.

This chapter is based on: Maheshwari, Gaurav, Aurélien Bellet, Pascal Denis, and Mikaela Keller. "Fair NLP Models with Differentially Private Text Encoders." In The 2023 Conference on Empirical Methods in Natural Language Processing. The codebase for the chapter is available at - `https://github.com/saist1993/BenchmarkingIntersectionalBias`.

5.1 Introduction

In the preceding chapter, our primary focus was on group fairness with most of the discussion and evaluation centered on single sensitive attributes like gender (e.g., Male vs. Female) or race (e.g., African-Americans vs. European-Americans). However, recent studies (Yang, Cisse, and Koyejo, 2020; Kirk et al., 2021) have demonstrated that even when fairness can be ensured at the level of each individual sensitive axis, significant unfairness can still exist at the intersection levels (e.g., Male European-Americans vs. Female African-Americans). For example, Buolamwini and Gebru (2018) showed that commercially available face recognition tools exhibit significantly higher error rates for darker-skinned females than for lighter-skinned males. Similar

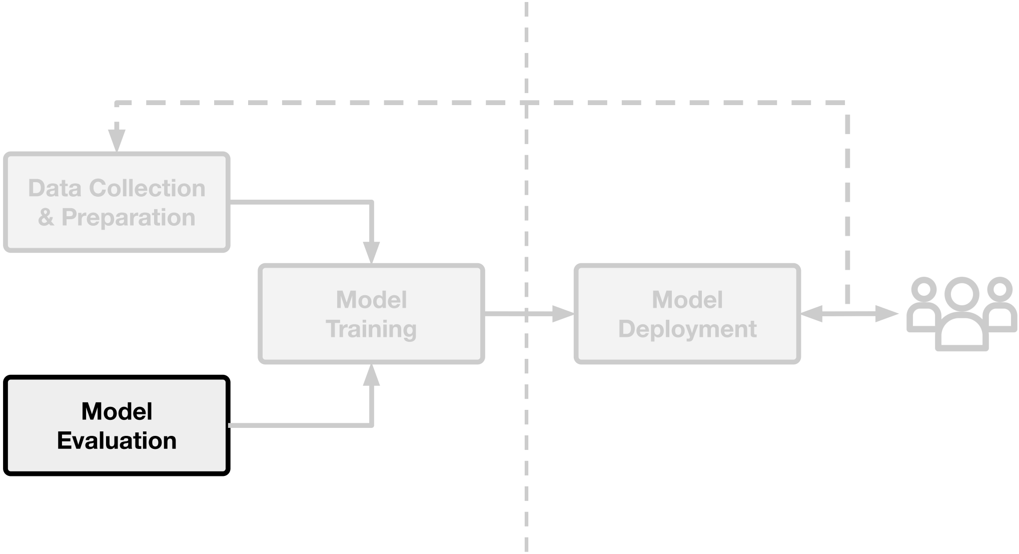


FIGURE 5.1: α-Intersectional Fairness focuses on model evaluation aspect of the machine learning pipeline.

observations have been made by several studies in NLP including contextual word representation Tan and Celis, 2019, and generative models Kirk et al., 2021. These findings resonate with the analytical framework of *intersectionality* Crenshaw, 1989, which argues that systems of inequality based on various attributes (like gender and race) may "intersect" to create unique effects.

Thus we shift our focus to intersectional fairness (see Section 3.3.3). More specifically, on capturing intersectional fairness and benchmarking various methods in this setting. The chapter is part of the model evaluation aspect of the machine learning pipeline (see Figure 5.1). To capture intersectional fairness, several measures have been proposed Kearns et al., 2018; Hébert-Johnson et al., 2018; Foulds et al., 2020. Amongst them the most commonly used (Lalor et al., 2022; Zhao et al., 2022; Subramanian et al., 2021) is Differential Fairness (DF) Foulds et al., 2020, which is the log-ratio of the best-performing group to the worst-performing group for a given performance measure (such as the True Positive Rate. For more details refer to Section 3.2.2). While DF has many desirable properties, in this work we emphasize that DF implements a "strictly egalitarian" view, i.e., it only considers the *relative* performance between the group and ignores their *absolute* performance. In particular, a trivial way to improve fairness as measured by DF is by harming the best-off group without improving the worst-off group. This phenomenon, known as *leveling down*, does not fit the desired fairness requirements in many practical use-cases Mittelstadt, Wachter, and Russell, 2023; Zietlow et al., 2022. Yet, we empirically observe that (i) popular fairness-promoting approaches tend to level down more in intersectional fairness, and (ii) this often goes unnoticed in the overall performance of the model due to the large number of groups induced by intersectional fairness.

To address these issues and explicitly capture the leveling down phenomena, we propose a generalization of DF, called **α-Intersectional Fairness** (IF$_\alpha$), which takes into account both the relative performance between the groups and the absolute performance of the groups. More precisely, IF$_\alpha$ is a weighted average between the relative and absolute performance of the groups, and allows the exploration of the whole trade-off between these two quantities by changing their relative importance via a weight $\alpha \in [0, 1]$. Our extensive benchmarks across various datasets show that many existing fairness-inducing methods aim for a different point in the aforementioned trade-off and generally show no consistent improvement over a simple unconstrained approach.

In summary, our primary contributions are as follows:

- We showcase the shortcomings of the existing intersectional fairness definition and propose a generalization called **α-Intersectional Fairness**. We analyze the properties and behavior of the proposed fairness measure, and contrast them with DF.
- We benchmark existing fairness approaches on multiple datasets and evaluate their performance with several fairness measures, including ours. On the one hand, we find that many fairness approaches optimize for existing fairness measures by harming both the worst-off and best-off groups or only the best-off group. On the other hand, our measure is more careful in showing improvements over a simple baseline than previous metrics, allowing the emphasis on cases of leveling down.

5.2 Setting

In this section, we begin by introducing our notations and then formally define problem statement.

5.2.1 Notations

In this study, we adopt and extend the notations proposed by Morina et al. (2019). Let p denote the number of distinct *sensitive axes* of interest, which generally correspond to socio-demographic features of a population. We refer to these sensitive axes as $A_1, \ldots, A_p$, each of which is a set of discrete-valued *sensitive attributes*. For instance, a dataset may be composed of gender, race, and age as the three sensitive axes, and each of these sensitive axes may be encoded by a set of sensitive attributes, such as gender: {male, female, non-binary}, race: {European American, African American}, and age: {under 45, above 45}. We define a *sensitive group* $\mathbf{g}$ as any p-dimensional vector in the Cartesian product set $\mathcal{G} = A_1 \times \cdots \times A_p$ of these sensitive axes. A sensitive group $\mathbf{g} \in \mathcal{G}$ can then be written as $(a_1, \ldots, a_p)$ with $a_j \in A_j$.

5.2.2 Problem Statement

Consider a feature space $\mathcal{X}$, a finite discrete label space $\mathcal{Y}$, and a set $\mathcal{G}$ representing all possible intersections of p sensitive axes as defined above. Let $\mathcal{D}$ be an unknown distribution over $\mathcal{X} \times \mathcal{Y} \times \mathcal{G}$ through which we sample i.i.d a finite dataset $\mathcal{T} = \{(x_i, y_i, \mathbf{g}_i)\}_{i=1}^{n}$ consisting of n examples. This sample can be rewritten as $\mathcal{T} = \bigcup_{\mathbf{g} \in \mathcal{G}} \mathcal{T}_{\mathbf{g}}$ where $\mathcal{T}_{\mathbf{g}}$ represents the subset of examples from group $\mathbf{g}$. The goal of fair machine learning is then to learn an accurate model $h_\theta \in \mathcal{H}$, with learnable parameters $\theta \in \mathbb{R}^D$, such that $h_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ is fair with respect to a given group fairness definition (See section 3.3) like Equal Opportunity (Hardt, Price, and Srebro, 2016), Equal Odds (Hardt, Price, and Srebro, 2016), Accuracy Parity Zafar et al., 2017b, etc.

Existing group fairness definitions generally consist of comparing a certain performance measure (See Section 3.2.2), such as True Positive Rate (TPR), False Positive

Rate (FPR) or accuracy, across groups. In the following, for the sake of generality, we abstract away from the particular measure and denote by $m(h_\theta, \mathcal{T}_\mathbf{g}) \in [0,1]$ the group-wise performance for model h_θ on the group of examples $\mathcal{T}_\mathbf{g}$, with the convention that **higher values of m correspond to better performance**. For instance, in the case of TPR (used to define Equal Opportunity) we define $m(h_\theta, \mathcal{T}_\mathbf{g}) = \mathrm{TPR}(h_\theta, \mathcal{T}_\mathbf{g})$, while for FPR, we define it as $m(h_\theta, \mathcal{T}_\mathbf{g}) = 1 - \mathrm{FPR}(h_\theta, \mathcal{T}_\mathbf{g})$.

5.3 Existing Intersectional Framework

While the literature on group fairness in machine learning initially considered a single sensitive axis (Section 3.3.1), several works have recently proposed fairness definitions for the intersectional setting (Gohar and Cheng, 2023) (Section 3.3.3). Kearns et al. (2018) proposed subgroup-fairness, which is based on the difference in performance of a particular group weighted by the size of the group. Several calibration and metric fairness-based variants were considered by Hébert-Johnson et al. (2018) and Yona and Rothblum (2018). A shortcoming of these notions is that they weight each group by its size, hence small groups may not be protected even though they are often the disadvantaged ones.

5.3.1 Differential Fairness

To circumvent the above issue as introduced in Section 3.3.3, Foulds et al. (2020) proposed Differential Fairness (DF), which puts a constraint on the relative performance between all pairs of groups. DF was originally proposed for statistical parity (Foulds et al., 2020), and was then extended by (Morina et al., 2019) to generalize other fairness definitions such as parity in False Positive Rates and Equal Odds. Below, we provide a general definition of DF based on an arbitrary group-wise performance measure m as defined in Section 5.2.2.[1]

Definition 4 (Differential Fairness)**.** A model h_θ is ϵ-differentially fair (*DF*) with respect to a group-wise performance measure m, if

$$\mathrm{DF}(h_\theta, m) \equiv \max_{\mathbf{g},\mathbf{g}' \in \mathcal{G}} \log \frac{m(h_\theta, \mathcal{T}_\mathbf{g})}{m(h_\theta, \mathcal{T}_{\mathbf{g}'})} \leq \epsilon.$$

It is important to note that DF only depends on the relative performance between the best-performing group and the worst-performing group.

5.3.2 Shortcomings of Differential Fairness

We now highlight what we believe to be a key shortcoming of DF in the context of intersectional fairness: **DF can be improved by leveling down, i.e., harming the best-off and/or worst-off group, without significantly affecting the overall performance of the model.** This problem is caused by the combination of two factors.

First, DF is a strictly egalitarian measure that only considers the relative performance between groups. This can lead to situations where a model that improves the performance across all groups is deemed more unfair by DF. To illustrate this, let the

[1] We note that, in their extension to parity in False Positive Rates, Morina et al. (2019) did not account for the fact that higher FPR means lower performance, hence harming all groups always leads to better fairness. Our general formulation in Definition 4 fixes this problem through the convention that higher m corresponds to better performance.

group-wise performance measure m to be the TPR and consider two models h_θ and $h_{\tilde{\theta}}$. Let the worst-off and best-off group-wise performance of h_θ be 0.50 and 0.60, respectively. For $h_{\tilde{\theta}}$, let it be 0.65 and 0.95. According to DF, h_θ is more fair than $h_{\tilde{\theta}}$ as the two groups are closer, while $h_{\tilde{\theta}}$ has better performance for both groups. In other words, h_θ is leveling down compared to $h_{\tilde{\theta}}$, but is deemed more fair. This exhibits the tension between the relative performance between groups, and the absolute performance of the groups.

The second factor is that in intersectional fairness, leveling down can have a negligible effect on the overall performance of a model on the full dataset. This is because the number of groups in intersectional fairness is typically quite large (exponential in the number of sensitive axes p). Therefore, the bulk of examples generally do not belong to either the worst-off or best-off group, leading to a situation where the performance of other groups accounts for most of the model's overall performance. This issue may be further exacerbated if the class proportions are imbalanced across groups.

5.4 α-Intersectional Fairness

In order to circumvent the above issue and effectively capture intersectional fairness while taking into account the leveling down phenomena, we propose α-Intersectional Fairness (IF$_\alpha$). Our definition is essentially a convex combination of two components, namely (i) Δ_{rel}, which takes into account the relative performance between the two groups, such as the ratio of their performance, and (ii) Δ_{abs}, which captures the leveling down effect by accounting for the absolute performance of the worst-off group.

More precisely, given a model h_θ and a group-wise performance measure m, let us first define a measure of fairness for a pair of groups $\mathbf{g}$ and $\mathbf{g}'$:

$$I_\alpha(\mathbf{g}, \mathbf{g}', h_\theta, m) = \alpha\Delta_{abs} + (1-\alpha)\Delta_{rel}, \tag{5.1}$$

where $\alpha \in [0,1]$ and

$$\begin{aligned}\Delta_{abs} &= \max\left(1 - m(h_\theta, \mathcal{T}_{\mathbf{g}}), 1 - m(h_\theta, \mathcal{T}_{\mathbf{g}'})\right),\\ \Delta_{rel} &= \frac{1 - \max\left(m(h_\theta, \mathcal{T}_{\mathbf{g}}), m(h_\theta, \mathcal{T}_{\mathbf{g}'})\right)}{1 - \min\left(m(h_\theta, \mathcal{T}_{\mathbf{g}}), m(h_\theta, \mathcal{T}_{\mathbf{g}'})\right)}.\end{aligned}$$

Now taking the maximum value of I_α over all pairs of groups, we get our proposed notion of α-Intersectional Fairness.

Definition 5 (α-Intersectional Fairness)**.** A model h_θ is (α, γ)-intersectionally fair (IF$_\alpha$) with respect to a group-wise performance measure m, if

$$\mathrm{IF}_\alpha(h_\theta, m) \equiv \max_{\mathbf{g}, \mathbf{g}' \in \mathcal{G}} I_\alpha(\mathbf{g}, \mathbf{g}', h_\theta, m) \leq \gamma.$$

Note that IF$_\alpha(h_\theta, m)$ can be equivalently obtained as the the value of I_α over the pair of worst performing and the best performing group, as shown by the following proposition.

Proposition 1. If a model h_θ is (α, γ)-intersectionally fair with respect to a group-wise performance measure m, then

$$\mathrm{IF}_\alpha(h_\theta, m) = I_\alpha(\mathbf{g}^w, \mathbf{g}^b, h_\theta, m) \leq \gamma,$$

where $\mathbf{g}^w = \arg\min_{\mathbf{g}\in\mathcal{G}} m(h_\theta, \mathcal{T}_\mathbf{g})$ and $\mathbf{g}^b = \arg\max_{\mathbf{g}\in\mathcal{G}} m(h_\theta, \mathcal{T}_\mathbf{g})$.

5.5 Design Choices of α-Intersectional Fairness

In this section, we discuss our design choices for Δ_{abs} and Δ_{rel}.

Choice of Δ_{rel} An alternate choice of Δ_{rel} is to utilize the performance difference between the groups instead of the above mentioned ratio. However, we advocate for the ratio as a superior choice for the following reasons:

- Scale-Invariant Comparison: The ratio enables comparing two models without the influence of the scale by normalizing the relative performance of a model. For instance, assume two models h_θ and $h_{\theta'}$ with the worst and the best group's performance for h_θ as 0.01 and 0.02 respectively, and 0.1 and 0.2 for $h_{\theta'}$. In this setting, the Δ_{rel} as the difference would always assign h_θ as fairer, even though both models are twice worse for the worst group compared to the best group. Note that our overall fairness measure accounts for the effect of scale through the inclusion of Δ_{abs}. This is in-contrast to DFwhich does not take scale into account.

- Alignment with the 80% rule: The ratio aligns with the well-known 80% rule (Commission et al., 1990), which states that there exists legal evidence of discrimination if the ratio of the probabilities for a favorable outcome between the disadvantaged sensitive group and the advantaged sensitive group is less than 0.8. By adopting the ratio as Δ_{rel}, our metric adheres to this established criterion.

- Influence of worst-case group: If Δ_{rel} represents the difference in performance, then at $\alpha = 0.5$ the model with better worst-case performance will always have a lower γ than the one with worse worst-case performance. In other words, at $\alpha = 0.5$, Δ_{abs} would always dominate Δ_{rel}. However, this contradicts the intuitive understanding that, at $\alpha = 0.5$, both Δ_{rel} and Δ_{abs} should exert an equal influence.

Choice of Δ_{abs} An alternate choice we explored for Δ_{abs} was the average performance of the two groups involved instead of just the worst-performing one. However, Proposition 1 does not hold in the average case. This implies that a pair of groups can exist for which I_α is larger than the pair of groups consisting of the worst and best-performing groups. Moreover, Proposition 1 is an essential building block for intersectional property which is described later.

5.6 Properties of α-Intersectional Fairness

In the following, we compare and contrast our fairness definition with DF when evaluating the fairness of the two models. We then investigate various properties of our proposed definition and discuss the impact of α.

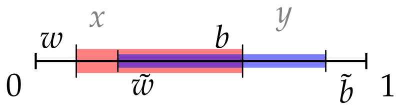


FIGURE 5.2: Group-wise performance range comparison. The range of group-wise performances of models h_θ and model $h_{\tilde{\theta}}$ are respectively $[w, b]$ and $[\tilde{w}, \tilde{b}]$. Note that the difference x (resp. y) between the best (resp. worst) group-wise performances of h_θ and $h_{\tilde{\theta}}$ can be positive or negative.

Comparing DF and IF$_\alpha$. The primary difference between DF and IF$_\alpha$ when comparing two models arises when one model adversely affects the worst-off group (Δ_{abs}) more than the other, despite having better relative performance (Δ_{rel}). In this case, DF would consistently consider one model more fair than the other, whereas IF$_\alpha$ enables the exploration of this tension by varying the relative importance of both criteria through α.

We formally capture this intuition as follows. Consider two models h_θ and $h_{\tilde{\theta}}$. Let the value of the worst-off and the best-off group's performance for the model h_θ be w and b, respectively. Similarly, for model $h_{\tilde{\theta}}$ let the worst and the best group's performance be $\tilde{w}$ and $\tilde{b}$, respectively. Without the loss of generality, $\tilde{w}$ and $\tilde{b}$ can be written as $\tilde{w} = w + x$ and $\tilde{b} = b + y$. Note that x and y can be either positive or negative as long as $\tilde{w} \leq \tilde{b}$. We visualize this setup in Figure 5.2. Based on this setup, we have following cases:

- $x \geq y \geq 0$: In this case, h_θ harms the worst-off group (absolute performance) more, and its relative performance is worse than $h_{\tilde{\theta}}$. In Figure 5.2, this corresponds to $\tilde{w} \geq w$ and the blue region is smaller than the red region. Here, $\text{IF}_\alpha(h_{\tilde{\theta}}, m) \leq \text{IF}_\alpha(h_\theta, m) \; \forall \, \alpha \, \in [0, 1]$, and $\text{DF}(h_{\tilde{\theta}}, m) \leq \text{DF}(h_\theta, m)$.
- $x \leq y \leq 0$: This is similar to the case above, but with $h_{\tilde{\theta}}$ harming the groups more than h_θ. In Figure 5.2, this corresponds to $\tilde{w} \leq w$ and the blue region is larger than the red region. Here, $\text{IF}_\alpha(h_{\tilde{\theta}}, m) \leq \text{IF}_\alpha(h_\theta, m) \; \forall \, \alpha \, \in [0, 1]$, and $\text{DF}(h_{\tilde{\theta}}, m) \leq \text{DF}(h_\theta, m)$.
- All other cases: In this setting, one of the model has better Δ_{abs} performance, while the other model has better Δ_{rel} performance. The fairness in this setting depends on the relative importance of absolute and relative performance for IF$_\alpha$, while for DF it exclusively depends on absolute performance. In Figure 5.2, this corresponds to $\tilde{w} \leq w$ and the blue region is smaller than the red region or vice-versa. Here, $\exists \alpha \in [0, 1]$ for which $\text{IF}_\alpha(h_{\tilde{\theta}}, m) \geq \text{IF}_\alpha(h_\theta, m)$ and vice versa. On the other hand, $\text{DF}(h_{\tilde{\theta}}, m) \leq \text{DF}(h_\theta, m)$ if $y \times m(h_\theta, \mathcal{T}_{\mathbf{g}^w}) \leq x \times m(h_\theta, \mathcal{T}_{\mathbf{g}^b})$, otherwise $\text{DF}(h_{\tilde{\theta}}, m) > \text{DF}(h_\theta, m)$.

To summarize, in the first two cases, one model harms the worst-off group (absolute performance), and the relative performance of that model is worse than the other. Thus, a good fairness measure should assign a higher unfairness to that model, which both DF and IF$_\alpha$ do. In the third case, one model performs better on the worst-off group, while the other model has a closer relative performance. The fairness in this setting depends on the relative importance of absolute and relative performance. Here, DF consistently assigns one model a higher fairness than the other, while IF$_\alpha$ enables to explore this tension and tune the relative importance of both criteria through α. For instance, the previous example in Section 5.2 falls in the third case.

On the one hand, DF will assign higher ϵ for h_{θ_1} in comparison to h_{θ_2}. On the other hand, IF$_\alpha$ will assign higher γ for h_{θ_1} for $\alpha \in (0.0, 0.81)$, while for all other α, the γ would be higher for h_{θ_2}. We illustrate the effect of α in more details below.

Impact of α: The parameter α allows to tune the relative importance of Δ_{abs} and Δ_{rel}. On the one end of the spectrum, $\alpha = 0$ corresponds to considering only the relative performance Δ_{rel}, while $\alpha = 1$ corresponds to considering only the absolute performance. At $\alpha = 0.0$ we recover the same relative ranking of unfairness as DF, and thus DF can be seen as a special case of IF$_\alpha$. In other words, for any three models h_{θ_1}, h_{θ_2}, and h_{θ_3} such that $\mathrm{DF}(h_{\theta_1}, m) \geq \mathrm{DF}(h_{\theta_2}, m) \geq \mathrm{DF}(h_{\theta_3}, m)$, then $\mathrm{IF}_0(h_{\theta_1}, m) \geq \mathrm{IF}_0(h_{\theta_2}, m) \geq \mathrm{IF}_0(h_{\theta_3}, m)$. On the other end, $\alpha = 1$ only considers the absolute performance Δ_{abs}), and $\alpha = 0.5$ corresponds to giving Δ_{abs} and Δ_{rel} an equal importance. In practice, it is useful to visualize the complete trade-off by plotting $\alpha \mapsto \mathrm{IF}_\alpha$ (see Section 5.7).

Intersectional Property: We have the following intersectional property.

Proposition 2. Let the model h_θ be (α, γ)-intersectionally fair over the set of groups defined by $\mathcal{G} = A_1 \times \cdots A_p$. Let $1 \leq s_1 \leq \cdots \leq s_k \leq p$, and $\mathcal{P} = A_{s_1} \times \cdots A_{s_k}$ be the Cartesian product of the sensitive axes where $s_j \in \mathbb{N}^+$. Then, h_θ is (α, γ)-intersectionally fair over $\mathcal{P}$.

In other words, the fairness value calculated over the intersectional groups also holds over independent and "gerrymandering" intersectional groups Yang, Cisse, and Koyejo, 2020. For instance, if a model is (α, γ)-intersectionally fair in a space defined by gender, race, and age, then it is also (α, γ)-intersectionally fair in the space defined by gender and race, or just gender. We delegate the proof to Appendix C.1.

Generalization Guarantees: α-Intersectional Fairness enjoys the same generalization guarantees as the ones shown for DF in (Foulds et al., 2020). Indeed, the result of Foulds et al. (2020) relies on a generalization analysis of the group-wise performance measure m, which directly translates into generalization guarantees for IF$_\alpha$.

Guidelines for setting α: α-Intersectional Fairness enables exploring the tradeoff between worst-case performance and relative performance across groups. Indeed, at alpha=0.0, only relative performance is considered, aligning with strictly egalitarian measures. On the other extreme, at alpha=1.0, solely the worst-off group performance is considered. Based on this, we recommend:
Setting $\alpha = 0.75$ (more focus towards worst case performance) in:

- Situations where the cost of misclassification is not similar for each group. In these cases, leveling down would disproportionately affect those subgroups for whom the cost is higher. One example can be seen in education system, where the cost of denying financial assistance has higher impact on minority (Nora and Horvath, 1989; Hinojosa, 2023).
- Cases where data for disadvantaged groups is unreliable due to historical underrepresentation and lack of opportunities. For instance, certain facial recognition systems exhibit a higher likelihood of error when analyzing images of dark-skinned female individuals (Buolamwini and Gebru, 2018). Similarly, Sap et al. (2019) found that the hate speech detection systems are biased against black people.

In such contexts, emphasizing improvement for these disadvantaged groups is more pivotal than uniform performance over all subgroups, in line with the ideas of affirmative action. These scenarios best align with strategies seeking Demographic Parity or Equalized Odds.

Setting $\alpha = 0.25$ (more focus on relative performance) in:

- Scenarios where no group is significantly worse off, but to make sure that the algorithm behaves similarly for all the groups involved. This is related to algorithmic bias, as presented by Mehrabi et al. (2022). Moreover, the misclassification costs are similar in this setting.
- Legal or regulatory requirements may mandate similar outcomes across groups, like the 4/5th employment rule.[2] However, one must exercise caution when extrapolating this to other contexts, as it can lead to the "portability trap" as discussed by (Selbst et al., 2019).

In such a context, the emphasis is on equality among the groups. In practice, these scenarios indicate places where the practitioner would advocate for Accuracy Parity.

Otherwise, we recommend setting $\alpha = 0.50$ (a neutral default) when no domain or context-specific insights are available. This is what we used in our experiments. Ultimately, the choice of alpha reflects an understanding of the domain, the inherent biases in the data, and the real-world consequences of misclassifications.

5.7 Experiments

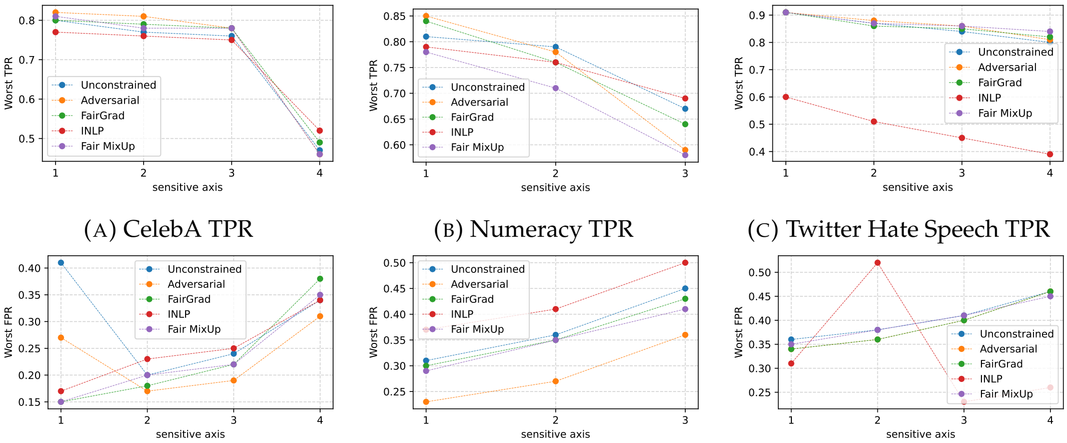
In this section, we present experiments[3] that showcase (i) the model's performance over the worst-off group as the number of sensitive axes increases, and (ii) the “leveling down" phenomenon observed in various fairness-promoting mechanisms, along with the effectiveness of α-Intersectional Fairness in uncovering it. However, before describing these experiments, we begin with an overview of the datasets, baselines, and fairness measures used.

Datasets: We benchmark over four datasets covering both text and images, with varying numbers of examples and sensitive groups:

- *Twitter Hate Speech*: The dataset is derived from multilingual Twitter Hate speech corpus (Huang et al., 2020) consisting of tweets annotated with 4 demographic factors (sensitive axes), namely age, race, gender, and country. The primary objective is to classify individual tweets as either hate speech or non-hate speech. In this work, we focus on the English subset and binarize all the demographic factors resulting in a total of 63 sensitive groups. Moreover, we only choose tweets where all the demographic factors are present. Consequently, our train, valid and test sets consists of 22, 818, 4, 512, and 5, 032 tweets.
- *CelebA* (Liu et al., 2015): The dataset consists of 202, 599 images of human faces, alongside 40 binary attributes for each image. We set ‘sex', ‘Young', ‘Attractive', and ‘Pale Skin' attributes as the sensitive axis for the images and ‘Smiling' as the class label. We split the dataset into 80% training and 20% test split. Furthermore, we set aside 20% of the training set as the validation split.

[2] https://www.law.cornell.edu/cfr/text/29/1607.4

[3] source code is available here: https://github.com/saist1993/BenchmarkingIntersectionalBias



(A) CelebA TPR

(B) Numeracy TPR

(C) Twitter Hate Speech TPR

(D) CelebA FPR

(E) Numeracy FPR

(F) Twitter Hate Speech FPR

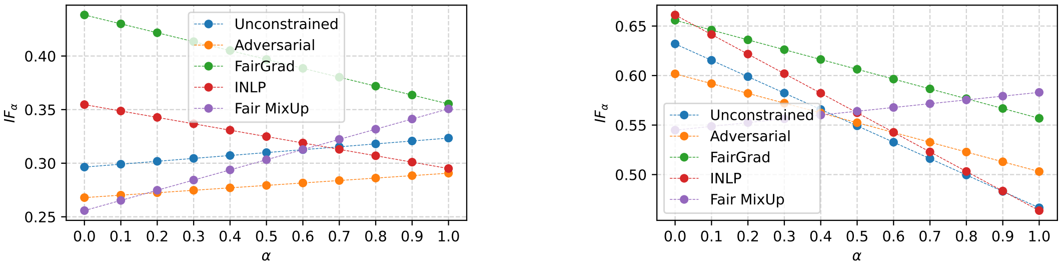
FIGURE 5.3: Test results over the worst-off group on *CelebA*, *Twitter Hate Speech*, and (b) *Numeracy* by varying the number of sensitive axes. For p binary sensitive axis in the dataset, the total number of sensitive groups are $p^3 - 1$. Note that in FPR, lower the value better it is, while for TPR opposite is true.

- *Psychometric dataset* (Abbasi et al., 2021): The dataset is a collection of $8,502$ free text responses alongside numerical scores over multiple psychometric dimensions. In this work, we focus on two dimensions:
  - *Numeracy* reflects the numerical comprehension capability of the individual.
  - *Anxiety* reflects the anxiety level as described by the patient.

  Both these datasets consists of free text responses and binarized scores by the medical expert. Moreover, each response is associated with gender, race, age, and income. We use same pre-processing as Lalor et al., 2022 and follow the same procedure to split the dataset as described above.

For improved readability, we present a subset of experiments in the main thesis. The remaining experiments are included in the Appendix.

Methods. We evaluate the fairness performance and accuracy of the following methods: (i) `Unconstrained` which is oblivious to any fairness measure and solely optimizes the model's accuracy; (ii) `Adversarial` implements standard adversarial learning approach (Li, Baldwin, and Cohn, 2018), where an adversary is added to the `Unconstrained` with the objective to predict the sensitive attributes; (iii) `FairGrad` (Maheshwari and Perrot, 2022) (introduced in the previous chapter), is an in-processing approach that iteratively learns group-specific weights based on the fairness level of the model; (iv) `INLP` (Ravfogel et al., 2020), is a post-processing approach that iteratively trains a classifier to predict the sensitive attributes and then projects the representation on the classifier's null space. To enforce fairness across multiple sensitive axes in this work, we follow the extension proposed by Subramanian et al. (2021); (v) `Fair MixUp` (Chuang and Mroueh, 2021) is a data augmentation mechanism that enforces fairness by regularizing the model on the paths of interpolated samples between the sensitive groups.



(A) CelebA FPR

(B) Anxiety FPR

FIGURE 5.4: Value of IF$_\alpha$ on the test set of CelebA, and Numeracy datasets for varying $\alpha \in [0, 1]$.

In all our experiments, we employ the same model architecture for all the approaches to have a fair comparison. Specifically, we use a three-hidden layer fully connected neural network with 128, 64, and 32 corresponding sizes. Furthermore, we use ReLU as the activation with dropout fixed to 0.5. We optimize cross-entropy loss in all cases with Adam (Kingma and Ba, 2015) as the optimizer using default parameters. Moreover, for Twitter Hate Speech and Numeracy datasets, we encode the text using bert-base-uncased Devlin et al., 2019 text encoder. For CelebA, an image dataset, we employ ResNet18 (He et al., 2016b) as the encoder. In all cases, we do not fine-tune the pre-trained encoders. Lastly, several previous studies have shown the effectiveness of equal sampling in improving fairness (Kamiran and Calders, 2009; Chawla et al., 2003; Kamiran and Calders, 2010; González-Zelaya et al., 2021). That is, to counter the imbalance in the training data, the data is resampled so that there is an equal number of examples from each group and class in the final training set. Through preliminary experiments, we determine that equal sampling improves the worst-case performance of several approaches, including `Unconstrained` in various settings. We thus incorporate it as a hyperparameter indicating a continuous scale between undersampling and oversampling. Note that we also incorporate a setting where no equal sampling is performed, and we take the distribution as it is.

Fairness performance measure. In this work we focus on True Positive Rate parity and False Positive Rate parity as the fairness measure. The corresponding group wise performance measure m for these fairness measures are TPR and FPR. Formally, m in case of TPR for a group $\mathbf{g}$ is:

$$m(h_\theta, \mathcal{T}_\mathbf{g}) = P(h_\theta(x) = 1 | y = 1) \ \forall x, y \ \in \mathcal{T}_\mathbf{g},$$

while the FPR for a group $\mathbf{g}$ is:

$$m(h_\theta, \mathcal{T}_\mathbf{g}) = 1 - P(h_\theta(x) = 0 | y = 1) \ \forall x, y \ \in \mathcal{T}_\mathbf{g}$$

In order to estimate the empirical probabilities, we employ the bootstrap estimation procedure as proposed by Morina et al. (2019). In total, we generate 1000 datasets by sampling from the original dataset with replacement. We then estimate the probabilities on this dataset using smoothed empirical estimation mechanism and then average the results over all the sampled datasets. In order to evaluate the utility of various methods, we employ balanced accuracy. Note that the choice of TPR Parity, and FPR Parity allows the derivation of several other fairness measures including Equal Opportunities, and Equalized Odds.

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.81 + 0.0	0.08 + 0.01	0.36 + 0.04	0.36 +/- 0.06	0.31 +/- 0.02
`Adversarial`	0.8 + 0.0	0.07 + 0.02	0.32 + 0.02	0.31 +/- 0.12	0.28 +/- 0.04
`FairGrad`	0.77 + 0.01	0.14 + 0.01	0.39 + 0.01	0.34 +/- 0.03	0.4 +/- 0.02
`INLP`	0.8 + 0.0	0.09 + 0.01	0.34 + 0.04	0.32 +/- 0.03	0.32 +/- 0.01
`Fair MixUp`	0.8 + 0.0	0.08 + 0.01	0.37 + 0.02	0.38 +/- 0.04	0.3 +/- 0.01

(A) Results on CelebA

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.63 + 0.01	0.27 + 0.04	0.5 + 0.03	0.38 +/- 0.05	0.55 +/- 0.06
`Adversarial`	0.62 + 0.01	0.28 + 0.05	0.53 + 0.09	0.43 +/- 0.04	0.55 +/- 0.06
`FairGrad`	0.63 + 0.01	0.33 + 0.04	0.59 + 0.06	0.49 +/- 0.05	0.61 +/- 0.03
`INLP`	0.63 + 0.01	0.27 + 0.04	0.49 + 0.03	0.36 +/- 0.03	0.56 +/- 0.05
`Fair MixUp`	0.61 + 0.02	0.3 + 0.03	0.61 + 0.07	0.58 +/- 0.03	0.56 +/- 0.03

(B) Results on Anxiety

TABLE 5.1: Test results on (a) *CelebA*, and (b) *Anxiety* using False Positive Rate while optimizing for DF. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness $\text{IF}_{\alpha=0.5}$. For both fairness definition, lower is better, while for balanced accuracy, higher is better. The Best Off and Worst Off, in both cases lower is better, represents the min FPR and max FPR. Results have been averaged over 5 different runs. We deem a method to exhibit leveling down if its performance on either the worst-off or best-off group is inferior to the performance of an unconstrained model which we have highlighted using cyan (🟦).

5.7.1 Worst-off performance and number of sensitive axis

In this experiment, we empirically evaluate the interplay between the number of sensitive groups and the harm towards the worst-off group. To this end, we iteratively increase the number of sensitive axes in the dataset and report the performance of the worst-off group for each approach. For instance, with CelebA we first randomly added gender (randomly chosen) when considering 1 sensitive axis. In the next iteration, we added race (randomly chosen) to the set with gender (previously added). Similarly, we then added age, and finally country. Note that for all the datasets, we start with a random choice of sensitive axis hoping to remove any form of selection bias. To select the optimal hyperparameters for this experiment, we follow the same procedure described in (Maheshwari et al., 2022) with the objective to select the hyperparameters with the best performance over the worst-off group.

We plot the results of this experiment in Figure 5.3. The results over the Anxiety dataset, which follow similar trend, can be found in the Appendix C.2. Based on these results, we observe that as the number of subgroups increases, the performance of the worst-off group becomes worse for all approaches in all settings. This can be attributed to the fact that the number of training examples available for each group decreases as the number of sensitive axis in the dataset increases. In terms of the performance of other approaches in comparison to `Unconstrained`, we find that fairness-inducing approaches generally perform better or similar to `Unconstrained` when 1 or 2 sensitive axes are considered. However, when 3 or more sensitive axis are considered, the performance of all approaches tends to converge to that of `Unconstrained`. For instance, in CelebA, on the one hand, with 1 sensitive axis, all approaches significantly outperform `Unconstrained` with the difference between the best-performing method and `Unconstrained` being 0.26. On the other hand, when 4 sensitive axes are considered, the difference between the best-performing method and `Unconstrained` is 0.03, with only `Adversarial` outperforming it.

In a similar fashion, when considering TPR over Numeracy dataset, `Unconstrained` performs significantly worse than `FairGrad` and `Adversarial` with 1 sensitive axis while outperforming all approaches apart from `INLP` when 3 sensitive axis are considered. Similar observations can be made for Numeracy and Twitter Hate Speech datasets in the FPR setting, with some minor exceptions. Overall we find that most fairness approaches start harming or do not improve the worst-off group as the number of sensitive axes grows in the dataset. Thus it is pivotal for an intersectional fairness measure to consider the harm induced by an approach.

5.7.2 Benchmarking Intersectional Fairness

In this experiment, we showcase the leveling down phenomena shown by various existing approaches. We also compare and contrast IF_α and DF. The results of this comparison over FPR parity can be found in Table 5.1a and 5.1b for CelebA and Anxiety respectively. The results of remaining two datasets over FPR parity, and all datasets over TPR Parity can be found in Appendix C.2. *In these experiment, we deem a method to exhibit leveling down if its performance on either the worst-off or best-off group is inferior to the performance of an unconstrained model.* In the results table, we highlight the methods that show leveling down in cyan (🟦).

We find that most of the methods have similar balanced accuracy across all the datasets, even if the fairness levels are different. This observation aligns with the arguments presented in Section 5.3 about the relationship between group fairness measure and the overall performance. In terms of fairness, most methods showcase leveling down. For instance, over the CelebA dataset, all methods apart from `Adversarial` shows leveling down. While in the case of Anxiety, all methods apart from `INLP` shows leveling down.

While comparing DF and $\text{IF}_{\alpha=0.5}$, we find that $\text{IF}_{\alpha=0.5}$ is more conservative in assigning fairness value, with most approaches performing similarly to Unconstrained. Moreover, leveling down cases may go unnoticed in DF. For instance, over the CelebA dataset, even though `FairGrad` and `INLP` showcases leveling down, the fairness value assigned by DF is lower for them than the one assigned to `Unconstrained`. Similar observation can be seen over Numeracy in case of `INLP`.

A particular advantage of IF_α over DF is that it equips the practitioner with a more nuanced view of the results. In Figure 5.4, we plot the complete trade-off between the relative and the absolute performance of groups by varying α. For instance, in CelebA FPR, `Fair MixUp` shows the lowest level of unfairness at $\alpha = 0.0$. However, as soon as the worst-off group's performance is considered, i.e., $\alpha > 0.0$, it rapidly becomes unfair with it being one of the most unfair method at $\alpha = 1.0$. Interestingly, in Anxiety, `INLP` starts as one of the worst-performing mechanisms. However, with $\alpha > 0.0$, it quickly outperforms most approaches.

These findings shed light on the trade-offs and complexities inherent in optimizing fairness while maintaining worst-off group performance. It highlights the need for comprehensive evaluation metrics and the importance of considering the performance of both advantaged and disadvantaged groups in the fairness analysis. Finally, we emphasize that methods do not always exhibit leveling down. In settings without leveling down, DF adequately captures unfairness, producing values similar to α-Intersectional Fairness. However, every method displays some degree of leveling down for some combinations of datasets and metrics. A robust fairness measure should expose unfairness universally, which our experiments demonstrate IF_α achieves.

5.8 Conclusion

We propose a new definition for measuring intersectional fairness in the group classification setting. We provide various comparative analyses of our proposed measure, and contrast it with existing ones. Through them, we show that our fairness definition can uncover various notions of harm, including notably, the leveling down phenomenon. We further show that many fairness-inducing methods show no significant improvement over a simple unconstrained approach. Through this work, we provide tools to the community to better uncover latent vectors of harm. Further, our findings chart a path for developing new fairness-inducing approaches which optimizes for fairness without harming the groups involved.

5.9 Limitations

While appealing, α-Intersectional Fairness also has limitations. One of the primary ones is that it assumes a minimum number of examples for each subgroup to estimate the fairness level of the model correctly. Moreover, it does not consider the data drift over time, as it assumes a static view of the problem. Thus we recommend checking the fairness level over time to account for it. Further, in this definition, setting up α is left to the practitioner and thus can be abused. In the future, we aim to develop mechanisms to validate α without access to the dataset or model.

Finally, we want to emphasize that a hypothetical perfectly fair model might not be devoid of social harm. Firstly, vectors of harm of using statistical models are not restricted to existing definitions of group fairness. Further, if some socio-economic groups are not present in a given dataset, existing fairness-inducing approaches are likely to not have any positive impact towards them when encountered upon deployment. Such is the case with commonly used datasets in the community, which over-simplify gender and race as binary features, ignoring people of mixed heritage, or non-binary gender, for example. In our experiments, we too have used these datasets, owing to their prevalence, and we urge the community to create dataset with non-binary attributes. That said, our measure works with non-binary sensitive attributes, with no modifications.

Chapter 6

Synthetic Data Generation for Intersectional Fairness

Abstract

In this chapter, we introduce a data augmentation approach specifically tailored to enhance intersectional fairness in classification tasks. Our method capitalizes on the hierarchical structure inherent to intersectionality, by viewing groups as intersections of their broader parent categories. This perspective allows us to augment data for smaller groups by learning a transformation function that combines data from these parent groups. Our empirical analysis, conducted on four diverse datasets, reveals that classifiers trained with this data augmentation approach achieve superior fairness levels and is more robust to "leveling down" when compared to methods focused solely on optimizing traditional group fairness metrics.

The codebase for the chapter is available at - https://github.com/saist1993/BenchmarkingIntersectionalBias.

