Investigating Trade-offs For Fair Machine Learning Systems

Max Hort

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Computer Science

University College London

January 27, 2023

I, Max Hort, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work. I list below the chapters based on publications:

Chapter 3

- Max Hort, Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. Bias Mitigation for Machine Learning Classifiers: A Comprehensive Survey. *arXiv preprint arXiv:2207.07068*, 2022

Chapter 4

- Max Hort, Jie M. Zhang, Federica Sarro, and Mark Harman. Fairea: A Model Behaviour Mutation Approach to Benchmarking Bias Mitigation Methods. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2021

Chapter 5

- Max Hort, Jie M Zhang, Federica Sarro, and Mark Harman. Search-based Automatic Repair for Fairness and Accuracy in Decision-making Software. *Under review*, 2022

Chapter 6

- Max Hort and Federica Sarro. Privileged and Unprivileged Groups: An Empirical Study on the Impact of the Age Attribute on Fairness. In *International Workshop on Equitable Data and Technology (FairWare '22)*. ACM, 2022

Additional Articles

Additionally, I co-authored the following articles, which are not included in this thesis:

1. Zhenpeng Chen, Jie M Zhang, Max Hort, Federica Sarro, and Mark Harman. Fairness testing: A comprehensive survey and analysis of trends. *arXiv preprint arXiv:2207.10223*, 2022

2. Max Hort and Federica Sarro. The Effect of Offspring Population Size on NSGA-II: A Preliminary Study. In *Proceedings of the 2021 Genetic and Evolutionary Computation Conference Companion*, 2021

3. Max Hort, Maria Kechagia, Federica Sarro, and Mark Harman. A Survey of Performance Optimization for Mobile Applications. *IEEE Transactions on Software Engineering (TSE)*, 2021

4. Max Hort and Federica Sarro. Optimising Word Embeddings With Search-Based Approaches. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 269–270, 2020

5. Emeralda Sesari, Max Hort, and Federica Sarro. An Empirical Study on the Fairness of Pre-trained Word Embeddings. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing*, 2022

6. James Zhong, Max Hort, and Federica Sarro. Py2Cy: A Genetic Improvement Tool To Speed Up Python. In *Genetic and Evolutionary Computation Conference Companion (GECCO '22 Companion)*, 2022

7. Max Hort, Rebecca Moussa, and Federica Sarro. Multi-objective search for gender-fair and semantically correct word embeddings. *Applied Soft Computing*, 133:109916, 2023

8. Minghua Ma, Zhao Tian, Max Hort, Federica Sarro, Hongyu Zhang, Qingwei Lin, and Dongmei Zhang. Enhanced fairness testing via generating effective initial individual discriminatory instances. *arXiv preprint arXiv:2209.08321*, 2022

9. Max Hort and Federica Sarro. Did You Do Your Homework? Raising Awareness on Software Fairness and Discrimination. In *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 1322–1326. IEEE, 2021

Abstract

Fairness in software systems aims to provide algorithms that operate in a non-discriminatory manner, with respect to protected attributes such as gender, race, or age. Ensuring fairness is a crucial non-functional property of data-driven Machine Learning systems. Several approaches (i.e., bias mitigation methods) have been proposed in the literature to reduce bias of Machine Learning systems. However, this often comes hand in hand with performance deterioration. Therefore, this thesis addresses trade-offs that practitioners face when debiasing Machine Learning systems.

At first, we perform a literature review to investigate the current state of the art for debiasing Machine Learning systems. This includes an overview of existing debiasing techniques and how they are evaluated (e.g., how is bias measured).

As a second contribution, we propose a benchmarking approach that allows for an evaluation and comparison of bias mitigation methods and their trade-offs (i.e., how much performance is sacrificed for improving fairness).

Afterwards, we propose a debiasing method ourselves, which modifies already trained Machine Learning models, with the goal to improve both, their fairness and accuracy.

Moreover, this thesis addresses the challenge of how to deal with fairness with regards to age. This question is answered with an empirical evaluation on real-world datasets.

Impact Statement

The work presented in this thesis has an impact on the study of trade-offs for data-driven Machine Learning systems in research and real world applications. At first, we performed a literature review to understand the state-of-the-art of bias mitigation methods. This literature review does not only help researchers and practitioners to see what approaches already exist and could be used in practical settings, but also how to proceed when proposing a novel debiasing method themselves.

To investigate the performance and quality of existing and newly-created bias mitigation methods, *Fairea*, our benchmarking approach, can be used. In addition to comparing bias mitigation methods, it supports practitioners in choosing bias mitigation methods for their use case.

Next, we devise a post-processing method for simultaneous improvements in performance and fairness, which empowers the applicability of bias mitigation methods in real-world settings, as practitioners are enabled to achieve fairness improvements without sacrificing performance.

The investigation of different age thresholds on fairness (i.e., at what age is the population divided in “young” and “old”), helps researchers that consider fairness with regards to age.

Acknowledgements

First and foremost I would like to thank my supervisors Professor Federica Sarro and Professor Mark Harman for their continuous encouragement and valuable guidance throughout my studies. I have learnt valuable lessons in software engineering and statistical tests. More so, I would like to thank them for their advice on research ideas and allowing the change to focus on fairness and making this thesis possible.

Also, I would like to thank all of the CRESTies and SOLARies who I have met during the past three years, many of which supported or collaborated on this research to make it sound and comprehensible.