6.1 Introduction

In the previous chapter, we highlighted that many fairness promoting methods improve intersectional fairness by compromising the performance over the best-off and/or the worst-off groups. This tendency, where a mechanism achieves better fairness at the expense of the involved groups, is termed "leveling down." A plausible explanation for this phenomenon is the limited data availability for specific subgroups. For instance, in the Twitter Hate Speech Dataset (Huang et al., 2020), the most underrepresented group has a mere 300 examples, contrasting starkly with the largest group which has over 6,000 examples. This data disparity, we posit, mirrors real-world challenges where gathering data for specific groups can be notably difficult.

To tackle the issue of data scarcity, we introduce a data augmentation method that utilizes the hierarchical structure characteristic of intersectionality. More precisely, we conceptualize each group as a combination of its parent groups. Figure 6.1 illustrates this hierarchical structure for the Twitter Hate Speech Dataset, showing how the group ′African American Male under 45′ is composed of ′Male′, ′African American′, ′Male under 45′, and ′African American under 45′ groups. The figure further highlights the data scarcity challenge, showing that the number of samples often decreases sharply

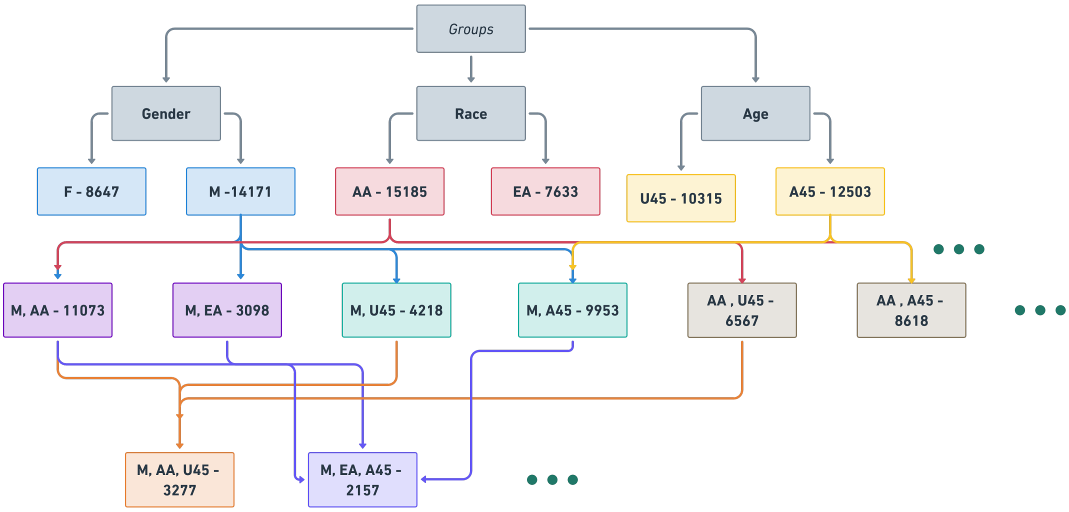


FIGURE 6.1: This figure illustrates a snippet of the hierarchical structure found in intersectional fairness along side group size for Twitter Hate Speech Dataset (Huang et al., 2020). In this context, 'F' and 'M' stand for Female and Male, respectively, while 'AA' and 'EA' represent African American and European American. Additionally, 'U45' and 'A45' denote age groups under 45 and above 45 years old, respectively. For instance, the group labeled 'M,AA,U45', represents African American men under 45 years old, and has parent groups identified as 'M,AA', 'M,U45', and 'AA,U45'. For each group, the number of examples is reported. The deeper we get in this hierarchical structure, the smaller the number of examples in each group.

as we consider more specific intersections. For example, the 'African American Male under 45' group has 3,277 instances, whereas the 'Male' group has 14,171 instances.

Our data generation mechanism tackles the problem of fairness at the data collection and preparation phase of the machine learning pipeline (See Figure 6.2). We hypothesizes that more specific groups can be augmented by modifying and combining the data from parent groups (which generally have more examples). For instance, data for the subgroup Female African American could be synthesized by combining and transforming examples from the Female and African American groups. To achieve this, we train a generative model optimizing a loss based on Maximum Mean Discrepancy (MMD) Gretton et al., 2012, which quantifies the difference between the generated and the original examples.

During training, we combine generated examples with original examples from the dataset, and additionally, as in previous chapters, we use equal sampling. The first step increases the diversity of examples the classifier is trained on, thereby improving generalization, while the latter ensures that equal importance is given to all subgroups instead of focusing more on larger groups. We empirically evaluate the quality and diversity of the generated examples and their impact on fairness and accuracy. Our results on various datasets show that our proposed approach consistently improves fairness, without harming the groups and at a minimal cost in accuracy.

The chapter is organized as follows. Section 6.2 provides background on MMD. Section 6.3 presents the setting and notations. Section 6.4 details our approach. Experimental results and conclusions are given in Sections 6.5 and 6.6.

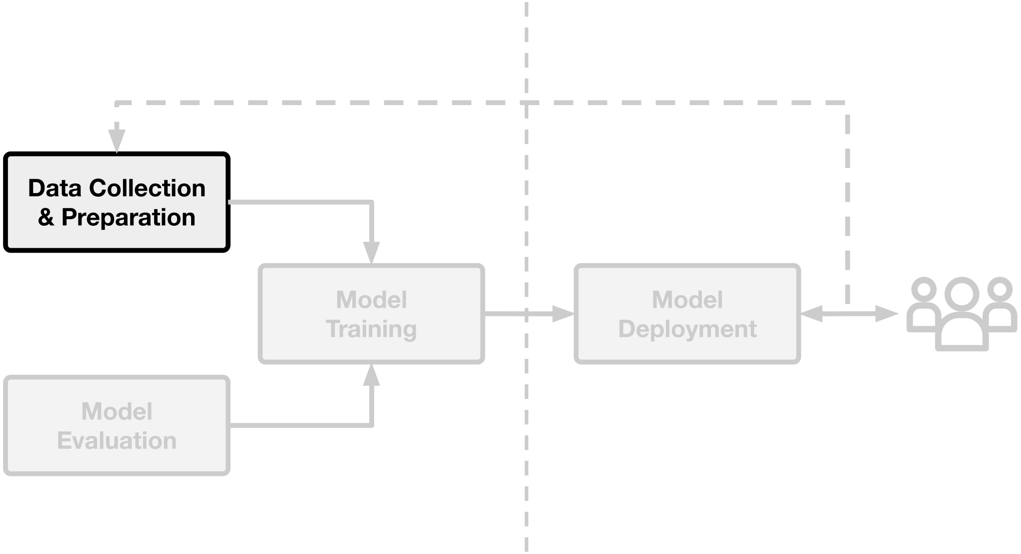


FIGURE 6.2: This chapter focuses on data pre-processing aspect of the machine learning pipeline.

6.2 Background: Maximum Mean Discrepancy

Maximum Mean Discrepancy (MMD) is a kernel-based divergence used to assess the similarity between distributions. In a nutshell, it involves identifying a function that, given two distributions $\mathcal{P}$ and $\mathcal{Q}$, yields larger values for samples drawn from $\mathcal{P}$ and smaller values for those from $\mathcal{Q}$. The difference in the mean value of this function for samples drawn from these two distributions provides an estimate of their similarity.

In this work, following the footsteps of Gretton et al. (2012), we use unit balls in characteristic reproducing kernel Hilbert spaces as the function class. Intuitively, the idea is to use the kernel trick to compute the differences in all moments of two distributions and then average the result. Formally, the MMD between two distributions $\mathcal{P}$ and $\mathcal{Q}$ is:

$$\begin{aligned} MMD^2(\mathcal{P},\mathcal{Q}) = \sup_{\|\Psi\|_H \leq 1} &|E_{Z\sim\mathcal{P}}[\Psi(Z)] \\ &- E_{Z'\sim\mathcal{Q}}[\Psi(Z')]| \\ = &E_{Z\sim\mathcal{P}}[k(Z,Z)] \\ &- 2E_{Z\sim\mathcal{P},Z'\sim\mathcal{Q}}[k(Z,Z')] \\ &+ E_{Z'\sim\mathcal{Q}}[k(Z',Z')] \end{aligned}$$

Here, k is the kernel derived from $\|\cdot\|_H$, the norm associated with corresponding Reproducing Kernel Hilbert Space H. In practice, we generally do not have access to true distributions but only samples, and thus the above equation is approximated as:

$$MMD^2(S_z, S_{z'}) = \frac{1}{m(m-1)} \sum_i \sum_{j\neq i} k(z_i, z_j)$$

$$\frac{1}{m(m-1)} \sum_i \sum_{j\neq i} k(z'_i, z'_j) +$$

$$\frac{1}{m(m)}\sum_i\sum_j k(z_i, z'_j)$$

where S_z (resp. $S_{z'}$) is a set of m samples drawn from $\mathcal{P}$ (resp. $\mathcal{Q}$). In this work, we use the radial basis function kernel $k : (z, z') \mapsto \exp(\|z - z'\|^2 / 2\sigma^2)$ where σ is the free parameter. In summary, MMD provides a simple and powerful way to compute the similarity between two distributions by using samples drawn from those distributions.

6.3 Problem Statement

This section introduces the notation used throughout the chapter and the problem statement.

Notations: In this chapter, we adopt and extend the notations proposed in the previous chapter. Let p denote the number of distinct *sensitive axes* of interest, which we denote as $\mathcal{A}_1, \ldots, \mathcal{A}_p$. Each of these sensitive axes is a set of discrete-valued *sensitive attributes*. For instance, a dataset may be composed of gender, race, and age as the three sensitive axes, and each of these sensitive axes may be encoded by a set of sensitive attributes, such as gender: {male, female}, race: {European American, African American}, and age: {under 45, above 45}.

Consider a feature space $\mathcal{X}$, a finite discrete label space $\mathcal{Y}$, and the sensitive axis space $\mathcal{A}_1 \cdots \mathcal{A}_p$ corresponding to sensitive axes as defined above. Let $\mathcal{D}$ be an unknown distribution over $\mathcal{X} \times \mathcal{Y} \times \mathcal{A}_1 \times \cdots \times \mathcal{A}_p$ which can be written as:

$$\mathcal{D} = P(X, Y, A_1, \cdots, A_p) \tag{6.1}$$

We define a *sensitive group* $\mathbf{g}$ as any p-dimensional vector in the Cartesian product set $\mathcal{G} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_p$ of these sensitive axes. For instance, a sensitive group $\mathbf{g} \in \mathcal{G}$ can be represented as $(a_1, \ldots, a_p)$ and the corresponding distribution is given by:

$$\begin{aligned}\mathcal{D}_{\mathbf{g}} &= P(X, Y, A_1 = a_1, \cdots, A_p = a_p)\\ &= P(X, Y, \mathbf{g})\end{aligned}$$

Additionally, we introduce a more general group than $\mathbf{g}$ called $\mathbf{g}^{\setminus i}$, referred to as the *parent group* where the i-th sensitive axis is not specified. It can be represented as $(a_1, \cdots, a_{i-1}, a_{i+1}, \cdots, a_p)$ where $i \in \{1, \ldots, p\}$. The distribution over such a group can be written as:

$$\begin{aligned}\mathcal{D}_{\mathbf{g}^{\setminus i}} = \sum_{a_i \in \mathcal{A}_i} & P(X, Y, A_1 = a_1,\\ & \cdots, A_i = a_i, \cdots, A_p = a_p)\end{aligned} \tag{6.2}$$

In our example above, if a group $\mathbf{g}$ is {male, European American, under 45}, then the corresponding parent groups are: ({male, European American}, {male, under 45}, {European American, under 45}).

Finally, in this work, we focus on classification problems and assume K distinct labels. We will denote the distribution of a group conditioned on same label k by $\mathcal{D}_{\mathbf{g}|Y=k}$.

Problem Statement: As standard in machine learning, $\mathcal{D}$ is generally unknown and instead we have access to a finite dataset $\mathcal{T} = \{(x_j, y_j, \mathbf{g}_j)\}_{j=1}^{n}$ consisting of n examples sample i.i.d from $\mathcal{D}$. This sample can be rewritten as $\mathcal{T} = \bigcup_{\mathbf{g}\in\mathcal{G}} \mathcal{T}_{\mathbf{g}}$ where $\mathcal{T}_{\mathbf{g}}$ represents the subset of examples from group $\mathbf{g}$. Examples belonging to parent group $\mathbf{g}^{\setminus i}$ are denoted by:

$$\mathcal{T}_{\mathbf{g}^{\setminus i}} = \bigcup_{a_i \in \mathcal{A}_i} \mathcal{T}_{a_1,\cdots,a_i,\cdots a_p} \tag{6.3}$$

The goal of fair machine learning is then to learn an accurate model $h_\theta \in \mathcal{H}$, with learnable parameters $\theta \in \mathbb{R}^D$, such that $h_\theta : \mathcal{X} \to \mathcal{Y}$ is fair with respect to a given group fairness definition like Equal Opportunity (Hardt, Price, and Srebro, 2016), Equal Odds (Hardt, Price, and Srebro, 2016), Accuracy Parity Zafar et al., 2017b, etc.

6.4 Approach

In this work, we introduce a novel approach for generating data that leverages the underlying structure of intersectional groups. We begin by the structural properties of interest, and then present our data generation mechanism. Note that in this work, we treat data as vectors, which allows us to encompass a range of modalities including images and text. In order to convert data into vector representations, we may use pre-trained encoders.

6.4.1 Structure of the data

Using the notations discussed in the previous section, we make the following simple but crucial observation the structure of the data:

$$\mathcal{T}_{\mathbf{g}} = \bigcap_{i=1}^{p} \mathcal{T}_{\mathbf{g}^{\setminus i}} \quad \text{and} \quad \mathcal{T}_{\mathbf{g}} \subset \mathcal{T}_{\mathbf{g}^{\setminus i}} \ \forall i \in \{1, \ldots, p\}.$$

In other words, the intersection of immediate parent groups constitutes the target group $\mathbf{g}$, with each parent group containing more examples than the target group itself. For example, all instances of the group Female African American are also part of both the Female and African American groups. Moreover, the common instances between the Female and African American groups collectively define the Female African American group.

6.4.2 Data Generation

Our goal is to learn a generative function $gen_{\theta,k}$ such that, given a dataset $\mathcal{T}$, a group $\mathbf{g}$, and task label k, the generated distribution $Z_{gen} \sim gen_{\theta,k}(\mathcal{T}, \mathbf{g})$ is similar to the true distribution $\mathcal{D}_{\mathbf{g}|Y=k}$. Based on the above observations, we propose to generate examples for group $\mathbf{g}$ by combining and transforming the examples from the corresponding parent groups. This can be achieved by appropriate parameterizations of $gen_{\theta,k}$ which we describe next.

Parameterization of the Generative Function: In this work, we explore the use of two simple choices for the generative function $gen_{\theta,k}(\mathcal{T}, \mathbf{g})$ that generates an example

$Z_{gen} = (X_{gen}, k, \mathbf{g})$ for a given group $\mathbf{g}$ and label k. The first parameterization is:

$$X_{gen} = \sum_{i=1}^{p} \lambda_i X_{\mathbf{g}^{\backslash i}} \tag{6.4}$$

where $Z_{\mathbf{g}^{\backslash i}} = (X_{\mathbf{g}^{\backslash i}}, k, \mathbf{g}^{\backslash i}) \sim \mathcal{D}_{\mathbf{g}^{\backslash i}|Y=k}$. In the above equation, $\lambda = (\lambda_1, \ldots, \lambda_p) \in \mathbb{R}^p$ are the parameters to optimize based on the loss we define below. In other words, we generate data for group $\mathbf{g}$ by forming weighted combinations of examples from its parent groups. A second parameterization that we consider is:

$$X_{gen} = \sum_{i=1}^{p} W \cdot X^T_{\mathbf{g}^{\backslash i}}, \tag{6.5}$$

where $\mathbf{W} \in \mathbb{R}^{d \times d}$ is a diagonal matrix with d parameters where d is the dimension of the encoded inputs. Here, we use a uniform combination of examples from parent groups, but learn weights for the different features of the representation.

Given the limited data available for many groups, we opt to share parameters across them instead of learning specific parameters for each group. This approach, combined with the relatively simple parameterizations of the generative function, serves to reduce the risk of overfitting (recall that in practice we have very limited data for many groups). However, we still learn a separate model for each label, i.e., $gen_{\theta,k}(\mathcal{T}, \mathbf{g})\ \forall k \in K$, to avoid the added complexity of jointly learning $\mathcal{X} \times \mathcal{Y}$.

Training the Generative Models: To train the generative model $gen_{\theta,k}$, we minimize the Maximum Mean Discrepancy between the generated samples and the samples from group $\mathbf{g}$. Additionally, we integrate into our objective the MMD between the generated samples and those from its parent groups. In our preliminary set of experiments, we found that this additional term brought more diversity in the generated examples. Consequently, the final loss function is formulated as follows:

$$\begin{aligned} L_{\mathbf{g},k}(\theta) = {} & MMD(S_{gen}, S_{\mathbf{g},k}) + \\ & \sum_{i=1}^{p} MMD(S_{gen}, S_{\mathbf{g}^{\backslash i},k}), \end{aligned} \tag{6.6}$$

where S_{gen} is a batch of examples generated from $gen_{\theta,k}$. $S_{\mathbf{g},k}$ and $S_{\mathbf{g}^{\backslash i},k}$ are batches of examples respectively drawn from $\mathcal{D}_{\mathbf{g}|Y=k}$ and $\mathcal{D}_{\mathbf{g}^{\backslash i}|Y=k}$. Since $\mathcal{D}_{\mathbf{g}|Y=k}$ and $\mathcal{D}_{\mathbf{g}^{\backslash i}|Y=k}$ are unknown, we approximate them with the empirical distribution by sampling with replacement from $\mathcal{T}_{\mathbf{g}|Y=k}$ and $\mathcal{T}_{\mathbf{g}^{\backslash i}|Y=k}$. Algorithm 4 details the precise training process to learn the generative function.

Training Classifiers on Augmented Data: After training the generative models $gen_{\theta,k}$, we use them to create additional training data. Specifically, for a group $\mathbf{g}$, we sample examples from its corresponding parent groups and pass these samples through the generative models as previously described. In this way, we can generate additional data for smaller groups that we use to augment the original training dataset, so as to enhance their representation in downstream tasks. As we will see in the next section, this helps to improve the fairness of the classifier.

Alternative formulations: An alternative approach to learn $gen_{\theta,k}$ involves using a generative adversarial network (GAN) (Goodfellow et al., 2014b). In this setup, the

Algorithm 4 Training the Generative Models

Input: Groups $\mathcal{G}$, Dataset $\mathcal{T}$, batch size b, number of iterations l and batch size b
Output: K trained generative models $\{gen_{\theta,k}\}_{k=1}^{K}$ capable of generating data for each label k

```
for _ in $l$ do
    Randomly sample a group $\mathbf{g}$ from $\mathcal{G}$
    for $k$ in $K$ do
        $S_{\mathbf{g},k} \leftarrow$ Sample $b$ examples from $\mathcal{T}_{\mathbf{g}|Y=k}$
        $S_{\mathbf{g}^{\backslash i},k} \leftarrow$ Sample $b$ examples from $\mathcal{T}_{\mathbf{g}^{\backslash i}|Y=k}$ $\forall i \in \{1, \ldots, p\}$
        $S_{gen} \leftarrow$ Sample $b$ examples from $gen_{\theta,k}(\mathcal{T}, \mathbf{g})$
        Compute the MMD loss using these examples as stated in Equation 6.6
        Backpropagate this loss to update the parameters of the model $gen_{\theta,k}$
    end for
end for
```

adversary aims to differentiate between two distributions, while the encoder strives to mislead the adversary. However, training GANs presents notable challenges (Thanh-Tung and Tran, 2020; Bau et al., 2019), including the risk of mode collapse, the complexity of nested optimization, and substantial computational demands. By contrast, MMD is more straightforward to implement and train, with significantly less computational burden. We also note that, while this work primarily employs MMD, our methodology can be adapted to work with other divergences between distributions, such as Sinkhorn Divergences and the Fisher-Rao Distance. We keep the exploration of other choices of divergences for future work.

6.5 Experiments

In this section, we present experiments designed to (i) assess the quality of the data generated by our approach, and (ii) examine the influence of this data on fairness with a focus on leveling down and performance of classifier over the worst-off group. We start by outlining the datasets, baselines, and fairness metrics employed in our experiments.

Datasets: Throughout this chapter, we experiment with four datasets, each differing in the number of examples, sensitive groups, and modality. These datasets are: (i) *Twitter Hate Speech* (Huang et al., 2020) – a collection of tweets annotated based on 4 demographic attributes, or sensitive axes, namely age, race, gender, and country; (ii) *CelebA* (Liu et al., 2015) – composed of human face images annotated with various attributes; (iii) *Numeracy* (Abbasi et al., 2021) – a compilation of free text responses that denote the numerical comprehension capabilities of individuals; and (iv) *Anxiety* (Abbasi et al., 2021) – a dataset indicative of a patient's anxiety levels. These are the same datasets used in the previous chapter. Furthermore, we rely on the same setup, pre-processing, and splits as in the preceding chapter.

Methods: We benchmark against the same methods as the ones in the previous chapter. More specifically, we experiment with: (i) `Unconstrained` which solely optimizes model accuracy and is oblivious to any fairness measure; (ii) `Adversarial` which adds an adversary (Li, Baldwin, and Cohn, 2018) to unconstrained, implementing standard adversarial learning approach (Li, Baldwin, and Cohn, 2018); (iii)

FairGrad (Maheshwari and Perrot, 2022),is an in-processing iterative approach as described in Chapter 4; (iv) INLP (Ravfogel et al., 2020) is a post-processing approach that iteratively trains a classifier and then projects the representation on the classifier's null space; (v) Fair MixUp (Chuang and Mroueh, 2021) enforces fairness by forcing the model to have similar predictions on the paths of interpolated samples between the sensitive groups; and (vi) Unconstrained + Augmented which is same as Unconstrained, but trained on the data generated via our proposed data generation mechanism.

In these experiments we employ the same non-linear architecture as described in Chapter 4 and 5. Specifically, we use a three-hidden layer fully connected neural network with 128, 64, and 32 corresponding sizes. Furthermore, we use ReLU as the activation with dropout fixed to 0.5. We optimize cross-entropy loss in all cases with Adam (Kingma and Ba, 2015) as the optimizer using default parameters. Finally, for text-based datasets we encode the text using bert-base-uncased Devlin et al., 2019 and for images we employ a pre-trained ResNet18[1] (He et al., 2016b). As in the previous chapter, we use equal sampling, where we sample equal number of examples for each group. We keep the number of examples as the hyperparameter ranging from 100 to 5000 indicating a continuous scale between undersampling regime (where we under sample from each group) and oversampling.

In order to generate data for Unconstrained + Augmented, we employ the generative function as described in Section 6.4.2. More specifically, our initial experiments suggest that employing a simpler model with fewer parameters (Equation 6.4) for the positive class, and a more complex model with a larger number of parameters for the negative class (Equation 6.5), leads to an enhanced fairness-accuracy trade-off, when using the False Positive rate as a measure of fairness. Consequently, for the positive class, we implement the function detailed in Equation 6.4, and for the negative class, we apply the model specified in Equation 6.5.

Fairness Metrics: In this chapter we employ IF_α introduced in the previous chapter, as well as Differential Fairness (DF), as fairness definitions. For the performance measure m associated with these definitions, we focus on False Positive Rate. Formally, for a group $\mathbf{g}$, m is given by:

$$m(h_\theta, \mathcal{T}_\mathbf{g}) = 1 - P(h_\theta(x) = 0 | (x,y) \in \mathcal{T}_\mathbf{g}, y = 1)$$

To estimate the empirical probabilities, we employ the bootstrap estimation procedure as proposed by Morina et al. (2019). Like in the previous chapter, we generate 1000 datasets by sampling from the original dataset with replacement. We then estimate the probabilities on this dataset using a smoothed empirical estimation mechanism and then average the results over all the sampled datasets.

Utility metric: In order to evaluate the utility of various methods, we employ balanced accuracy.

6.5.1 Quality of Generated Data

In this experiment, we assess both the quality and diversity of the generated data. Specifically, our goal is to generate data that resembles the overall distribution of

[1] https://pytorch.org/vision/stable/models.html

Dataset	Gen-Real	Real-Real
CelebA	0.46 ± 0.014	0.48 ± 0.00
Numeracy	0.51 ± 0.01	0.58 ± 0.01
Anxiety	0.51 ± 0.00	0.59 ± 0.02
Twitter Hate Speech	0.47 ± 0.01	0.53 ± 0.01

TABLE 6.1: Analyzing the similarity of a generated sample with existing sample.

real data, while ensuring the generated examples remain distinct from the original samples. To achieve this, we propose two evaluations:

- **Diversity:** To gauge the diversity of the generated dataset, for each generated example, we identify the most similar example in the real dataset. If the generated sample closely resembles a real one, the distance between the generated and real examples will be substantially smaller than between distinct real examples.
- **Distinguishability:** To assess the difference between the generated and real datasets, we train a classifier to differentiate between them. If the classifier's accuracy approaches that of a random guess, it suggests the empirical distributions of the generated and real data are analogous.

In both experiments, we report metrics based on the entire dataset rather than computing averages for individual groups and then aggregating these averages.

Diversity

In this experiment, we use cosine similarity as a measure of closeness, which for two vectors **a** and **b** is defined as:

$$\text{cosine\_similarity}(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \, \|\mathbf{b}\|}. \tag{6.7}$$

We generate 1000 examples and randomly sample an equal number from the real dataset. For each of these real examples, we identify its closest counterpart within the real dataset to establish a baseline (Real-Real). Subsequently, for each generated example, we determine its nearest match in the real dataset (Gen-Real). The results of this experiment can be found in Table 6.1.

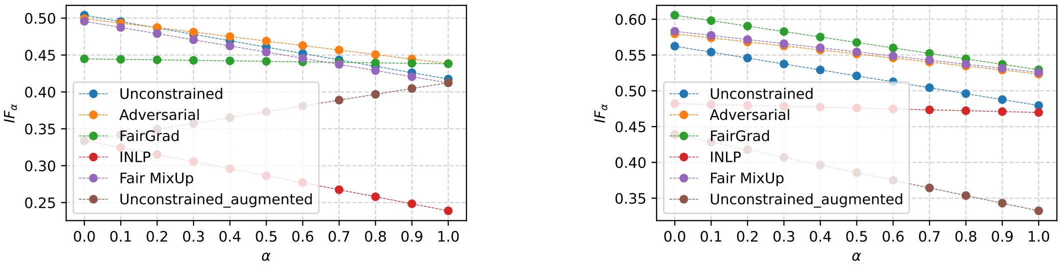
For all datasets, the distance between the generated and real examples is comparable to that between two real examples. In every dataset, the Gen-Real closeness is less than the Real-Real proximity. Based on these results, we conclude that the generated examples are not mere replicas of the real samples.

Distinguishability

In this study, we frame distinguishability as a binary classification task where we train a two-layer MLP classifier aimed at distinguishing between real and generated samples. As in prior experiments, we compile a dataset by selecting 1000 real instances and generating an equivalent number of samples. This dataset is subsequently partitioned into training and evaluation subsets with a ratio of 80% to 20%.

Dataset	Accuracy
CelebA	0.52 ± 0.011
Numeracy	0.64 ± 0.012
Anxiety	0.64 ± 0.019
Twitter Hate Speech	0.57 ± 0.022

TABLE 6.2: Accuracy of a classifier to distinguish between real and generated sample over various datasets. The value of 0.5 represents a random classifier, while 1.0 is a perfect classifier.



(A) Twitter Hate Speech FPR

(B) Anxiety FPR

FIGURE 6.3: Value of IF_α on the test set of Twitter Hate Speech, and Numeracy datasets for varying $\alpha \in [0, 1]$.

Results are presented in Table 6.2. The mean accuracy of the classifier is approximately 0.59, suggesting that the generated samples have a distribution similar, but not identical to, the real instances. In our preliminary experiments we found that by modulating the generator complexity (i.e by employing more complex and higher parameter models), we could achieve near-random distinguishability. However, such adjustments led to an unfavorable fairness-accuracy trade-off. We conjecture this may arise because near-random indistinguishability in the generated samples causes them to inherit biases from the real data.

6.5.2 Fairness-Accuracy Trade-offs

In this experiment, we explore the impact of generated data on the fairness-privacy trade-off. To that end, we train `Unconstrained` just over the generated data which we refer to as `Unconstrained + Augmented` in this chapter. Specifically, we delve into the leveling down phenomenon as detailed in the preceding chapter. To recap, a method is considered to exhibit leveling down if its performance for the worst-off or best-off group is inferior to that of an unconstrained model.

The outcomes of this experiment are presented in Table 6.3. Detailed results for CelebA and Numeracy, both of which display a similar trend, are provided in the Appendix D. In terms of accuracy, `Unconstrained + Augmented` exhibits a slight drop for the Anxiety dataset. However, its accuracy is on par with the Unconstrained model when evaluated on Twitter Hate Speech. In terms of performance for both best-off and worst-off groups, `Unconstrained + Augmented` outperforms competing methods. Notably, `Unconstrained + Augmented` does not show any signs of leveling down across all datasets. When assessing IF_α, `Unconstrained + Augmented` consistently achieves the best fairness results among the datasets. We also plot the complete trade-off between relative and absolute performance of groups by varying α in Figure 6.3.

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$IF\alpha = 0.5$ ↓
`Unconstrained`	0.63 + 0.01	0.25 + 0.02	0.51 + 0.03	0.43 +/- 0.09	0.52 +/- 0.03
`Adversarial`	0.63 + 0.01	0.27 + 0.06	0.55 + 0.12	0.48 +/- 0.05	0.55 +/- 0.04
`FairGrad`	0.63 + 0.01	0.29 + 0.05	0.56 + 0.12	0.48 +/- 0.07	0.57 +/- 0.04
`INLP`	0.63 + 0.01	0.22 + 0.02	0.49 + 0.03	0.42 +/- 0.07	0.48 +/- 0.03
`Fair MixUp`	0.61 + 0.01	0.28 + 0.02	0.55 + 0.06	0.47 +/- 0.09	0.55 +/- 0.02
`Unconstrained + Augmented`	0.6 + 0.0	0.13 + 0.08	0.35 + 0.12	0.29 +/- 0.32	0.39 +/- 0.11

(A) Results on Anxiety

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$IF\alpha = 0.5$ ↓
`Unconstrained`	0.81 + 0.0	0.18 + 0.01	0.46 + 0.01	0.42 +/- 0.05	0.46 +/- 0.02
`Adversarial`	0.79 + 0.01	0.18 + 0.01	0.48 + 0.04	0.46 +/- 0.08	0.47 +/- 0.02
`FairGrad`	0.8 + 0.0	0.17 + 0.01	0.49 + 0.03	0.49 +/- 0.1	0.44 +/- 0.02
`INLP`	0.66 + 0.0	0.08 + 0.02	0.26 + 0.02	0.22 +/- 0.25	0.29 +/- 0.04
`Fair MixUp`	0.81 + 0.01	0.18 + 0.02	0.46 + 0.02	0.42 +/- 0.09	0.45 +/- 0.04
`Unconstrained + Augmented`	0.81 + 0.0	0.12 + 0.02	0.44 + 0.02	0.45 +/- 0.18	0.37 +/- 0.04

(B) Results on Twitter Hate Speech

TABLE 6.3: Test results on (a) *Anxiety*, and (b) *Twitter Hate Speech* using False Positive Rate. We select hyper parameters based on $IF\alpha = 0.5$ value. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness $IF\alpha = 0.5$. For both fairness definitions, lower is better, while for balanced accuracy, higher is better. Best Off and Worst Off represent the min FPR and max FPR across groups (in both cases, lower is better). Results have been averaged over 5 different runs.

For the Anxiety dataset, `Unconstrained + Augmented` gives the best trade-off for every value of α. In the case of Twitter Hate Speech, `INLP` achieves superior results. However, it is worth noting that `INLP`'s accuracy is 14 points below `Unconstrained + Augmented`.

6.5.3 Impact of Intersectionality

In this experiment, we examine the influence of intersectionality on our approach and its effect on worst-case performance. To achieve this, we iteratively introduce more sensitive axes and plot the fairness-accuracy trade-off. For example, akin to the experiment in the preceding chapter using CelebA, we initially introduced gender (selected at random) as a single sensitive axis. In the subsequent step, we incorporated race (also selected randomly) alongside the previously added gender. Similarly, we then added age, and finally country.

The results of this experiment can be found in Figure 6.4. With fewer groups (2 sensitive axes), the model's performance on the generated dataset closely matches that on the real dataset. However, as the number of axes increases, the performance difference becomes more pronounced. Furthermore, we find that the performance of the model remains relatively stable despite the increase in sensitive axes, further underscoring the effectiveness of our proposed approach.

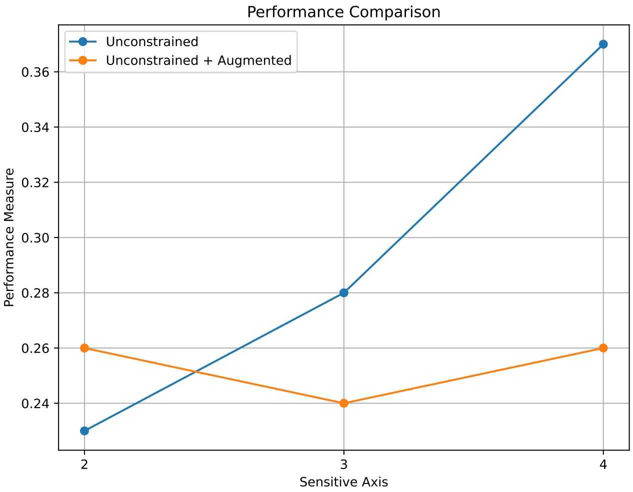


FIGURE 6.4: FPR for the worst-off group on the test data of *CelebA* (the lower, the better) by varying the number of sensitive axes.

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$IF\alpha = 0.5$ ↓
`Unconstrained + Augmented`	0.6 + 0.0	0.13 + 0.08	0.35 + 0.12	0.29 +/- 0.32	0.39 +/- 0.11
`Unconstrained + Augmented` Parent-of-Parent	0.59 + 0.01	0.16 + 0.08	0.40 + 0.15	0.34 +/- 0.32	0.43 +/- 0.08

(A) Results on Anxiety

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	$IF\alpha = 0.5$ ↓
`Unconstrained + Augmented`	0.69 + 0.02	0.14 + 0.05	0.39 + 0.11	0.34 +/- 0.24	0.44 +/- 0.07
`Unconstrained + Augmented` Parent-of-Parent	0.69 + 0.01	0.17 + 0.08	0.44 + 0.15	0.39 +/- 0.32	0.44 +/- 0.11

(B) Results on Numeracy

TABLE 6.4: Test results on (a) *Anxiety*, and (b) *Twitter Hate Speech* using False Positive Rate while optimizing for DF. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness $IF\alpha = 0.5$. For both fairness definition, lower is better, while for balanced accuracy, higher is better. The Best Off and Worst Off, in both cases lower is better, represents the min FPR and max FPR. Results have been averaged over 5 different runs.

6.5.4 Impact of Abstract Groups

Until now in this chapter, we have focused on generating data for a group by combining and manipulating data from the immediate parents of the group. However, this notion can be further extended to get parents of parents for a given group. For example, for group **g** defined as {male, European American, under 45}. The immediate parent groups are: ({male, European American}, {male, under 45}, {European American, under 45}), while the parents of these parent groups are ({male}, {European American}, {under 45}). In this experiment, we explore the impact of generating data from the parents of immediate parents, as opposed to solely from the immediate parent set.

Table 6.4 presents the results of this experiment. We observe that training on the parents of parent groups neither enhances the accuracy nor the fairness of the classifier. Furthermore, the performance on the Anxiety closely resembles that of an unconstrained model. We hypothesize that this occurs because considering more abstract groups approximates a scenario where no groups are considered, which is similar to an unconstrained setting.

6.6 Conclusion

In this chapter, we introduce a data augmentation mechanism that leverages the hierarchical structure inherent to intersectional settings. Our extensive experiments demonstrate that this method not only generates diverse data but also enhances the classifier's performance across both the best-off and worst-off groups. In the future, we plan to extend our approach to a broader range of performance metrics, delve into zero-shot fairness, and explore more sophisticated sampling mechanisms.

Chapter 7

Fair NLP Models with Differentially Private Text Encoders

Abstract

Encoded text representations often capture sensitive attributes about individuals (e.g., race or gender), which raise privacy concerns and can make downstream models unfair to certain groups. In this chapter, we propose FEDERATE, an approach that combines ideas from differential privacy and adversarial training to learn private text representations which also induces fairer models. We empirically evaluate the trade-off between the privacy of the representations and the fairness and accuracy of the downstream model on four NLP datasets. Our results show that FEDERATE consistently improves upon previous methods, and thus suggest that privacy and fairness can positively reinforce each other.

This chapter is based on the article - Gaurav Maheshwari, Pascal Denis, Mikaela Keller, and Aurélien Bellet. 2022. Fair NLP Models with Differentially Private Text Encoders. In Findings of the Association for Computational Linguistics: EMNLP 2022, pages 6913–6930, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. The codebase for the chapter is available at - https://github.com/saist1993/DPNLP.

7.1 Introduction

In this thesis, thus far, we have primarily focused on issues of fairness in machine learning models. However, these systems have also been shown to leak sensitive information about the data of individuals used for training or inference and thus pose privacy risks (Shokri et al., 2017). Societal pressure as well as recent regulations push for enforcing both privacy and fairness in real-world deployments, which is challenging as these notions are multi-faceted concepts that need to be tailored to the context. Moreover, privacy and fairness can be at odds with one another: recent studies have shown that preventing a model from leaking information about its training data negatively impacts the fairness of the model and vice versa (Bagdasaryan, Poursaeed, and Shmatikov, 2019; Pujol et al., 2020; Cummings et al., 2019; Chang and Shokri, 2020).

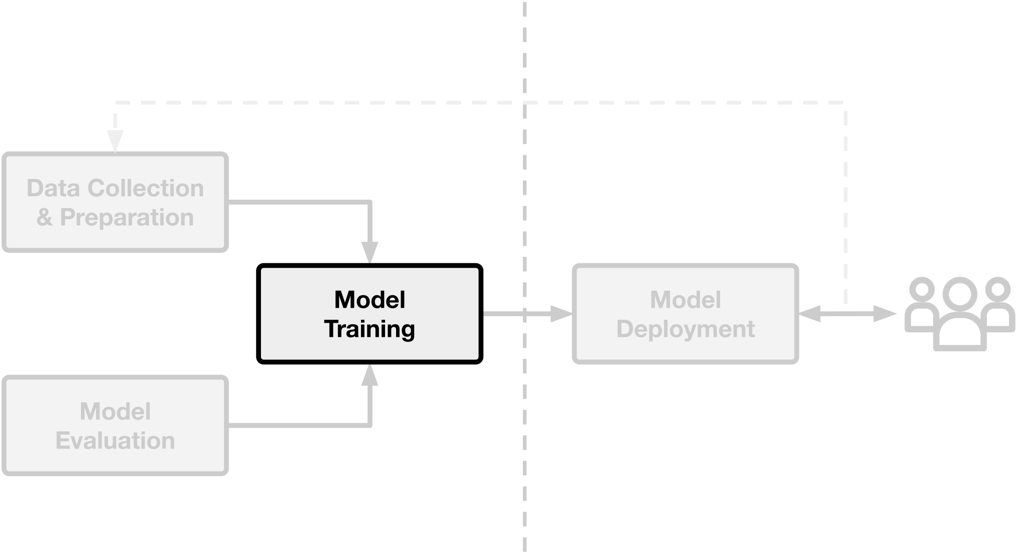


FIGURE 7.1: FEDERATE focuses on model training aspect of the machine learning pipeline.

In this chapter we study fairness and privacy and their interplay in the NLP context during model training phase (See Figure 7.1), where these two notions have often been considered independently from one another. Modern NLP heavily relies on learning or fine-tuning encoded representations of text. Unfortunately, such representations often leak sensitive attributes (e.g., gender, race, or age) present explicitly or implicitly in the input text, even when such attributes are known to be irrelevant to the task Song and Raghunathan, 2020. Moreover, the presence of such information in the representations may lead to unfair downstream models, as has been shown on various NLP tasks such as occupation prediction from text bios De-Arteaga et al., 2019, coreference resolution Zhao et al., 2018, or sentiment analysis Kiritchenko and Mohammad, 2018.

Privatizing encoded representations is thus an important, yet challenging problem for which existing approaches based on subspace projection (Bolukbasi et al., 2016; Wang et al., 2020; Karve, Ungar, and Sedoc, 2019; Ravfogel et al., 2020) or adversarial learning (Li, Baldwin, and Cohn, 2018; Coavoux, Narayan, and Cohen, 2018; Han, Baldwin, and Cohn, 2021) do not provide a satisfactory solution. In particular, these methods lack any formal privacy guarantee, and it has been shown that an adversary can still recover sensitive attributes from the resulting representations with high accuracy Elazar and Goldberg, 2018; Gonen and Goldberg, 2019.

Instead of relying on adversarial learning to prevent attribute leakage, Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021) recently propose to add random noise to text representations so as to satisfy differential privacy (DP), a mathematical definition which comes with rigorous guarantees (Dwork et al., 2006). However, we uncover a critical error in their privacy analysis which drastically weakens their privacy claims. Moreover, their approach harms accuracy and fairness compared to adversarial learning.

To circumvent these issues, we propose a novel approach (called FEDERATE) to learn private text representations and fair models by combining ideas from DP with an adversarial training mechanism. More specifically, we propose a flexible end-to-end architecture in which (i) the output of an arbitrary text encoder is normalized and perturbed using random noise to make the resulting encoder differentially private, and (ii) on top of the encoder, we combine a classifier branch with an adversarial branch to actively induce fairness, improve accuracy and further hide specific sensitive attributes. Like Chapter 4, `FEDERATE` is an in-processing fairness promoting method which focuses on the model training aspect of the machine learning pipeline.

We empirically evaluate the privacy-fairness-accuracy trade-offs achieved by our proposed mechanism over four datasets and find that it simultaneously leads to more private representations and fairer models than state-of-the-art methods while maintaining comparable accuracy. Beyond the superiority of our approach, our results bring valuable insights on the complementarity of DP and adversarial learning and the compatibility of privacy and fairness. On the one hand, DP drastically reduces undesired leakage from adversarially trained representations, and has a stabilizing effect on the training dynamics of adversarial learning. On the other hand, adversarial learning improves the accuracy and fairness of models trained over DP text representations.

Our main contributions are as follows:

- We propose a new approach, `FEDERATE`, which combines a DP encoder with adversarial learning to learn fair and accurate models from private representations.
- We identify and fix (with a formal proof) a critical mistake in the privacy analysis of previous work on learning DP text representations.
- We empirically show that `FEDERATE` leads to more private representations and fairer models than state-of-the-art methods while maintaining comparable accuracy.
- Unlike previous studies, our empirical results suggest that privacy and fairness are compatible in our setting, and even mutually reinforce each other.

7.2 Background: Differential Privacy

Differential Privacy (DP) (Dwork et al., 2006) provides a rigorous mathematical definition of the privacy leakage associated with an algorithm. It does not depend on assumptions about the attacker's capabilities and comes with a powerful algorithmic framework. For these reasons, it has become a de-facto standard in privacy currently used by the US Census Bureau Abowd, 2018 and several big tech companies (Erlingsson, Pihur, and Korolova, 2014; Fanti, Pihur, and Erlingsson, 2016; Ding, Kulkarni, and Yekhanin, 2017). This section gives a brief overview of DP, focusing on the aspects needed to understand our approach (see Dwork and Roth (2014) for an in-depth review of DP).

Over the last few years, two main models for DP have emerged:

- Central DP (CDP) Dwork et al., 2006, where raw user data is collected and processed by a trusted curator, which then releases the result of the computation to a third party or the public.
- Local DP (LDP) Kasiviswanathan et al., 2011 which removes the need for a trusted curator by having each user locally perturb their data before sharing it.

Our work aims to create an encoder that leads to a private embedding of an input text, which can then be shared with an untrusted curator for learning or inference. We thus consider LDP, defined as follows.

Definition 6 (Local Differential Privacy)**.** A randomized algorithm $M : X \to O$ is ϵ-differentially private if for all pairs of inputs $x, x' \in X$ and all possible outputs $o \in O$:

$$\Pr[M(x) = o] \leq e^{\epsilon} \Pr[M(x') = o]. \tag{7.1}$$

LDP ensures that the probability of observing a particular output o of M should not depend too much on whether the input is x or x'. The strength of privacy is controlled by ϵ, which bounds the log-ratio of these probabilities for any x, x'. Setting $\epsilon = 0$ corresponds to perfect privacy, while $\epsilon \to \infty$ does not provide any privacy guarantees (as one may be able to uniquely associate an observed output to a particular input). In our approach described in Section 7.3, x will be an input text and M will be an encoding function which transforms x into a private vector representation that can be safely shared with untrusted parties.

Laplace mechanism. As clearly seen from Definition 6, an algorithm needs to be randomized to satisfy DP. A classical approach to achieve ϵ-DP for vector data is the Laplace mechanism Dwork et al., 2006. Given the desired privacy guarantee ϵ and an input vector $\mathbf{x} \in \mathbb{R}^D$, this mechanism adds centered Laplace noise $\text{Lap}(\frac{\Delta}{\epsilon})$ independently to each dimension of $\mathbf{x}$. The noise scale $\frac{\Delta}{\epsilon}$ is calibrated to ϵ and the *L1-sensitivity* Δ of inputs:

$$\Delta = \max_{\mathbf{x},\mathbf{x}' \in X} \|\mathbf{x} - \mathbf{x}'\|_1. \tag{7.2}$$

In this chapter, we propose an architecture in which the Laplace mechanism is applied on top of a trainable encoder to get private representations of input texts, and is further combined with adversarial training to learn fair models.

7.3 Approach

We consider a scenario similar to Coavoux, Narayan, and Cohen (2018), where a user locally encodes its input data (text) x into an intermediate representation $E_{priv}(x)$ which is then shared with an untrusted curator to predict the label y associated with x using a classifier C. Additionally, an attacker (which may be the untrusted curator or an eavesdropper) may observe the intermediate representation $E_{priv}(x)$ and try to infer some sensitive (discrete) attribute z about x (e.g., gender, race etc.). Our goal is to learn an encoder E_{priv} and classifier C such that (i) the attacker performs poorly at inferring z from $E_{priv}(x)$, (ii) the classifier $C(E_{priv}(x))$ is fair with respect to z according to some fairness definition, and (iii) C accurately predicts the label y.

To achieve the above goals we introduce FEDERATE (Fair modEls with DiffERentiAlly private Text Encoders), which combines two components: a differentially private encoder and an adversarial branch. Figure 7.2 shows an overview of our proposed architecture.

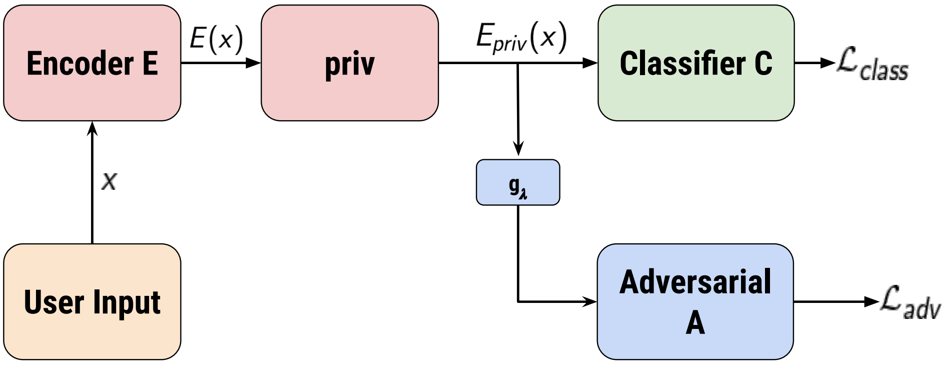


FIGURE 7.2: Overview of our FEDERATE approach. The text input x is transformed to $E(x) \in \mathbb{R}^D$ by the text encoder E. The encoded input is then made private by the privacy layer *priv*, which involves normalization and addition of Laplace noise. The resulting private representation $E_{priv}(x) \in \mathbb{R}^D$ is then used by the main task classifier C. It also serves as input to the adversarial layer A which is connected to the main branch via a radient reversal layer g_λ. The light red boxes represent the Differentially Private Encoder (Sec. 7.3.1), and the light blue boxes represent the Adversarial component (Sec. 7.3.2).

7.3.1 Differentially Private Encoder

We propose a generic private encoder construction $E_{priv} = priv \circ E$ composed of two main components. The first component E can be any encoder which maps the text input to some vector space of dimension D. It can be a pre-trained language model along with a few trainable layers, or it can be trained from scratch. The second component *priv* is a randomized mapping which transforms the encoded input to a differentially private representation. Given the desired privacy guarantee $\epsilon > 0$, this mapping is obtained by applying the Laplace mechanism (see Section 7.2) to a normalized version of the encoded representation $E(x)$:

$$priv(E(x)) = E(x)/\|E(x)\|_1 + \boldsymbol{\ell}, \tag{7.3}$$

where each entry of $\boldsymbol{\ell} \in \mathbb{R}^D$ is sampled independently from $\mathrm{Lap}(\frac{2}{\epsilon})$. We will prove that $E_{priv} = priv \circ E$ satisfies ϵ-DP in Section 7.3.4.

7.3.2 Adversarial Component

To improve the fairness of the downstream classifier C, we model the adversary by another classifier A which aims to predict z from the privately encoded input $E_{priv}(x)$. The encoder E_{priv} is optimized to fool A while maximizing the accuracy of the downstream classifier C. Specifically, given $\lambda > 0$, we train E_{priv}, C and A (parameterized by θ_E, θ_C, and θ_A respectively) to optimize the following objective:

$$\min_{\theta_E,\theta_C} \max_{\theta_A} \mathcal{L}_{class}(\theta_E, \theta_C) - \lambda \mathcal{L}_{adv}(\theta_E, \theta_A), \tag{7.4}$$

where $\mathcal{L}_{class}(\theta_E, \theta_C)$ is the cross-entropy loss for the $C \circ E_{priv}$ branch and $\mathcal{L}_{adv}(\theta_E, \theta_A)$ is the cross-entropy loss for the $A \circ E_{priv}$ branch. For an in-depth introduction to Adversarial Learning, please refer to Chapter 2.

Algorithm 5 Training procedure of FEDERATE (one epoch).

Input: Model architecture composed of encoder E (parameterized by θ_E), classifier C (parameterized by θ_C), adversary A (parameterized by θ_A), loss function L.
Output: Trained model.
Data: Samples S=$\{x^i, y^i, z^i\}_{i=1}^m$ where x^i is the input text, y^i is the task label, and z^i is the sensitive attribute.

```
1: for $i \leftarrow 0$ to $m$ do
2:     Encode: $\mathbf{x}^i \leftarrow E(x^i)$
3:     Normalize: $\mathbf{x}^i \leftarrow \frac{\mathbf{x}^i}{\|\mathbf{x}^i\|_1}$
4:     Privatize: $\mathbf{x}^i_{priv} \leftarrow \mathbf{x}^i + \boldsymbol{\ell}$, where each entry of the vector $\boldsymbol{\ell} \in \mathbb{R}^D$ is sampled
       independently from a centered Laplace distribution with scale $\frac{2}{\epsilon}$
5:     Adversarial prediction: $\hat{z}^i \leftarrow A(\mathbf{x}^i_{priv})$
6:     Update $\theta_A$ by backpropagating the loss $L(z^i, \hat{z}^i)$
7:     Task classification: $\hat{y}^i \leftarrow C(\mathbf{x}^i_{priv})$
8:     Update $\theta_E$ and $\theta_C$ by backpropagating the loss $L(y^i, \hat{y}^i) - \lambda \cdot L(z^i, \hat{z}^i)$
9: end for
```

7.3.3 Training

We train the private encoder E_{priv} and the classifier C from a set of public tuples (x, y, z) by optimizing (7.4) with backpropagation using a gradient reversal layer g_λ Ganin and Lempitsky, 2015. The latter acts like an identity function in the forward pass but scales the gradients passed through it by $-\lambda$ in the backward pass. This results in E_{priv} receiving opposite gradients to A. The pseudo-code of the training procedure of FEDERATE in Algorithm 5. Note that the combination of Steps 2-3-4 corresponds to E_{priv} as defined in Sec. 7.3.

7.3.4 Privacy Analysis

We show the following privacy guarantee.

Theorem 1. Our encoder E_{priv} and the downstream predictions $C \circ E_{priv}$ satisfy ϵ-DP.

Proof. We start by proving that our noisy encoder $E_{priv} : X \to \mathbb{R}^D$ satisfies ϵ-DP. Recall that for any input text $x \in X$

$$E_{priv}(x) = priv \circ E(x) = E(x)/\|E(x)\|_1 + \boldsymbol{\ell},$$

where each entry of $\boldsymbol{\ell} \in \mathbb{R}^D$ is sampled independently from Lap($\frac{2}{\epsilon}$), the centered Laplace distribution with scale $2/\epsilon$. Let $\tilde{E}(x) = E(x)/\|E(x)\|_1$. The L1 sensitivity of $\tilde{E}$ is

$$\Delta_{\tilde{E}} = \max_{x, x' \in X} \|\tilde{E}(x) - \tilde{E}(x)'\|_1.$$

Since for any $x \in X$ we have $\|\tilde{E}(x)\|_1 = 1$, the triangle inequality gives $\Delta_{\tilde{E}} \leq 2$. The ϵ-DP guarantee then follows from the application of the Laplace mechanism Dwork et al., 2006. Formally, let

$$p(y) = \frac{\epsilon}{4} e^{-\frac{|y|\epsilon}{2}}$$

denote the p.d.f. of Lap$(2/\epsilon)$. Consider two arbitrary input texts $x, x' \in X$ and let $\tilde{\mathbf{x}} = \tilde{E}(x) \in \mathbb{R}^D$ and $\tilde{\mathbf{x}}' = \tilde{E}(x') \in \mathbb{R}^D$ be their normalized encoded representations.

Then, for any possible encoded output $\mathbf{e} = (e_1, \ldots, e_D) \in \mathbb{R}^D$, we have:

$$\begin{align}
\frac{\Pr[E_{priv}(x) = \mathbf{e}]}{\Pr[E_{priv}(x') = \mathbf{e}]} &= \prod_{d=1}^{D} \frac{p(e_d - \tilde{x}_d)}{p(e_d - \tilde{x}_d{}')} \tag{7.5}\\
&= \prod_{d=1}^{D} \frac{e^{-\frac{\epsilon}{2}|e_d - \tilde{x}_d|}}{e^{-\frac{\epsilon}{2}|e_d - \tilde{x}'_d|}} \nonumber\\
&= e^{\frac{\epsilon}{2}\sum_{d=1}^{D} |e_d - \tilde{x}'_d| - |e_d - \tilde{x}_d|} \nonumber\\
&\leq e^{\frac{\epsilon}{2}\sum_{d=1}^{D} |\tilde{x}_d - \tilde{x}'_d|} \tag{7.6}\\
&= e^{\frac{\epsilon}{2}\|\tilde{\mathbf{x}} - \tilde{\mathbf{x}}'\|_1} \nonumber\\
&\leq e^{\frac{\epsilon}{2}\Delta_{\tilde{E}}} = e^{\epsilon}, \tag{7.7}
\end{align}$$

where (7.5) follows from the independence of the noise across dimensions, (7.6) uses the triangle inequality, and (7.7) from the definition of $\Delta_{\tilde{E}}$ and the fact that $\Delta_{\tilde{E}} \leq 2$ as shown above.

□

The above inequality shows that E_{priv} satisfies ϵ-DP as per Definition 6. The fact that $C \circ E_{priv}$ also satisfies ϵ-DP follows from the post-processing property of DP, which ensures that the composition of any function with an ϵ-DP algorithm also satisfies ϵ-DP Dwork and Roth, 2014.

Error in previous work. We found a critical error in the privacy analysis of previous work on differential private text encoders (Lyu, He, and Li, 2020; Plant, Gkatzia, and Giuffrida, 2021). In a nutshell, they incorrectly state that normalizing each entry of the encoded representation in $[0, 1]$ allows to bound the sensitivity of their representation by 1, while it can in fact be as large as D (the dimension of the representation). As a result, the privacy guarantees are dramatically weaker than what the authors claim: the ϵ values they report should be multiplied by D. In contrast, the L1 normalization we use in (7.3) ensures that the sensitivity of E is bounded by 2. We provide more details in Appendix B.1.

Interestingly, Habernal (2021) recently identified an error in ADePT Krishna, Gupta, and Dupuy, 2021, a differentially private auto-encoder for text rewriting. However, the error in ADePT is different from the one in Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021): the problem with ADePT is that it calibrates the noise to L2 sensitivity, while the Laplace mechanism requires L1 sensitivity. These errors call for greater scrutiny of differential privacy-based approaches in NLP—our work contributes to this goal.

7.4 Related Work

`FEDERATE` is an in-processing approach (see Section 3.5) that induces fairness by removing sensitive attributes from the encoder representations. In this section, we first describe two common techniques for removing sensitive attributes. We then discuss research exploring the interplay between fairness and differential privacy.

Adversarial learning. In order to improve model fairness or to prevent leaking sensitive attributes, several approaches employ adversarial-based training. For instance, Li, Baldwin, and Cohn (2018) propose to use a different adversary for each protected attribute, while Coavoux, Narayan, and Cohen (2018) consider additional loss components to improve the privacy-accuracy trade-off of the learned representation. Han, Baldwin, and Cohn (2021) introduce multiple adversaries focusing on different aspects of the representation by encouraging orthogonality between pairs of adversaries. Recently, Chowdhury et al. (2021) propose an adversarial scrubbing mechanism. However, they purely focus on information leakage, and not on fairness. Moreover, unlike our approach, these methods do not offer formal privacy guarantees. In fact, it has been observed that one can recover the sensitive attributes from the representations by training a post-hoc non linear classifier (Elazar and Goldberg, 2018). This is confirmed by our empirical results in Section 7.5. Several works have also explored the use of adversarial learning in inducing fairness. For instance, Beutel et al. (2017) explore the effect of data distribution during fair adversarial training, while Madras et al. (2018) propose various adversarial objective and connects them with different group fairness measure. However, unlike our work, they do not consider fairness and privacy at the same time.

Sub-space projection. A related line of work focuses on debiasing text representations using projection methods Bolukbasi et al., 2016; Wang et al., 2020; Karve, Ungar, and Sedoc, 2019. The general approach involves identifying and removing a sub-space associated with sensitive attributes. However, they rely on a manual selection of words in the vocabulary which is difficult to generalize to new attributes. Furthermore, Gonen and Goldberg (2019) showed that sensitive attributes still remain present even after applying these approaches.

Recently, Ravfogel et al. (2020) propose Iterative Null space Projection (INLP). It involves iteratively training a linear classifier to predict sensitive attributes followed by projecting the representation on the classifier's null space. On the same lines, Ravfogel et al. (2022) proposed a linear minmax game based mechanism to remove information which they showcase to be a better formulation than null space projection. However, these methods can only remove linear information from the representation. By leveraging DP, our approach provides robust guarantees that do not depend on the expressiveness of the adversary, thereby providing protection against a wider range of attacks.

DP and fairness. Recent work has studied the interplay between DP and (group) fairness in the setting where one seeks to prevent a model from leaking information about individual training points. Empirically, this is evaluated through membership inference attacks, where an attacker uses the model to determine whether a given data point was in the training set (Shokri et al., 2017). While Kulynych et al. (2022) observed that DP reduces disparate vulnerability to such attacks, it has also been shown that DP can exacerbate unfairness Bagdasaryan, Poursaeed, and Shmatikov, 2019; Pujol et al., 2020. Conversely, Chang and Shokri (2020) showed that enforcing a fair model leads to more privacy leakage for the unprivileged group. This tension between DP and fairness is further confirmed by a formal incompatibility result between ϵ-DP and fairness proved by Cummings et al. (2019), albeit in a restrictive setting. Some recent work attempts to train models under both DP and fairness constraints (Cummings et al., 2019; Xu, Du, and Wu, 2020; Liu et al., 2020), but this typically comes at the cost of enforcing weaker privacy guarantees for some groups.

Finally, Jagielski et al. (2019) train a fair model under DP constraints only for the sensitive attribute.

A fundamental difference between this line of work and our approach lies in the kind of privacy we provide. While the above approaches study (central) DP as a way to design algorithms which protect training points from membership inference attacks on the model, we construct a private encoder such that the encoded representation does not leak sensitive attributes of the input. Thus, unlike previous work, we provide privacy guarantees with respect to the model's intermediate representation for data unseen at training time, and empirically observe that in this case privacy and fairness are compatible and even mutually reinforce each other.

DP representations for NLP. In a setting similar to ours, Lyu, He, and Li (2020) propose to use DP to privatize model's intermediate representation. Unlike their method, we actively promote fairness by using an adversarial training mechanism, which leads to more private representations and fairer models in practice. Importantly, we also uncover a critical error in their privacy analysis (see Sec. 7.3.4). Concurrent to and independently from our work, Plant, Gkatzia, and Giuffrida (2021) propose an adversarial-driven DP training mechanism. However, they do not consider fairness, whereas we focus on enforcing both fairness and privacy. Moreover, their method has the same incorrect analysis as Lyu, He, and Li (2020).

7.5 Experiments

Recall that we are interested in approaches that are not only accurate but also fair and private at the same time. However, these three dimensions are not independent and are not straightforwardly amenable to a single evaluation metric. Thus, we present experiments aiming at (i) showcasing the privacy-fairness-accuracy tradeoffs of different approaches and then (ii) analyzing privacy-accuracy and fairness-accuracy tradeoffs separately. We begin by describing the datasets and the metrics.

Datasets. We consider 4 different datasets:

- *Twitter Sentiment* (Blodgett, Green, and O'Connor, 2016) consists of 200k tweets annotated with a binary sentiment label and a binary "race" attribute corresponding to African American English (AAE) vs. Standard American English (SAE) speakers.
- *Bias in Bios* De-Arteaga et al., 2019 consists of 393,423 textual biographies annotated with an occupation label (28 classes) and a binary gender attribute.
- *CelebA* Liu et al., 2015 is a binary classification dataset with a binary sensitive attribute (gender).
- *Adult Income* Kohavi, 1996 consists of 48,842 instances with binary sensitive attribute (gender).

Our setup for the first two dataset is similar to Ravfogel et al. (2020) and Han, Baldwin, and Cohn (2021). Appendix B.2 provides detailed description of these datasets, including sizes, pre-processing, and the challenges they pose to privacy and fairness tasks. Similar to Chapter 4, we postpone the results for *Adult Income* and

CelebA dataset to Appendix B.2.5 as they exhibit similar trends. The preprocessed versions of the datasets can be downloaded from this URL.[1]

Fairness metrics. For Twitter Sentiment we report the True Positive Rate Gap (TPR-gap), which measures the true positive rate difference between the two sensitive groups and is closely related to the notion of equal opportunity (see Section 3.3.1). Formally, denoting the binary ground truth $y \in \mathcal{Y}$, $\hat{y}$ the predicted label and $\mathcal{G} \in \{\mathbf{g}, \mathbf{g}'\}$ the sensitive attribute, TPR-gap is defined as:

$$\text{TPR-gap} = P(\hat{y} = 1 | y = 1, \mathbf{g}) - P(\hat{y} = 1 | y = 1, \mathbf{g}').$$

For Bias in Bios, which has 28 classes, we follow Romanov et al. (2019) and report the root mean square of TPR-gaps (GRMS) over all occupations $y \in \mathcal{Y}$ to obtain a single number:

$$\text{GRMS} = \sqrt{(1/|O|) \textstyle\sum_{y \in \mathcal{Y}} (\text{TPR-gap}_y)^2}. \tag{7.8}$$

Note that for GRMS essentially boils down to TPR-gap in binary setting.

Privacy metrics. We report two metrics for privacy:

- **Leakage**: The accuracy of a two-layer classifier which predicts the sensitive attribute from the encoded representation.
- **Minimum Description Length (MDL)** Voita and Titov, 2020, which quantifies the amount of "effort" required by such a classifier to achieve a certain accuracy. A higher MDL means that it is more difficult to retrieve the sensitive attribute from the representation. The metric depends on the dataset and the representation dimension, and thus cannot be compared across different datasets.

We provide more details about these metrics in Sec. B.2.1.

Methods and model architectures. We compare `FEDERATE` to the following methods: (i) `Adversarial` implements standard adversarial learning Li, Baldwin, and Cohn, 2018, which is equivalent to our approach without the *priv* layer, (ii) `Adversarial + Multiple` Han, Baldwin, and Cohn, 2021 implements multiple adversaries, (iii) `INLP` Ravfogel et al., 2020 is a subspace projection approach, and (iv) `Noise` learns DP text representations as proposed by Lyu, He, and Li (2020) but with corrected privacy analysis: this corresponds to our approach without the adversarial component. These methods have been described in details in Section 7.4 and their hyperparametrs in Appendix B.2.4. We also report the performance of two simple baselines: `Random` simply predicts a random label, and `Unconstrained` optimizes the classification performance without special consideration for privacy or fairness.

To provide a fair comparison, all methods use the same architecture for the encoder, the classifier and (when applicable) the adversarial branches. In order to evaluate across varying model complexities, we employ different architectures for the different datasets. For Twitter Sentiment, we follow the architecture employed by Han, Baldwin, and Cohn (2021), while for Bias in Bios we use a deeper architecture. The

[1] https://drive.google.com/uc?id=1ZmUE-g6FmzPPbZyw3EOki7z4bpzbKGWk

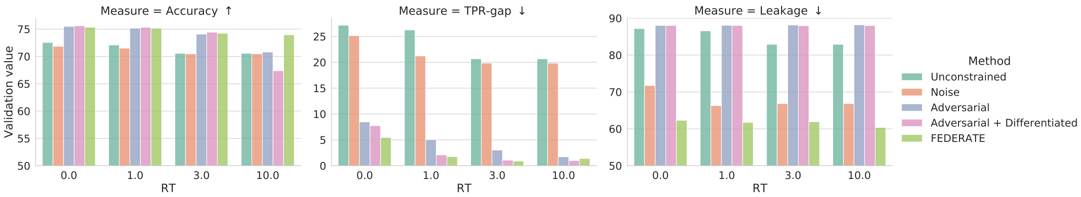


FIGURE 7.3: Validation accuracy, fairness and privacy of various approaches for different relaxation threshold (RT) (see Section 7.5.1) on Twitter Sentiment. When RT is increased, we select models with potentially lower accuracy on the validation set but are more fair (lower TPR-gap). Our approach FEDERATE consistently achieves better accuracy-fairness-privacy trade-offs than its competitors across all RTs.

exact architecture, hyperparameters, and their tuning details are provided in Appendix B.2.3-B.2.4. We implement FEDERATE in PyTorch Paszke et al., 2019. Our implementation, training, and evaluation scripts are available here.[2]

7.5.1 Accuracy-Fairness-Privacy Trade-off

In this first set of experiments, we explore the tridimensional trade-off between accuracy, fairness, and privacy and the inherent tension between them. These metrics are potentially all equally important and represent different information about the system on different scales. Thus, they cannot be trivially combined into a single metric. Moreover, this trade-off is influenced by the choice of method but also some of its hyperparameters (e.g., the value of ϵ and λ in our approach). Previous studies Han, Baldwin, and Cohn, 2021; Lyu, He, and Li, 2020 essentially selected hyperparameter values that maximize validation accuracy, which may lead to undesirable or suboptimal trade-offs. For instance, we found that this strategy does not always induce a fairer model than the Unconstrained baseline, and that it is often possible to obtain significantly more fair models at a negligible cost in accuracy.

Based on these observations, we propose to use a Relaxation Threshold (RT): instead of selecting the hyperparameters with highest validation accuracy α^*, we consider all models with accuracy in the range $[\alpha^* - \text{RT}, \alpha^*]$. We then select the hyperparameters with best fairness score within that range.[3]

Figure 7.3 presents the (validation) accuracy, fairness and privacy scores related to different RT for each method on Twitter Sentiment. The first thing to note is that FEDERATE achieves the best fairness and privacy results with accuracy higher or comparable to competing approaches. We also observe that setting RT$=0.0$ (i.e., choosing the model with highest validation accuracy) leads to a significantly more unfair model in all approaches, while fairness generally improves with increasing RT. This improvement comes at a negligible or small cost in accuracy. In terms of privacy, we find no significant differences across RTs.

We now showcase detailed results with RT fixed to 1.0 which is found to provide good trade-offs for all approaches in Figure 7.3, see Table 7.1a for Twitter Sentiment and Table 7.1b for Bias in Bios (and Appendix B.2.5 for additional results). For both datasets, we observe that all adversarial approaches induce a fairer model than Unconstrained or Noise, with FEDERATE performing best. In terms of accuracy,

[2] https://github.com/saist1993/DPNLP.

[3] We can also incorporate privacy into our hyperparameter selection strategy but, for the datasets and methods in our study, we found no significant change in Leakage across different hyperparameters.

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓	MDL ↑
Random	50.00 ± 0.00	0.00 ± 0.00	-	31.3 ± 0.10
Unconstrained	72.09 ± 0.73	26.26 ± 0.87	86.56 ± 0.83	15.21 ± 0.88
INLP	67.62 ± 0.57	9.19 ± 1.08	80.27 ± 2.50	24.82 ± 3.28
Noise	71.52 ± 0.51	21.23 ± 2.50	66.29 ± 3.55	21.10 ± 1.81
Adversarial	75.16 ± 0.65	5.03 ± 2.94	88.06 ± 0.20	16.16 ± 1.05
Adversarial + Multiple	75.32 ± 0.60	2.09 ± 1.18	88.03 ± 0.47	15.85 ± 1.46
FEDERATE	75.15 ± 0.59	1.75 ± 1.41	61.74 ± 5.05	22.94 ± 1.25

(A) Results on Twitter Sentiment dataset.

Method	Accuracy ↑	GRMS ↓	Leakage ↓	MDL ↑
Random	3.53 ± 0.01	0.00 ± 0.00	–	265.44 ± 0.13
Unconstrained	79.29 ± 0.32	15.88 ± 0.80	75.92 ± 2.73	173.99 ± 7.08
INLP	75.96 ± 0.47	12.81 ± 0.09	59.91 ± 0.08	253.36 ± 1.05
Noise	77.88 ± 0.32	13.89 ± 0.31	62.23 ± 0.99	241.22 ± 2.97
Adversarial	79.02 ± 0.20	13.06 ± 0.39	69.47 ± 1.64	206.78 ± 13.02
Adversarial + Multiple	79.30 ± 0.20	13.38 ± 0.63	68.24 ± 1.12	222.35 ± 10.04
FEDERATE	77.79 ± 0.11	11.02 ± 0.55	56.92 ± 0.98	257.94 ± 1.93

(B) Results on Bias in Bios dataset.

TABLE 7.1: Test results on (a) *Twitter Sentiment*, and (b) *Bias in Bios* with fixed Relaxation Threshold of 1.0. Fairness is measured with TPR-Gap or GRMS (lower is better), while privacy is measured by Leakage (lower is better) and MDL (higher is better). The MDL achieved by Random gives an upper bound for that particular dataset. Results have been averaged over 5 different seeds. Our proposed FEDERATE approach is the only method which achieves high levels of both fairness and privacy while maintaining competitive accuracy.

all adversarial approaches perform similarly on Twitter Sentiment. Interestingly, they achieve higher accuracy than Unconstrained. We attribute this to a significant mismatch in the train and test distribution due to class imbalance. On Bias in Bios, we observe a small drop in accuracy of our proposed approach in comparison to Adversarial, albeit with a corresponding gain in fairness. We hypothesize that this is due to the choice of possible hyperparameters for FEDERATE (we did not consider very large values of ϵ which would recover Adversarial), meaning that FEDERATE pushes for more fairness (and privacy) at a potential cost of some accuracy. We explore the pairwise trade-offs (fairness-accuracy and privacy-accuracy) in more details in Section 7.5.2.

In terms of both privacy metrics, FEDERATE significantly outperforms all adversarial methods on both datasets. In fact, in line with previous studies Han, Baldwin, and Cohn, 2021, the leakage and MDL of purely adversarial methods are similar to that of Unconstrained. On both datasets, Noise achieves slightly weaker privacy than FEDERATE with much worse accuracy and fairness. FEDERATE also consistently outperforms INLP in all dimensions.

In summary, the results show that FEDERATE stands out as the only approach that can simultaneously induce a fairer model *and* make its representation private while maintaining high accuracy. Furthermore, these results empirically demonstrate that

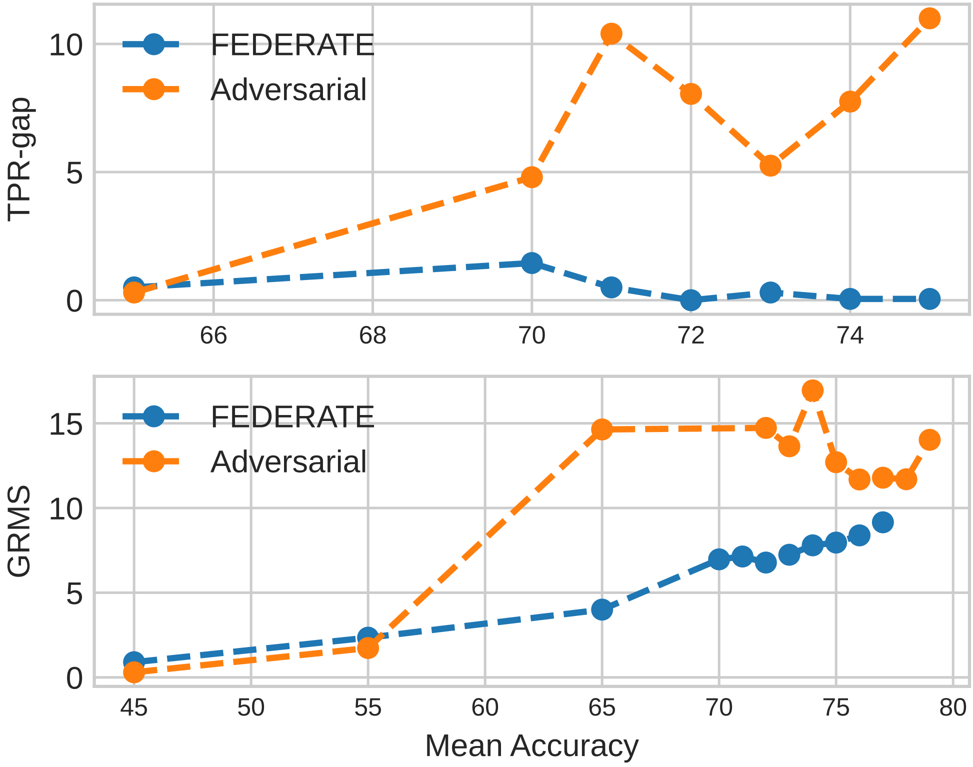


FIGURE 7.4: Fairness-accuracy trade-off on Twitter Sentiment (top) and Bias in Bios (bottom). A missing point means that the accuracy interval was not found within our hyperparameter search. FEDERATE provides better fairness across most accuracy intervals in comparison to Adversarial over both datasets.

our measures of privacy and fairness are indeed compatible with one another and can even reinforce each other.

7.5.2 Pairwise Trade-offs

In the previous experiments, we explored the tridimensional trade-off and found FEDERATE to attain better trade-offs than all other methods. Here, we take a closer look at the pairwise fairness-accuracy and privacy-accuracy trade-offs separately. We find that FEDERATE outperforms the Adversarial and Noise approach in their corresponding dimension, suggesting that FEDERATE is a better choice even for bidimensional trade-offs. This experiment also validates the superiority of combining adversarial learning and DP over using either approach alone.

Fairness-accuracy trade-off. We plot best validation fairness scores over different accuracy intervals for the two datasets in Figure 7.4. The interval is denoted by its mean accuracy (i.e., $[71.5, 72.5]$ is represented by 72). We then find the corresponding best fairness score for the interval. We observe:

- *Better fairness-accuracy trade-off:* FEDERATE provides better fairness than the Adversarial approach for almost all accuracy intervals. In the case of Bias in Bios, Adversarial is able to achieve higher accuracy (albeit with a loss in fairness). We note that this high accuracy regime can be matched by FEDERATE with a larger ϵ.

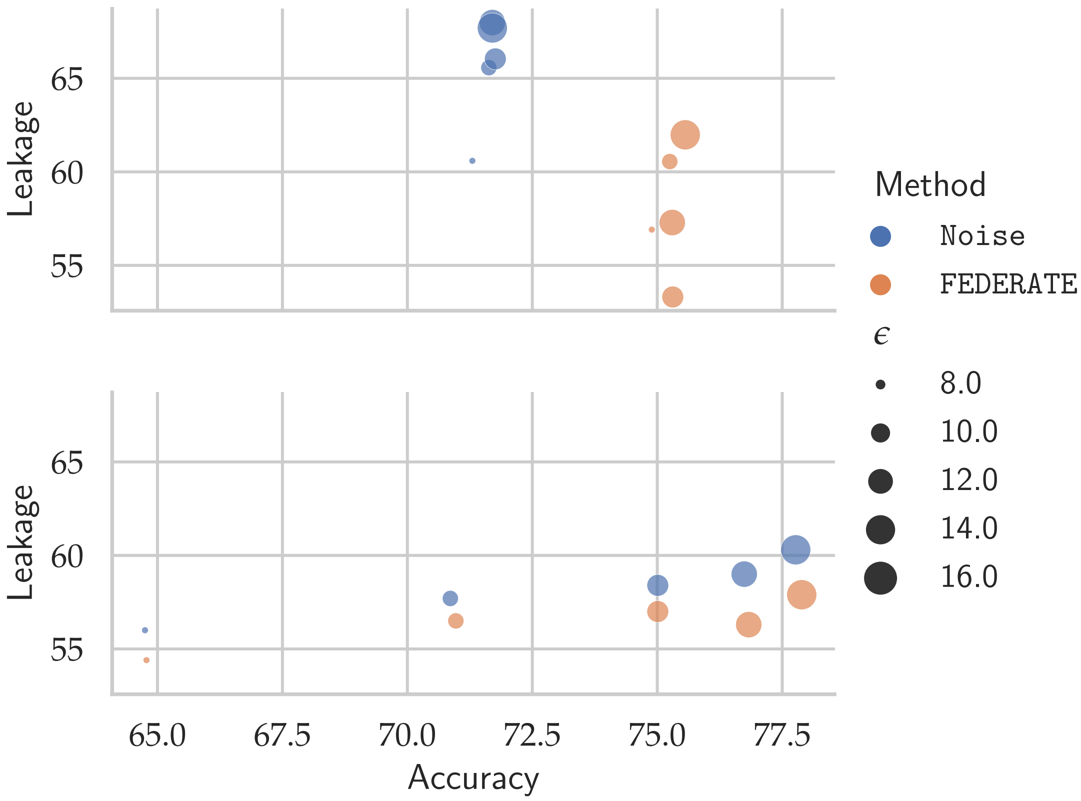


FIGURE 7.5: Privacy-accuracy trade-off on Twitter Sentiment (top) and Bias in Bios (bottom), with associated values of ϵ. FEDERATE gives lower leakage and better or comparable accuracy to Noise over both datasets.

- *Smoother fairness-accuracy trade-off:* Interestingly, FEDERATE enables a smoother exploration of the accuracy-fairness trade-off space than Adversarial. As adversarial models are notoriously difficult to train, this suggests that the introduction of DP noise has a stabilizing effect on the training dynamics of the adversarial component.

Privacy-accuracy trade-off. We plot privacy and accuracy with respect to ϵ, the parameter controlling the theoretical privacy level in Figure 7.5. In general, the value of ϵ correlates well with the empirical leakage. On Bias in Bios, FEDERATE and Noise are comparable in both accuracy and privacy. However, for Twitter Sentiment, our approach outperforms Noise in both accuracy and privacy for every ϵ. We hypothesize this difference in the accuracy to be a case of mismatch between train-test split, suggesting FEDERATE to be more robust to these distributional shifts. These observations suggest that FEDERATE either improves upon Noise in privacy-accuracy tradeoff or remains comparable. For completeness, we also present the same results as a table in Appendix B.2.5.

7.6 Limitations

A current limitation of this work in the context of fairness is that it is not designed to work with a specific definition of fairness, such as equal odds. Instead, it enforces fairness by removing certain protected information, which can correlate with specific fairness notions. Similarly, we do not provide formal fairness guarantees for our method as we do for privacy. We also do not provide privacy of training data, i.e., protection against reconstruction attacks. It is also necessary for the practitioner to monitor the fairness levels of the model over time, as due to data drift and other changes, the model's fairness level might change.

7.7 Conclusion and Perspectives

In this chapter, we proposed a DP-driven adversarial learning approach for NLP. Through our experiments, we showed that our method simultaneously induces private representations and fair models, with a mutually reinforcing effect between privacy and fairness. We also find that our approach improves upon competitors on each dimension separately. While we focused on privatizing sensitive attributes like race or gender, our approach can be used to remove other types of unwanted information from text representations, such as tenses or POS tag information, which might not be relevant for certain NLP tasks.

Chapter 8

Conclusion

In this chapter, we first summarize our contributions and then outline few potential future work and extensions.

8.1 Summary

In this thesis, we investigated the problem of fairness in machine learning. Specifically, we introduced measures and methods to mitigate allocation harm at different stages of a machine learning pipeline. We presented two in-processing fairness approaches that address the problem at the time of training. Further, we introduced a new evaluation measure for intersectional fairness which is robust to the phenomena of "leveling down". Finally, we proposed a data generation mechanism that leverages the structure of intersectional fairness to enhance performance for the most disadvantaged groups.

In Chapter 4, we introduced FairGrad, an iterative approach which dynamically learns group-specific weights based on fairness levels. Specifically, it increases the weights of disadvantaged groups, thereby enhancing their influence on the final loss function and decreases weights for the opposite scenario. FairGrad is straightforward to implement, necessitating only minor modifications to existing infrastructures. It is versatile, supporting multiple fairness measures and accommodating both approximate fairness and multi-class scenarios. Through experiments across over 10 datasets and 6 baselines, we demonstrated that FairGrad is an effective in-processing method, offering wide applicability with limited computational overhead.

We then shifted our attention to intersectional fairness setting in Chapter 5 where we benchmarked various fairness-inducing methods. Our experiments revealed that several approaches exhibit “leveling down" behavior, implying that they optimize for current fairness measures by harming the involved groups. We believe this occurs because existing fairness measures take a strictly egalitarian view. Consequently, we introduced a novel intersectional fairness measure, IF_α, devised to address leveling down by considering both relative and absolute performance. Furthermore, we illustrated its various properties and highlight its relationship with other fairness methods.

In Chapter 6, we introduced a data generation mechanism aimed at enhancing performance for the most disadvantaged groups in intersectional setting. Specifically, we developed a generative function that exploits the hierarchial structure of the intersectional setting, and augment data for sensitive groups by modifying and merging data from more general groups. Our experiments across three datasets demonstrated that this data generation yields diverse samples and improves performance for the most underrepresented groups.

Lastly, in Chapter 7, we explore the relationship between fairness and privacy in the context of NLP. We introduce FEDERATE, a novel approach that combines adversarial learning with differential privacy. Our evaluations delve into the privacy-fairness tradeoff, revealing that FEDERATE can simultaneously learn private representations and models that are fairer than contemporary methods but also maintain comparable accuracy. Furthermore, our findings underscore that privacy and fairness can coexist and even positively reinforce each other in specific scenarios.

8.2 Future Works and Perspective

I now outline few potential extensions of the works proposed as part of this thesis.

Joint Optimization of Fairness Across the Machine Learning Pipeline In this thesis, our focus has been on individual stages of the machine learning pipeline, where we propose various methods to promote fairness at each step. However, the cumulative impact of integrating these methods on overall fairness remains an open question. I will delve into these combined effects, along with exploring novel approaches that optimize fairness across multiple stages simultaneously. Additionally, I plan to examine how the characteristics of a task influence the most effective place for fairness intervention. This analysis could lead to gains in both fairness and computational efficiency, as well as a more nuanced understanding of the problem.

Generalized benchmark for group fairness: In Chapter 4, we experimented with over 10 datasets, 4 fairness measures, 6 different baselines, and 5 distinct seeds. Through our experiments we found significant sensitivity to seeds, hyperparameters, and hypothesis classes, echoing observations from other studies. Given the subtle differences in experimental settings across many research works, direct comparisons between them prove challenging. Consequently, there's a pressing need for an open-source benchmarking framework with standardized data splits, seeds, and other configurations. Such a benchmark would provide insights into the field's progression. For example, while many approaches claim to achieve state-of-the-art results, our experiments showed that no single fairness-enforcing method consistently outperformed others in terms of both accuracy and fairness across a wide range of settings. An open-source benchmark would allow practitioners to compare their findings more easily.

Intersectionality: The analytical framework of *intersectionality* (Crenshaw, 1989) posits that inequalities based on attributes like gender and race might "intersect", giving rise to unique combined effects. In Chapter 6, our initial results indicate that data for smaller groups can be derived from their corresponding parent groups. While the newly generated data is distinct, it remains nearly indistinguishable from the actual data. Additionally, employing this generated data enhances classifier performance for sensitive groups. However, these findings seem to deviate from the traditional intersectionality framework, as our data generation for a specific group involves manipulating and combining data from its related parent groups. This might suggest that our transformation function either captures these unique identities because they are already present in the parent groups (albeit in a latent manner) or that the dataset does not adequately mirror the intersectionality framework. Future research should delve deeper into this phenomenon where new identities stem from pre-existing ones.

Missing Sensitive Attributes: In this thesis, we have operated under the assumption that all sensitive attributes are available for every instance. Nevertheless, there are situations where such information might be inaccessible due to legal constraints, such as the GDPR. Additionally, even when this data is available, it may not be comprehensive for all instances given the costs and challenges associated with collecting sensitive attributes. Hence, it becomes crucial to develop methods that can ensure fairness in scenarios with absent or incomplete sensitive attributes. Recent studies, such as those by Lahoti et al. (2020) and Hashimoto et al. (2018), have started addressing this situation. Similarly, FairGrad could be adapted to such contexts by incorporating a classifier to predict group membership, in tandem with the dynamic group re-weighting procedure during training. Another potential strategy might involve initially training a classifier on data with missing attributes, followed by finetuning it using FairGrad on data containing sensitive attributes. Exploring data generation techniques in these scenarios would also be an area worth pursuing.

Effect of Training Techniques on Fairness: As deep learning architectures for NLP have significantly advanced, various specialized training methods have emerged in the literature. We identify several techniques that might have implications for fairness and privacy:

- Prompting and Fine-tuning: In the last year, several advancements have been made to classify data efficiently via large language models. These include methods such as prompting (Li and Liang, 2021), fine-tuning (Lialin, Deshpande, and Rumshisky, 2023), and adapters (Houlsby et al., 2019). However, several questions related to fairness have yet to be explored in-depth. These include:
  - The fairness implications of these techniques. For example, it is important to investigate whether fine-tuning uniformly improves a model's performance across all groups, or if certain subgroups are disproportionately affected.
  - The interplay between a language model's inherent bias and dataset bias. As discussed in Chapter 1, biases can arise at both dataset and model training levels. Given that large language models, pretrained on vast datasets, already contain various biases, a critical research direction would be examining the interaction between dataset bias and the biases within these large models.
  - The impact of in-context examples used during prompting on fairness. In few-shot prompting, where a model is exposed to a limited number of examples before generalizing to unseen data, the impact over accuracy is well-documented (Brown et al., 2020; Touvron et al., 2023). However, the influence of biases in these examples on model predictions remains unexplored. For instance, an intriguing question is whether combining

stereotypical and anti-stereotypical examples leads to improved fairness, or if focusing solely on anti-stereotypical examples is more effective.

- Distillation: The rise of large language models has prompted significant research interest in distilling these models (Sanh et al., 2019; Rashid et al., 2020; Gou et al., 2021) aiming to reduce computational complexity and memory requirements. Several works have shown that the distilled model often achieves higher or similar accuracy as the original model. However, the effects of distillation on subgroups are still unclear.
- Interpretability: The need for understanding complex architecture has gathered broad interest in creating training routines and mechanism which are interpretable. This generally involves methods like attention visualization (Vashishth et al., 2019), and highlighting parts of the text (Ventura et al., 2021) which played a significant role in classification. However, these methods typically generate explanations based on input data and the training dataset, potentially increasing the model's vulnerability to privacy breaches and data leakage.

Appendix A

FairGrad: Fairness Aware Gradient Descent (Appendix)

In this appendix, we provide details that were omitted in Chapter 4. First, in in Section A.1, we show that several well known group fairness measures are compatible with FairGrad. In Section A.2, we prove Lemma 1. Next, in Section A.3, we derive the update rules for FairGrad with ϵ-fairness. Finally, in Section A.4, we provide additional experiments.

A.1 Reformulation of Various Group Fairness Notion

In this section, we present several group fairness notions which respect our fairness definition presented in Section 4.2.1.

Example 2 (Equalized Odds (EOdds) (Hardt, Price, and Srebro, 2016)). A model h_θ is fair for Equalized Odds when the probability of predicting the correct label is independent of the sensitive attribute, that is, $\forall l \in \mathcal{Y}, \forall \mathbf{g} \in \mathcal{G}$

$$\widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, \mathbf{g}, y = l\right) = \widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, y = l\right).$$

It means that we need to partition the space into $K = |\mathcal{Y} \times \mathcal{G}|$ groups and, $\forall l \in \mathcal{Y}, \forall \mathbf{g} \in \mathcal{G}$, we define $\widehat{F}_{(l,\mathbf{g})}$ as

$$\begin{aligned}\widehat{F}_{(l,\mathbf{g})}(\mathcal{T}, h_\theta) &= \widehat{\mathbb{P}}\left(h_\theta(x) \neq l \,|\, y = l\right) - \widehat{\mathbb{P}}\left(h_\theta(x) \neq l \,|\, \mathbf{g}, y = l\right) \\ &= \sum_{(l,\mathbf{g}') \neq (l,\mathbf{g})} \widehat{\mathbb{P}}\left(\mathbf{g}' | y = l\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq l \,|\, \mathbf{g}', y = l\right) \\ &\quad - (1 - \widehat{\mathbb{P}}\left(\mathbf{g} | y = l\right)) \widehat{\mathbb{P}}\left(h_\theta(x) \neq l \,|\, \mathbf{g}, y = l\right)\end{aligned}$$

where the law of total probability was used to obtain the last equation. Thus, Equalized Odds satisfies all our assumptions with $C_{(l,\mathbf{g})}^{(l,\mathbf{g})} = \widehat{\mathbb{P}}\left(\mathbf{g} | y = l\right) - 1$, $C_{(l,\mathbf{g})}^{(l,\mathbf{g}')} = \widehat{\mathbb{P}}\left(\mathbf{g}' | y = l\right)$, $C_{(l,\mathbf{g})}^{(l',\mathbf{g}')} = 0$ with $\mathbf{g}' \neq \mathbf{g}$ and $l' \neq l$, and $C_{(l,\mathbf{g})}^{0} = 0$.

Example 3 (Equality of Opportunity (EOpp) (Hardt, Price, and Srebro, 2016)). A model h_θ is fair for Equality of Opportunity when the probability of predicting the correct label is independent of the sensitive attribute for a given subset $\mathcal{Y}' \subset \mathcal{Y}$ of labels called the desirable outcomes, that is, $\forall l \in \mathcal{Y}', \forall \mathbf{g} \in \mathcal{G}$

$$\widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, \mathbf{g}, y = l\right) = \widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, y = l\right).$$

It means that we need to partition the space into $K = |\mathcal{Y} \times \mathcal{G}|$ groups and, $\forall l \in \mathcal{Y}, \forall \mathbf{g} \in \mathcal{G}$, we define $\widehat{F}_{(l,\mathbf{g})}$ as

$$\widehat{F}_{(l,\mathbf{g})}(\mathcal{T}, h_\theta) = \begin{cases} \widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, \mathbf{g}, y = l\right) & \\ \qquad - \widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, y = l\right) & \forall (l, \mathbf{g}) \in \mathcal{Y}' \times \mathcal{G} \\ 0 & \forall (l, \mathbf{g}) \in \mathcal{Y} \times \mathcal{G} \setminus \mathcal{Y}' \times \mathcal{G} \end{cases}$$

which can then be rewritten in the correct form in the same way as Equalized Odds, the only difference being that $C^{\cdot}_{(l,\mathbf{g})} = 0, \forall (l, \mathbf{g}) \in \mathcal{Y} \times \mathcal{G} \setminus \mathcal{Y}' \times \mathcal{G}$.

Example 4 (Demographic Parity (DP) (Calders, Kamiran, and Pechenizkiy, 2009)). A model h_θ is fair for Demographic Parity when the probability of predicting a binary label is independent of the sensitive attribute, that is, $\forall l \in \mathcal{Y}, \forall \mathbf{g} \in \mathcal{G}$

$$\widehat{\mathbb{P}}\left(h_\theta(x) = l \,|\, \mathbf{g}\right) = \widehat{\mathbb{P}}\left(h_\theta(x) = l\right).$$

It means that we need to partition the space into $K = |\mathcal{Y} \times \mathcal{G}|$ groups and, $\forall l \in \mathcal{Y}, \forall \mathbf{g} \in \mathcal{G}$, we define $\widehat{F}_{(l,\mathbf{g})}$ as

$$\begin{aligned} \widehat{F}_{(l,\mathbf{g})}(\mathcal{T}, h_\theta) &= \widehat{\mathbb{P}}\left(h_\theta(x) \neq l\right) - \widehat{\mathbb{P}}\left(h_\theta(x) \neq l \,|\, \mathbf{g}\right) \\ &= \left(\widehat{\mathbb{P}}\left(y = l, \mathbf{g}\right) - \widehat{\mathbb{P}}\left(y = l \,|\, \mathbf{g}\right)\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}, y = l\right) \\ &\quad + \sum_{(l,\mathbf{g}') \neq (l,\mathbf{g})} \widehat{\mathbb{P}}\left(y = l, \mathbf{g}'\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}', y = l\right) \\ &\quad + \left(\widehat{\mathbb{P}}\left(y = \bar{l} \,|\, \mathbf{g}\right) - \widehat{\mathbb{P}}\left(y = \bar{l}, \mathbf{g}\right)\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}, y = \bar{l}\right) \\ &\quad - \sum_{(\bar{l},\mathbf{g}') \neq (\bar{l},\mathbf{g})} \widehat{\mathbb{P}}\left(y = \bar{l}, \mathbf{g}'\right) \widehat{\mathbb{P}}\left(h_\theta(x) \neq y \,|\, \mathbf{g}', y = \bar{l}\right) \\ &\quad \widehat{\mathbb{P}}\left(y = \bar{l}\right) - \widehat{\mathbb{P}}\left(y = \bar{l} \,|\, \mathbf{g}\right) \end{aligned}$$

where the law of total probability was used to obtain the last equation. Thus, Demographic Parity satisfies all our assumptions with $C^{(l,\mathbf{g})}_{(l,\mathbf{g})} = \widehat{\mathbb{P}}\left(y = l, \mathbf{g}\right) - \widehat{\mathbb{P}}\left(y = l \,|\, \mathbf{g}\right)$, $C^{(l,\mathbf{g}')}_{(l,\mathbf{g})} = \widehat{\mathbb{P}}\left(y = l, \mathbf{g}'\right)$ with $\mathbf{g}' \neq \mathbf{g}$, $C^{(\bar{l},\mathbf{g})}_{(l,\mathbf{g})} = \widehat{\mathbb{P}}\left(y = \bar{l} \,|\, \mathbf{g}\right) - \widehat{\mathbb{P}}\left(y = \bar{l}, \mathbf{g}\right)$, $C^{(\bar{l},\mathbf{g}')}_{(l,\mathbf{g})} = -\widehat{\mathbb{P}}\left(y = \bar{l}, \mathbf{g}'\right)$ with $\mathbf{g}' \neq \mathbf{g}$, and $C^{0}_{(l,\mathbf{g})} = \widehat{\mathbb{P}}\left(y = \bar{l}\right) - \widehat{\mathbb{P}}\left(y = \bar{l} \,|\, \mathbf{g}\right)$.

A.2 Proof of Lemma 1

Lemma (Negative weights are necessary.)**.** Assume that the fairness notion under consideration is Accuracy Parity. Let h_θ^* be the most accurate and fair model. Then using negative weights is necessary as long as

$$\min_{\substack{h_\theta \in \mathcal{H} \\ h_\theta \text{unfair}}} \max_{\mathcal{T}_k} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) < \widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y\right).$$

Proof. To prove this Lemma, one first need to notice that, for Accuracy Parity, since $\sum_{k=1}^{K} \widehat{\mathbb{P}}\left(\mathcal{T}_k\right) = 1$ we have that

$$\sum_{k'=1}^{K} C_k^{k'} = \left(\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) - 1\right) + \sum_{\substack{k'=1 \\ k' \neq k}}^{K} \widehat{\mathbb{P}}\left(\mathcal{T}_{k'}\right) = 0.$$

This implies that

$$\sum_{k=1}^{K} \left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'} \right] = 1.$$

This implies that, whatever our choice of λ, the weights will always sum to one. In other words, since we also have that $\sum_{k=1}^{K} \lambda_k C_k^0 = 0$ by definition, for a given hypothesis h_θ, we have that

$$\max_{\lambda_1, \ldots, \lambda_K \in \mathbb{R}} \sum_{k=1}^{K} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) \left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) + \sum_{k'=1}^{K} C_{k'}^{k} \lambda_{k'} \right] \tag{A.1}$$

$$= \max_{\substack{w_1, \ldots, w_K \in \mathbb{R} \\ s.t. \sum_k w_k = 1}} \sum_{k=1}^{K} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) w_k \tag{A.2}$$

where, given $w_1, \ldots, w_K$, the original values of lambda can be obtained by solving the linear system $C\lambda = w$ where

$$C = \begin{pmatrix} C_1^1 & \ldots & C_K^1 \\ \vdots & & \vdots \\ C_1^K & \ldots & C_K^K \end{pmatrix}, \quad \lambda = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_K \end{pmatrix}, \quad w = \begin{pmatrix} w_1 - \widehat{\mathbb{P}}\left(\mathcal{T}_1\right) \\ \vdots \\ w_K - \widehat{\mathbb{P}}\left(\mathcal{T}_K\right) \end{pmatrix}$$

which is guaranteed to have infinitely many solutions since the rank of the matrix C is $K-1$ and the rank of the augmented matrix $(C|w)$ is also $K-1$. Here we are using the fact that $\widehat{\mathbb{P}}\left(\mathcal{T}_k\right) \neq 0, \forall k$ since all the groups have to be represented to be taken into account.

We will now assume that all the weights are positive, that is $w_k \geq 0, \forall k$. Then, the best strategy to solve Problem (A.2) is to put all the weight on the worst off group k, that is set $w_k = 1$ and $w_{k'} = 0, \forall k' \neq k$. It implies that

$$\max_{\substack{w_1, \ldots, w_K \in \mathbb{R} \\ s.t. \sum_k w_k = 1}} \sum_{k=1}^{K} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) w_k = \max_{k} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right).$$

Furthermore, notice that, for fair models with respect to Accuracy Parity, we have that $\widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) = \widehat{\mathbb{P}}\left(h_\theta(x) \neq y\right), \forall k$. Thus, if it holds that

$$\min_{\substack{h_\theta \in \mathcal{H} \\ h_\theta \text{unfair}}} \max_{\mathcal{T}_k} \widehat{\mathbb{P}}\left(h_\theta(x) \neq y | \mathcal{T}_k\right) < \widehat{\mathbb{P}}\left(h_\theta^*(x) \neq y\right)$$

where h_θ^* is the most accurate and fair model, then the optimal solution of Problem (3) in the main chapter will be unfair. It implies that, in this case, using positive weights is not sufficient and negative weights are necessary. □

A.3 FairGrad for ϵ-fairness

To derive FairGrad for ϵ-fairness we first consider the following standard optimization problem

$$\begin{aligned}\arg\min_{h_\theta\in\mathcal{H}} & \widehat{\mathbb{P}}\left(h_\theta(x)\neq y\right)\\ \text{s.t. } & \forall k\in[K], \widehat{F}_k(\mathcal{T},h_\theta)\leq\epsilon\\ & \forall k\in[K], \widehat{F}_k(\mathcal{T},h_\theta)\geq-\epsilon.\end{aligned}$$

We, once again, use a standard multipliers approach to obtain the following unconstrained formulation:

$$\mathcal{L}\left(h_\theta,\lambda_1,\dots,\lambda_K,\delta_1,\dots,\delta_K\right)=\widehat{\mathbb{P}}\left(h_\theta(x)\neq y\right)+\sum_{k=1}^{K}\lambda_k\left(\widehat{F}_k(\mathcal{T},h_\theta)-\epsilon\right)-\delta_k\left(\widehat{F}_k(\mathcal{T},h_\theta)+\epsilon\right) \tag{A.3}$$

where $\lambda_1,\dots,\lambda_K$ and $\delta_1,\dots,\delta_K$ are the multipliers that belong to $\mathbb{R}^+$, that is the set of positive reals. Once again, to solve this problem, we will use an alternating approach where the hypothesis and the multipliers are updated one after the other.

Updating the Multipliers. To update the values $\lambda_1,\dots,\lambda_K$, we will use a standard gradient ascent procedure. Hence, noting that the gradient of the previous formulation is

$$\begin{aligned}\nabla_{\lambda_1,\dots,\lambda_K}\mathcal{L}\left(h_\theta,\lambda_1,\dots,\lambda_K,\delta_1,\dots,\delta_K\right)&=\begin{pmatrix}\widehat{F}_1(\mathcal{T},h_\theta)-\epsilon\\ \vdots\\ \widehat{F}_K(\mathcal{T},h_\theta)-\epsilon\end{pmatrix}\\ \nabla_{\delta_1,\dots,\delta_K}\mathcal{L}\left(h_\theta,\lambda_1,\dots,\lambda_K,\delta_1,\dots,\delta_K\right)&=\begin{pmatrix}-\widehat{F}_1(\mathcal{T},h_\theta)-\epsilon\\ \vdots\\ -\widehat{F}_K(\mathcal{T},h_\theta)-\epsilon\end{pmatrix}\end{aligned}$$

we have the following update rule $\forall k\in[K]$

$$\begin{aligned}\lambda_k^{T+1}&=\max\left(0,\lambda_k^T+\eta\left(\widehat{F}_k\left(\mathcal{T},h_\theta^T\right)-\epsilon\right)\right)\\ \delta_k^{T+1}&=\max\left(0,\delta_k^T-\eta\left(\widehat{F}_k\left(\mathcal{T},h_\theta^T\right)+\epsilon\right)\right)\end{aligned}$$

where η is a fairness rate that controls the importance of each weight update.

Updating the Model. To update the parameters $\theta\in\mathbb{R}^D$ of the model h_θ, we proceed as before, using a gradient descent approach. However, first, we notice that given the fairness notions that we consider, Equation (A.3) is equivalent to

$$\mathcal{L}\left(h_\theta,\lambda_1,\dots,\lambda_K,\delta_1,\dots,\delta_K\right)=\sum_{k=1}^{K}\widehat{\mathbb{P}}\left(h_\theta(x)\neq y|\mathcal{T}_k\right)\left[\widehat{\mathbb{P}}\left(\mathcal{T}_k\right)+\sum_{k'=1}^{K}C_{k'}^{k}\left(\lambda_{k'}-\delta_{k'}\right)\right] \tag{A.4}$$

$$-\sum_{k=1}^{K}\left(\lambda_k+\delta_k\right)\epsilon+\sum_{k=1}^{K}(\lambda_k-\delta_k)C_k^0.$$

Since the additional terms in the optimization problem do not depend on h_θ, the main difference between exact and ϵ-fairness is the nature of the weights. More precisely, at iteration t, the update rule becomes

$$\theta^{T+1} = \theta^{T} - \eta_\theta \sum_{k=1}^{K} \left[\widehat{\mathbb{P}} \left(\mathcal{T}_k \right) + \sum_{k'=1}^{K} C_{k'}^{k} \left(\lambda_{k'} - \delta_{k'} \right) \right] \nabla_\theta \widehat{\mathbb{P}} \left(h_\theta(x) \neq y | \mathcal{T}_k \right)$$

where η_θ is a learning rate. Once again, we obtain a simple re-weighting scheme where the weights depend on the current fairness level of the model through $\lambda_1, \dots, \lambda_K$ and $\delta_1, \dots, \delta_K$, the relative size of each group through $\widehat{\mathbb{P}} \left(\mathcal{T}_k \right)$, and the fairness notion through the constants C.

A.4 Extended Experiments

In this section, we provide additional details related to the baselines and the hyper-parameters tuning procedure. We then provide descriptions of the datasets and finally the results.

A.4.1 Baselines

- **Adversarial**: One of the common ways of removing sensitive information from the model's representation is via adversarial learning. Broadly, it consists of three components, namely an encoder, a task classifier, and an adversary. On the one hand, the objective of the adversary is to predict sensitive information from the encoder. On the other hand, the encoder aims to create representations that are useful for the downstream task (task classifier) and, at the same time, fool the adversary. The adversary is generally connected to the encoder via a gradient reversal layer (Ganin and Lempitsky, 2015) which acts like an identity function during the forward pass and scales the loss with a parameter $-\lambda$ during the backward pass. In our setting, the encoder is a Multi-Layer Perceptron with two hidden layers of size 64 and 128 respectively, and the task classifier is another Multi-Layer Perceptron with a single hidden layer of size 32. The adversary is the same as the main task classifier. We use a ReLU as the activation function with the dropout set to 0.2 and employ batch normalization with default PyTorch parameters. As a part of the hyper-parameter tuning, we did a grid search over λ, varying it between 0.1 to 3.0 with an interval of 0.2.

- **BiFair (Ozdayi, Kantarcioglu, and Iyer, 2021)**: For this baseline, we fix the weight parameter to be of length 8 as suggested in the code released by the authors[1]. In this fixed setting, we perform a grid search over the following hyper-parameters:
  - Batch Size: 128, 256, 512
  - Weight Decay: 0.0, 0.001
  - Fairness Loss Weight: 0.5, 1, 2, 4
  - Inner Loop Length: 5, 25, 50

[1] https://github.com/TinfoilHat0/BiFair

- **Constraints**: We use the implementation available in the TensorFlow Constrained Optimization[2] library with default hyper-parameters.
- **FairBatch**: We use the implementation publicly released by the authors[3].
- **Weighted ERM**: We reweigh each example in the dataset based on inverse of the proportion of the sensitive group it belongs to.
- **Reduction**: We use the implementation available in the Fairlearn[4] with default hyper-parameters.

In our initial experiments, we varied the batch size, and learning rates for both Constraints and FairBatch. However, we found that the default hyper-parameters as specified by the authors result in the best performances. In the spirit of being comparable in terms of hyper-parameter search budget, we also fix all hyper-parameters of FairGrad, apart from the batch size and weight decay. We experiment with two different batch sizes namely, 64 or 512 for the standard fairness dataset. Similarly, we also experiment with three weight decay values namely, 0.0, 0.001 and 0.01. Note that we also vary weight decay and batch sizes for FairBatch, Adversarial, Unconstrained, and BiFair.

For all our experiments, apart from BiFair, we use Batch Gradient Descent as the optimizer with a learning rate of 0.1 and a gradient clipping of 0.05 to avoid exploding gradients. For BiFair, we employ the Adam optimizer as suggested by the authors with a learning rate of 0.001. For FairGrad, FairBatch and Unconstrained, we considered 6 hyper-parameters combinations. For BiFair, we considered 72 such combinations, while for Adversarial, there were 90 combinations.

A.4.2 Datasets

Here, we provide additional details on the datasets used in our experiments. We begin by describing the standard fairness datasets for which we follow the pre-processing procedure described in Lohaus, Perrot, and Von Luxburg (2020).

- **Adult**[5]: The dataset (Kohavi, 1996) is composed of 45222 instances, with 14 features each describing several attributes of a person. The objective is to predict the income of a person (below or above 50k) while remaining fair with respect to gender (binary in this case). Following the pre-processing step of Wu, Zhang, and Wu (2019), only 9 features were used for training.
- **CelebA**[6]: The dataset (Liu et al., 2015) consists of $202,599$ images, along with 40 binary attributes associated with each image. We use 38 of these as features while keeping gender as the sensitive attribute and "Smiling" as the class label.
- **Dutch**[7]: The dataset (Žliobaite, Kamiran, and Calders, 2011) is composed of $60,420$ instances with each instance described by 12 features. We predict "Low Income" or "High Income" as dictated by the occupation as the main classification task and gender as the sensitive attribute.

[2] https://github.com/google-research/tensorflow_constrained_optimization

[3] https://github.com/yuji-roh/fairbatch

[4] https://fairlearn.org/

[5] https://archive.ics.uci.edu/ml/datasets/adult

[6] https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[7] https://sites.google.com/site/conditionaldiscrimination/

- **Compas**[8]: The dataset (Larson et al., 2016) contains 6172 data points, where each data point has 53 features. The goal is to predict if the defendant will be arrested again within two years of the decision. The sensitive attribute is race, which has been merged into "White" and "Non White" categories.
- **Communities and Crime**[9]: The dataset (Redmond and Baveja, 2002) is composed of 1994 instances with 128 features, of which 29 have been dropped. The objective is to predict the number of violent crimes in the community, with race being the sensitive attribute.
- **German Credit**[10]: The dataset (Dua, Graff, et al., 2017) consists of 1000 instances, with each having 20 attributes. The objective is to predict a person's creditworthiness (binary), with gender being the sensitive attribute.
- **Gaussian**[11]: It is a toy dataset with binary task label and binary sensitive attribute, introduced in Lohaus, Perrot, and Von Luxburg (2020). It is constructed by drawing points from different Gaussian distributions. We follow the same mechanism as described in Lohaus, Perrot, and Von Luxburg (2020), and sample 50000 data points for each class.
- **Adult Folktables**[12]: This dataset (Ding et al., 2021) is an updated version of the original Adult Income dataset. We use California census data with gender as the sensitive attribute. There are 195665 instances, with 9 features describing several attributes of a person. We use the same preprocessing step as recommended by the authors.

For all these datasets, we use a 20% of the data as a test set and 80% as a train set. We further divide the train set into two and keep 25% of the training examples as a validation set. For each repetition, we randomly shuffle the data before splitting it, and thus we had unique splits for each random seed. We use the following seeds: $10, 20, 30, 40, 50$ for all our experiments. As a last pre-processing step, we centered and scaled each feature independently by substracting the mean and dividing by the standard deviation both of which were estimated on the training set.

Twitter Sentiment Analysis[13]: The dataset (Blodgett, Green, and O'Connor, 2016) consists of 200k tweets with binary sensitive attribute (race) and binary sentiment score. We follow the setup proposed by Han, Baldwin, and Cohn (2021) and Elazar and Goldberg (2018) and create bias in the dataset by changing the proportion of each subgroup (race-sentiment) in the training set. With two sentiment classes being happy and sad, and two race classes being AAE and SAE, the training data consists of 40% AAE-happy, 10% AAE-sad, 10% SAE-happy, and 40% SAE-sad. The test set remains balanced. The tweets are encoded using the DeepMoji (Felbo et al., 2017) encoder with no fine-tuning, which has been pre-trained over millions of tweets to predict their emoji, thereby predicting the sentiment. Note that the train-test splits are pre-defined and thus do not change based on the random seed of the repetition.

A.4.3 Detailed Results

[8] https://github.com/propublica/compas-analysis

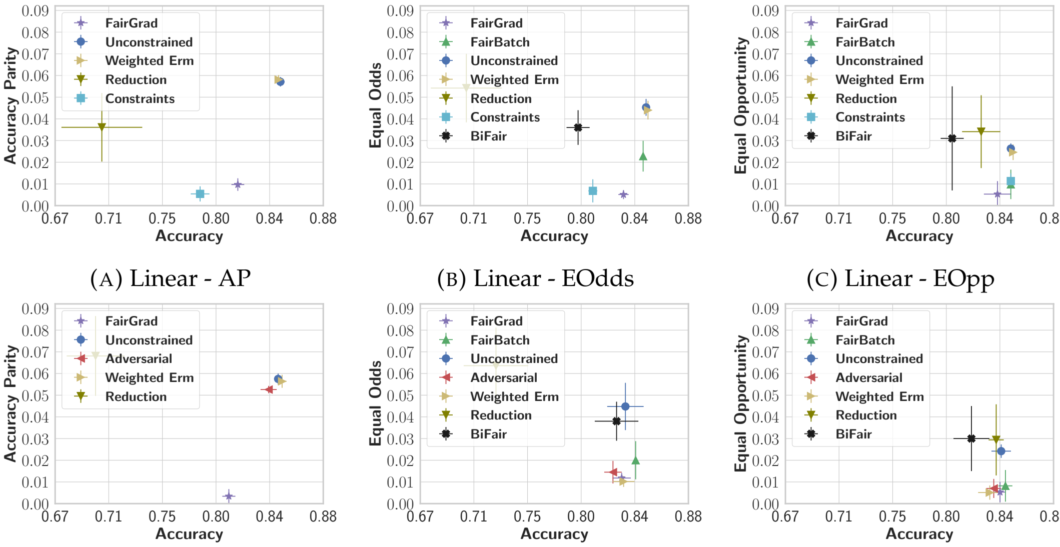
[9] http://archive.ics.uci.edu/ml/datasets/communities+and+crime

[10] https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29

[11] https://github.com/mlohaus/SearchFair/blob/master/examples/get_synthetic_data.py

[12] https://github.com/zykls/folktables

[13] https://slanglab.cs.umass.edu/TwitterAAE/



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

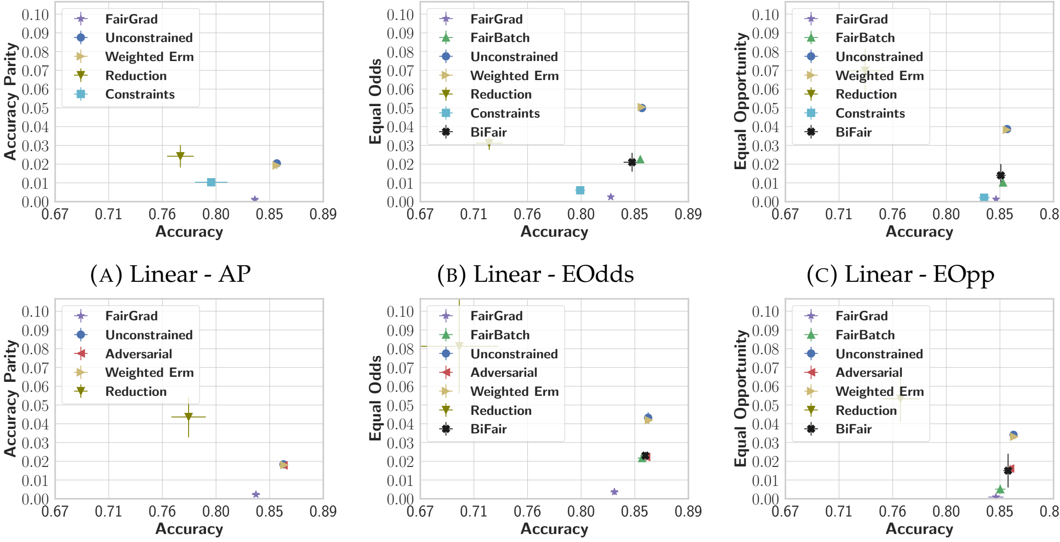
FIGURE A.1: Results for the Adult dataset with different fairness measures.

TABLE A.1: Results for the Adult dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8456 ± 0.0033	**AP**	0.0571 ± 0.0022	0.077 ± 0.0029	-0.0373 ± 0.0017
Constant	0.751 ± 0.0	**AP**	0.102 ± 0.0	0.138 ± 0.0	0.067 ± 0.0
Weighted ERM	0.8442 ± 0.0016	**AP**	0.0581 ± 0.0021	0.0783 ± 0.0028	-0.0379 ± 0.0014
Constrained	0.783 ± 0.007	**AP**	0.005 ± 0.003	0.007 ± 0.005	0.004 ± 0.002
Reduction	0.7064 ± 0.0315	**AP**	0.0361 ± 0.0158	0.0235 ± 0.0103	-0.0487 ± 0.0214
FairGrad	0.8124 ± 0.005	**AP**	0.0097 ± 0.0029	0.0131 ± 0.004	-0.0063 ± 0.0019
Unconstrained	0.846 ± 0.0028	**Eodds**	0.0453 ± 0.0039	0.048 ± 0.0043	-0.0878 ± 0.01
Constant	0.748 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8475 ± 0.0024	**Eodds**	0.044 ± 0.0043	0.0477 ± 0.0031	-0.0837 ± 0.0124
Constrained	0.805 ± 0.004	**Eodds**	0.007 ± 0.005	0.019 ± 0.017	0.002 ± 0.001
BiFair	0.793 ± 0.009	**Eodds**	0.036 ± 0.008	0.085 ± 0.027	-0.03 ± 0.016
FairBatch	0.8437 ± 0.0013	**Eodds**	0.0228 ± 0.0071	0.0411 ± 0.0105	-0.0245 ± 0.0183
Reduction	0.7059 ± 0.0277	**Eodds**	0.0542 ± 0.0158	0.0711 ± 0.0189	-0.1055 ± 0.022
FairGrad	0.8284 ± 0.004	**Eodds**	0.0051 ± 0.0021	0.0078 ± 0.0068	-0.0078 ± 0.0054
Unconstrained	0.8457 ± 0.0028	**Eopp**	0.0263 ± 0.0024	0.0157 ± 0.0011	-0.0893 ± 0.0083
Constant	0.754 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8475 ± 0.0024	**Eopp**	0.0246 ± 0.0036	0.0148 ± 0.002	-0.0837 ± 0.0124
Constrained	0.846 ± 0.002	**Eopp**	0.011 ± 0.004	0.039 ± 0.012	0.0 ± 0.0
BiFair	0.8 ± 0.009	**Eopp**	0.031 ± 0.024	0.019 ± 0.014	-0.107 ± 0.083
FairBatch	0.8457 ± 0.0016	**Eopp**	0.0098 ± 0.0068	0.0225 ± 0.0174	-0.0166 ± 0.0241
Reduction	0.8226 ± 0.0149	**Eopp**	0.0341 ± 0.0168	0.116 ± 0.0575	-0.0204 ± 0.0098
FairGrad	0.8353 ± 0.0106	**Eopp**	0.0053 ± 0.006	0.0177 ± 0.021	-0.0037 ± 0.0033

TABLE A.2: Results for the Adult dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.8438 ± 0.0025	**AP**	0.0575 ± 0.0025	0.0776 ± 0.0033	-0.0375 ± 0.0018
Constant	0.751 ± 0.0	**AP**	0.102 ± 0.0	0.138 ± 0.0	0.067 ± 0.0
Weighted ERM	0.8469 ± 0.0035	**AP**	0.0564 ± 0.003	0.0761 ± 0.0038	-0.0368 ± 0.0021
Adversarial	0.8364 ± 0.0063	**AP**	0.0526 ± 0.0017	0.0709 ± 0.0025	-0.0343 ± 0.0009
Reduction	0.7015 ± 0.0225	**AP**	0.0681 ± 0.0184	0.0444 ± 0.0122	-0.0917 ± 0.0247
FairGrad	0.8054 ± 0.0051	**AP**	0.0034 ± 0.0033	0.0033 ± 0.0031	-0.0036 ± 0.0042
Unconstrained	0.8299 ± 0.0142	**Eodds**	0.0448 ± 0.0109	0.0404 ± 0.0136	-0.0977 ± 0.0422
Constant	0.748 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8285 ± 0.0085	**Eodds**	0.0102 ± 0.0025	0.0196 ± 0.0102	-0.0099 ± 0.0047
Adversarial	0.8202 ± 0.0068	**Eodds**	0.0145 ± 0.0052	0.0288 ± 0.0177	-0.0153 ± 0.0067
BiFair	0.823 ± 0.017	**Eodds**	0.038 ± 0.009	0.09 ± 0.034	-0.038 ± 0.015
FairBatch	0.8379 ± 0.0009	**Eodds**	0.02 ± 0.0088	0.0327 ± 0.0153	-0.0244 ± 0.0218
Reduction	0.729 ± 0.0252	**Eodds**	0.0636 ± 0.0176	0.0673 ± 0.0203	-0.115 ± 0.0334
FairGrad	0.827 ± 0.0071	**Eodds**	0.0118 ± 0.0024	0.022 ± 0.014	-0.0165 ± 0.0135
Unconstrained	0.8382 ± 0.0076	**Eopp**	0.0242 ± 0.0031	0.0145 ± 0.0017	-0.0822 ± 0.0108
Constant	0.754 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8293 ± 0.0091	**Eopp**	0.0051 ± 0.0033	0.0141 ± 0.0137	-0.0062 ± 0.0038
Adversarial	0.8324 ± 0.0058	**Eopp**	0.007 ± 0.0044	0.0139 ± 0.0159	-0.0144 ± 0.0133
BiFair	0.815 ± 0.014	**Eopp**	0.03 ± 0.015	0.019 ± 0.009	-0.103 ± 0.053
FairBatch	0.8415 ± 0.0054	**Eopp**	0.0082 ± 0.0073	0.0157 ± 0.0121	-0.017 ± 0.0271
Reduction	0.8343 ± 0.0059	**Eopp**	0.0294 ± 0.0164	0.0779 ± 0.0662	-0.0396 ± 0.0455
FairGrad	0.8373 ± 0.0043	**Eopp**	0.0053 ± 0.0047	0.0099 ± 0.0146	-0.0112 ± 0.0127



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

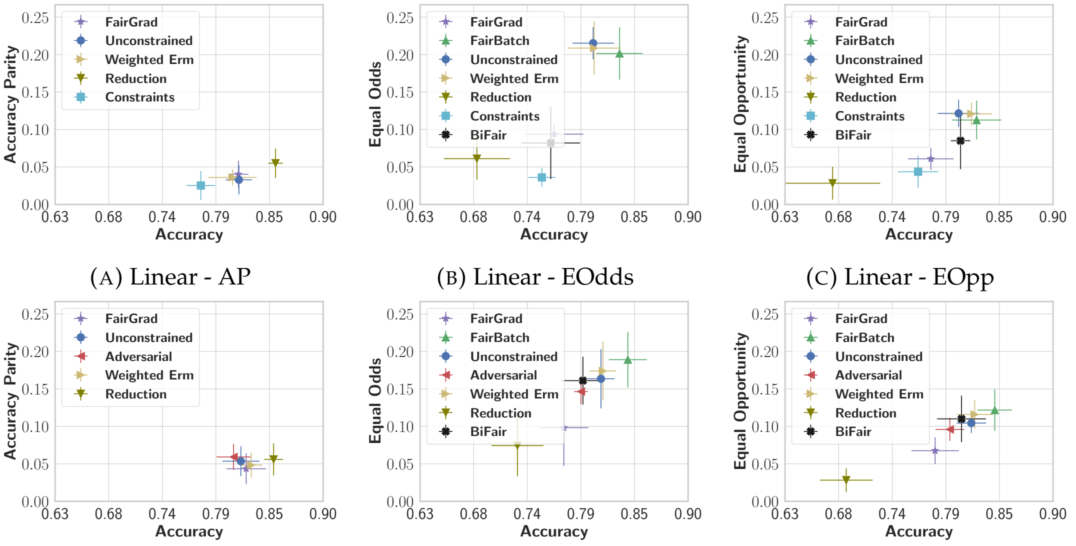
FIGURE A.2: Results for the CelebA dataset with different fairness measures.

TABLE A.3: Results for the CelebA dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8532 ± 0.0009	**AP**	0.0204 ± 0.0022	0.017 ± 0.0019	-0.0238 ± 0.0025
Constant	0.516 ± 0.0	**AP**	0.072 ± 0.0	0.084 ± 0.0	0.06 ± 0.0
Weighted ERM	0.853 ± 0.0008	**AP**	0.0193 ± 0.0021	0.0161 ± 0.0018	-0.0225 ± 0.0023
Constrained	0.799 ± 0.013	**AP**	0.01 ± 0.001	0.012 ± 0.002	0.009 ± 0.001
Reduction	0.7734 ± 0.011	**AP**	0.0242 ± 0.006	0.0282 ± 0.0071	-0.0201 ± 0.005
FairGrad	0.835 ± 0.0028	**AP**	0.0012 ± 0.0009	0.0011 ± 0.0007	-0.0014 ± 0.0011
Unconstrained	0.8532 ± 0.0009	**Eodds**	0.0499 ± 0.0019	0.0538 ± 0.0024	-0.1011 ± 0.0033
Constant	0.518 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.853 ± 0.0009	**Eodds**	0.0504 ± 0.0019	0.0532 ± 0.0024	-0.1001 ± 0.0032
Constrained	0.802 ± 0.004	**Eodds**	0.006 ± 0.001	0.01 ± 0.003	0.002 ± 0.001
BiFair	0.845 ± 0.007	**Eodds**	0.021 ± 0.005	0.02 ± 0.003	-0.036 ± 0.009
FairBatch	0.8518 ± 0.0009	**Eodds**	0.0226 ± 0.0017	0.0218 ± 0.0028	-0.0411 ± 0.0053
Reduction	0.7268 ± 0.011	**Eodds**	0.0312 ± 0.0036	0.0628 ± 0.0089	-0.0334 ± 0.0047
FairGrad	0.8274 ± 0.002	**Eodds**	0.0025 ± 0.0009	0.0038 ± 0.0018	-0.0046 ± 0.0026
Unconstrained	0.8532 ± 0.0009	**Eopp**	0.0387 ± 0.0014	0.0538 ± 0.0024	-0.1011 ± 0.0033
Constant	0.518 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.853 ± 0.0008	**Eopp**	0.0383 ± 0.0014	0.0531 ± 0.0024	-0.0999 ± 0.0032
Constrained	0.834 ± 0.005	**Eopp**	0.002 ± 0.001	0.005 ± 0.002	0.0 ± 0.0
BiFair	0.848 ± 0.004	**Eopp**	0.014 ± 0.006	0.02 ± 0.009	-0.037 ± 0.017
FairBatch	0.8498 ± 0.001	**Eopp**	0.0102 ± 0.0016	0.0142 ± 0.0022	-0.0268 ± 0.0042
Reduction	0.7358 ± 0.0159	**Eopp**	0.0698 ± 0.0118	0.1824 ± 0.0313	-0.0968 ± 0.0158
FairGrad	0.844 ± 0.0022	**Eopp**	0.0013 ± 0.0009	0.0025 ± 0.0021	-0.0028 ± 0.0018

TABLE A.4: Results for the CelebA dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.8587 ± 0.0015	**AP**	0.0184 ± 0.0014	0.0154 ± 0.0012	-0.0215 ± 0.0016
Constant	0.516 ± 0.0	**AP**	0.072 ± 0.0	0.084 ± 0.0	0.06 ± 0.0
Weighted ERM	0.8593 ± 0.0018	**AP**	0.018 ± 0.0017	0.015 ± 0.0014	-0.021 ± 0.0019
Adversarial	0.8588 ± 0.0012	**AP**	0.0178 ± 0.0014	0.0148 ± 0.0012	-0.0208 ± 0.0015
Reduction	0.7802 ± 0.0142	**AP**	0.0436 ± 0.0108	0.0508 ± 0.0123	-0.0364 ± 0.0092
FairGrad	0.8359 ± 0.0033	**AP**	0.0023 ± 0.0012	0.0025 ± 0.0015	-0.0021 ± 0.0009
Unconstrained	0.8583 ± 0.0012	**Eodds**	0.0432 ± 0.003	0.0475 ± 0.0028	-0.0893 ± 0.0049
Constant	0.518 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8589 ± 0.0009	**Eodds**	0.0419 ± 0.0021	0.0459 ± 0.0025	-0.0864 ± 0.0038
Adversarial	0.8567 ± 0.0014	**Eodds**	0.0223 ± 0.002	0.0272 ± 0.0039	-0.0511 ± 0.0073
BiFair	0.856 ± 0.004	**Eodds**	0.023 ± 0.002	0.028 ± 0.005	-0.052 ± 0.009
FairBatch	0.8533 ± 0.0037	**Eodds**	0.0217 ± 0.0014	0.0197 ± 0.0026	-0.0321 ± 0.005
Reduction	0.7021 ± 0.0323	**Eodds**	0.0813 ± 0.0253	0.1777 ± 0.0426	-0.0946 ± 0.0238
FairGrad	0.8304 ± 0.0031	**Eodds**	0.0037 ± 0.0017	0.0048 ± 0.0018	-0.0055 ± 0.0023
Unconstrained	0.8585 ± 0.0016	**Eopp**	0.0341 ± 0.002	0.0473 ± 0.003	-0.0889 ± 0.0052
Constant	0.518 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.859 ± 0.0009	**Eopp**	0.0331 ± 0.0014	0.046 ± 0.0023	-0.0866 ± 0.0035
Adversarial	0.8557 ± 0.0019	**Eopp**	0.0161 ± 0.002	0.0223 ± 0.0029	-0.0419 ± 0.0053
BiFair	0.854 ± 0.004	**Eopp**	0.015 ± 0.009	0.021 ± 0.012	-0.039 ± 0.022
FairBatch	0.8475 ± 0.0043	**Eopp**	0.0051 ± 0.0024	0.007 ± 0.0033	-0.0131 ± 0.0063
Reduction	0.765 ± 0.0149	**Eopp**	0.0533 ± 0.0124	0.1393 ± 0.033	-0.0738 ± 0.0167
FairGrad	0.8439 ± 0.0063	**Eopp**	0.0009 ± 0.0008	0.002 ± 0.0022	-0.0016 ± 0.0011



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

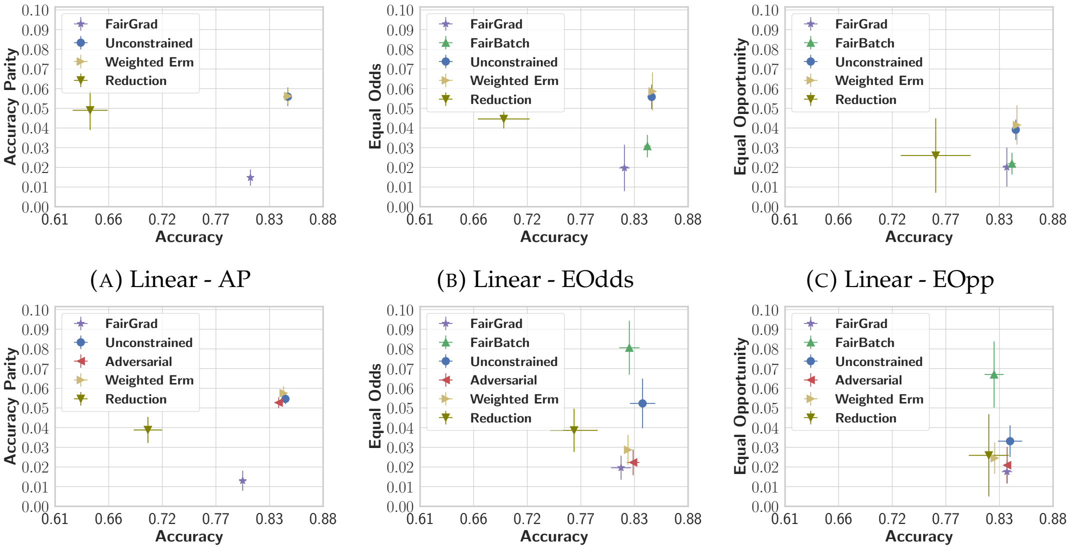
FIGURE A.3: Results for the Crime dataset with different fairness measures.

TABLE A.5: Results for the Crime dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8145 ± 0.0136	**AP**	0.0329 ± 0.0195	0.0258 ± 0.0162	-0.0399 ± 0.0229
Constant	0.734 ± 0.0	**AP**	0.272 ± 0.0	0.377 ± 0.0	0.168 ± 0.0
Weighted ERM	0.808 ± 0.0246	**AP**	0.0361 ± 0.0108	0.0284 ± 0.0091	-0.0438 ± 0.0129
Constrained	0.775 ± 0.015	**AP**	0.025 ± 0.019	0.031 ± 0.025	0.019 ± 0.014
Reduction	0.8521 ± 0.0075	**AP**	0.055 ± 0.0197	0.0426 ± 0.0147	-0.0673 ± 0.0253
FairGrad	0.814 ± 0.0102	**AP**	0.0403 ± 0.0181	0.0316 ± 0.0147	-0.049 ± 0.0218
Unconstrained	0.8035 ± 0.0212	**Eodds**	0.2152 ± 0.0215	0.1038 ± 0.0231	-0.396 ± 0.0433
Constant	0.677 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8045 ± 0.0271	**Eodds**	0.2086 ± 0.0357	0.0974 ± 0.0165	-0.3747 ± 0.0679
Constrained	0.751 ± 0.014	**Eodds**	0.036 ± 0.012	0.088 ± 0.043	0.007 ± 0.004
BiFair	0.76 ± 0.03	**Eodds**	0.082 ± 0.048	0.048 ± 0.03	-0.163 ± 0.092
FairBatch	0.8306 ± 0.0237	**Eodds**	0.2015 ± 0.035	0.1054 ± 0.0333	-0.3704 ± 0.067
Reduction	0.6842 ± 0.0339	**Eodds**	0.0611 ± 0.0281	0.0349 ± 0.0111	-0.1291 ± 0.047
FairGrad	0.7634 ± 0.03	**Eodds**	0.0938 ± 0.0144	0.0491 ± 0.016	-0.1927 ± 0.0362
Unconstrained	0.804 ± 0.0215	**Eopp**	0.1215 ± 0.0183	0.1009 ± 0.0238	-0.3852 ± 0.0549
Constant	0.697 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8171 ± 0.0213	**Eopp**	0.1209 ± 0.0154	0.0985 ± 0.0106	-0.3851 ± 0.0599
Constrained	0.762 ± 0.021	**Eopp**	0.044 ± 0.021	0.138 ± 0.066	0.0 ± 0.0
BiFair	0.806 ± 0.01	**Eopp**	0.085 ± 0.038	0.073 ± 0.042	-0.268 ± 0.112
FairBatch	0.8225 ± 0.0252	**Eopp**	0.1126 ± 0.0259	0.1002 ± 0.0281	-0.3501 ± 0.0821
Reduction	0.6747 ± 0.0488	**Eopp**	0.0283 ± 0.022	0.0413 ± 0.0375	-0.0718 ± 0.0829
FairGrad	0.7755 ± 0.0233	**Eopp**	0.0609 ± 0.0149	0.0507 ± 0.0166	-0.193 ± 0.0456

TABLE A.6: Results for the Crime dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.8165 ± 0.019	**AP**	0.0535 ± 0.0199	0.0423 ± 0.0155	-0.0648 ± 0.0251
Constant	0.734 ± 0.0	**AP**	0.272 ± 0.0	0.377 ± 0.0	0.168 ± 0.0
Weighted ERM	0.8271 ± 0.0114	**AP**	0.0483 ± 0.0167	0.0382 ± 0.0139	-0.0584 ± 0.02
Adversarial	0.809 ± 0.0175	**AP**	0.0592 ± 0.0173	0.0464 ± 0.0135	-0.0719 ± 0.0223
Reduction	0.8501 ± 0.0096	**AP**	0.0559 ± 0.0215	0.0432 ± 0.0166	-0.0685 ± 0.0269
FairGrad	0.822 ± 0.0203	**AP**	0.0434 ± 0.0206	0.0341 ± 0.0162	-0.0526 ± 0.0252
Unconstrained	0.8115 ± 0.014	**Eodds**	0.1635 ± 0.0395	0.0854 ± 0.014	-0.3326 ± 0.0649
Constant	0.677 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8135 ± 0.0137	**Eodds**	0.1739 ± 0.0394	0.0861 ± 0.0212	-0.3309 ± 0.0778
Adversarial	0.791 ± 0.007	**Eodds**	0.1464 ± 0.0168	0.0797 ± 0.0192	-0.3001 ± 0.0296
BiFair	0.793 ± 0.022	**Eodds**	0.161 ± 0.032	0.091 ± 0.025	-0.339 ± 0.048
FairBatch	0.8391 ± 0.0195	**Eodds**	0.189 ± 0.0368	0.1106 ± 0.0313	-0.3828 ± 0.0671
Reduction	0.7258 ± 0.0267	**Eodds**	0.0743 ± 0.0409	0.0553 ± 0.014	-0.1556 ± 0.0976
FairGrad	0.7734 ± 0.0251	**Eodds**	0.0982 ± 0.0513	0.0511 ± 0.0179	-0.2016 ± 0.0771
Unconstrained	0.817 ± 0.0152	**Eopp**	0.1044 ± 0.0133	0.0856 ± 0.0123	-0.3321 ± 0.0489
Constant	0.697 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8205 ± 0.0184	**Eopp**	0.1159 ± 0.0191	0.0955 ± 0.019	-0.368 ± 0.0642
Adversarial	0.795 ± 0.0148	**Eopp**	0.0959 ± 0.0153	0.0802 ± 0.0227	-0.3036 ± 0.042
BiFair	0.807 ± 0.025	**Eopp**	0.11 ± 0.031	0.091 ± 0.031	-0.351 ± 0.097
FairBatch	0.8411 ± 0.0177	**Eopp**	0.1217 ± 0.0277	0.1083 ± 0.0311	-0.3784 ± 0.0891
Reduction	0.6887 ± 0.0271	**Eopp**	0.0282 ± 0.0159	0.034 ± 0.0281	-0.0788 ± 0.0619
FairGrad	0.7799 ± 0.0243	**Eopp**	0.0675 ± 0.0179	0.0556 ± 0.0147	-0.2143 ± 0.0592



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

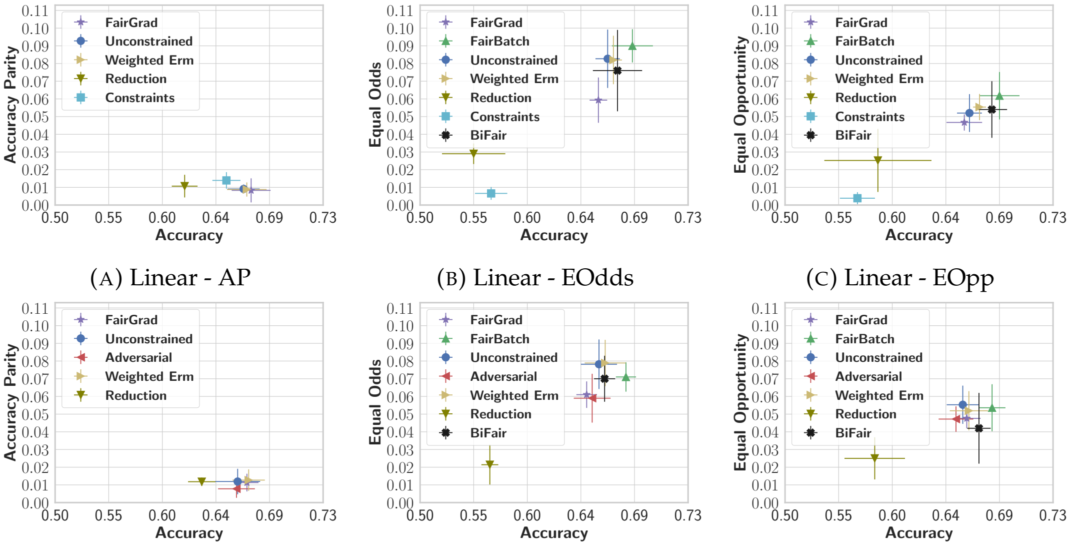
FIGURE A.4: Results for the Adult with multiple groups dataset with different fairness measures.

TABLE A.7: Results for the Adult with multiple groups dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8451 ± 0.0042	**AP**	0.0559 ± 0.0047	0.0985 ± 0.0111	-0.042 ± 0.003
Constant	0.754 ± 0.0	**AP**	0.097 ± 0.0	0.159 ± 0.0	0.024 ± 0.0
Weighted ERM	0.8454 ± 0.0032	**AP**	0.0562 ± 0.0042	0.0993 ± 0.0117	-0.0426 ± 0.0018
Reduction	0.6436 ± 0.0178	**AP**	0.049 ± 0.01	0.0493 ± 0.017	-0.0661 ± 0.0113
FairGrad	0.807 ± 0.0022	**AP**	0.0148 ± 0.0041	0.0256 ± 0.0048	-0.0107 ± 0.0045
Unconstrained	0.844 ± 0.0011	**Eodds**	0.0558 ± 0.0062	0.0578 ± 0.0069	-0.1586 ± 0.0621
Constant	0.75 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8448 ± 0.0038	**Eodds**	0.0586 ± 0.0097	0.0567 ± 0.0048	-0.1702 ± 0.0776
FairBatch	0.8396 ± 0.0034	**Eodds**	0.0308 ± 0.0057	0.0565 ± 0.0116	-0.0641 ± 0.0234
Reduction	0.6932 ± 0.0264	**Eodds**	0.0446 ± 0.0048	0.0806 ± 0.043	-0.0896 ± 0.0278
FairGrad	0.8162 ± 0.0052	**Eodds**	0.0197 ± 0.0118	0.0373 ± 0.0233	-0.0493 ± 0.0403
Unconstrained	0.8431 ± 0.002	**Eopp**	0.0391 ± 0.0052	0.0297 ± 0.0131	-0.169 ± 0.0565
Constant	0.762 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8443 ± 0.0038	**Eopp**	0.0415 ± 0.01	0.0316 ± 0.0145	-0.1767 ± 0.0797
FairBatch	0.8392 ± 0.004	**Eopp**	0.0219 ± 0.0055	0.05 ± 0.0133	-0.0749 ± 0.0285
Reduction	0.7615 ± 0.0357	**Eopp**	0.026 ± 0.0189	0.0487 ± 0.0378	-0.1115 ± 0.0867
FairGrad	0.834 ± 0.0044	**Eopp**	0.0201 ± 0.0099	0.0442 ± 0.0415	-0.0679 ± 0.0808

TABLE A.8: Results for the Adult with multiple groups dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8427 ± 0.0041	**AP**	0.0546 ± 0.0026	0.0966 ± 0.0098	-0.0421 ± 0.0022
Constant	0.754 ± 0.0	**AP**	0.097 ± 0.0	0.159 ± 0.0	0.024 ± 0.0
Weighted ERM	0.8408 ± 0.0031	**AP**	0.0575 ± 0.0035	0.101 ± 0.0106	-0.0443 ± 0.0026
Adversarial	0.8358 ± 0.0043	**AP**	0.0527 ± 0.0028	0.0889 ± 0.0066	-0.0401 ± 0.0022
Reduction	0.7025 ± 0.0144	**AP**	0.0388 ± 0.0066	0.054 ± 0.0151	-0.0525 ± 0.0099
FairGrad	0.7991 ± 0.0036	**AP**	0.013 ± 0.0051	0.0257 ± 0.0138	-0.0125 ± 0.0043
Unconstrained	0.8347 ± 0.0129	**Eodds**	0.0523 ± 0.0126	0.0495 ± 0.0166	-0.1772 ± 0.0512
Constant	0.75 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8199 ± 0.002	**Eodds**	0.0287 ± 0.0076	0.0274 ± 0.0177	-0.1013 ± 0.0543
Adversarial	0.8251 ± 0.0064	**Eodds**	0.0223 ± 0.0065	0.0451 ± 0.0308	-0.0667 ± 0.0559
FairBatch	0.8212 ± 0.0103	**Eodds**	0.0806 ± 0.0137	0.0522 ± 0.0076	-0.2545 ± 0.0525
Reduction	0.7649 ± 0.0241	**Eodds**	0.0386 ± 0.011	0.044 ± 0.02	-0.0954 ± 0.0465
FairGrad	0.8128 ± 0.0102	**Eodds**	0.0196 ± 0.0061	0.0392 ± 0.0176	-0.0443 ± 0.0342
Unconstrained	0.8373 ± 0.0123	**Eopp**	0.0331 ± 0.008	0.0183 ± 0.0045	-0.1587 ± 0.0643
Constant	0.762 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8216 ± 0.0031	**Eopp**	0.0245 ± 0.008	0.0243 ± 0.0196	-0.1016 ± 0.0543
Adversarial	0.8343 ± 0.0036	**Eopp**	0.0209 ± 0.0093	0.0327 ± 0.013	-0.0927 ± 0.0589
FairBatch	0.821 ± 0.0097	**Eopp**	0.067 ± 0.0168	0.047 ± 0.0113	-0.2484 ± 0.0535
Reduction	0.8156 ± 0.0204	**Eopp**	0.0259 ± 0.0209	0.0472 ± 0.0325	-0.0968 ± 0.1117
FairGrad	0.8341 ± 0.0053	**Eopp**	0.0176 ± 0.0059	0.0302 ± 0.0272	-0.0731 ± 0.0543



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

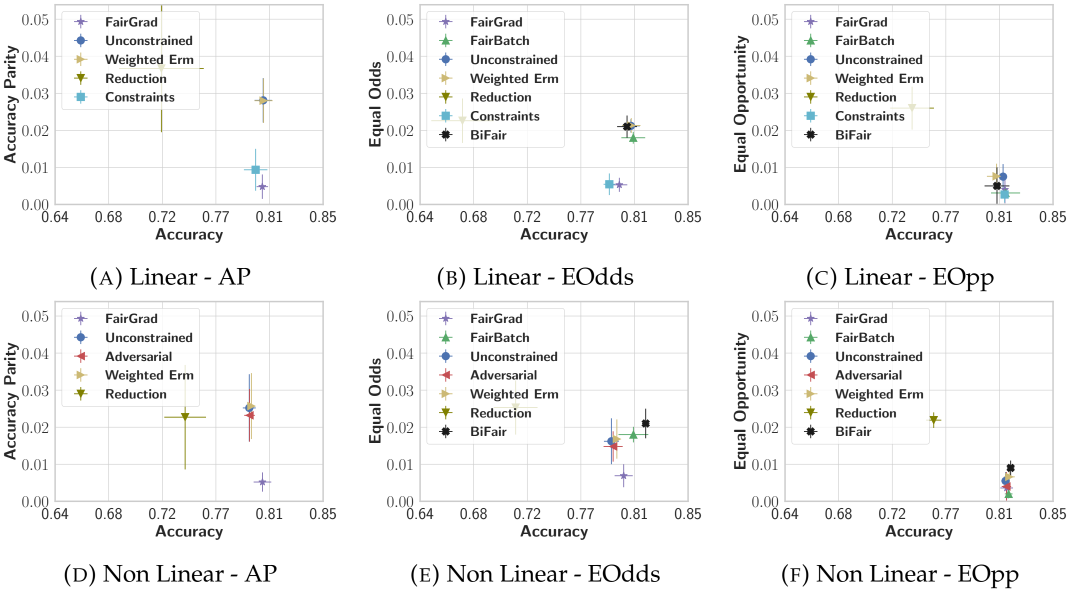
FIGURE A.5: Results for the Compas dataset with different fairness measures.

TABLE A.9: Results for the Compas dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.6644 ± 0.0137	**AP**	0.0091 ± 0.0025	0.0076 ± 0.0031	-0.0107 ± 0.004
Constant	0.545 ± 0.0	**AP**	0.066 ± 0.0	0.085 ± 0.0	0.047 ± 0.0
Weighted ERM	0.6671 ± 0.0169	**AP**	0.0088 ± 0.004	0.0061 ± 0.0028	-0.0115 ± 0.0051
Constrained	0.65 ± 0.012	**AP**	0.014 ± 0.005	0.018 ± 0.006	0.009 ± 0.003
Reduction	0.6141 ± 0.011	**AP**	0.0107 ± 0.0064	0.009 ± 0.006	-0.0124 ± 0.0086
FairGrad	0.6708 ± 0.0166	**AP**	0.0083 ± 0.0068	0.0057 ± 0.0048	-0.0108 ± 0.0088
Unconstrained	0.6636 ± 0.0104	**Eodds**	0.0827 ± 0.0165	0.0758 ± 0.0133	-0.1553 ± 0.0259
Constant	0.527 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.6685 ± 0.0073	**Eodds**	0.082 ± 0.0137	0.0697 ± 0.0115	-0.1618 ± 0.0222
Constrained	0.564 ± 0.014	**Eodds**	0.007 ± 0.004	0.014 ± 0.011	0.002 ± 0.001
BiFair	0.672 ± 0.021	**Eodds**	0.076 ± 0.023	0.071 ± 0.025	-0.15 ± 0.039
FairBatch	0.6847 ± 0.0175	**Eodds**	0.09 ± 0.0094	0.0854 ± 0.0149	-0.1727 ± 0.0304
Reduction	0.5493 ± 0.027	**Eodds**	0.029 ± 0.0058	0.0268 ± 0.0062	-0.0622 ± 0.0219
FairGrad	0.6557 ± 0.0075	**Eodds**	0.0593 ± 0.0128	0.0524 ± 0.0102	-0.1241 ± 0.0202
Unconstrained	0.6609 ± 0.0106	**Eopp**	0.052 ± 0.0107	0.062 ± 0.0145	-0.1461 ± 0.0286
Constant	0.55 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.6695 ± 0.0055	**Eopp**	0.0554 ± 0.0074	0.0659 ± 0.0107	-0.1557 ± 0.0194
Constrained	0.565 ± 0.015	**Eopp**	0.004 ± 0.003	0.011 ± 0.009	0.0 ± 0.0
BiFair	0.68 ± 0.013	**Eopp**	0.054 ± 0.016	0.064 ± 0.022	-0.15 ± 0.044
FairBatch	0.6865 ± 0.0171	**Eopp**	0.0618 ± 0.0134	0.0715 ± 0.0173	-0.1755 ± 0.0364
Reduction	0.5828 ± 0.0457	**Eopp**	0.0252 ± 0.0178	0.03 ± 0.0216	-0.0707 ± 0.0498
FairGrad	0.6565 ± 0.0152	**Eopp**	0.0467 ± 0.0046	0.0554 ± 0.0071	-0.1313 ± 0.0119

TABLE A.10: Results for the Compas dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.6593 ± 0.0192	**AP**	0.0119 ± 0.0072	0.0095 ± 0.004	-0.0144 ± 0.0107
Constant	0.545 ± 0.0	**AP**	0.066 ± 0.0	0.085 ± 0.0	0.047 ± 0.0
Weighted ERM	0.6687 ± 0.0138	**AP**	0.0127 ± 0.0061	0.011 ± 0.0034	-0.0145 ± 0.0099
Adversarial	0.6583 ± 0.0157	**AP**	0.0078 ± 0.0051	0.0066 ± 0.0044	-0.009 ± 0.0069
Reduction	0.6287 ± 0.0117	**AP**	0.0118 ± 0.0024	0.0103 ± 0.0062	-0.0134 ± 0.0024
FairGrad	0.6672 ± 0.0099	**AP**	0.0113 ± 0.005	0.0095 ± 0.0023	-0.0131 ± 0.0082
Unconstrained	0.6562 ± 0.0154	**Eodds**	0.0782 ± 0.014	0.0715 ± 0.0136	-0.1521 ± 0.0277
Constant	0.527 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.6615 ± 0.0175	**Eodds**	0.0789 ± 0.0131	0.0726 ± 0.0077	-0.1496 ± 0.0313
Adversarial	0.6504 ± 0.0157	**Eodds**	0.059 ± 0.0138	0.0549 ± 0.0107	-0.1294 ± 0.0183
BiFair	0.661 ± 0.009	**Eodds**	0.07 ± 0.013	0.068 ± 0.018	-0.133 ± 0.016
FairBatch	0.6792 ± 0.0086	**Eodds**	0.071 ± 0.0083	0.0663 ± 0.0091	-0.1508 ± 0.0304
Reduction	0.5631 ± 0.0072	**Eodds**	0.0214 ± 0.0112	0.024 ± 0.0102	-0.0489 ± 0.0363
FairGrad	0.6457 ± 0.0088	**Eodds**	0.061 ± 0.0075	0.0564 ± 0.0065	-0.127 ± 0.0081
Unconstrained	0.6552 ± 0.0137	**Eopp**	0.0553 ± 0.0108	0.0659 ± 0.015	-0.1552 ± 0.0281
Constant	0.55 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.6604 ± 0.0163	**Eopp**	0.0519 ± 0.0111	0.0618 ± 0.0148	-0.1458 ± 0.0299
Adversarial	0.6494 ± 0.0148	**Eopp**	0.0472 ± 0.0072	0.0563 ± 0.0108	-0.1327 ± 0.0183
BiFair	0.669 ± 0.01	**Eopp**	0.042 ± 0.02	0.05 ± 0.025	-0.117 ± 0.055
FairBatch	0.6802 ± 0.0114	**Eopp**	0.0536 ± 0.0133	0.062 ± 0.0167	-0.1526 ± 0.0367
Reduction	0.5801 ± 0.0258	**Eopp**	0.025 ± 0.0119	0.0296 ± 0.0145	-0.0702 ± 0.0333
FairGrad	0.6586 ± 0.0118	**Eopp**	0.0476 ± 0.0056	0.0563 ± 0.0067	-0.1339 ± 0.0163



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

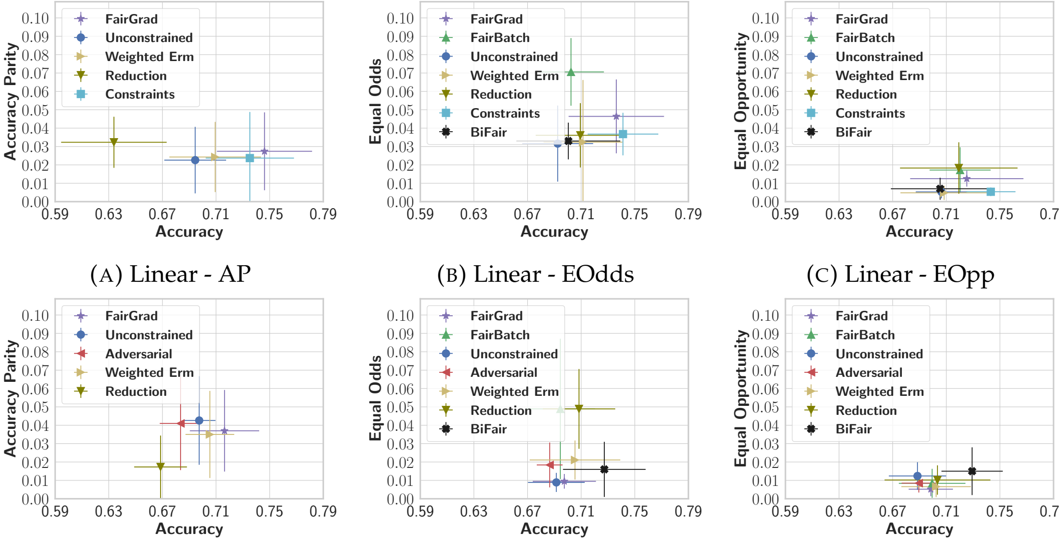
FIGURE A.6: Results for the Dutch dataset with different fairness measures.

TABLE A.11: Results for the Dutch dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8049 ± 0.007	**AP**	0.0281 ± 0.006	0.0281 ± 0.006	-0.0282 ± 0.0061
Constant	0.524 ± 0.0	**AP**	0.151 ± 0.0	0.152 ± 0.0	0.15 ± 0.0
Weighted ERM	0.8052 ± 0.0073	**AP**	0.028 ± 0.006	0.028 ± 0.006	-0.0281 ± 0.006
Constrained	0.799 ± 0.009	**AP**	0.009 ± 0.006	0.009 ± 0.006	0.009 ± 0.006
Reduction	0.723 ± 0.0341	**AP**	0.0367 ± 0.0172	0.0368 ± 0.0172	-0.0367 ± 0.0172
FairGrad	0.8042 ± 0.0046	**AP**	0.0048 ± 0.0033	0.0048 ± 0.0033	-0.0048 ± 0.0032
Unconstrained	0.8071 ± 0.0072	**Eodds**	0.0212 ± 0.0018	0.0322 ± 0.009	-0.0256 ± 0.0052
Constant	0.522 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8074 ± 0.0074	**Eodds**	0.0213 ± 0.002	0.032 ± 0.0086	-0.0254 ± 0.0051
Constrained	0.79 ± 0.005	**Eodds**	0.005 ± 0.003	0.009 ± 0.005	0.002 ± 0.002
BiFair	0.804 ± 0.008	**Eodds**	0.021 ± 0.003	0.025 ± 0.004	-0.033 ± 0.01
FairBatch	0.809 ± 0.0096	**Eodds**	0.018 ± 0.0016	0.0262 ± 0.0039	-0.0211 ± 0.004
Reduction	0.6716 ± 0.0251	**Eodds**	0.0226 ± 0.006	0.0333 ± 0.0107	-0.0404 ± 0.0213
FairGrad	0.7978 ± 0.0064	**Eodds**	0.0053 ± 0.0019	0.007 ± 0.0019	-0.009 ± 0.0049
Unconstrained	0.8129 ± 0.0021	**Eopp**	0.0075 ± 0.0034	0.0107 ± 0.0049	-0.0193 ± 0.0086
Constant	0.524 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8077 ± 0.0078	**Eopp**	0.0076 ± 0.0034	0.011 ± 0.0049	-0.0196 ± 0.0087
Constrained	0.814 ± 0.003	**Eopp**	0.003 ± 0.002	0.007 ± 0.006	0.0 ± 0.0
BiFair	0.808 ± 0.01	**Eopp**	0.005 ± 0.005	0.008 ± 0.007	-0.012 ± 0.012
FairBatch	0.8149 ± 0.0117	**Eopp**	0.0031 ± 0.0014	0.0044 ± 0.002	-0.0079 ± 0.0036
Reduction	0.7397 ± 0.0176	**Eopp**	0.026 ± 0.0058	0.0669 ± 0.0149	-0.0372 ± 0.0083
FairGrad	0.8144 ± 0.0021	**Eopp**	0.004 ± 0.0037	0.006 ± 0.0052	-0.0099 ± 0.0097

TABLE A.12: Results for the Dutch dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.7937 ± 0.0052	**AP**	0.0252 ± 0.0091	0.0252 ± 0.009	-0.0252 ± 0.0091
Constant	0.524 ± 0.0	**AP**	0.151 ± 0.0	0.152 ± 0.0	0.15 ± 0.0
Weighted ERM	0.7954 ± 0.0023	**AP**	0.0257 ± 0.0089	0.0257 ± 0.0089	-0.0257 ± 0.0089
Adversarial	0.7939 ± 0.0043	**AP**	0.0232 ± 0.0071	0.0232 ± 0.0071	-0.0232 ± 0.007
Reduction	0.7421 ± 0.0168	**AP**	0.0227 ± 0.0141	0.0227 ± 0.0142	-0.0227 ± 0.0141
FairGrad	0.8043 ± 0.0071	**AP**	0.0052 ± 0.0026	0.0052 ± 0.0026	-0.0052 ± 0.0026
Unconstrained	0.7914 ± 0.006	**Eodds**	0.0162 ± 0.0062	0.0193 ± 0.0071	-0.0263 ± 0.0142
Constant	0.522 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7958 ± 0.0027	**Eodds**	0.0168 ± 0.0053	0.0202 ± 0.0048	-0.0261 ± 0.0131
Adversarial	0.7928 ± 0.0077	**Eodds**	0.0148 ± 0.0041	0.0202 ± 0.0066	-0.0211 ± 0.006
BiFair	0.819 ± 0.003	**Eodds**	0.021 ± 0.004	0.03 ± 0.005	-0.028 ± 0.007
FairBatch	0.8091 ± 0.012	**Eodds**	0.018 ± 0.0021	0.0254 ± 0.0058	-0.0248 ± 0.0062
Reduction	0.7144 ± 0.0176	**Eodds**	0.0253 ± 0.0073	0.0347 ± 0.0123	-0.0323 ± 0.0064
FairGrad	0.8013 ± 0.0073	**Eodds**	0.0069 ± 0.0031	0.0099 ± 0.0038	-0.0095 ± 0.0068
Unconstrained	0.8149 ± 0.0034	**Eopp**	0.0055 ± 0.0024	0.0079 ± 0.0035	-0.014 ± 0.0061
Constant	0.524 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8179 ± 0.0044	**Eopp**	0.0066 ± 0.0026	0.0095 ± 0.0037	-0.017 ± 0.0065
Adversarial	0.8156 ± 0.0038	**Eopp**	0.004 ± 0.0039	0.0058 ± 0.0057	-0.0102 ± 0.01
BiFair	0.819 ± 0.003	**Eopp**	0.009 ± 0.002	0.012 ± 0.003	-0.022 ± 0.006
FairBatch	0.8174 ± 0.0031	**Eopp**	0.002 ± 0.0012	0.0029 ± 0.0017	-0.0052 ± 0.0031
Reduction	0.7571 ± 0.0061	**Eopp**	0.0219 ± 0.0021	0.0563 ± 0.0054	-0.0313 ± 0.0028
FairGrad	0.8158 ± 0.0051	**Eopp**	0.0036 ± 0.0031	0.0051 ± 0.0045	-0.0092 ± 0.0079



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

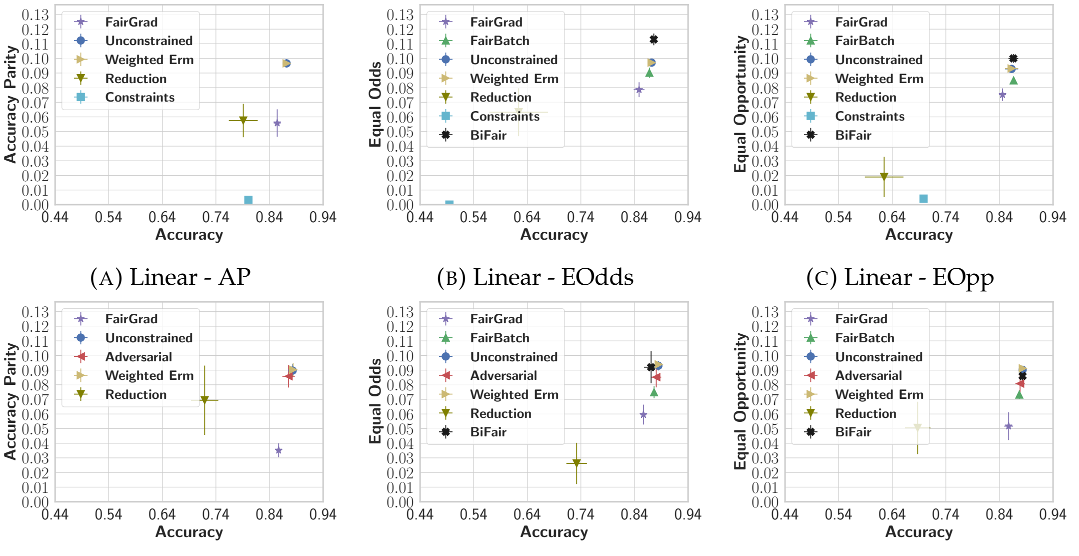
FIGURE A.7: Results for the German dataset with different fairness measures.

TABLE A.13: Results for the German dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.692 ± 0.0232	**AP**	0.0226 ± 0.0181	0.0169 ± 0.0111	-0.0284 ± 0.0256
Constant	0.73 ± 0.0	**AP**	0.05 ± 0.0	0.069 ± 0.0	0.031 ± 0.0
Weighted ERM	0.707 ± 0.0344	**AP**	0.0243 ± 0.0191	0.0186 ± 0.0113	-0.0299 ± 0.027
Constrained	0.733 ± 0.033	**AP**	0.024 ± 0.025	0.032 ± 0.033	0.015 ± 0.017
Reduction	0.631 ± 0.0396	**AP**	0.0323 ± 0.0139	0.0286 ± 0.0202	-0.036 ± 0.0185
FairGrad	0.744 ± 0.0357	**AP**	0.0274 ± 0.0212	0.0215 ± 0.0123	-0.0334 ± 0.0306
Unconstrained	0.69 ± 0.0266	**Eodds**	0.0316 ± 0.0207	0.0499 ± 0.0341	-0.0618 ± 0.0471
Constant	0.7 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.709 ± 0.0296	**Eodds**	0.0324 ± 0.0338	0.0461 ± 0.046	-0.055 ± 0.0626
Constrained	0.739 ± 0.027	**Eodds**	0.037 ± 0.012	0.072 ± 0.025	0.01 ± 0.004
BiFair	0.698 ± 0.039	**Eodds**	0.033 ± 0.01	0.052 ± 0.023	-0.059 ± 0.029
FairBatch	0.7 ± 0.0247	**Eodds**	0.0706 ± 0.0184	0.1102 ± 0.0489	-0.1134 ± 0.0518
Reduction	0.707 ± 0.0335	**Eodds**	0.0361 ± 0.0175	0.0716 ± 0.056	-0.0576 ± 0.0266
FairGrad	0.734 ± 0.0358	**Eodds**	0.0464 ± 0.0201	0.0784 ± 0.0232	-0.0721 ± 0.0496
Unconstrained	0.704 ± 0.0193	**Eopp**	0.0053 ± 0.0035	0.0096 ± 0.004	-0.0116 ± 0.0117
Constant	0.7 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.706 ± 0.0328	**Eopp**	0.0048 ± 0.0039	0.0097 ± 0.0091	-0.0096 ± 0.0092
Constrained	0.741 ± 0.019	**Eopp**	0.005 ± 0.002	0.015 ± 0.006	0.0 ± 0.0
BiFair	0.703 ± 0.037	**Eopp**	0.007 ± 0.006	0.014 ± 0.015	-0.013 ± 0.015
FairBatch	0.718 ± 0.0229	**Eopp**	0.0172 ± 0.0124	0.0272 ± 0.0187	-0.0416 ± 0.0396
Reduction	0.717 ± 0.0441	**Eopp**	0.0183 ± 0.014	0.036 ± 0.0254	-0.0372 ± 0.0407
FairGrad	0.723 ± 0.0425	**Eopp**	0.0125 ± 0.0043	0.0212 ± 0.0087	-0.0288 ± 0.0162

TABLE A.14: Results for the German dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.695 ± 0.0122	**AP**	0.0426 ± 0.0241	0.0314 ± 0.0144	-0.0537 ± 0.0345
Constant	0.73 ± 0.0	**AP**	0.05 ± 0.0	0.069 ± 0.0	0.031 ± 0.0
Weighted ERM	0.703 ± 0.0183	**AP**	0.035 ± 0.0237	0.0265 ± 0.0138	-0.0436 ± 0.0338
Adversarial	0.681 ± 0.0156	**AP**	0.041 ± 0.0254	0.0327 ± 0.0165	-0.0492 ± 0.0368
Reduction	0.666 ± 0.0198	**AP**	0.0173 ± 0.0171	0.0131 ± 0.0115	-0.0215 ± 0.0231
FairGrad	0.714 ± 0.026	**AP**	0.037 ± 0.0222	0.0291 ± 0.0119	-0.0448 ± 0.0331
Unconstrained	0.689 ± 0.0213	**Eodds**	0.0089 ± 0.0052	0.0117 ± 0.0045	-0.0144 ± 0.0116
Constant	0.7 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.703 ± 0.034	**Eodds**	0.0211 ± 0.0106	0.0305 ± 0.0186	-0.0372 ± 0.0158
Adversarial	0.684 ± 0.0097	**Eodds**	0.0184 ± 0.0122	0.0263 ± 0.0201	-0.0339 ± 0.0237
BiFair	0.725 ± 0.031	**Eodds**	0.016 ± 0.015	0.021 ± 0.018	-0.027 ± 0.018
FairBatch	0.692 ± 0.026	**Eodds**	0.0489 ± 0.0382	0.0607 ± 0.0446	-0.0882 ± 0.0983
Reduction	0.706 ± 0.0272	**Eodds**	0.0489 ± 0.0217	0.0742 ± 0.0266	-0.0717 ± 0.051
FairGrad	0.695 ± 0.0237	**Eodds**	0.0095 ± 0.004	0.0121 ± 0.0046	-0.0175 ± 0.0076
Unconstrained	0.686 ± 0.0215	**Eopp**	0.0124 ± 0.0075	0.0227 ± 0.0128	-0.0269 ± 0.0227
Constant	0.7 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7 ± 0.0261	**Eopp**	0.0066 ± 0.0057	0.0131 ± 0.0071	-0.0133 ± 0.0173
Adversarial	0.687 ± 0.0129	**Eopp**	0.0085 ± 0.0051	0.0203 ± 0.0147	-0.0137 ± 0.0099
BiFair	0.727 ± 0.023	**Eopp**	0.015 ± 0.013	0.023 ± 0.019	-0.036 ± 0.038
FairBatch	0.697 ± 0.025	**Eopp**	0.0084 ± 0.0079	0.0235 ± 0.0226	-0.0102 ± 0.0094
Reduction	0.701 ± 0.0397	**Eopp**	0.0102 ± 0.008	0.0242 ± 0.024	-0.0167 ± 0.0134
FairGrad	0.696 ± 0.0166	**Eopp**	0.0052 ± 0.0038	0.0093 ± 0.0064	-0.0115 ± 0.0108



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

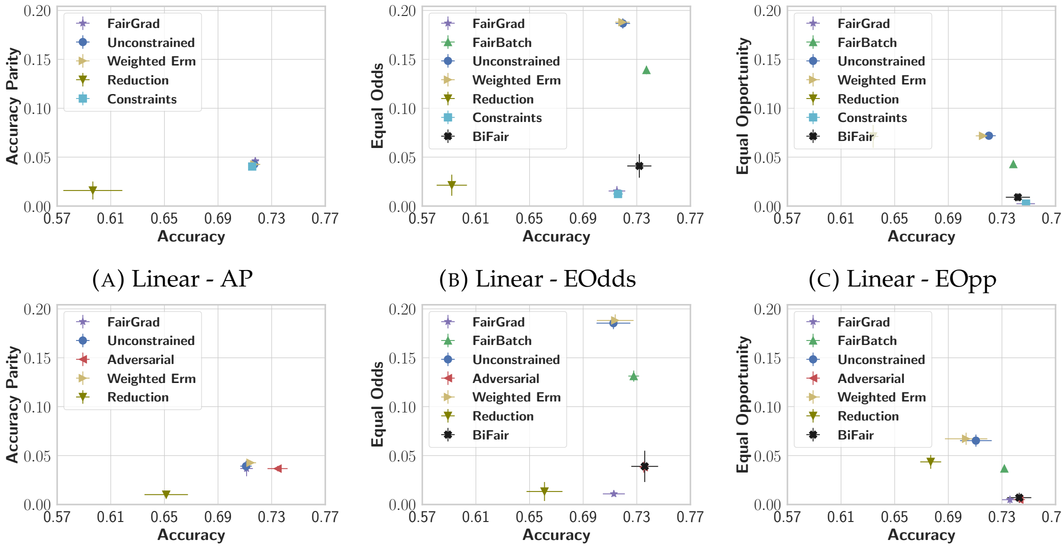
FIGURE A.8: Results for the Gaussian dataset with different fairness measures.

TABLE A.15: Results for the Gaussian dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.8689 ± 0.0037	**AP**	0.0966 ± 0.0029	0.0957 ± 0.0028	-0.0974 ± 0.0036
Constant	0.497 ± 0.0	**AP**	0.001 ± 0.0	0.001 ± 0.0	0.001 ± 0.0
Weighted ERM	0.869 ± 0.0039	**AP**	0.0966 ± 0.0026	0.0957 ± 0.0023	-0.0974 ± 0.0034
Constrained	0.799 ± 0.004	**AP**	0.003 ± 0.002	0.003 ± 0.002	0.003 ± 0.002
Reduction	0.7891 ± 0.0266	**AP**	0.0575 ± 0.0114	0.057 ± 0.0118	-0.0579 ± 0.0111
FairGrad	0.8516 ± 0.0064	**AP**	0.0558 ± 0.0094	0.0553 ± 0.0093	-0.0562 ± 0.0096
Unconstrained	0.869 ± 0.0037	**Eodds**	0.0971 ± 0.0026	0.1872 ± 0.0067	-0.1896 ± 0.0056
Constant	0.499 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.869 ± 0.0039	**Eodds**	0.0971 ± 0.0023	0.1869 ± 0.0063	-0.1894 ± 0.0051
Constrained	0.497 ± 0.003	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
BiFair	0.873 ± 0.004	**Eodds**	0.113 ± 0.004	0.21 ± 0.007	-0.213 ± 0.004
FairBatch	0.8649 ± 0.0025	**Eodds**	0.0902 ± 0.0035	0.1717 ± 0.0046	-0.1719 ± 0.0079
Reduction	0.6241 ± 0.054	**Eodds**	0.0632 ± 0.0164	0.0732 ± 0.0198	-0.074 ± 0.0226
FairGrad	0.8459 ± 0.01	**Eodds**	0.0786 ± 0.0051	0.1504 ± 0.0102	-0.1527 ± 0.0142
Unconstrained	0.8598 ± 0.0121	**Eopp**	0.0928 ± 0.0012	0.1845 ± 0.0041	-0.1869 ± 0.0041
Constant	0.498 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8599 ± 0.0121	**Eopp**	0.0931 ± 0.0011	0.1849 ± 0.004	-0.1874 ± 0.004
Constrained	0.698 ± 0.005	**Eopp**	0.004 ± 0.002	0.008 ± 0.005	0.0 ± 0.0
BiFair	0.863 ± 0.009	**Eopp**	0.1 ± 0.003	0.2 ± 0.007	-0.202 ± 0.006
FairBatch	0.8635 ± 0.0024	**Eopp**	0.085 ± 0.0023	0.17 ± 0.0032	-0.1702 ± 0.0065
Reduction	0.6251 ± 0.0355	**Eopp**	0.0189 ± 0.0138	0.0379 ± 0.0271	-0.0378 ± 0.0282
FairGrad	0.8431 ± 0.0065	**Eopp**	0.0752 ± 0.0043	0.1494 ± 0.0087	-0.1514 ± 0.0094

TABLE A.16: Results for the Gaussian dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.88 ± 0.0038	**AP**	0.0897 ± 0.0045	0.0888 ± 0.0035	-0.0905 ± 0.0055
Constant	0.497 ± 0.0	**AP**	0.001 ± 0.0	0.001 ± 0.0	0.001 ± 0.0
Weighted ERM	0.8809 ± 0.0048	**AP**	0.0903 ± 0.0045	0.0894 ± 0.0033	-0.0911 ± 0.0057
Adversarial	0.8725 ± 0.0115	**AP**	0.0858 ± 0.0077	0.0851 ± 0.0076	-0.0866 ± 0.0081
Reduction	0.718 ± 0.0251	**AP**	0.0694 ± 0.0237	0.0699 ± 0.0236	-0.0689 ± 0.0239
FairGrad	0.8542 ± 0.0047	**AP**	0.0352 ± 0.0047	0.0349 ± 0.0048	-0.0355 ± 0.0046
Unconstrained	0.8814 ± 0.0024	**Eodds**	0.093 ± 0.0032	0.1807 ± 0.0066	-0.183 ± 0.005
Constant	0.499 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8821 ± 0.0031	**Eodds**	0.0939 ± 0.0013	0.1826 ± 0.0042	-0.185 ± 0.0033
Adversarial	0.8775 ± 0.0091	**Eodds**	0.0852 ± 0.007	0.1643 ± 0.0125	-0.1666 ± 0.0146
BiFair	0.868 ± 0.013	**Eodds**	0.092 ± 0.011	0.167 ± 0.035	-0.168 ± 0.031
FairBatch	0.8735 ± 0.0032	**Eodds**	0.0749 ± 0.0041	0.1455 ± 0.0059	-0.1456 ± 0.0056
Reduction	0.7309 ± 0.0189	**Eodds**	0.0262 ± 0.0141	0.0438 ± 0.0257	-0.0435 ± 0.0265
FairGrad	0.8539 ± 0.0056	**Eodds**	0.0596 ± 0.0068	0.1013 ± 0.0147	-0.1025 ± 0.0144
Unconstrained	0.8801 ± 0.004	**Eopp**	0.0902 ± 0.0017	0.1792 ± 0.0041	-0.1816 ± 0.0053
Constant	0.498 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.8805 ± 0.0046	**Eopp**	0.0912 ± 0.0008	0.1812 ± 0.0024	-0.1837 ± 0.0045
Adversarial	0.8754 ± 0.0086	**Eopp**	0.0808 ± 0.0066	0.1605 ± 0.0128	-0.1628 ± 0.0143
BiFair	0.88 ± 0.003	**Eopp**	0.086 ± 0.005	0.17 ± 0.013	-0.172 ± 0.009
FairBatch	0.874 ± 0.0035	**Eopp**	0.0733 ± 0.0029	0.1465 ± 0.0054	-0.1467 ± 0.0066
Reduction	0.6868 ± 0.0234	**Eopp**	0.0505 ± 0.0179	0.1015 ± 0.0359	-0.1005 ± 0.036
FairGrad	0.8543 ± 0.0082	**Eopp**	0.0517 ± 0.0095	0.1028 ± 0.0191	-0.1041 ± 0.0192



(A) Linear - AP (B) Linear - EOdds (C) Linear - EOpp

(D) Non Linear - AP (E) Non Linear - EOdds (F) Non Linear - EOpp

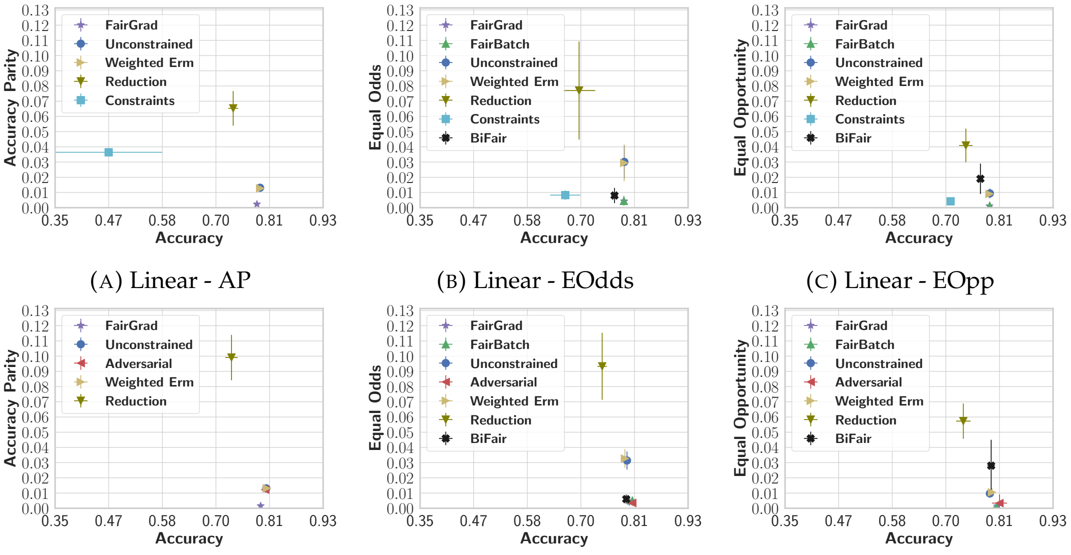
FIGURE A.9: Results for the Twitter Sentiment dataset with different fairness measures.

TABLE A.17: Results for the Twitter Sentiment dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.7211 ± 0.004	**AP**	0.0426 ± 0.0011	0.0426 ± 0.0011	-0.0426 ± 0.0011
Constant	0.5 ± 0.0	**AP**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7212 ± 0.0044	**AP**	0.0426 ± 0.0011	0.0426 ± 0.0011	-0.0426 ± 0.0011
Constrained	0.72 ± 0.002	**AP**	0.04 ± 0.003	0.04 ± 0.003	0.04 ± 0.003
Reduction	0.6008 ± 0.022	**AP**	0.0159 ± 0.0092	0.0159 ± 0.0092	-0.0159 ± 0.0092
FairGrad	0.7219 ± 0.0027	**AP**	0.0462 ± 0.0021	0.0462 ± 0.0021	-0.0462 ± 0.0021
Unconstrained	0.7237 ± 0.0054	**Eodds**	0.1867 ± 0.0052	0.2287 ± 0.0078	-0.2288 ± 0.0078
Constant	0.5 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7234 ± 0.0054	**Eodds**	0.188 ± 0.0033	0.2314 ± 0.0056	-0.2315 ± 0.0056
Constrained	0.72 ± 0.004	**Eodds**	0.012 ± 0.002	0.019 ± 0.005	0.006 ± 0.005
BiFair	0.736 ± 0.009	**Eodds**	0.041 ± 0.012	0.056 ± 0.022	-0.056 ± 0.022
FairBatch	0.7413 ± 0.0014	**Eodds**	0.1391 ± 0.0043	0.1755 ± 0.0084	-0.1756 ± 0.0084
Reduction	0.5962 ± 0.0113	**Eodds**	0.0213 ± 0.0108	0.0314 ± 0.0211	-0.0314 ± 0.021
FairGrad	0.7193 ± 0.0062	**Eodds**	0.0154 ± 0.0051	0.0204 ± 0.0098	-0.0204 ± 0.0098
Unconstrained	0.7244 ± 0.0051	**Eopp**	0.0719 ± 0.0012	0.1439 ± 0.0023	-0.1438 ± 0.0023
Constant	0.5 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.72 ± 0.0054	**Eopp**	0.0718 ± 0.0013	0.1437 ± 0.0026	-0.1436 ± 0.0026
Constrained	0.752 ± 0.004	**Eopp**	0.002 ± 0.001	0.005 ± 0.001	0.0 ± 0.0
BiFair	0.746 ± 0.009	**Eopp**	0.009 ± 0.004	0.017 ± 0.009	-0.017 ± 0.009
FairBatch	0.7426 ± 0.001	**Eopp**	0.0429 ± 0.0005	0.0858 ± 0.0011	-0.0858 ± 0.0011
Reduction	0.6381 ± 0.0039	**Eopp**	0.0712 ± 0.0117	0.1424 ± 0.0234	-0.1425 ± 0.0234
FairGrad	0.7518 ± 0.0069	**Eopp**	0.0024 ± 0.002	0.0049 ± 0.004	-0.0049 ± 0.004

TABLE A.18: Results for the Twitter Sentiment dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	ACCURACY ↑	FAIRNESS			
		MEASURE	MEAN ABS. ↓	MAXIMUM	MINIMUM
Unconstrained	0.715 ± 0.0043	**AP**	0.0392 ± 0.0055	0.0392 ± 0.0055	-0.0392 ± 0.0055
Constant	0.5 ± 0.0	**AP**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7183 ± 0.0042	**AP**	0.0427 ± 0.0019	0.0427 ± 0.0019	-0.0427 ± 0.0019
Adversarial	0.7385 ± 0.0075	**AP**	0.0367 ± 0.0027	0.0367 ± 0.0027	-0.0368 ± 0.0027
Reduction	0.6555 ± 0.0162	**AP**	0.0101 ± 0.0038	0.0101 ± 0.0038	-0.0101 ± 0.0038
FairGrad	0.7154 ± 0.0047	**AP**	0.0368 ± 0.0079	0.0367 ± 0.0078	-0.0368 ± 0.0079
Unconstrained	0.7167 ± 0.0126	**Eodds**	0.1854 ± 0.0061	0.2349 ± 0.0091	-0.235 ± 0.0091
Constant	0.5 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.718 ± 0.0137	**Eodds**	0.1882 ± 0.0062	0.2379 ± 0.0073	-0.2381 ± 0.0073
Adversarial	0.7393 ± 0.0024	**Eodds**	0.0382 ± 0.0056	0.06 ± 0.0151	-0.06 ± 0.0151
BiFair	0.74 ± 0.01	**Eodds**	0.039 ± 0.016	0.058 ± 0.017	-0.058 ± 0.017
FairBatch	0.7318 ± 0.004	**Eodds**	0.1313 ± 0.0057	0.1724 ± 0.0055	-0.1725 ± 0.0055
Reduction	0.6653 ± 0.0134	**Eodds**	0.0133 ± 0.0097	0.0199 ± 0.0172	-0.0199 ± 0.0173
FairGrad	0.717 ± 0.0082	**Eodds**	0.0109 ± 0.0027	0.0165 ± 0.0053	-0.0165 ± 0.0053
Unconstrained	0.7147 ± 0.0118	**Eopp**	0.0653 ± 0.0062	0.1306 ± 0.0124	-0.1306 ± 0.0124
Constant	0.5 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7074 ± 0.0158	**Eopp**	0.0672 ± 0.0062	0.1346 ± 0.0125	-0.1345 ± 0.0125
Adversarial	0.7471 ± 0.0042	**Eopp**	0.005 ± 0.0035	0.0099 ± 0.007	-0.0099 ± 0.007
BiFair	0.747 ± 0.009	**Eopp**	0.007 ± 0.005	0.013 ± 0.01	-0.013 ± 0.01
FairBatch	0.7359 ± 0.0011	**Eopp**	0.0368 ± 0.0012	0.0736 ± 0.0025	-0.0736 ± 0.0025
Reduction	0.681 ± 0.0078	**Eopp**	0.0436 ± 0.0071	0.0871 ± 0.0143	-0.0871 ± 0.0143
FairGrad	0.7401 ± 0.0059	**Eopp**	0.0049 ± 0.0041	0.0099 ± 0.0083	-0.0099 ± 0.0083



(A) Linear - AP

(B) Linear - EOdds

(C) Linear - EOpp

(D) Non Linear - AP

(E) Non Linear - EOdds

(F) Non Linear - EOpp

FIGURE A.10: Results for the Folktables Adult dataset with different fairness measures.

TABLE A.19: Results for the Folktables Adult dataset with Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (L)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.7905 ± 0.0033	**AP**	0.0131 ± 0.0021	0.0123 ± 0.0021	-0.0138 ± 0.0022
Constant	0.666 ± 0.0	**AP**	0.053 ± 0.0	0.056 ± 0.0	0.051 ± 0.0
Weighted ERM	0.7906 ± 0.0032	**AP**	0.0127 ± 0.0023	0.0119 ± 0.0022	-0.0134 ± 0.0024
Constrained	0.467 ± 0.115	**AP**	0.036 ± 0.003	0.039 ± 0.003	0.034 ± 0.003
Reduction	0.733 ± 0.0106	**AP**	0.0653 ± 0.0114	0.0614 ± 0.011	-0.0692 ± 0.0118
FairGrad	0.7837 ± 0.0049	**AP**	0.0023 ± 0.0009	0.0023 ± 0.001	-0.0022 ± 0.0008
Unconstrained	0.789 ± 0.0026	**Eodds**	0.0301 ± 0.011	0.0377 ± 0.0153	-0.0458 ± 0.0184
Constant	0.667 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7886 ± 0.0032	**Eodds**	0.0294 ± 0.012	0.0364 ± 0.0169	-0.0443 ± 0.0206
Constrained	0.663 ± 0.032	**Eodds**	0.008 ± 0.003	0.013 ± 0.004	0.004 ± 0.002
BiFair	0.768 ± 0.007	**Eodds**	0.008 ± 0.005	0.011 ± 0.006	-0.011 ± 0.008
FairBatch	0.788 ± 0.0027	**Eodds**	0.0045 ± 0.0033	0.0069 ± 0.0065	-0.0063 ± 0.0049
Reduction	0.6922 ± 0.0346	**Eodds**	0.077 ± 0.0322	0.0761 ± 0.0257	-0.0903 ± 0.0378
FairGrad	0.7885 ± 0.0027	**Eodds**	0.0043 ± 0.0019	0.0073 ± 0.0037	-0.0068 ± 0.0045
Unconstrained	0.7902 ± 0.0038	**Eopp**	0.0094 ± 0.0031	0.0162 ± 0.0053	-0.0215 ± 0.0071
Constant	0.667 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7893 ± 0.0031	**Eopp**	0.009 ± 0.003	0.0155 ± 0.0051	-0.0206 ± 0.0069
Constrained	0.706 ± 0.002	**Eopp**	0.004 ± 0.0	0.01 ± 0.001	0.0 ± 0.0
BiFair	0.77 ± 0.002	**Eopp**	0.019 ± 0.01	0.033 ± 0.017	-0.044 ± 0.023
FairBatch	0.79 ± 0.0031	**Eopp**	0.0012 ± 0.0015	0.0022 ± 0.0026	-0.0026 ± 0.0034
Reduction	0.7388 ± 0.0144	**Eopp**	0.0409 ± 0.0111	0.0932 ± 0.025	-0.0704 ± 0.0194
FairGrad	0.7893 ± 0.0026	**Eopp**	0.0011 ± 0.0009	0.0024 ± 0.002	-0.0021 ± 0.0016

TABLE A.20: Results for the Folktables Adult dataset with Non Linear Models. All the results are averaged over 5 runs. Here MEAN ABS., MAXIMUM, and MINIMUM represent the mean absolute fairness value, the fairness level of the most well-off group, and the fairness level of the worst-off group, respectively.

METHOD (NL)	**ACCURACY ↑**	**FAIRNESS**			
		MEASURE	**MEAN ABS. ↓**	**MAXIMUM**	**MINIMUM**
Unconstrained	0.8037 ± 0.0037	**AP**	0.0131 ± 0.0017	0.0123 ± 0.0016	-0.0139 ± 0.0017
Constant	0.666 ± 0.0	**AP**	0.053 ± 0.0	0.056 ± 0.0	0.051 ± 0.0
Weighted ERM	0.8046 ± 0.0049	**AP**	0.0131 ± 0.0014	0.0123 ± 0.0014	-0.0138 ± 0.0015
Adversarial	0.8016 ± 0.0053	**AP**	0.0122 ± 0.0016	0.0115 ± 0.0015	-0.0129 ± 0.0016
Reduction	0.7293 ± 0.0133	**AP**	0.0991 ± 0.0149	0.0932 ± 0.0139	-0.1051 ± 0.016
FairGrad	0.7917 ± 0.0025	**AP**	0.0016 ± 0.0011	0.0016 ± 0.0011	-0.0016 ± 0.001
Unconstrained	0.7947 ± 0.0078	**Eodds**	0.0314 ± 0.0059	0.0373 ± 0.0058	-0.0454 ± 0.0066
Constant	0.667 ± 0.0	**Eodds**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7902 ± 0.0049	**Eodds**	0.0327 ± 0.0061	0.04 ± 0.0067	-0.0488 ± 0.0077
Adversarial	0.806 ± 0.0047	**Eodds**	0.0035 ± 0.0018	0.0051 ± 0.0021	-0.0053 ± 0.0028
BiFair	0.793 ± 0.006	**Eodds**	0.006 ± 0.003	0.007 ± 0.003	-0.007 ± 0.004
FairBatch	0.8061 ± 0.0044	**Eodds**	0.0051 ± 0.0015	0.0087 ± 0.0048	-0.0084 ± 0.0029
Reduction	0.7416 ± 0.01	**Eodds**	0.0933 ± 0.022	0.1517 ± 0.0311	-0.1244 ± 0.026
FairGrad	0.7997 ± 0.0087	**Eodds**	0.0045 ± 0.0029	0.0067 ± 0.0045	-0.0071 ± 0.0058
Unconstrained	0.7902 ± 0.0044	**Eopp**	0.0097 ± 0.0026	0.0168 ± 0.0045	-0.0222 ± 0.006
Constant	0.667 ± 0.0	**Eopp**	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
Weighted ERM	0.7947 ± 0.0022	**Eopp**	0.0105 ± 0.0027	0.0181 ± 0.0047	-0.024 ± 0.0062
Adversarial	0.8108 ± 0.0161	**Eopp**	0.0034 ± 0.0057	0.0041 ± 0.0057	-0.0095 ± 0.017
BiFair	0.793 ± 0.008	**Eopp**	0.028 ± 0.017	0.048 ± 0.029	-0.064 ± 0.039
FairBatch	0.8038 ± 0.0063	**Eopp**	0.0008 ± 0.0005	0.0014 ± 0.0009	-0.0018 ± 0.0012
Reduction	0.7334 ± 0.0155	**Eopp**	0.0573 ± 0.0116	0.1307 ± 0.0265	-0.0986 ± 0.0199
FairGrad	0.8058 ± 0.0035	**Eopp**	0.0014 ± 0.0014	0.003 ± 0.0031	-0.0026 ± 0.0024

Appendix B

Fair NLP Models with Differentially Private Text Encoders

In this appendix, we provide details that were omitted in Chapter 7. First, in Section B.1, we describe in more details the error in privacy analysis of previous works. We then describe experimental settings and extended results in Section B.2.

B.1 Error in Privacy Analysis of Previous Work

As briefly mentioned in Section 7.3.4, we found a critical error in the differential privacy analysis made in previous work by Lyu, He, and Li (2020). This error is then reproduced in subsequent work by Plant, Gkatzia, and Giuffrida (2021). In this section, we explain this error and its consequences for the formal privacy guarantees of these methods, and provide a correction.

Recall from Section 7.2 that to achieve ϵ-DP with the Laplace mechanism, one must calibrate the scale of the Laplace noise needed to the L1 sensitivity of the encoded representation (see Eq. 7.2). This sensitivity bounds the worst-case change in L1 norm for any two arbitrary encoded user inputs $\mathbf{x}$ and $\mathbf{x}'$ of dimension D.

In order to bound the L1 sensitivity, Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021) propose to bound each entry of the encoded input $\mathbf{x} \in \mathbb{R}^D$ in the $[0, 1]$ range. Specifically, they normalize as follows:

$$\mathbf{x} \leftarrow \mathbf{x} - \min(\mathbf{x}) / (\max(\mathbf{x}) - \min(\mathbf{x})) \tag{B.1}$$

where $\min(\mathbf{x})$ and $\max(\mathbf{x})$ are respectively the minimum and maximum values in the vector $\mathbf{x}$. Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021) incorrectly claim that this allows to bound the L1 sensitivity by 1 and thus add Laplace noise of scale $\frac{1}{\epsilon}$. In fact, the sensitivity can be as large as D, as can be seen by considering the two inputs $\mathbf{x} = [0, 1, \dots, 1]_D$ and $\mathbf{x}' = [1, 0, \dots, 0]$ for which $\|\mathbf{x} - \mathbf{x}'\|_1 = D$. Therefore, to achieve ϵ-DP, the scale of the Laplace noise should be $\frac{D}{\epsilon}$ (i.e., D times larger than what the authors use). As a consequence, the differential privacy provided by their method are D times worse than claimed by Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021): the ϵ values they report should be multiplied by D, which leads to essentially void privacy guarantees.

While Lyu, He, and Li (2020) claim to follow the approach of Shokri and Shmatikov (2015), they missed the fact that Shokri and Shmatikov (2015) do account for multiple dimensions by scaling the noise to the number of entries (denoted by c in their paper) that are submitted to the server, see pseudo-code in Figure 12 of Shokri and Shmatikov (2015). In contrast to Lyu, He, and Li (2020) and Plant, Gkatzia, and Giuffrida (2021), our normalization in Eq. 7.3 guarantees by design that the L1 sensitivity is bounded by 2.

B.2 Experiments

B.2.1 Privacy metric

Leakage: We compute the leakage using a sklearn's MLPClassifier. We use the validation set of the original dataset as the train and the test set of the original dataset as the test.

Minimum Description Length (MDL) is a information-theoretic probing measure which captures the strength of regularity in the data. In this work, we employ the online coding approach (Voita and Titov, 2020) to calculate MDL. Online coding captures the regularity by characterizing the effort required to achieve a certain level of accuracy. Here, a portion of data is transmitted to the receiver at each step, which then uses all the data in the previous steps to understand the regularity in the current step. The regularity is obtained by training the model on the previously received data and then evaluating it on the current portion of the data.

Borrowing, the terminology from Voita and Titov (2020), consider a dataset D consisting of $\{(x_1, y_1), \cdots, (x_n, y_n)\}$ pairs, where the x_i's are the data representation, and the y_i's are the task label. In our case, x_i is the output of the encoder, and y_i is the sensitive attribute associated with the underlying text. Following the standard information theory setting, consider a sender Alice who wants to transmit labels $y_{1:n} = \{y_1 \cdots, y_n\}$ to a receiver Bob, and both of them have access to the data representation $x_{1:n} = \{x_1 \cdots, x_n\}$. In order to transmit labels $y_{1:n}$ efficiently (as few bits possible), Alice encodes $y_{1:n}$ using a model $p(y|x)$. According to Shannon-Huffman code, the minimum bits required to transmit these labels losslessly is:

$$L_p(y_{1:n}|x_{1:n}) = -\sum_{i=1}^{n} \log_2 p(y_i|x_i).$$

In the online coding setting of MDL, the labels are transmitted in blocks of n timesteps $t_0 < t_1 < \cdots t_n$. Alice starts by encoding $y_{1:t_1}$ with a uniform code, then both Alice and Bob learn a model $p_{\theta_1}(y|x)$ that predicts y from x using data $\{(x_i, y_i)\}_{i=1}^{t1}$. Alice then uses this model to communicate the next data block $y_{t_1:t_2}$, and both learns a new model using larger chunk of data $\{(x_i, y_i)\}_{i=1}^{t2}$. This continues till the whole set of labels $y_{1:n}$ is transmitted. The total code length required for transmission using this setting is given as:

$$\begin{aligned} L_{online}(y_{1:n}|x_{1:n}) = t_1 \log_2 C - \\ \sum_{i=1}^{n-1} \log_2 p_{\theta_i}(y_{t_i+1:t_i}|x_{t_i+1:t_i}). \end{aligned} \tag{B.2}$$

where $y_i \in \{1, 2, \cdots, C\}$. In our case, the online code length $L_{online}(y_{1:n}|x_{1:n})$ is shorter, if it is easier for probing model to perform well with fewer training instances. This implies that the sensitive information is more easily available in the encoder's representation.

We compute MDL using sklearn's MLPClassifier at timesteps corresponding to 0.1%, 0.2%, 0.4%, 0.8%, 1.6%, 3.2%, 6.25%, 12.5%, 25%, 50% and 100% of each dataset as suggested by Voita and Titov (2020).

B.2.2 Datasets

Twitter Sentiment (Blodgett, Green, and O'Connor, 2016) consists of 200k tweets annotated with a binary sentiment label and a binary "race" attribute corresponding to African American English (AAE) vs. Standard American English (SAE) speakers. The initial representation of tweets are obtained from a Deepmoji encoder Felbo et al., 2017. The dataset is evenly balanced with respect to the four sentiment-race subgroup combinations. To create bias in the training data, we follow Elazar and Goldberg (2018) and change the race proportion in each sentiment class to have 40% AAE-happy, 10% AAE-sad, 10% SAE-happy, and 40% SAE-sad. Test data remains balanced. This setup is particularly challenging regarding privacy and fairness, as the model may exploit the correlation between the protected attribute and the main class label, which is reinforced due to skewing. The mismatch between the train-test distribution is also relevant for our setup, where the system may be trained on publicly available datasets or collected via an opt-in policy and may therefore not closely resemble the test distribution. This dataset is made available for research purposes only.[1]

Bias in Bios De-Arteaga et al., 2019 consists of 393,423 textual biographies annotated with an occupation label (28 classes) and a binary gender attribute. Similar to Ravfogel et al. (2020), we encode each biography with BERT Devlin et al., 2019, using the last hidden state over the CLS token. We use the same train-valid-test split as De-Arteaga et al. (2019). As the dataset was collected by scrapping the web, it tends to reflect common gender stereotypes and contains explicit gender indicators (e.g., pronouns), making it more challenging to prevent models from relying on these gendered words. It is also more complex than Twitter Sentiment in terms of the number of classes. Dataset is released under MIT License.[2]

CelebA Liu et al., 2015 consists of over 200,000 images of the human face, alongside with 40 binary attributes labels describing the content of the images. Following the standard setting as described in Lohaus, Perrot, and Von Luxburg, 2020, we use 38 of these attributes as features, "Smiling" as the class label, and "Sex" as the sensitive attribute. We use 60% of the data as train, 20% as validation, and the remaining as the test split. This dataset is available for non-commercial research purposes.[3]

Adult Income Kohavi, 1996 consists of a U.S. 1994 Census database segment and has 48842 instances with 14 features each. We apply the pre-processing as proposed by Wu, Zhang, and Wu, 2019 resulting in a total of 9 features for each instance. The objective is to predict whether a given data point earns more than fifty thousand U.S. dollars or less. We consider sex (binary) as the sensitive attribute. Like CelebA, We

[1] http://slanglab.cs.umass.edu/TwitterAAE/

[2] https://github.com/Microsoft/biosbias

[3] https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

use 60% of the data as train, 20% as validation, and the remaining as the test split. The license of the dataset is unknown, however it is commonly used in several fairness papers and is available at Dua, Graff, et al., 2017.

B.2.3 Model Architecture

Twitter Sentiment. The encoder consists of two layers with ReLU activation and a fixed dropout of 0.1. The classifier is linear, and the adversarial branch consists of three layers. We use a fixed dropout of 0.1 in all the layers with ReLU activation, apart from the last layer.

Bias in Bios. The encoder consists of three layers and a fixed dropout of 0.1. The classifier also consists of three layers, and the adversarial branch consists of two layers. We use a fixed dropout of 0.1 in all the layers with ReLU activation, apart from the last layer.

In case of *Adult Income* and *CelebA* dataset we use the same model as for *Twitter Sentiment*.

B.2.4 Hyperparameters

For all our experiments, we use Adam optimizer with a learning rate of 0.001 and batch size of 2000. We give additional tuning details of the different methods below. A single experiment takes about 30 minutes to run on Intel Xenon CPU. We will also provide the PyTorch model description in the README of the source code for easier reproduction.

- `Adversarial`: We perform a grid search over λ varying it between 0.1 to 3.0 with an interval of 0.2. Moreover, following previous work Lample et al., 2017; Adi et al., 2019, instead of a constant λ, we increase it over the epochs using the update scheme $\lambda_i = 2/(1 + e^{-p_i}) - 1$, where p_i is the scaled version of the epoch number. We also experimented with increasing the λ linearly, as well as keeping it constant, but found the above update scheme to perform the best in various settings. We also use this scheme in all other adversarial approaches.

- `Adversarial + Multiple`: Similar to `Adversarial`, we vary λ between 0.1 to 3.0 with an interval of 0.2. Apart from λ, `Adversarial + Multiple` has an additional hyperparameter λ_{ort} which corresponds to the weight given to the orthogonality loss component. We vary λ_{ort} between 0.1 and 1.0. Here, we do a simultaneous grid search over λ and λ_{ort} resulting in 150 runs for each seed. We fix the number of the adversary to three which is the same as the original implementation by Han, Baldwin, and Cohn, 2021.

- `FEDERATE`: In order to have comparable number of runs to `Adversarial + Multiple`, we experiments with following ϵ values: $8, 9, 10, 11, 12, 13, 14, 15, 16, 20$. Similar to above approach, we do a simultaneous grid search over λ and ϵ resulting in 150 runs for each seed.

- `INLP`: In the case of `INLP`, we always debias the representation after the penultimate classifier layer and before the final layer, which is consistent with the setting considered by the authors (Ravfogel et al., 2020). We also observe that this choice empirically led to the best results. We vary the number of iterations as a part of hyperparameter tuning. For Bias in Bios we vary the iterations

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓	MDL ↑
`Random`	50.00 ± 0.00	0.00 ± 0.00	–	104.64 ± 0.11
`Unconstrained`	85.70 ± 0.21	12.25 ± 2.07	81.3 ± 0.89	67.82 ± 1.46
`INLP`	84.81 ± 0.47	12.69 ± 4.66	66.00 ± 1.32	100.17 ± 1.65
`Noise`	85.12 ± 0.47	12.49 ± 0.58	59.01 ± 0.65	103.93 ± 0.24
`Adversarial`	85.34 ± 0.22	7.83 ± 0.97	87.00 ± 2.22	46.61 ± 5.52
`Adversarial + Multiple`	84.92 ± 0.12	5.79 ± 1.44	84.38 ± 2.07	51.11 ± 4.06
`FEDERATE`	84.81 ± 0.34	2.68 ± 0.60	65.49 ± 3.48	98.53 ± 4.51

TABLE B.1: Test results on CelebA dataset with fixed Relaxation Threshold of 1.0. Fairness is measured by TPR-Gap (lower is better), while privacy is measured by Leakage (lower is better) and MDL (higher is better). The MDL achieved by `Random` gives an upper bound for that particular dataset. The results have been averaged over 5 different seeds.

between 15 and 45, while for Twitter Sentiment we vary between 2 to 7. We found that in case of Bias in Bios, performing less than 15 iterations resulted in the same behaviour as `Unconstrained` model over validation set while more than 45 iterations resulted in a random classifier. We observed the same in the Twitter Sentiment before 2 and after 7 iterations, respectively.

B.2.5 Extended Experiments

Tables B.1–B.2 present detailed results on CelebA and Adult Income dataset respectively. In terms of fairness over both the datasets, we observe that adversarial-based approaches induce a more fair model than Unconstrained or Noise, with `FEDERATE` outperforming all other methods. Interestingly, unlike Twitter Sentiment and Bias in Bios, all approaches have comparable accuracy, including `Noise` and `INLP`. We believe this to be the case due to these datasets being relatively more challenging than CelebA and Adult Income. As observed previously, purely adversarial-based approaches leak significantly more information than the DP-based approaches in terms of privacy. We observe that `Noise` and `INLP` performs marginally better in privacy than `FEDERATE`; however, they suffer significantly in the fairness metric. In fact, they induce fairness levels which are similar to `Unconstrained`.

Overall, the results show `FEDERATE` as the only viable choice to induce a fairer model and make its representation private while maintaining comparable accuracy. These observations are in line with previous experiments described in Sec. 7.5.1

B.2.6 Additional Results

Tables B.3–B.5 present detailed results on Twitter Sentiment with different relaxation thresholds, which were summarized in Figure 7.3.

Table B.6 provides the detailed privacy-fairness results which were summarized in Figure 7.5.

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓	MDL ↑
Random	50.00 ± 0.00	0.00 ± 0.00	-	20.15 ± 0.083
Unconstrained	83.41 ± 0.32	12.73 ± 7.17	78.19 ± 1.0	16.38 ± 0.46
INLP	83.11 ± 0.51	3.91 ± 2.43	74.54 ± 0.67	19.93 ± 0.35
Noise	82.87 ± 0.37	8.01 ± 1.18	68.12 ± 0.94	19.38 ± 0.33
Adversarial	83.14 ± 0.53	7.02 ± 3.31	78.2 ± 0.18	16.1 ± 0.36
Adversarial + Multiple	83.14 ± 0.25	3.55 ± 2.16	81.37 ± 0.98	13.5 ± 1.09
FEDERATE	82.29 ± 0.9	2.73 ± 2.18	70.25 ± 4.81	18.1 ± 2.79

Table B.2: Test results on Adult Income dataset with fixed Relaxation Threshold of 1.0. Fairness is measured by TPR-Gap (lower is better), while privacy is measured by Leakage (lower is better) and MDL (higher is better). The MDL achieved by Random gives an upper bound for that particular dataset. The results have been averaged over 5 different seeds.

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓
Unconstrained	72.54 ± 0.57	27.17 ± 1.76	87.18 ± 0.32
Noise	71.87 ± 0.56	25.14 ± 3.47	71.75 ± 2.99
Adversarial	75.49 ± 0.71	8.47 ± 3.5	88.03 ± 0.24
Adversarial + Multiple	75.6 ± 0.53	7.74 ± 4.17	88.01 ± 0.28
FEDERATE	75.34 ± 0.56	5.46 ± 3.59	62.31 ± 5.69

Table B.3: Test set results on Twitter Sentiment dataset (scores averaged over 5 different seeds, RT=0.0).

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓
Unconstrained	70.57 ± 0.98	20.68 ± 0.99	82.91 ± 1.65
Noise	70.47 ± 0.43	19.84 ± 0.91	66.83 ± 3.32
Adversarial	74.09 ± 1.56	3.03 ± 2.65	88.14 ± 0.18
Adversarial + Multiple	74.44 ± 0.62	1.07 ± 0.74	87.98 ± 0.36
FEDERATE	74.24 ± 1.25	0.89 ± 0.46	61.92 ± 5.04

Table B.4: Test set results on Twitter Sentiment dataset (scores averaged over 5 different seeds, RT=3.0).

Method	Accuracy ↑	TPR-gap ↓	Leakage ↓
Unconstrained	70.57 ± 0.98	20.68 ± 0.99	82.91 ± 1.65
Noise	70.47 ± 0.43	19.84 ± 0.91	66.83 ± 3.32
Adversarial	70.8 ± 2.77	1.72 ± 1.5	88.2 ± 0.24
Adversarial + Multiple	67.39 ± 1.16	1.0 ± 0.8	88.01 ± 0.12
FEDERATE	73.97 ± 1.6	1.4 ± 1.22	60.38 ± 5.46

Table B.5: Test set results on Twitter Sentiment dataset (scores averaged over 5 different seeds, RT=10.0).

Method	ϵ	Twitter Sentiment		Bias in Bios	
		Accuracy ↑	Leakage ↓	Accuracy ↑	Leakage ↓
`Noise`	8.0	71.3	60.59	64.75	56
`FEDERATE`	8.0	74.89	56.91	64.78	54.4
`Noise`	10.0	71.63	65.57	70.86	57.7
`FEDERATE`	10.0	75.25	60.55	70.97	56.5
`Noise`	12.0	71.76	66.04	75.01	58.4
`FEDERATE`	12.0	75.31	53.31	75.01	57
`Noise`	14.0	71.7	67.98	76.74	59
`FEDERATE`	14.0	75.3	57.29	76.83	56.3
`Noise`	16.0	71.7	67.69	77.77	60.3
`FEDERATE`	16.0	75.56	61.98	77.89	57.9

TABLE B.6: Accuracy-privacy trade-off for different noise level (as captured by ϵ).

Appendix C

Fair Without Leveling Down: α-Intersectional Fairness

C.1 Intersectional Property

In this section, we prove the intersectional property stated in Section 5.4. The proof follows the same procedure as described by Foulds et al. (2020). The intersectional property states that:

Proposition. Let the model h_θ be (α, γ)-intersectionally fair over the set of groups defined by $\mathcal{G} = A_1 \times \cdots A_p$. Let $1 \leq s_1 \leq \cdots \leq s_k \leq p$, and $\mathcal{P} = A_{s_1} \times \cdots A_{s_k}$ be the Cartesian product of the sensitive axes where $s_j \in \mathbb{N}^+$. Then, h_θ is (α, γ)-intersectionally fair over $\mathcal{P}$.

The essential idea of the proof is to show that the maximum and the minimum group wise performance in $\mathcal{P}$ is bounded by the maximum and the minimum group wise performance in $\mathcal{G}$. After proving the above, then using Proposition 1, we can show that IF_α over $\mathcal{G}$ is higher than IF_α over $\mathcal{P}$.

Define $E = A_1 \times \ldots \times A_{a-1} \times A_{a+1} \ldots \times A_{k-1} \times A_{k+1} \times \ldots \times A_p$, the Cartesian product of the protected attributes included in $\mathcal{G}$ but not in $\mathcal{P}$. Then for any model h_θ, $y \in \text{Range}(h_\theta)$,

$$
\begin{aligned}
& \max_{\mathbf{g} \in \mathcal{P}: P(\mathbf{g}|\theta) > 0} P_{h_\theta}(h_\theta(\mathbf{x}) = y | \mathcal{P} = \mathbf{g}) \\
= & \max_{\mathbf{g} \in \mathcal{P}: P(\mathbf{g}|\theta) > 0} \sum_{\mathbf{e} \in E} P_{h_\theta}(h_\theta(\mathbf{x}) = y | E = \mathbf{e}, \mathbf{g}) \\
& \qquad\qquad\qquad\qquad P_{h_\theta}(E = e | \mathbf{g}) \\
\leq & \max_{\mathbf{g} \in \mathcal{P}: P(\mathbf{g}|\theta) > 0} \sum_{\mathbf{e} \in E} \max_{\mathbf{e}' \in E : P_{h_\theta}(E = \mathbf{e}' | \mathbf{g}) > 0} \\
& \big(P_{h_\theta}(h_\theta(\mathbf{x}) = y | E = \mathbf{e}', \mathbf{g})\big) \times P_\theta(E = \mathbf{e} | \mathbf{g}) \\
= & \max_{\mathbf{g} \in \mathcal{P}: P(\mathbf{g}|\theta) > 0} \max_{\mathbf{e}' \in E : P_\theta(E = \mathbf{e}' | \mathbf{g}, \theta) > 0} \\
& P_{h_\theta}(h_\theta(\mathbf{x}) = y | E = \mathbf{e}', \mathbf{g}) \\
= & \max_{\mathbf{s}' \in \mathcal{G}: P(\mathbf{s}'|\theta) > 0} P_{M,\theta}(M(\mathbf{x}) = y | \mathbf{s}')
\end{aligned}
$$

Method	BA ↑	Best Off ↑	Worst Off ↑	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.8 + 0.01	0.84 + 0.01	0.45 + 0.04	0.62 +/- 0.03	0.43 +/- 0.01
`Adversarial`	0.8 + 0.01	0.84 + 0.01	0.46 + 0.04	0.6 +/- 0.04	0.44 +/- 0.01
`FairGrad`	0.78 + 0.01	0.85 + 0.02	0.44 + 0.04	0.66 +/- 0.02	0.43 +/- 0.03
`INLP`	0.8 + 0.0	0.85 + 0.03	0.52 + 0.05	0.49 +/- 0.04	0.41 +/- 0.02
`Fair MixUp`	0.79 + 0.01	0.85 + 0.03	0.48 + 0.05	0.57 +/- 0.05	0.43 +/- 0.04

(A) Results on CelebA

Method	BA ↑	Best Off ↑	Worst Off ↑	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.68 + 0.02	0.87 + 0.05	0.61 + 0.04	0.36 +/- 0.01	0.38 +/- 0.09
`Adversarial`	0.7 + 0.01	0.81 + 0.05	0.55 + 0.08	0.39 +/- 0.03	0.45 +/- 0.07
`FairGrad`	0.68 + 0.02	0.88 + 0.04	0.64 + 0.09	0.32 +/- 0.03	0.35 +/- 0.07
`INLP`	0.68 + 0.01	0.84 + 0.05	0.66 + 0.1	0.24 +/- 0.03	0.44 +/- 0.08
`Fair MixUp`	0.7 + 0.01	0.81 + 0.05	0.54 + 0.05	0.41 +/- 0.02	0.44 +/- 0.07

(B) Results on Numeracy

Method	BA ↑	Best Off ↑	Worst Off ↑	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.79 + 0.01	0.96 + 0.01	0.77 + 0.03	0.22 +/- 0.03	0.2 +/- 0.03
`Adversarial`	0.76 + 0.0	0.97 + 0.01	0.81 + 0.04	0.18 +/- 0.04	0.21 +/- 0.03
`FairGrad`	0.76 + 0.02	0.95 + 0.01	0.78 + 0.03	0.2 +/- 0.04	0.25 +/- 0.03
`INLP`	0.67 + 0.01	0.73 + 0.03	0.38 + 0.03	0.65 +/- 0.05	0.56 +/- 0.03
`Fair MixUp`	0.76 + 0.01	0.98 + 0.0	0.84 + 0.02	0.15 +/- 0.02	0.16 +/- 0.01

(C) Results on Twitter Hate Speech

Method	BA ↑	Best Off ↑	Worst Off ↑	DF ↓	$\text{IF}_{\alpha=0.5}$ ↓
`Unconstrained`	0.63 + 0.01	0.77 + 0.02	0.47 + 0.07	0.49 +/- 0.05	0.5 +/- 0.02
`Adversarial`	0.63 + 0.01	0.82 + 0.05	0.51 + 0.1	0.47 +/- 0.06	0.45 +/- 0.05
`FairGrad`	0.63 + 0.01	0.76 + 0.01	0.47 + 0.06	0.48 +/- 0.04	0.52 +/- 0.02
`INLP`	0.63 + 0.01	0.76 + 0.02	0.51 + 0.04	0.4 +/- 0.01	0.51 +/- 0.03
`Fair MixUp`	0.62 + 0.01	0.75 + 0.07	0.45 + 0.07	0.51 +/- 0.03	0.52 +/- 0.06

(D) Results on Anxiety

TABLE C.1: Test results on (a) *CelebA*, (b) *Numeracy*, and (c) *Twitter Hate Speech* using True Positive Rate while optimizing for DF. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness $\text{IF}_{\alpha=0.5}$. For both fairness definition, lower is better, while for balanced accuracy, higher is better. The Best Off and Worst Off, in both cases higher is better, represents the min TPR and max TPR. Results have been averaged over 5 different runs. We have also highlighted methods which showcase leveling down using cyan (🟦).

By a similar argument, $\min_{\mathbf{g}\in\mathcal{P}:P(\mathbf{g}|\theta)>0} P_{h_\theta}(h_\theta(\mathbf{x}) = y|\mathcal{P} = \mathbf{g}) \geq \min_{\mathbf{g}'\in\mathcal{G}:P(\mathbf{g}'|\theta)>0} P_{h_\theta}(h_\theta(\mathbf{x}) = y|\mathbf{g}')$. Applying Corollary 1, we hence bound γ in $\mathcal{P}$ by the γ in $\mathcal{G}$

Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	IF$_{\alpha=0.5}$ ↓
`Unconstrained`	0.7 + 0.01	0.22 + 0.03	0.5 + 0.04	0.44 +/- 0.1	0.51 +/- 0.04
`Adversarial`	0.71 + 0.01	0.14 + 0.03	0.38 + 0.02	0.33 +/- 0.22	0.42 +/- 0.08
`FairGrad`	0.7 + 0.02	0.19 + 0.06	0.51 + 0.07	0.5 +/- 0.22	0.45 +/- 0.07
`INLP`	0.68 + 0.01	0.27 + 0.08	0.52 + 0.08	0.42 +/- 0.13	0.58 +/- 0.06
`Fair MixUp`	0.7 + 0.01	0.22 + 0.05	0.48 + 0.03	0.41 +/- 0.17	0.52 +/- 0.06

(A) Results on Numeracy

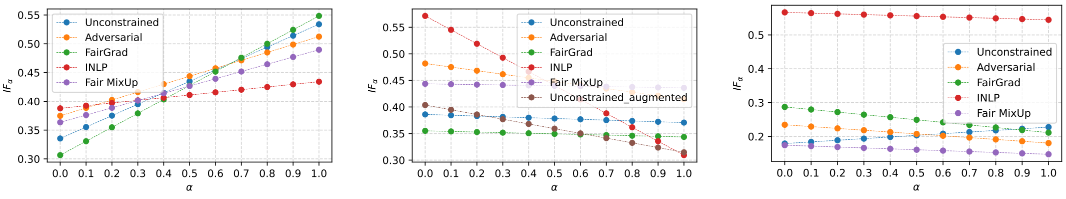
Method	BA ↑	Best Off ↓	Worst Off ↓	DF ↓	IF$_{\alpha=0.5}$ ↓
`Unconstrained`	0.81 + 0.01	0.18 + 0.02	0.47 + 0.02	0.44 +/- 0.04	0.46 +/- 0.03
`Adversarial`	0.8 + 0.01	0.18 + 0.02	0.46 + 0.02	0.42 +/- 0.03	0.47 +/- 0.04
`FairGrad`	0.79 + 0.01	0.19 + 0.03	0.51 + 0.04	0.5 +/- 0.03	0.47 +/- 0.04
`INLP`	0.67 + 0.01	0.18 + 0.1	0.38 + 0.18	0.28 +/- 0.02	0.47 +/- 0.1
`Fair MixUp`	0.81 + 0.01	0.18 + 0.02	0.49 + 0.04	0.47 +/- 0.02	0.46 +/- 0.03

(B) Results on Twitter Hate Speech

TABLE C.2: Test results on (a) *Numeracy*, and (b) *Twitter Hate Speech* using False Positive Rate while optimizing for DF. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness IF$_{\alpha=0.5}$. For both fairness definition, lower is better, while for balanced accuracy, higher is better. The Best Off and Worst Off, in both cases lower is better, represents the min FPR and max FPR. Results have been averaged over 5 different runs. We have also highlighted methods which showcase leveling down using cyan (🟦).

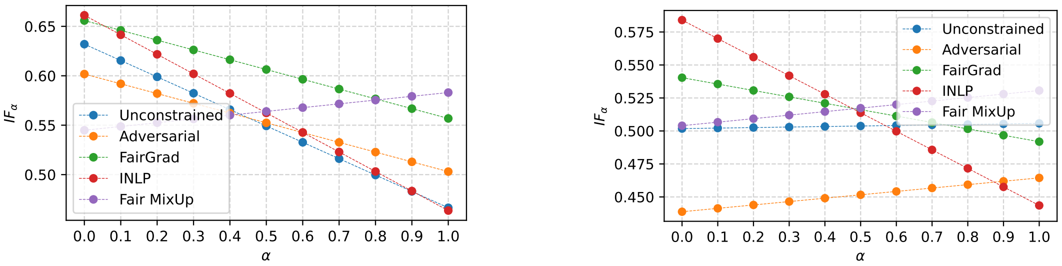
C.2 Extended Experiments

In this section, we detail the additional results. Table C.1 provides results for the True Positive Rate (TPR) fairness measure, as outlined in the Experiment Section 5.7.2. In Figure C.3, we vary the number of sensitive axes and plot the worst-case performance for Anxiety in FPR and TPR settings. Finally, Table C.2 displays results related to the FPR parity fairness measure, focusing on the Twitter Hate Speech and Numeracy datasets. Notably, for TPR, each method exhibits leveling down in at least one dataset. For example, `Adversarial` shows leveling down in the Numeracy dataset, whereas `INLP` does so in both the Twitter Hate Speech and Anxiety datasets. Similarly, as with FPR, *DF* does not consistently identify leveling down. As evidence, while both `FairGrad` and `INLP` demonstrate leveling down, they show a better fairness level than `Unconstrained`.



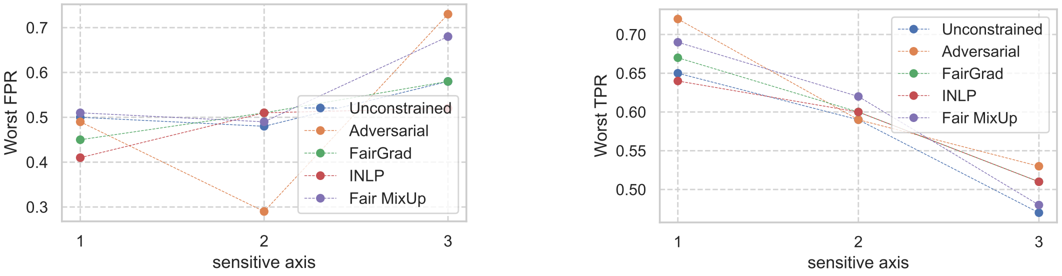
(A) CelebA TPR (B) Numeracy TPR (C) Twitter Hate Speech TPR

FIGURE C.1: Value of IF_α on the test set of CelebA, Numeracy, and Twitter Hate Speech datasets for varying $\alpha \in [0, 1]$.



(A) Anxiety FPR (B) Anxiety TPR

FIGURE C.2: Value of IF_α on the test set of Anxiety datasets for varying $\alpha \in [0, 1]$.



(A) Anxiety FPR (B) Anxiety TPR

FIGURE C.3: Test results over the worst-off group on *Anxiety* by varying the number of sensitive axes. For p binary sensitive axis in the dataset, the total number of sensitive groups are $p^3 - 1$. Note that in FPR, lower the value better it is, while for TPR opposite is true.

Appendix D

Synthetic Data Generation for Intersectional Fairness

D.1 Extended Experiments

In this section, we detail the additional experiments over the *CelebA* and *Numeracy* datasets. Table D.1 shows results for fixed value of α. While Figure D.1 plot the trade-off between relative and absolute performance of groups by varying α.

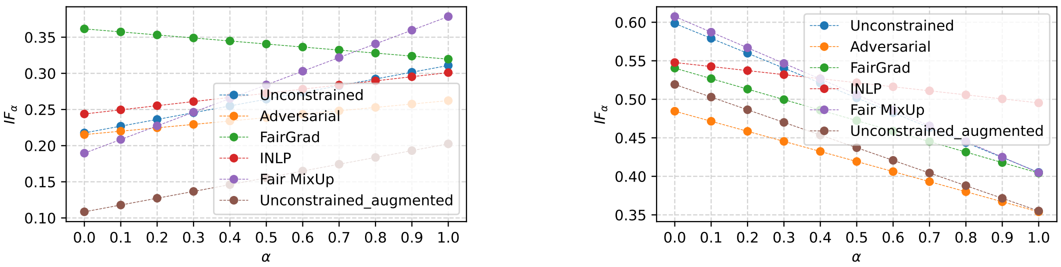
Method	BA	Best Off	Worst Off	DF	$IF\alpha = 0.5$
Unconstrained	0.81 + 0.0	0.06 + 0.02	0.34 + 0.01	0.35 +/- 0.38	0.26 +/- 0.04
Adversarial	0.81 + 0.01	0.05 + 0.01	0.3 + 0.03	0.31 +/- 0.19	0.24 +/- 0.03
FairGrad	0.76 + 0.0	0.1 + 0.01	0.35 + 0.04	0.33 +/- 0.12	0.34 +/- 0.02
INLP	0.81 + 0.01	0.07 + 0.01	0.35 + 0.03	0.36 +/- 0.16	0.27 +/- 0.01
Fair MixUp	0.81 + 0.0	0.06 + 0.0	0.4 + 0.07	0.45 +/- 0.19	0.28 +/- 0.02
Unconstrained + Augmented	0.76 + 0.01	0.02 + 0.0	0.21 + 0.03	0.22 +/- 0.21	0.16 +/- 0.01

(A) Results on CelebA

Method	BA	Best Off	Worst Off	DF	$IF\alpha = 0.5$
Unconstrained	0.7 + 0.01	0.21 + 0.05	0.46 + 0.06	0.38 +/- 0.13	0.5 +/- 0.06
Adversarial	0.69 + 0.02	0.15 + 0.03	0.39 + 0.04	0.33 +/- 0.16	0.42 +/- 0.05
FairGrad	0.7 + 0.01	0.19 + 0.05	0.45 + 0.09	0.39 +/- 0.12	0.47 +/- 0.06
INLP	0.69 + 0.0	0.23 + 0.02	0.52 + 0.02	0.47 +/- 0.05	0.52 +/- 0.02
Fair MixUp	0.69 + 0.01	0.21 + 0.04	0.45 + 0.05	0.36 +/- 0.09	0.51 +/- 0.04
Unconstrained + Augmented	0.69 + 0.02	0.14 + 0.05	0.39 + 0.11	0.34 +/- 0.24	0.44 +/- 0.07

(B) Results on Numeracy

TABLE D.1: Test results on (a) *CelebA*, (and b) *Numeracy* using False Positive Rate. We select hyper parameters based on $IF\alpha = 0.5$ value. The utility of various approaches is measured by balanced accuracy (BA), whereas fairness is measured by differential fairness DF and intersectional fairness $IF\alpha = 0.5$. For both fairness definitions, lower is better, while for balanced accuracy, higher is better. Best Off and Worst Off represent the min FPR and max FPR across groups (in both cases, lower is better). Results have been averaged over 5 different runs.



(A) CelebA FPR

(B) Anxiety FPR

FIGURE D.1: Value of IF_α on the test set of CelebA, and Numeracy datasets for varying $\alpha \in [0, 1]$.

Bibliography

Abadi, Martín et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. URL: https://www.tensorflow.org/.

Abbasi, Ahmed et al. (2021). "Constructing a Psychometric Testbed for Fair Natural Language Processing". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by Marie-Francine Moens et al. Association for Computational Linguistics, pp. 3748–3758. DOI: 10.18653/v1/2021.emnlp-main.304. URL: https://doi.org/10.18653/v1/2021.emnlp-main.304.

Abowd, John M (2018). "The US Census Bureau adopts differential privacy". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2867–2867.

Adi, Yossi et al. (2019). "To Reverse the Gradient or Not: an Empirical Comparison of Adversarial and Multi-task Learning in Speech Recognition". In: *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*. IEEE, pp. 3742–3746. DOI: 10.1109/ICASSP.2019.8682468. URL: https://doi.org/10.1109/ICASSP.2019.8682468.

Adler, Philip et al. (2018). "Auditing black-box models for indirect influence". In: *Knowl. Inf. Syst.* 54.1, pp. 95–122. DOI: 10.1007/s10115-017-1116-3. URL: https://doi.org/10.1007/s10115-017-1116-3.

Agarap, Abien Fred (2018). "Deep Learning using Rectified Linear Units (ReLU)". In: *CoRR* abs/1803.08375. arXiv: 1803.08375. URL: http://arxiv.org/abs/1803.08375.

Agarwal, Alekh et al. (2018). "A reductions approach to fair classification". In: *International Conference on Machine Learning*. PMLR, pp. 60–69.

Ahmed, Ali, Mark Granberg, and Shantanu Khanna (2021). "Gender discrimination in hiring: an experimental reexamination of the Swedish case". In: *PloS one* 16.1, e0245513.

Aïvodji, Ulrich et al. (2021). "Local Data Debiasing for Fairness Based on Generative Adversarial Training". In: *Algorithms* 14.3, p. 87. DOI: 10.3390/A14030087. URL: https://doi.org/10.3390/a14030087.

Albanese, Jay S (1984). "Concern about variation in criminal sentences: A cyclical history of reform". In: *J. Crim. L. & Criminology* 75, p. 260.

Alyoubi, Wejdan L, Wafaa M Shalash, and Maysoon F Abulkhair (2020). "Diabetic retinopathy detection through deep learning techniques: A review". In: *Informatics in Medicine Unlocked* 20, p. 100377.

Baeza-Yates, Ricardo (2018). “Bias on the web”. In: *Commun. ACM* 61.6, pp. 54–61. DOI: 10.1145/3209581. URL: https://doi.org/10.1145/3209581.

Bagdasaryan, Eugene, Omid Poursaeed, and Vitaly Shmatikov (2019). “Differential Privacy Has Disparate Impact on Model Accuracy”. In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach et al., pp. 15453–15462. URL: https://proceedings.neurips.cc/paper/2019/hash/fc0de4e0396fff257ea362983c2dda5a-Abstract.html.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2015). “Neural Machine Translation by Jointly Learning to Align and Translate”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: http://arxiv.org/abs/1409.0473.

Barocas, Solon, Moritz Hardt, and Arvind Narayanan (2019). *Fairness and Machine Learning: Limitations and Opportunities*. http://www.fairmlbook.org. fairmlbook.org.

Barocas, Solon et al. (2017). “Big data, data science, and civil rights”. In: *arXiv preprint arXiv:1706.03102*.

Bau, David et al. (2019). “Seeing what a gan cannot generate”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4502–4511.

Ben-Tal, Aharon et al. (2012). “Efficient methods for robust classification under uncertainty in kernel matrices”. In: *J. Mach. Learn. Res.* 13, pp. 2923–2954. DOI: 10.5555/2503308.2503335. URL: https://dl.acm.org/doi/10.5555/2503308.2503335.

Bender, Emily M. et al. (2021). “On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?” In: *FAccT ’21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*. Ed. by Madeleine Clare Elish, William Isaac, and Richard S. Zemel. ACM, pp. 610–623. DOI: 10.1145/3442188.3445922. URL: https://doi.org/10.1145/3442188.3445922.

Berk, Richard et al. (2021). “Fairness in criminal justice risk assessments: The state of the art”. In: *Sociological Methods & Research* 50.1, pp. 3–44.

Beutel, Alex et al. (2017). “Data Decisions and Theoretical Implications when Adversarially Learning Fair Representations”. In: *CoRR* abs/1707.00075. arXiv: 1707.00075. URL: http://arxiv.org/abs/1707.00075.

Biddle, Dan (2006). *Adverse impact and test validation: A practitioner’s guide to valid and defensible employment testing*. Gower Publishing, Ltd.

Blanzeisky, William and Pádraig Cunningham (2022). “Using Pareto simulated annealing to address algorithmic bias in machine learning”. In: *Knowl. Eng. Rev.* 37, e5. DOI: 10.1017/S0269888922000029. URL: https://doi.org/10.1017/S0269888922000029.

Blodgett, Su Lin, Lisa Green, and Brendan O’Connor (2016). “Demographic Dialectal Variation in Social Media: A Case Study of African-American English”. In.

Bolukbasi, Tolga et al. (2016). “Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. Ed. by Daniel D. Lee et al., pp. 4349–4357.

Boser, Bernhard E., Isabelle Guyon, and Vladimir Vapnik (1992). “A Training Algorithm for Optimal Margin Classifiers”. In: *Proceedings of the Fifth Annual ACM Conference on Computational Learning Theory, COLT 1992, Pittsburgh, PA, USA, July 27-29, 1992*. Ed. by David Haussler. ACM, pp. 144–152. DOI: 10.1145/130385.130401. URL: https://doi.org/10.1145/130385.130401.

Boulitsakis-Logothetis, Stelios (2022). “Fairness-Aware Naive Bayes Classifier for Data with Multiple Sensitive Features”. In: *CoRR* abs/2202.11499. arXiv: 2202.11499. URL: https://arxiv.org/abs/2202.11499.

Bradbury, James et al. (2018). *JAX: composable transformations of Python+NumPy programs*. Version 0.3.13. URL: http://github.com/google/jax.

Brown, Tom B. et al. (2020). “Language Models are Few-Shot Learners”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Buchanan, Elizabeth (2012). “Ethical decision-making and internet research”. In: *Association of Internet Researchers*.

Buolamwini, Joy and Timnit Gebru (2018). “Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification”. In: *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*. Ed. by Sorelle A. Friedler and Christo Wilson. Vol. 81. Proceedings of Machine Learning Research. PMLR, pp. 77–91. URL: http://proceedings.mlr.press/v81/buolamwini18a.html.

Burrell, Jenna (2016). “How the machine ‘thinks’: Understanding opacity in machine learning algorithms”. In: *Big data & society* 3.1, p. 2053951715622512.

Calders, Toon, Faisal Kamiran, and Mykola Pechenizkiy (2009). “Building classifiers with independency constraints”. In: *2009 IEEE International Conference on Data Mining Workshops*. IEEE, pp. 13–18.

Calders, Toon and Sicco Verwer (2010). “Three naive Bayes approaches for discrimination-free classification”. In: *Data Min. Knowl. Discov.* 21.2, pp. 277–292. DOI: 10.1007/s10618-010-0190-x. URL: https://doi.org/10.1007/s10618-010-0190-x.

Caliskan, Aylin, Joanna J Bryson, and Arvind Narayanan (2017). “Semantics derived automatically from language corpora contain human-like biases”. In: *Science* 356.6334, pp. 183–186.

Caton, Simon and Christian Haas (2020). “Fairness in Machine Learning: A Survey”. In: *CoRR* abs/2010.04053. arXiv: 2010.04053. URL: https://arxiv.org/abs/2010.04053.

Chakraborty, Joymallya et al. (2020). “Fairway: a way to build fair ML software”. In: *ESEC/FSE ’20: 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, Virtual Event, USA, November 8-13, 2020*. Ed. by Prem Devanbu, Myra B. Cohen, and Thomas Zimmermann. ACM, pp. 654–665. DOI: 10.1145/3368089.3409697. URL: https://doi.org/10.1145/3368089.3409697.

Chan, David (2011). “Perceptions of fairness”. In.

Chang, Hongyan and Reza Shokri (2020). “On the Privacy Risks of Algorithmic Fairness”. In: *CoRR* abs/2011.03731. arXiv: 2011.03731. URL: https://arxiv.org/abs/2011.03731.

Chang, Kai-Wei, Vinodkumar Prabhakaran, and Vicente Ordonez (2019). “Bias and fairness in natural language processing”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): Tutorial Abstracts*.

Chapman, Elizabeth N, Anna Kaatz, and Molly Carnes (2013). “Physicians and implicit bias: how doctors may unwittingly perpetuate health care disparities”. In: *Journal of general internal medicine* 28, pp. 1504–1510.

Chawla, Nitesh V et al. (2002). “SMOTE: synthetic minority over-sampling technique”. In: *Journal of artificial intelligence research* 16, pp. 321–357.

Chawla, Nitesh V. et al. (2003). “SMOTEBoost: Improving Prediction of the Minority Class in Boosting”. In: *Knowledge Discovery in Databases: PKDD 2003, 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, Cavtat-Dubrovnik, Croatia, September 22-26, 2003, Proceedings*. Ed. by Nada Lavrac et al. Vol. 2838. Lecture Notes in Computer Science. Springer, pp. 107–119. DOI: 10.1007/978-3-540-39804-2_12. URL: https://doi.org/10.1007/978-3-540-39804-2_12.

Chen, Irene Y., Fredrik D. Johansson, and David A. Sontag (2018). “Why Is My Classifier Discriminatory?” In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al., pp. 3543–3554.

Chen, Jarvis T and Nancy Krieger (2021). “Revealing the unequal burden of COVID-19 by income, race/ethnicity, and household crowding: US county versus zip code analyses”. In: *Journal of Public Health Management and Practice* 27.1, S43–S56.

Chen, Zhenpeng et al. (2022). “MAAT: a novel ensemble approach to addressing fairness and performance bugs for machine learning software”. In: *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ESEC/FSE 2022, Singapore, Singapore, November 14-18, 2022*. Ed. by Abhik Roychoudhury, Cristian Cadar, and Miryung Kim. ACM, pp. 1122–1134. DOI: 10.1145/3540250.3549093. URL: https://doi.org/10.1145/3540250.3549093.

Chiappa, Silvia (2019). “Path-specific counterfactual fairness”. In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 33. 01, pp. 7801–7808.

Chiappa, Silvia and William S. Isaac (2018). “A Causal Bayesian Networks Viewpoint on Fairness”. In: *Privacy and Identity Management. Fairness, Accountability, and Transparency in the Age of Big Data - 13th IFIP WG 9.2, 9.6/11.7, 11.6/SIG 9.2.2 International Summer School, Vienna, Austria, August 20-24, 2018, Revised Selected Papers*. Ed. by Eleni Kosta et al. Vol. 547. IFIP Advances in Information and Communication Technology. Springer, pp. 3–20. DOI: 10.1007/978-3-030-16744-8_1. URL: https://doi.org/10.1007/978-3-030-16744-8_1.

Choi, Keunwoo et al. (2017). “Convolutional recurrent neural networks for music classification”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. IEEE, pp. 2392–2396. DOI: 10.1109/ICASSP.2017.7952585. URL: https://doi.org/10.1109/ICASSP.2017.7952585.

Chouldechova, Alexandra (2017). “Fair Prediction with Disparate Impact: A Study of Bias in Recidivism Prediction Instruments”. In: *Big Data* 5.2, pp. 153–163. DOI: 10.1089/big.2016.0047. URL: https://doi.org/10.1089/big.2016.0047.

Chowdhury, Somnath Basu Roy et al. (2021). “Adversarial Scrubbing of Demographic Information for Text Classification”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by Marie-Francine Moens et al. Association for Computational Linguistics, pp. 550–562. DOI: 10.18653/v1/2021.emnlp-main.43. URL: https://doi.org/10.18653/v1/2021.emnlp-main.43.

Chuang, Ching-Yao and Youssef Mroueh (2021). “Fair Mixup: Fairness via Interpolation”. In: *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. URL: https://openreview.net/forum?id=DNl5s5BXeBn.

Ciampiconi, Lorenzo et al. (2023). “A survey and taxonomy of loss functions in machine learning”. In: *CoRR* abs/2301.05579. DOI: 10.48550/arXiv.2301.05579. arXiv: 2301.05579. URL: https://doi.org/10.48550/arXiv.2301.05579.

Cleary, T Anne (1966). “Test bias: Validity of the Scholastic Aptitude Test for Negro and White students in integrated colleges”. In: *ETS Research Bulletin Series* 1966.2, pp. i–23.

Coavoux, Maximin, Shashi Narayan, and Shay B. Cohen (2018). “Privacy-preserving Neural Representations of Text”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Ed. by Ellen Riloff et al. Association for Computational Linguistics, pp. 1–10. DOI: 10.18653/v1/d18-1001. URL: https://doi.org/10.18653/v1/d18-1001.

Commission, Equal Employment Opportunity et al. (1990). “Uniform guidelines on employee selection procedures”. In: *Fed Register* 1, pp. 216–243.

Commission, European (2018). *Communication Artificial Intelligence for Europe*.

Cortes, Corinna and Vladimir Vapnik (1995). “Support-Vector Networks”. In: *Mach. Learn.* 20.3, pp. 273–297. DOI: 10.1007/BF00994018. URL: https://doi.org/10.1007/BF00994018.

Cotter, Andrew, Heinrich Jiang, and Karthik Sridharan (2019). “Two-player games for efficient non-convex constrained optimization”. In: *Algorithmic Learning Theory*. PMLR, pp. 300–332.

Cotter, Andrew et al. (2019). “Optimization with Non-Differentiable Constraints with Applications to Fairness, Recall, Churn, and Other Goals”. In: *J. Mach. Learn. Res.* 20, 172:1–172:59. URL: http://jmlr.org/papers/v20/18-616.html.

Crenshaw, Kimberle (1989). “Demarginalizing the Intersection of Race and Sex: A Black Feminist Critique of Antidiscrimination Doctrine, Feminist Theory and Antiracist Politics”. In: *The University of Chicago Legal Forum* 140, pp. 139–167.

Cummings, Rachel et al. (2019). “On the Compatibility of Privacy and Fairness”. In: *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization, UMAP 2019, Larnaca, Cyprus, June 09-12, 2019*. Ed. by George Angelos Papadopoulos et al. ACM, pp. 309–315. DOI: 10.1145/3314183.3323847. URL: https://doi.org/10.1145/3314183.3323847.

Dablain, Damien, Bartosz Krawczyk, and Nitesh V. Chawla (2022). “Towards A Holistic View of Bias in Machine Learning: Bridging Algorithmic Fairness and Imbalanced Learning”. In: *CoRR* abs/2207.06084. DOI: 10.48550/arXiv.2207.06084. arXiv: 2207.06084. URL: https://doi.org/10.48550/arXiv.2207.06084.

d’Alessandro, Brian, Cathy O’Neil, and Tom LaGatta (2019). “Conscientious Classification: A Data Scientist’s Guide to Discrimination-Aware Classification”. In: *CoRR* abs/1907.09013. arXiv: 1907.09013. URL: http://arxiv.org/abs/1907.09013.

Dalvi, Nilesh N. et al. (2004). “Adversarial classification”. In: *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*. Ed. by Won Kim et al. ACM, pp. 99–108. DOI: 10.1145/1014052.1014066. URL: https://doi.org/10.1145/1014052.1014066.

Danks, David and Alex John London (2017). “Algorithmic Bias in Autonomous Systems”. In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. Ed. by Carles Sierra. ijcai.org, pp. 4691–4697. DOI: 10.24963/ijcai.2017/654. URL: https://doi.org/10.24963/ijcai.2017/654.

Darlington, Richard B (1971). “Another look at “cultural fairness” 1”. In: *Journal of educational measurement* 8.2, pp. 71–82.

Datta, Anupam et al. (2017). “Proxy non-discrimination in data-driven systems”. In: *arXiv preprint arXiv:1707.08120*.

De-Arteaga, Maria et al. (2019). “Bias in Bios: A Case Study of Semantic Representation Bias in a High-Stakes Setting”. In: *Proceedings of the Conference on Fairness,* *Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*. Ed. by danah boyd and Jamie H. Morgenstern. ACM, pp. 120–128. DOI: 10.1145/3287560.3287572. URL: https://doi.org/10.1145/3287560.3287572.

Devine, Patricia and Ashby Plant (2012). *Advances in experimental social psychology*. Academic Press.

Devlin, Jacob et al. (2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/n19-1423. URL: https://doi.org/10.18653/v1/n19-1423.

DeVos, Alicia et al. (2022). "Toward User-Driven Algorithm Auditing: Investigating users' strategies for uncovering harmful algorithmic behavior". In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–19.

Ding, Bolin, Janardhan Kulkarni, and Sergey Yekhanin (2017). "Collecting Telemetry Data Privately". In: *NIPS*.

Ding, Frances et al. (2021). "Retiring Adult: New Datasets for Fair Machine Learning". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc'Aurelio Ranzato et al., pp. 6478–6490. URL: https://proceedings.neurips.cc/paper/2021/hash/32e54441e6382a7fbacbbbaf3c450059-Abstract.html.

Donini, Michele et al. (2018). "Empirical risk minimization under fairness constraints". In: *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 2796–2806.

Dressel, Julia and Hany Farid (2018). "The accuracy, fairness, and limits of predicting recidivism". In: *Science advances* 4.1, eaao5580.

Dua, Dheeru, Casey Graff, et al. (2017). "UCI machine learning repository". In.

Duchi, John C., Elad Hazan, and Yoram Singer (2011). "Adaptive Subgradient Methods for Online Learning and Stochastic Optimization". In: *J. Mach. Learn. Res.* 12, pp. 2121–2159. DOI: 10.5555/1953048.2021068. URL: https://dl.acm.org/doi/10.5555/1953048.2021068.

Dutta, Sanghamitra et al. (2020). "Is There a Trade-Off Between Fairness and Accuracy? A Perspective Using Mismatched Hypothesis Testing". In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 2803–2813. URL: http://proceedings.mlr.press/v119/dutta20a.html.

Dwork, Cynthia and Aaron Roth (2014). "The Algorithmic Foundations of Differential Privacy". In: *Found. Trends Theor. Comput. Sci.* 9.3-4, pp. 211–407. DOI: 10.1561/0400000042. URL: https://doi.org/10.1561/0400000042.

Dwork, Cynthia et al. (2006). "Calibrating Noise to Sensitivity in Private Data Analysis". In: *Theory of Cryptography, Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006, Proceedings*. Ed. by Shai Halevi and Tal Rabin. Vol. 3876. Lecture Notes in Computer Science. Springer, pp. 265–284. DOI: 10.1007/11681878_14. URL: https://doi.org/10.1007/11681878_14.

Dwork, Cynthia et al. (2012). "Fairness through awareness". In: *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*. Ed. by Shafi Goldwasser. ACM, pp. 214–226. DOI: 10.1145/2090236.2090255. URL: https://doi.org/10.1145/2090236.2090255.

Dwork, Cynthia et al. (2018). “Decoupled Classifiers for Group-Fair and Efficient Machine Learning”. In: *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*. Ed. by Sorelle A. Friedler and Christo Wilson. Vol. 81. Proceedings of Machine Learning Research. PMLR, pp. 119–133. URL: http://proceedings.mlr.press/v81/dwork18a.html.

Elazar, Yanai and Yoav Goldberg (2018). “Adversarial Removal of Demographic Attributes from Text Data”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Ed. by Ellen Riloff et al. Association for Computational Linguistics, pp. 11–21.

Ensign, Danielle et al. (2018). “Runaway Feedback Loops in Predictive Policing”. In: *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*. Ed. by Sorelle A. Friedler and Christo Wilson. Vol. 81. Proceedings of Machine Learning Research. PMLR, pp. 160–171. URL: http://proceedings.mlr.press/v81/ensign18a.html.

Erlingsson, Úlfar, Vasyl Pihur, and Aleksandra Korolova (2014). “RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response”. In: *CCS*.

Ethayarajh, Kawin, David Duvenaud, and Graeme Hirst (2019). “Understanding Undesirable Word Embedding Associations”. In: *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28-August 2, 2019, Volume 1: Long Papers*. Ed. by Anna Korhonen, David R. Traum, and Lluís Màrquez. Association for Computational Linguistics, pp. 1696–1705. DOI: 10.18653/v1/p19-1166. URL: https://doi.org/10.18653/v1/p19-1166.

Evans, Richard and Jim Gao (2016). “Deepmind ai reduces google data centre cooling bill by 40%”. In: *DeepMind blog* 20, p. 158.

Everson, George (1919). “Human element in justice”. In: *J. Am. Inst. Crim. L. & Criminology* 10, p. 90.

Fanti, Giulia, Vasyl Pihur, and Úlfar Erlingsson (2016). “Building a RAPPOR with the unknown: Privacy-preserving learning of associations and data dictionaries”. In: *PoPETs*.

Felbo, Bjarke et al. (2017). “Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm”. In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. Ed. by Martha Palmer, Rebecca Hwa, and Sebastian Riedel. Association for Computational Linguistics, pp. 1615–1625.

Feldman, Michael et al. (2015). “Certifying and Removing Disparate Impact”. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*. Ed. by Longbing Cao et al. ACM, pp. 259–268. DOI: 10.1145/2783258.2783311. URL: https://doi.org/10.1145/2783258.2783311.

Fish, Benjamin, Jeremy Kun, and Ádám Dániel Lelkes (2016). “A Confidence-Based Approach for Balancing Fairness and Accuracy”. In: *Proceedings of the 2016 SIAM International Conference on Data Mining, Miami, Florida, USA, May 5-7, 2016*. Ed. by Sanjay Chawla Venkatasubramanian and Wagner Meira Jr. SIAM, pp. 144–152. DOI: 10.1137/1.9781611974348.17. URL: https://doi.org/10.1137/1.9781611974348.17.

Floridi, Luciano and Massimo Chiriatti (2020). “GPT-3: Its Nature, Scope, Limits, and Consequences”. In: *Minds Mach.* 30.4, pp. 681–694. DOI: 10.1007/S11023-020-09548-1. URL: https://doi.org/10.1007/s11023-020-09548-1.

Foulds, James R. et al. (2020). “An Intersectional Definition of Fairness”. In: *36th IEEE International Conference on Data Engineering, ICDE 2020, Dallas, TX, USA, April 20-24, 2020*. IEEE, pp. 1918–1921. DOI: 10.1109/ICDE48307.2020.00203. URL: https://doi.org/10.1109/ICDE48307.2020.00203.

Friedler, Sorelle A. et al. (2019). “A comparative study of fairness-enhancing interventions in machine learning”. In: *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*. Ed. by danah boyd and Jamie H. Morgenstern. ACM, pp. 329–338. DOI: 10.1145/3287560.3287589. URL: https://doi.org/10.1145/3287560.3287589.

Galhotra, Sainyam, Yuriy Brun, and Alexandra Meliou (2017). “Fairness testing: testing software for discrimination”. In: *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering, ESEC/FSE 2017, Paderborn, Germany, September 4-8, 2017*. Ed. by Eric Bodden et al. ACM, pp. 498–510. DOI: 10.1145/3106237.3106277. URL: https://doi.org/10.1145/3106237.3106277.

Ganin, Yaroslav and Victor Lempitsky (2015). “Unsupervised Domain Adaptation by Backpropagation”. In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Francis Bach and David Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, pp. 1180–1189.

Ganin, Yaroslav et al. (2016). “Domain-Adversarial Training of Neural Networks”. In: *J. Mach. Learn. Res.* 17, 59:1–59:35. URL: http://jmlr.org/papers/v17/15-239.html.

Gentile, Claudio and Manfred K. Warmuth (1998). “Linear Hinge Loss and Average Margin”. In: *Advances in Neural Information Processing Systems 11, [NIPS Conference, Denver, Colorado, USA, November 30 - December 5, 1998]*. Ed. by Michael J. Kearns, Sara A. Solla, and David A. Cohn. The MIT Press, pp. 225–231. URL: http://papers.nips.cc/paper/1610-linear-hinge-loss-and-average-margin.

Gillen, Stephen et al. (2018). “Online Learning with an Unknown Fairness Metric”. In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al., pp. 2605–2614. URL: https://proceedings.neurips.cc/paper/2018/hash/50905d7b2216bfeccb5b41016357176b-Abstract.html.

Gohar, Usman and Lu Cheng (2023). “A Survey on Intersectional Fairness in Machine Learning: Notions, Mitigation, and Challenges”. In: *CoRR* abs/2305.06969. DOI: 10.48550/arXiv.2305.06969. arXiv: 2305.06969. URL: https://doi.org/10.48550/arXiv.2305.06969.

Gonen, Hila and Yoav Goldberg (Aug. 2019). “Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings But do not Remove Them”. In: *Proceedings of the 2019 Workshop on Widening NLP*. Florence, Italy: Association for Computational Linguistics, pp. 60–63. URL: https://aclanthology.org/W19-3621.

González-Zelaya, Vladimiro et al. (2021). “Optimising Fairness Through Parametrised Data Sampling.” In: *EDBT*, pp. 445–450.

Goodfellow, Ian et al. (2014a). “Generative adversarial nets”. In: *Advances in neural information processing systems* 27.

Goodfellow, Ian J. et al. (2014b). “Generative Adversarial Networks”. In: *CoRR* abs/1406.2661. arXiv: 1406.2661. URL: http://arxiv.org/abs/1406.2661.

Gopalan, Parikshit et al. (2022). “Low-Degree Multicalibration”. In: *Conference on Learning Theory, 2-5 July 2022, London, UK*. Ed. by Po-Ling Loh and Maxim Raginsky. Vol. 178. Proceedings of Machine Learning Research. PMLR, pp. 3193–3234. URL: https://proceedings.mlr.press/v178/gopalan22a.html.

Gou, Jianping et al. (2021). "Knowledge distillation: A survey". In: *International Journal of Computer Vision* 129.6, pp. 1789–1819.

Gretton, Arthur et al. (2012). "A Kernel Two-Sample Test". In: *J. Mach. Learn. Res.* 13, pp. 723–773. DOI: 10.5555/2503308.2188410. URL: https://dl.acm.org/doi/10.5555/2503308.2188410.

Guion, Robert M (1966). "Employment tests and discriminatory hiring". In: *Industrial Relations: A Journal of Economy and Society* 5.2, pp. 20–37.

Guo, Tianmei et al. (2017). "Simple convolutional neural network on image classification". In: *2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA)*. IEEE, pp. 721–724.

Habernal, Ivan (2021). "When differential privacy meets NLP: The devil is in the detail". In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by Marie-Francine Moens et al. Association for Computational Linguistics, pp. 1522–1528. DOI: 10.18653/v1/2021.emnlp-main.114. URL: https://doi.org/10.18653/v1/2021.emnlp-main.114.

Hall, Melissa et al. (2022). "A Systematic Study of Bias Amplification". In: *CoRR* abs/2201.11706. arXiv: 2201.11706. URL: https://arxiv.org/abs/2201.11706.

Han, Xudong, Timothy Baldwin, and Trevor Cohn (2021). "Diverse Adversaries for Mitigating Bias in Training". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*. Ed. by Paola Merlo, Jörg Tiedemann, and Reut Tsarfaty. Association for Computational Linguistics, pp. 2760–2765.

Harber, Kent D et al. (2012). "Students' race and teachers' social support affect the positive feedback bias in public schools." In: *Journal of Educational Psychology* 104.4, p. 1149.

Hardt, Moritz, Eric Price, and Nati Srebro (2016). "Equality of opportunity in supervised learning". In: *Advances in neural information processing systems* 29, pp. 3315–3323.

Hashimoto, Tatsunori B. et al. (2018). "Fairness Without Demographics in Repeated Loss Minimization". In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 1934–1943. URL: http://proceedings.mlr.press/v80/hashimoto18a.html.

He, Kaiming et al. (2016a). "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society, pp. 770–778. DOI: 10.1109/CVPR.2016.90. URL: https://doi.org/10.1109/CVPR.2016.90.

— (2016b). "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778.

Hébert-Johnson, Úrsula et al. (2018). "Multicalibration: Calibration for the (Computationally-Identifiable) Masses". In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 1944–1953. URL: http://proceedings.mlr.press/v80/hebert-johnson18a.html.

Hinojosa, Elisa Reyes (2023). "Unequal Access to Higher Education: Student Loan Debt Disproportionately Impacts Minority Students". In: *Scholar* 25, p. 63.

Hoffmann, Diane E and Anita J Tarzian (2001). “The girl who cried pain: a bias against women in the treatment of pain”. In: *Journal of Law, Medicine & Ethics* 29.1, pp. 13–27.

Hood, Rodney G (2001). “Confronting racial and ethnic disparities in health care”. In: *Academic Medicine* 76.6, pp. 584–585.

Houlsby, Neil et al. (2019). “Parameter-Efficient Transfer Learning for NLP”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 2790–2799. URL: http://proceedings.mlr.press/v97/houlsby19a.html.

Huang, Xiaolei et al. (May 2020). “Multilingual Twitter Corpus and Baselines for Evaluating Demographic Bias in Hate Speech Recognition”. English. In: *Proceedings of the Twelfth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 1440–1448. ISBN: 979-10-95546-34-4. URL: https://aclanthology.org/2020.lrec-1.180.

Huber, Peter J (1965). “A robust version of the probability ratio test”. In: *The Annals of Mathematical Statistics*, pp. 1753–1758.

Hutchinson, Ben and Margaret Mitchell (2019). “50 Years of Test (Un)fairness: Lessons for Machine Learning”. In: *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*. Ed. by danah boyd and Jamie H. Morgenstern. ACM, pp. 49–58. DOI: 10.1145/3287560.3287600. URL: https://doi.org/10.1145/3287560.3287600.

Iofinova, Eugenia, Nikola Konstantinov, and Christoph H. Lampert (2022). “FLEA: Provably Robust Fair Multisource Learning from Unreliable Training Data”. In: *Trans. Mach. Learn. Res.* 2022. URL: https://openreview.net/forum?id=XsPopigZXV.

Iosifidis, Vasileios, Besnik Fetahu, and Eirini Ntoutsi (2019). “FAE: A Fairness-Aware Ensemble Framework”. In: *2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019*. Ed. by Chaitanya K. Baru et al. IEEE, pp. 1375–1380. DOI: 10.1109/BigData47090.2019.9006487. URL: https://doi.org/10.1109/BigData47090.2019.9006487.

Iosifidis, Vasileios and Eirini Ntoutsi (2018). “Dealing with bias via data augmentation in supervised learning scenarios”. In: *Jo Bates Paul D. Clough Robert Jäschke* 24.11.

— (2019). “Adafair: Cumulative fairness adaptive boosting”. In: *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 781–790.

Jagielski, Matthew et al. (2019). “Differentially Private Fair Learning”. In: *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, pp. 3000–3008. URL: http://proceedings.mlr.press/v97/jagielski19a.html.

Jentzsch, Sophie F. et al. (2019). “Semantics Derived Automatically from Language Corpora Contain Human-like Moral Choices”. In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*. Ed. by Vincent Conitzer, Gillian K. Hadfield, and Shannon Vallor. ACM, pp. 37–44. DOI: 10.1145/3306618.3314267. URL: https://doi.org/10.1145/3306618.3314267.

Jiang, Heinrich and Ofir Nachum (2020). “Identifying and correcting label bias in machine learning”. In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 702–712.

Jones, Gareth P. et al. (2020). "Metrics and methods for a systematic comparison of fairness-aware machine learning algorithms". In: *CoRR* abs/2010.03986. arXiv: 2010.03986. URL: https://arxiv.org/abs/2010.03986.

Jung, Christopher et al. (2020). "Fair prediction with endogenous behavior". In: *Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 677–678.

Kamiran, Faisal and Toon Calders (2009). "Classifying without discriminating". In: *2009 2nd international conference on computer, control and communication*. IEEE, pp. 1–6.

— (2010). "Classification with no discrimination by preferential sampling". In: *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*. Citeseer.

— (2012). "Data preprocessing techniques for classification without discrimination". In: *Knowledge and Information Systems* 33.1, pp. 1–33.

Kamiran, Faisal, Toon Calders, and Mykola Pechenizkiy (2010). "Discrimination Aware Decision Tree Learning". In: *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*. Ed. by Geoffrey I. Webb et al. IEEE Computer Society, pp. 869–874. DOI: 10.1109/ICDM.2010.50. URL: https://doi.org/10.1109/ICDM.2010.50.

Kamishima, Toshihiro et al. (2012). "Fairness-Aware Classifier with Prejudice Remover Regularizer". In: *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II*. Ed. by Peter A. Flach, Tijl De Bie, and Nello Cristianini. Vol. 7524. Lecture Notes in Computer Science. Springer, pp. 35–50. DOI: 10.1007/978-3-642-33486-3_3. URL: https://doi.org/10.1007/978-3-642-33486-3_3.

Kanamori, Kentaro and Hiroki Arimura (2021). "Fairness-aware decision tree editing based on mixed-integer linear optimization". In: *Transactions of the Japanese Society for Artificial Intelligence* 36.4, B–L13_1.

Kappeler, Armin et al. (2016). "Video Super-Resolution With Convolutional Neural Networks". In: *IEEE Trans. Computational Imaging* 2.2, pp. 109–122. DOI: 10.1109/TCI.2016.2532323. URL: https://doi.org/10.1109/TCI.2016.2532323.

Karpathy, Andrej et al. (2014). "Large-Scale Video Classification with Convolutional Neural Networks". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*. IEEE Computer Society, pp. 1725–1732. DOI: 10.1109/CVPR.2014.223. URL: https://doi.org/10.1109/CVPR.2014.223.

Karve, Saket, Lyle Ungar, and João Sedoc (Aug. 2019). "Conceptor Debiasing of Word Representations Evaluated on WEAT". In: *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*. Florence, Italy: Association for Computational Linguistics, pp. 40–48. DOI: 10.18653/v1/W19-3806. URL: https://aclanthology.org/W19-3806.

Kasiviswanathan, Shiva Prasad et al. (2011). "What Can We Learn Privately?" In: *SIAM J. Comput.* 40.3, pp. 793–826. DOI: 10.1137/090756090. URL: https://doi.org/10.1137/090756090.

Katzman, Jared et al. (2023). "Representational Harms in Image Tagging". In: *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence (2023)*. Vol. 5.

Kay, Matthew, Cynthia Matuszek, and Sean A. Munson (2015). "Unequal Representation and Gender Stereotypes in Image Search Results for Occupations". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, CHI 2015, Seoul, Republic of Korea, April 18-23, 2015*. Ed. by Bo Begole et al. ACM, pp. 3819–3828. DOI: 10.1145/2702123.2702520. URL: https://doi.org/10.1145/2702123.2702520.

Kearns, Michael J. et al. (2018). “Preventing Fairness Gerrymandering: Auditing and Learning for Subgroup Fairness”. In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 2569–2577. URL: http://proceedings.mlr.press/v80/kearns18a.html.

Kiefer, Jack and Jacob Wolfowitz (1952). “Stochastic estimation of the maximum of a regression function”. In: *The Annals of Mathematical Statistics*, pp. 462–466.

Kilbertus, Niki et al. (2017). “Avoiding discrimination through causal reasoning”. In: *Advances in neural information processing systems* 30.

Kim, Michael P., Omer Reingold, and Guy N. Rothblum (2018). “Fairness Through Computationally-Bounded Awareness”. In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al., pp. 4847–4857. URL: https://proceedings.neurips.cc/paper/2018/hash/c8dfece5cc68249206e4690fc4737a8d-Abstract.html.

Kingma, Diederik P. and Jimmy Ba (2015). “Adam: A Method for Stochastic Optimization”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. URL: http://arxiv.org/abs/1412.6980.

Kiritchenko, Svetlana and Saif M Mohammad (2018). “Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems”. In: *NAACL HLT 2018*, p. 43.

Kirk, Hannah Rose et al. (2021). “Bias Out-of-the-Box: An Empirical Analysis of Intersectional Occupational Biases in Popular Generative Language Models”. In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc’Aurelio Ranzato et al., pp. 2611–2624. URL: https://proceedings.neurips.cc/paper/2021/hash/1531beb762df4029513ebf9295e0d34f-Abstract.html.

Kleinberg, Jon et al. (2018). “Algorithmic fairness”. In: *Aea papers and proceedings*. Vol. 108, pp. 22–27.

Kleinberg, Jon M., Sendhil Mullainathan, and Manish Raghavan (2017). “Inherent Trade-Offs in the Fair Determination of Risk Scores”. In: *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA*. Ed. by Christos H. Papadimitriou. Vol. 67. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 43:1–43:23. DOI: 10.4230/LIPIcs.ITCS.2017.43. URL: https://doi.org/10.4230/LIPIcs.ITCS.2017.43.

Kobayashi, Kenji and Yuri Nakao (2022). “One-vs.-one mitigation of intersectional bias: A general method for extending fairness-aware binary classification”. In: *New Trends in Disruptive Technologies, Tech Ethics and Artificial Intelligence: The DITTET Collection 1*. Springer, pp. 43–54.

Kohavi, Ron (1996). “Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid”. In: *Kdd*. Ed. by Evangelos Simoudis, Jiawei Han, and Usama M. Fayyad. AAAI Press, pp. 202–207.

Krasanakis, Emmanouil et al. (2018). “Adaptive sensitive reweighting to mitigate bias in fairness-aware classification”. In: *Proceedings of the 2018 World Wide Web Conference*, pp. 853–862.

Krishna, Satyapriya, Rahul Gupta, and Christophe Dupuy (Apr. 2021). “ADePT: Autoencoder based Differentially Private Text Transformation”. In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Online: Association for Computational Linguistics, pp. 2435–2439.

DOI: 10.18653/v1/2021.eacl-main.207. URL: https://aclanthology.org/2021.eacl-main.207.

Kulynych, Bogdan et al. (2022). “Disparate Vulnerability to Membership Inference Attacks”. In: *PETS*. URL: http://arxiv.org/abs/1906.00389.

Kusner, Matt J et al. (2017). “Counterfactual fairness. arXiv e-prints, Article”. In: *arXiv preprint arXiv:1703.06856*.

Kusner, Matt J. et al. (2018). “Causal Interventions for Fairness”. In: *CoRR* abs/1806.02380. arXiv: 1806.02380. URL: http://arxiv.org/abs/1806.02380.

Lahoti, Preethi, Krishna P. Gummadi, and Gerhard Weikum (2019). “iFair: Learning Individually Fair Data Representations for Algorithmic Decision Making”. In: *35th IEEE International Conference on Data Engineering, ICDE 2019, Macao, China, April 8-11, 2019*. IEEE, pp. 1334–1345. DOI: 10.1109/ICDE.2019.00121. URL: https://doi.org/10.1109/ICDE.2019.00121.

Lahoti, Preethi et al. (2020). “Fairness without Demographics through Adversarially Reweighted Learning”. In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al. URL: https://proceedings.neurips.cc/paper/2020/hash/07fc15c9d169ee48573edd749d25945d-Abstract.html.

Lalor, John et al. (2022). “Benchmarking Intersectional Biases in NLP”. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*. Ed. by Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz. Association for Computational Linguistics, pp. 3598–3609. DOI: 10.18653/v1/2022.naacl-main.263. URL: https://doi.org/10.18653/v1/2022.naacl-main.263.

Lample, Guillaume et al. (2017). “Fader Networks: Manipulating Images by Sliding Attributes”. In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon et al., pp. 5967–5976. URL: https://proceedings.neurips.cc/paper/2017/hash/3fd60983292458bf7dee75f12d5e9e05-Abstract.html.

Larson, Jeff et al. (2016). “How we analyzed the COMPAS recidivism algorithm”. In: *ProPublica (5 2016)* 9.1, pp. 3–3.

LeCun, Yann et al. (1989). “Backpropagation applied to handwritten zip code recognition”. In: *Neural computation* 1.4, pp. 541–551.

Lee, Nicol Turner (2018). “Detecting racial bias in algorithms and machine learning”. In: *Journal of Information, Communication and Ethics in Society* 16.3, pp. 252–260.

Lerman, Kristina and Tad Hogg (2014). “Leveraging position bias to improve peer recommendation”. In: *PloS one* 9.6, e98914.

Li, Peizhao and Hongfu Liu (2022). “Achieving Fairness at No Utility Cost via Data Reweighing with Influence”. In: *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, pp. 12917–12930. URL: https://proceedings.mlr.press/v162/li22p.html.

Li, Qing et al. (2014). “Medical image classification with convolutional neural network”. In: *13th International Conference on Control Automation Robotics & Vision, ICARCV 2014, Singapore, December 10-12, 2014*. IEEE, pp. 844–848. DOI: 10.1109/ICARCV.2014.7064414. URL: https://doi.org/10.1109/ICARCV.2014.7064414.

Li, Tianyi et al. (2022a). "'Propose and Review': Interactive Bias Mitigation for Machine Classifiers". In: *Available at SSRN 4139244*.

Li, Xiang Lisa and Percy Liang (2021). "Prefix-Tuning: Optimizing Continuous Prompts for Generation". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*. Ed. by Chengqing Zong et al. Association for Computational Linguistics, pp. 4582–4597. DOI: 10.18653/v1/2021.acl-long.353. URL: https://doi.org/10.18653/v1/2021.acl-long.353.

Li, Yanhui et al. (2022b). "Training Data Debugging for the Fairness of Machine Learning Software". In: *44th IEEE/ACM 44th International Conference on Software Engineering, ICSE 2022, Pittsburgh, PA, USA, May 25-27, 2022*. ACM, pp. 2215–2227. DOI: 10.1145/3510003.3510091. URL: https://doi.org/10.1145/3510003.3510091.

Li, Yitong, Timothy Baldwin, and Trevor Cohn (July 2018). "Towards Robust and Privacy-preserving Text Representations". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 25–30. DOI: 10.18653/v1/P18-2005. URL: https://aclanthology.org/P18-2005.

Li, Yuanlong et al. (2019). "Transforming cooling optimization for green data center via deep reinforcement learning". In: *IEEE transactions on cybernetics* 50.5, pp. 2002–2013.

Lialin, Vladislav, Vijeta Deshpande, and Anna Rumshisky (2023). "Scaling Down to Scale Up: A Guide to Parameter-Efficient Fine-Tuning". In: *CoRR* abs/2303.15647. DOI: 10.48550/arXiv.2303.15647. arXiv: 2303.15647. URL: https://doi.org/10.48550/arXiv.2303.15647.

Liu, Wenyan et al. (2020). "Fair Differential Privacy Can Mitigate the Disparate Impact on Model Accuracy". In: *CoRR*.

Liu, Ziwei et al. (2015). "Deep learning face attributes in the wild". In: *Proceedings of the IEEE international conference on computer vision*, pp. 3730–3738.

Lohaus, Michael, Michaël Perrot, and Ulrike Von Luxburg (2020). "Too relaxed to be fair". In: *International Conference on Machine Learning*. PMLR, pp. 6360–6369.

Lum, Kristian and William Isaac (2016). "To predict and serve?" In: *Significance* 13.5, pp. 14–19.

Lum, Kristian and James E. Johndrow (2016). "A statistical framework for fair predictive algorithms". In: *CoRR* abs/1610.08077. arXiv: 1610.08077. URL: http://arxiv.org/abs/1610.08077.

Lyu, Lingjuan, Xuanli He, and Yitong Li (2020). "Differentially Private Representation for NLP: Formal Guarantee and An Empirical Study on Privacy and Fairness". In: *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*. Ed. by Trevor Cohn, Yulan He, and Yang Liu. Vol. EMNLP 2020. Findings of ACL. Association for Computational Linguistics, pp. 2355–2365. DOI: 10.18653/v1/2020.findings-emnlp.213. URL: https://doi.org/10.18653/v1/2020.findings-emnlp.213.

Madras, David et al. (2018). "Learning Adversarially Fair and Transferable Representations". In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 3381–3390. URL: http://proceedings.mlr.press/v80/madras18a.html.

Madry, Aleksander et al. (2018). "Towards Deep Learning Models Resistant to Adversarial Attacks". In: *6th International Conference on Learning Representations, ICLR*

2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings. OpenReview.net. URL: https://openreview.net/forum?id=rJzIBfZAb.

Maheshwari, Gaurav and Michaël Perrot (2022). "FairGrad: Fairness Aware Gradient Descent". In: *CoRR* abs/2206.10923. DOI: 10.48550/arXiv.2206.10923. arXiv: 2206.10923. URL: https://doi.org/10.48550/arXiv.2206.10923.

Maheshwari, Gaurav et al. (2022). "Fair NLP Models with Differentially Private Text Encoders". In: *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Association for Computational Linguistics, pp. 6913–6930. URL: https://aclanthology.org/2022.findings-emnlp.514.

Mangold, Paul et al. (2022). "Differential Privacy has Bounded Impact on Fairness in Classification". In: *arXiv preprint arXiv:2210.16242*.

Mathur, Ajay and Giles M. Foody (2008). "Multiclass and Binary SVM Classification: Implications for Training and Classification Users". In: *IEEE Geosci. Remote. Sens. Lett.* 5.2, pp. 241–245. DOI: 10.1109/LGRS.2008.915597. URL: https://doi.org/10.1109/LGRS.2008.915597.

May, Chandler et al. (2019). "On Measuring Social Biases in Sentence Encoders". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Association for Computational Linguistics, pp. 622–628. DOI: 10.18653/v1/n19-1063. URL: https://doi.org/10.18653/v1/n19-1063.

Mehrabi, Ninareh et al. (2022). "A Survey on Bias and Fairness in Machine Learning". In: *ACM Comput. Surv.* 54.6, 115:1–115:35. DOI: 10.1145/3457607. URL: https://doi.org/10.1145/3457607.

Menon, Aditya Krishna and Robert C. Williamson (2018). "The cost of fairness in binary classification". In: *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*. Ed. by Sorelle A. Friedler and Christo Wilson. Vol. 81. Proceedings of Machine Learning Research. PMLR, pp. 107–118. URL: http://proceedings.mlr.press/v81/menon18a.html.

Metcalf, Jacob and Kate Crawford (2016). "Where are human subjects in big data research? The emerging ethics divide". In: *Big Data & Society* 3.1, p. 2053951716650211.

Michalski, Ryszard Stanislaw, Jaime Guillermo Carbonell, and Tom M Mitchell (2013). *Machine learning: An artificial intelligence approach*. Springer Science & Business Media.

Mishler, Alan and Edward H. Kennedy (2022). "FADE: FAir Double Ensemble Learning for Observable and Counterfactual Outcomes". In: *FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022*. ACM, p. 1053. DOI: 10.1145/3531146.3533167. URL: https://doi.org/10.1145/3531146.3533167.

Mittelstadt, Brent D., Sandra Wachter, and Chris Russell (2023). "The Unfairness of Fair Machine Learning: Levelling down and strict egalitarianism by default". In: *CoRR* abs/2302.02404. DOI: 10.48550/arXiv.2302.02404. arXiv: 2302.02404. URL: https://doi.org/10.48550/arXiv.2302.02404.

Mohri, Mehryar, Afshin Rostamizadeh, and Ameet Talwalkar (2018). *Foundations of machine learning*. MIT press.

Monteiro, Wellington Rodrigo and Gilberto Reynoso-Meza (2021). *Proposal of a Fair Voting Classifier Using Multi-Objective Optimization*.

Morina, Giulio et al. (2019). "Auditing and Achieving Intersectional Fairness in Classification Problems". In: *CoRR* abs/1911.01468. arXiv: 1911.01468. URL: http://arxiv.org/abs/1911.01468.

Mulsa, Rodrigo Alejandro Chávez and Gerasimos Spanakis (2020). "Evaluating Bias In Dutch Word Embeddings". In: *CoRR* abs/2011.00244. arXiv: 2011.00244. URL: https://arxiv.org/abs/2011.00244.

Nadeem, Moin, Anna Bethke, and Siva Reddy (Aug. 2021). "StereoSet: Measuring stereotypical bias in pretrained language models". In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 5356–5371. DOI: 10.18653/v1/2021.acl-long.416. URL: https://aclanthology.org/2021.acl-long.416.

Nangia, Nikita et al. (2020). "CrowS-Pairs: A Challenge Dataset for Measuring Social Biases in Masked Language Models". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Ed. by Bonnie Webber et al. Association for Computational Linguistics, pp. 1953–1967. DOI: 10.18653/v1/2020.emnlp-main.154. URL: https://doi.org/10.18653/v1/2020.emnlp-main.154.

Neal, Brady et al. (2018). "A Modern Take on the Bias-Variance Tradeoff in Neural Networks". In: *CoRR* abs/1810.08591. arXiv: 1810.08591. URL: http://arxiv.org/abs/1810.08591.

Nissenbaum, Helen (1996). "Accountability in a computerized society". In: *Science and engineering ethics* 2, pp. 25–42.

Nora, Amaury and Fran Horvath (1989). "Financial assistance: Minority enrollments and persistence". In: *Education and Urban Society* 21.3, pp. 299–311.

Noriega-Campero, Alejandro et al. (2019). "Active Fairness in Algorithmic Decision Making". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*. Ed. by Vincent Conitzer, Gillian K. Hadfield, and Shannon Vallor. ACM, pp. 77–83. DOI: 10.1145/3306618.3314277. URL: https://doi.org/10.1145/3306618.3314277.

Oh, Changdae et al. (2022). "Learning Fair Representation via Distributional Contrastive Disentanglement". In: *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022*. Ed. by Aidong Zhang and Huzefa Rangwala. ACM, pp. 1295–1305. DOI: 10.1145/3534678.3539232. URL: https://doi.org/10.1145/3534678.3539232.

O'Neil, Cathy (2016). *Weapons of math destruction: How big data increases inequality and threatens democracy*. eng. First edition. New York: Crown. ISBN: 978-0-451-49733-8.

Oneto, Luca et al. (2019). "Taking Advantage of Multitask Learning for Fair Classification". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*. Ed. by Vincent Conitzer, Gillian K. Hadfield, and Shannon Vallor. ACM, pp. 227–237. DOI: 10.1145/3306618.3314255. URL: https://doi.org/10.1145/3306618.3314255.

Otterbacher, Jahna, Jo Bates, and Paul Clough (2017). "Competent Men and Warm Women: Gender Stereotypes and Backlash in Image Search Results". In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: Association for Computing Machinery, 6620–6631. ISBN: 9781450346559. DOI: 10.1145/3025453.3025727. URL: https://doi.org/10.1145/3025453.3025727.

Ozdayi, Mustafa Safa, Murat Kantarcioglu, and Rishabh Iyer (2021). "BiFair: Training Fair Models with Bilevel Optimization". In: *arXiv preprint arXiv:2106.04757*.

Padh, Kirtan et al. (2021). "Addressing fairness in classification with a model-agnostic multi-objective algorithm". In: *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence, UAI 2021, Virtual Event, 27-30 July 2021*. Ed. by Cassio P. de Campos, Marloes H. Maathuis, and Erik Quaeghebeur. Vol. 161. Proceedings of Machine Learning Research. AUAI Press, pp. 600–609. URL: https://proceedings.mlr.press/v161/padh21a.html.

Parraga, Otávio et al. (2022). "Debiasing Methods for Fairer Neural Models in Vision and Language Research: A Survey". In: *CoRR* abs/2211.05617. DOI: 10.48550/arXiv.2211.05617. arXiv: 2211.05617. URL: https://doi.org/10.48550/arXiv.2211.05617.

Paszke, Adam et al. (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach et al. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

Patel, Veer and Manan Shah (2022). "Artificial intelligence and machine learning in drug discovery and development". In: *Intelligent Medicine* 2.3, pp. 134–140.

Pedreschi, Dino, Salvatore Ruggieri, and Franco Turini (2008). "Discrimination-aware data mining". In: *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*. Ed. by Ying Li, Bing Liu, and Sunita Sarawagi. ACM, pp. 560–568. DOI: 10.1145/1401890.1401959. URL: https://doi.org/10.1145/1401890.1401959.

Pessach, Dana and Erez Shmueli (2023). "A Review on Fairness in Machine Learning". In: *ACM Comput. Surv.* 55.3, 51:1–51:44. DOI: 10.1145/3494672. URL: https://doi.org/10.1145/3494672.

Petersen, Nancy S and Melvin R Novick (1976). "An evaluation of some models for culture-fair selection". In: *Journal of Educational Measurement*, pp. 3–29.

Petrovic, Andrija et al. (2022). "FAIR: Fair adversarial instance re-weighting". In: *Neurocomputing* 476, pp. 14–37. DOI: 10.1016/j.neucom.2021.12.082. URL: https://doi.org/10.1016/j.neucom.2021.12.082.

Phelan, Sean M et al. (2015). "Impact of weight bias and stigma on quality of care and outcomes for patients with obesity". In: *Obesity reviews* 16.4, pp. 319–326.

Pin Calmon, Flávio du et al. (2018). "Data Pre-Processing for Discrimination Prevention: Information-Theoretic Optimization and Analysis". In: *IEEE J. Sel. Top. Signal Process.* 12.5, pp. 1106–1119. DOI: 10.1109/JSTSP.2018.2865887. URL: https://doi.org/10.1109/JSTSP.2018.2865887.

Plant, Richard, Dimitra Gkatzia, and Valerio Giuffrida (2021). "CAPE: Context-Aware Private Embeddings for Private Language Learning". In: *arXiv preprint arXiv:2108.12318*.

Pleiss, Geoff et al. (2017). "On Fairness and Calibration". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon et al., pp. 5680–5689. URL: https://proceedings.neurips.cc/paper/2017/hash/b8b9c74ac526fffbeb2d39ab038d1cd7-Abstract.html.

Pujol, David et al. (2020). "Fair decision making using privacy-protected data". In: *FAT* '20: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, January 27-30, 2020*. Ed. by Mireille Hildebrandt et al. ACM, pp. 189–199. DOI: 10.1145/3351095.3372872. URL: https://doi.org/10.1145/3351095.3372872.

Qian, Ning (1999). "On the momentum term in gradient descent learning algorithms". In: *Neural Networks* 12.1, pp. 145–151. DOI: 10.1016/S0893-6080(98)00116-6. URL: https://doi.org/10.1016/S0893-6080(98)00116-6.

Qian, Rebecca et al. (2022). “Perturbation Augmentation for Fairer NLP”. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Association for Computational Linguistics, pp. 9496–9521. DOI: 10.18653/V1/2022.EMNLP-MAIN.646. URL: https://doi.org/10.18653/v1/2022.emnlp-main.646.

Quadrianto, Novi, Viktoriia Sharmanska, and Oliver Thomas (2018). “Neural Styling for Interpretable Fair Representations”. In: *CoRR* abs/1810.06755. arXiv: 1810.06755. URL: http://arxiv.org/abs/1810.06755.

Quillian, Lincoln et al. (2017). “Hiring discrimination against Black Americans hasn’t declined in 25 years”. In: *Harvard Business Review* 11.

Raff, Edward and Jared Sylvester (2018). “Gradient Reversal against Discrimination: A Fair Neural Network Learning Approach”. In: *5th IEEE International Conference on Data Science and Advanced Analytics, DSAA 2018, Turin, Italy, October 1-3, 2018*. Ed. by Francesco Bonchi et al. IEEE, pp. 189–198. DOI: 10.1109/DSAA.2018.00029. URL: https://doi.org/10.1109/DSAA.2018.00029.

Raita, Yoshihiko et al. (2019). “Emergency department triage prediction of clinical outcomes using machine learning models”. In: *Critical care* 23.1, pp. 1–13.

Ramesh, Aditya et al. (2022). “Hierarchical Text-Conditional Image Generation with CLIP Latents”. In: *CoRR* abs/2204.06125. DOI: 10.48550/ARXIV.2204.06125. arXiv: 2204.06125. URL: https://doi.org/10.48550/arXiv.2204.06125.

Ramesh, Dadi and Suresh Kumar Sanampudi (2022). “An automated essay scoring systems: a systematic literature review”. In: *Artificial Intelligence Review* 55.3, pp. 2495–2527.

Rashid, Ahmad et al. (2020). “Towards zero-shot knowledge distillation for natural language processing”. In: *arXiv preprint arXiv:2012.15495*.

Ravfogel, Shauli et al. (2020). “Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by Dan Jurafsky et al. Association for Computational Linguistics, pp. 7237–7256. DOI: 10.18653/v1/2020.acl-main.647. URL: https://doi.org/10.18653/v1/2020.acl-main.647.

Ravfogel, Shauli et al. (2022). “Linear Adversarial Concept Erasure”. In: *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*. Ed. by Kamalika Chaudhuri et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, pp. 18400–18421. URL: https://proceedings.mlr.press/v162/ravfogel22a.html.

Redmond, Michael and Alok Baveja (2002). “A data-driven software tool for enabling cooperative information sharing among police departments”. In: *European Journal of Operational Research* 141.3, pp. 660–678.

Rennie, Jason DM (2005). “Smooth hinge classification”. In: *Proceeding of Massachusetts Institute of Technology*.

Ribeiro, Marco Túlio, Sameer Singh, and Carlos Guestrin (2016). “"Why Should I Trust You?": Explaining the Predictions of Any Classifier”. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*. Ed. by Balaji Krishnapuram et al. ACM, pp. 1135–1144. DOI: 10.1145/2939672.2939778. URL: https://doi.org/10.1145/2939672.2939778.

Robbins, Herbert E. (1951). “A Stochastic Approximation Method”. In: *Annals of Mathematical Statistics* 22, pp. 400–407. URL: https://api.semanticscholar.org/CorpusID:16945044.

Roh, Yuji et al. (2020). "FairBatch: Batch Selection for Model Fairness". In: *International Conference on Learning Representations*.

Roh, Yuji et al. (2021). "Sample Selection for Fair and Robust Training". In: *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*. Ed. by Marc'Aurelio Ranzato et al., pp. 815–827. URL: https://proceedings.neurips.cc/paper/2021/hash/07563a3fe3bbe7e3ba84431ad9d055af-Abstract.html.

Romanov, Alexey et al. (June 2019). "What's in a Name? Reducing Bias in Bios without Access to Protected Attributes". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4187–4195. DOI: 10.18653/v1/N19-1424. URL: https://aclanthology.org/N19-1424.

Roth, Derek (2018). "A Comparison of Fairness-Aware Machine Learning Algorithms". PhD thesis.

Roy, Arjun, Vasileios Iosifidis, and Eirini Ntoutsi (2021). "Multi-Fair Pareto Boosting". In: *CoRR* abs/2104.13312. arXiv: 2104.13312. URL: https://arxiv.org/abs/2104.13312.

Ruder, Sebastian (2016). "An overview of gradient descent optimization algorithms". In: *CoRR* abs/1609.04747. arXiv: 1609.04747. URL: http://arxiv.org/abs/1609.04747.

Sabbaghi, Shiva Omrani and Aylin Caliskan (2022). "Measuring Gender Bias in Word Embeddings of Gendered Languages Requires Disentangling Grammatical Gender Signals". In: *AIES '22: AAAI/ACM Conference on AI, Ethics, and Society, Oxford, United Kingdom, May 19 - 21, 2021*. Ed. by Vincent Conitzer et al. ACM, pp. 518–531. DOI: 10.1145/3514094.3534176. URL: https://doi.org/10.1145/3514094.3534176.

Sadeghi, Bashir, Runyi Yu, and Vishnu Boddeti (2019). "On the Global Optima of Kernelized Adversarial Representation Learning". In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, pp. 7970–7978. DOI: 10.1109/ICCV.2019.00806. URL: https://doi.org/10.1109/ICCV.2019.00806.

Salimi, Babak, Bill Howe, and Dan Suciu (2019). "Data Management for Causal Algorithmic Fairness". In: *IEEE Data Eng. Bull.* 42.3, pp. 24–35. URL: http://sites.computer.org/debull/A19sept/p24.pdf.

Salimi, Babak et al. (2019). "Capuchin: Causal Database Repair for Algorithmic Fairness". In: *CoRR* abs/1902.08283. arXiv: 1902.08283. URL: http://arxiv.org/abs/1902.08283.

Sanh, Victor et al. (2019). "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter". In: *arXiv preprint arXiv:1910.01108*.

Sap, Maarten et al. (2019). "The risk of racial bias in hate speech detection". In: *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1668–1678.

Schröder, Sarah et al. (2021). "Evaluating metrics for bias in word embeddings". In: *arXiv preprint arXiv:2111.07864*.

Selbst, Andrew D et al. (2019). "Fairness and abstraction in sociotechnical systems". In: *Proceedings of the conference on fairness, accountability, and transparency*, pp. 59–68.

Shalev-Shwartz, Shai and Shai Ben-David (2014). *Understanding Machine Learning - From Theory to Algorithms*. Cambridge University Press. ISBN: 978-1-10-705713-5.

Shalev-Shwartz, Shai and Shai Ben-David (2014). *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Sharma, Shubham et al. (2020). “Data Augmentation for Discrimination Prevention and Bias Disambiguation”. In: *AIES ’20: AAAI/ACM Conference on AI, Ethics, and Society, New York, NY, USA, February 7-8, 2020*. Ed. by Annette N. Markham et al. ACM, pp. 358–364. DOI: 10.1145/3375627.3375865. URL: https://doi.org/10.1145/3375627.3375865.

Shelby, Renee et al. (2022). “Identifying Sociotechnical Harms of Algorithmic Systems: Scoping a Taxonomy for Harm Reduction”. In: *arXiv preprint arXiv:2210.05791*.

Shokri, Reza and Vitaly Shmatikov (2015). “Privacy-Preserving Deep Learning”. In: *CCS*.

Shokri, Reza et al. (2017). “Membership Inference Attacks Against Machine Learning Models”. In: *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. IEEE Computer Society, pp. 3–18. DOI: 10.1109/SP.2017.41. URL: https://doi.org/10.1109/SP.2017.41.

Singh, Arashdeep et al. (2022). “Developing a Novel Fair-Loan Classifier through a Multi-Sensitive Debiasing Pipeline: DualFair”. In: *Mach. Learn. Knowl. Extr.* 4.1, pp. 240–253. DOI: 10.3390/make4010011. URL: https://doi.org/10.3390/make4010011.

Singh, Jagendra and Vijay Kumar Bohat (2021). “Neural Network Model for Recommending Music Based on Music Genres”. In: *2021 International Conference on Computer Communication and Informatics (ICCCI)*. IEEE, pp. 1–6.

Sinha, Arvind Kumar, Md Amir Khusru Akhtar, and Ashwani Kumar (2021). “Resume screening using natural language processing and machine learning: A systematic review”. In: *Machine Learning And Information Processing: Proceedings Of ICMLIP 2020*, pp. 207–214.

Song, Congzheng and Ananth Raghunathan (2020). “Information Leakage in Embedding Models”. In: *CCS ’20: 2020 ACM SIGSAC Conference on Computer and Communications Security, Virtual Event, USA, November 9-13, 2020*. Ed. by Jay Ligatti et al. ACM, pp. 377–390. DOI: 10.1145/3372297.3417270. URL: https://doi.org/10.1145/3372297.3417270.

Sprietsma, Maresa (2013). “Discrimination in grading: Experimental evidence from primary school teachers”. In: *Empirical economics* 45, pp. 523–538.

Stanczak, Karolina and Isabelle Augenstein (2021). “A Survey on Gender Bias in Natural Language Processing”. In: *CoRR* abs/2112.14168. arXiv: 2112.14168. URL: https://arxiv.org/abs/2112.14168.

Strandburg, Katherine J (2019). “Rulemaking and inscrutable automated decision tools”. In: *Columbia Law Review* 119.7, pp. 1851–1886.

Su, Yixuan et al. (2021). “Plan-then-Generate: Controlled Data-to-Text Generation via Planning”. In: *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*. Ed. by Marie-Francine Moens et al. Association for Computational Linguistics, pp. 895–909. DOI: 10.18653/V1/2021.FINDINGS-EMNLP.76. URL: https://doi.org/10.18653/v1/2021.findings-emnlp.76.

Subramanian, Shivashankar et al. (2021). “Evaluating Debiasing Techniques for Intersectional Biases”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*. Ed. by Marie-Francine Moens et al. Association for Computational Linguistics, pp. 2492–2498. DOI: 10.18653/v1/2021.emnlp-main.193. URL: https://doi.org/10.18653/v1/2021.emnlp-main.193.

Suresh, Harini and John V. Guttag (2019). “A Framework for Understanding Unintended Consequences of Machine Learning”. In: *CoRR* abs/1901.10002. arXiv: 1901.10002. URL: http://arxiv.org/abs/1901.10002.

— (2021). "A Framework for Understanding Sources of Harm throughout the Machine Learning Life Cycle". In: *EAAMO 2021: ACM Conference on Equity and Access in Algorithms, Mechanisms, and Optimization, Virtual Event, USA, October 5 - 9, 2021*. ACM, 17:1–17:9. DOI: 10.1145/3465416.3483305. URL: https://doi.org/10.1145/3465416.3483305.

Sutton, Richard S (1986). "Two problems with backpropagation and other steepest-descent learning procedures for networks". In: *Proc. of Eightth Annual Conference of the Cognitive Science Society*, pp. 823–831.

Sweeney, Latanya (2013). "Discrimination in online ad delivery: Google ads, black names and white names, racial discrimination, and click advertising". In: *Queue* 11.3, pp. 10–29.

Tan, Yi Chern and L. Elisa Celis (2019). "Assessing Social and Intersectional Biases in Contextualized Word Representations". In: *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*. Ed. by Hanna M. Wallach et al., pp. 13209–13220. URL: https://proceedings.neurips.cc/paper/2019/hash/201d546992726352471cfea6b0df0a48-Abstract.html.

Thanh, Binh Luong, Salvatore Ruggieri, and Franco Turini (2011). "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, San Diego, CA, USA, August 21-24, 2011*. Ed. by Chid Apté, Joydeep Ghosh, and Padhraic Smyth. ACM, pp. 502–510. DOI: 10.1145/2020408.2020488. URL: https://doi.org/10.1145/2020408.2020488.

Thanh-Tung, Hoang and Truyen Tran (2020). "Catastrophic forgetting and mode collapse in GANs". In: *2020 international joint conference on neural networks (ijcnn)*. IEEE, pp. 1–10.

Tibshirani, Robert (1996). "Regression shrinkage and selection via the lasso". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58.1, pp. 267–288.

Touvron, Hugo et al. (2023). "LLaMA: Open and Efficient Foundation Language Models". In: *CoRR* abs/2302.13971. DOI: 10.48550/ARXIV.2302.13971. arXiv: 2302.13971. URL: https://doi.org/10.48550/arXiv.2302.13971.

Valera, Isabel, Adish Singla, and Manuel Gomez Rodriguez (2018). "Enhancing the Accuracy and Fairness of Human Decision Making". In: *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*. Ed. by Samy Bengio et al., pp. 1774–1783. URL: https://proceedings.neurips.cc/paper/2018/hash/0a113ef6b61820daa5611c870ed8d5ee-Abstract.html.

Vashishth, Shikhar et al. (2019). "Attention interpretability across nlp tasks". In: *arXiv preprint arXiv:1909.11218*.

Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon et al., pp. 5998–6008. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Ventura, Francesco et al. (2021). "Explaining the Deep Natural Language Processing by Mining Textual Interpretable Features". In: *arXiv preprint arXiv:2106.06697*.

Verma, Sahil and Julia Rubin (2018). "Fairness definitions explained". In: *Proceedings of the International Workshop on Software Fairness, FairWare@ICSE 2018, Gothenburg, Sweden, May 29, 2018*. Ed. by Yuriy Brun, Brittany Johnson, and Alexandra Meliou.

ACM, pp. 1–7. DOI: 10.1145/3194770.3194776. URL: https://doi.org/10.1145/3194770.3194776.

Voita, Elena and Ivan Titov (2020). "Information-Theoretic Probing with Minimum Description Length". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Ed. by Bonnie Webber et al. Association for Computational Linguistics, pp. 183–196. DOI: 10.18653/v1/2020.emnlp-main.14. URL: https://doi.org/10.18653/v1/2020.emnlp-main.14.

Wald, Abraham (1945). "Statistical decision functions which minimize the maximum risk". In: *Annals of Mathematics*, pp. 265–280.

Wang, Angelina, Vikram V Ramaswamy, and Olga Russakovsky (2022). "Towards intersectionality in machine learning: Including more identities, handling underrepresentation, and performing evaluation". In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 336–349.

Wang, Angelina and Olga Russakovsky (2021). "Directional Bias Amplification". In: *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, pp. 10882–10893. URL: http://proceedings.mlr.press/v139/wang21t.html.

Wang, Hao-Chuan, Chun-Yen Chang, and Tsai-Yen Li (2008). "Assessing creative problem-solving with automated text grading". In: *Computers & Education* 51.4, pp. 1450–1466.

Wang, Jingbo, Yannan Li, and Chao Wang (2022). "Synthesizing Fair Decision Trees via Iterative Constraint Solving". In: *Computer Aided Verification - 34th International Conference, CAV 2022, Haifa, Israel, August 7-10, 2022, Proceedings, Part II*. Ed. by Sharon Shoham and Yakir Vizel. Vol. 13372. Lecture Notes in Computer Science. Springer, pp. 364–385. DOI: 10.1007/978-3-031-13188-2_18. URL: https://doi.org/10.1007/978-3-031-13188-2_18.

Wang, Tianlu et al. (2019). "Balanced Datasets Are Not Enough: Estimating and Mitigating Gender Bias in Deep Image Representations". In: *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE, pp. 5309–5318. DOI: 10.1109/ICCV.2019.00541. URL: https://doi.org/10.1109/ICCV.2019.00541.

Wang, Tianlu et al. (2020). "Double-Hard Debias: Tailoring Word Embeddings for Gender Bias Mitigation". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*. Ed. by Dan Jurafsky et al. Association for Computational Linguistics, pp. 5443–5453. DOI: 10.18653/v1/2020.acl-main.484. URL: https://doi.org/10.18653/v1/2020.acl-main.484.

Wang, Xingyou, Weijie Jiang, and Zhiyong Luo (2016). "Combination of Convolutional and Recurrent Neural Network for Sentiment Analysis of Short Texts". In: *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*. Ed. by Nicoletta Calzolari, Yuji Matsumoto, and Rashmi Prasad. ACL, pp. 2428–2437. URL: https://aclanthology.org/C16-1229/.

Wang, Yequan et al. (2018). "Sentiment Analysis by Capsules". In: *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*. Ed. by Pierre-Antoine Champin et al. ACM, pp. 1165–1174. DOI: 10.1145/3178876.3186015. URL: https://doi.org/10.1145/3178876.3186015.

Weber, Max (2016). "Economy and society". In: *Democracy: A Reader*. Columbia University Press, pp. 247–251.

Webster, Kellie et al. (2020). "Measuring and Reducing Gendered Correlations in Pre-trained Models". In: *CoRR* abs/2010.06032. arXiv: 2010.06032. URL: https://arxiv.org/abs/2010.06032.

Weerts, Hilde et al. (2023). *Fairlearn: Assessing and Improving Fairness of AI Systems*. arXiv: 2303.16626 [cs.LG].

Weidinger, Laura et al. (2021). "Ethical and social risks of harm from language models". In: *arXiv preprint arXiv:2112.04359*.

Weidinger, Laura et al. (2022). "Taxonomy of risks posed by language models". In: *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pp. 214–229.

Williams, Robert L et al. (1980). "The war against testing: A current status report". In: *The Journal of Negro Education* 49.3, pp. 263–273.

Woodworth, Blake E. et al. (2017). "Learning Non-Discriminatory Predictors". In: *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*. Ed. by Satyen Kale and Ohad Shamir. Vol. 65. Proceedings of Machine Learning Research. PMLR, pp. 1920–1953. URL: http://proceedings.mlr.press/v65/woodworth17a.html.

Wu, Chuhan et al. (2022). "Semi-FairVAE: Semi-supervised Fair Representation Learning with Adversarial Variational Autoencoder". In: *CoRR* abs/2204.00536. DOI: 10.48550/arXiv.2204.00536. arXiv: 2204.00536. URL: https://doi.org/10.48550/arXiv.2204.00536.

Wu, Yongkai, Lu Zhang, and Xintao Wu (2019). "On convexity and bounds of fairness-aware classification". In: *The World Wide Web Conference*, pp. 3356–3362.

Wu, Zongze, Dani Lischinski, and Eli Shechtman (2021). "StyleSpace Analysis: Disentangled Controls for StyleGAN Image Generation". In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*. Computer Vision Foundation / IEEE, pp. 12863–12872. DOI: 10.1109/CVPR46437.2021.01267.

Xu, Depeng, Wei Du, and Xintao Wu (2020). "Removing Disparate Impact of Differentially Private Stochastic Gradient Descent on Model Accuracy". In: *CoRR* abs/2003.03699. arXiv: 2003.03699. URL: https://arxiv.org/abs/2003.03699.

Xu, Depeng et al. (2018). "FairGAN: Fairness-aware Generative Adversarial Networks". In: *IEEE International Conference on Big Data (IEEE BigData 2018), Seattle, WA, USA, December 10-13, 2018*. Ed. by Naoki Abe et al. IEEE, pp. 570–575. DOI: 10.1109/BIGDATA.2018.8622525. URL: https://doi.org/10.1109/BigData.2018.8622525.

Xu, Depeng et al. (2019a). "Achieving Causal Fairness through Generative Adversarial Networks". In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*. Ed. by Sarit Kraus. ijcai.org, pp. 1452–1458. DOI: 10.24963/ijcai.2019/201. URL: https://doi.org/10.24963/ijcai.2019/201.

Xu, Depeng et al. (2019b). "FairGAN$^+$: Achieving Fair Data Generation and Classification through Generative Adversarial Nets". In: *2019 IEEE International Conference on Big Data (IEEE BigData), Los Angeles, CA, USA, December 9-12, 2019*. Ed. by Chaitanya K. Baru et al. IEEE, pp. 1401–1406. DOI: 10.1109/BIGDATA47090.2019.9006322. URL: https://doi.org/10.1109/BigData47090.2019.9006322.

Yan, Shen, Hsien-Te Kao, and Emilio Ferrara (2020). "Fair Class Balancing: Enhancing Model Fairness without Observing Sensitive Attributes". In: *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*. Ed. by Mathieu d'Aquin et al. ACM, pp. 1715–

1724. DOI: 10.1145/3340531.3411980. URL: https://doi.org/10.1145/3340531.3411980.

Yang, Forest, Mouhamadou Cisse, and Oluwasanmi Koyejo (2020). "Fairness with Overlapping Groups; a Probabilistic Perspective". In: *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*. Ed. by Hugo Larochelle et al.

Yang, Zitong et al. (2020). "Rethinking Bias-Variance Trade-off for Generalization of Neural Networks". In: *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119. Proceedings of Machine Learning Research. PMLR, pp. 10767–10777. URL: http://proceedings.mlr.press/v119/yang20j.html.

Yeung, Karen (2018). "Algorithmic regulation: A critical interrogation". In: *Regulation & governance* 12.4, pp. 505–523.

Yona, Gal and Guy N. Rothblum (2018). "Probably Approximately Metric-Fair Learning". In: *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*. Ed. by Jennifer G. Dy and Andreas Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, pp. 5666–5674. URL: http://proceedings.mlr.press/v80/yona18a.html.

Yurochkin, Mikhail, Amanda Bower, and Yuekai Sun (2020). "Training individually fair ML models with sensitive subspace robustness". In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net. URL: https://openreview.net/forum?id=B1gdkxHFDH.

Zafar, Muhammad Bilal et al. (2017a). "Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment". In: *Proceedings of the 26th international conference on world wide web*, pp. 1171–1180.

Zafar, Muhammad Bilal et al. (2017b). "Fairness constraints: Mechanisms for fair classification". In: *Artificial Intelligence and Statistics*. PMLR, pp. 962–970.

Zehlike, Meike, Philipp Hacker, and Emil Wiedemann (2020). "Matching code and law: achieving algorithmic fairness with optimal transport". In: *Data Min. Knowl. Discov.* 34.1, pp. 163–200. DOI: 10.1007/s10618-019-00658-8. URL: https://doi.org/10.1007/s10618-019-00658-8.

Zemel, Richard S. et al. (2013). "Learning Fair Representations". In: *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*. Vol. 28. JMLR Workshop and Conference Proceedings. JMLR.org, pp. 325–333. URL: http://proceedings.mlr.press/v28/zemel13.html.

Zhang, Hongyi et al. (2018). "mixup: Beyond Empirical Risk Minimization". In: *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net. URL: https://openreview.net/forum?id=r1Ddp1-Rb.

Zhang, Lu, Yongkai Wu, and Xintao Wu (2017). "A Causal Framework for Discovering and Removing Direct and Indirect Discrimination". In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. Ed. by Carles Sierra. ijcai.org, pp. 3929–3935. DOI: 10.24963/ijcai.2017/549. URL: https://doi.org/10.24963/ijcai.2017/549.

Zhang, Tong (2004). "Solving large scale linear prediction problems using stochastic gradient descent algorithms". In: *Machine Learning, Proceedings of the Twenty-first International Conference (ICML 2004), Banff, Alberta, Canada, July 4-8, 2004*. Ed. by Carla E. Brodley. Vol. 69. ACM International Conference Proceeding Series. ACM. DOI: 10.1145/1015330.1015332. URL: https://doi.org/10.1145/1015330.1015332.

Zhang, Zhifei, Yang Song, and Hairong Qi (2017). "Age progression/regression by conditional adversarial autoencoder". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5810–5818.

Zhao, Eric et al. (2022). "Scaling Fair Learning to Hundreds of Intersectional Groups". In.

Zhao, Jieyu et al. (2018). "Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods". In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*. Ed. by Marilyn A. Walker, Heng Ji, and Amanda Stent. Association for Computational Linguistics, pp. 15–20. DOI: 10.18653/v1/n18-2003. URL: https://doi.org/10.18653/v1/n18-2003.

Zietlow, Dominik et al. (2022). "Leveling Down in Computer Vision: Pareto Inefficiencies in Fair Deep Classifiers". In: *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*. IEEE, pp. 10400–10411. DOI: 10.1109/CVPR52688.2022.01016. URL: https://doi.org/10.1109/CVPR52688.2022.01016.

Zliobaite, Indre (2015). "On the relation between accuracy and fairness in binary classification". In: *arXiv preprint arXiv:1505.05723*.

Žliobaite, Indre, Faisal Kamiran, and Toon Calders (2011). "Handling conditional discrimination". In: *2011 IEEE 11th international conference on data mining*. IEEE, pp. 992–1001.