Last but by no means the least, I am grateful for my friends, and family, who bore with me during times of lockdowns.

Contents

List of Figures

List of Tables

1

Introduction

You strive for your justice, and I strive for mine.

– ペイン

Software Engineering (SE) research is concerned with designing, implementing, maintaining and testing software in a systematic manner. While software systems do not explicitly incorporate discrimination, they are not spared from biased decisions and unfairness. Social discrimination occurs when a decision about a person is unfairly biased with regards to sensitive attributes such as race or gender. This suppresses opportunities of deprived groups or individuals (e.g., in education, or finance) [14, 15]. For example, Machine Learning (ML) software, which nowadays is widely used in critical decision-making software such as software for loan applicant [16] and justice risk assessment [17, 18], has shown to exhibit discriminatory behaviours [19]. The decisions made by ML software can cause discrimination on two levels: individual and group-level. Individual discrimination occurs when a decision making software makes different predictions for two individuals with a high degree of similarity (e.g., a ML system making different predictions if the gender of an individual would change). Discrimination on a group-level describes situations in which one population group receives favourable treatment over another (e.g., white defendants have been found less likely to be labelled as future criminal by risk assessment software than non-white defendants [17]). Such discriminatory behaviours can be highly detrimental, affecting human rights [20], profit and revenue [21], and can also fall under regulatory control [19, 22, 23].

Fairness is an important non-functional property of software [24] and has been widely studied in the past few years in both software engineering [24,25,26,27,28] and machine learning literature [14,29,30,31]. Generally speaking, software fairness aims to provide algorithms that operate in a non-discriminatory manner [32]. Some approaches adapt the training data to reduce data bias [28, 33, 34, 35], some create classification models that consider fairness during the training process [29,36,37,38,39], others apply changes to the model prediction outcomes to reduce bias [15, 31, 40, 41, 42]. In particular, this thesis focuses on bias mitigation methods that aim to improve the fairness of ML systems with respect to two population groups in binary decision systems (e.g., when applying for a credit at a bank, male and female applicants should be approved with an equal probability). While bias mitigation methods are able to increase the fairness of decision making software, the improvement often comes at the cost of a lower performance or accuracy [29]. This circumstance is called the “fairness-accuracy” trade-off [34,35,43].

1.1 Problem Statement

This thesis addresses the problem of trade-offs when applying bias mitigation methods for improving the fairness of ML systems and associated performance deterioration with respect to other objectives, such as accuracy. This problem is addressed in three stages: assessing the state-of-the-art methods for bias mitigation as well as empirical configurations for evaluating their performance; identifying and measuring their performance with regards to their fairness-accuracy trade-off; creating an approach to achieve fairness and accuracy improvements.

To assess state-of-the-art bias mitigation methods, a literature survey is performed that collects and investigates fairness measurements, datasets and benchmarking practices. The performance of existing bias mitigation methods is evaluated in an empirical study with a baseline approach, which represents a naive bias mitigation behaviour and is more competitive than existing baselines. Lastly, modifications of two ML classifiers (Logistic Regression and Decision Trees) are performed, guided by a multi-objective search procedure. The goal of this multi-objective search is simultaneously improve fairness and accuracy, challenging the notion of *fairness-accuracy trade-offs*.

1.2 Objectives

The purpose of this thesis is to provide insights and guidelines for software engineers and practitioners on how to address fairness in software systems and potential harms to performance accuracy that come along with it. The objectives can be summarized as follows:

1. Analyze state-of-the-art methods for bias reduction.

2. Measure and compare the quality of bias mitigation methods.

3. Improve the accuracy and fairness of Machine Learning software.

1.3 Contributions

The contributions of this thesis are:

1. A comprehensive literature review on existing bias mitigation methods (Chapter 3).

2. *Fairea*, a baseline approach for evaluating fairness-accuracy trade-offs (Chapter 4).

3. An empirical study on the fairness-accuracy trade-offs of existing bias mitigation methods with *Fairea*.

4. A novel post-processing methods for modifying Logistic Regression and Decision Tree classifiers (Chapter 5).

5. An empirical investigation of different age bias with regards to different age thresholds to determine young and old population groups (Chapter 6).

1.4 Thesis Overview

The rest of this thesis is organized as follows. Chapter 2 provides background information and positions this thesis within the field of software fairness. Moreover, it presents details and fairness measurements, datasets and methods that are utilized by later chapters (Chapter 4-5).

Chapter 3 presents a literature review on existing bias mitigation methods as well as details on their empirical evaluation, such as datasets investigated and metrics used to measure fairness. In Chapter 4, *Fairea*, a benchmarking approach for bias mitigation methods is proposed. *Fairea* is able to compare bias mitigation methods that sacrifice accuracy for fairness improvements with a single quantitative measure. While Chapter 4 presents a categorization and benchmarking approach for bias mitigation methods, Chapter 5 introduces a novel post-processing methods for ML classifiers with the goal to simultaneously improve accuracy and fairness. Following, Chapter 6 considers the task of choosing population groups when dealing with the sensitive attribute “age” (e.g., at what age do young and old people separate). Lastly, Chapter 7 summarizes and discusses the results.

2

Background

This chapter provides details on how to measure and improve the fairness of ML model, as well as existing works on fairness in the Software Engineering domain. It also presents fairness metrics, bias mitigation methods and datasets used in later chapters of this thesis.

First, Section 2.1 outlines definitions of fairness and describes how bias and subsequent fairness improvements can be measured. Afterwards, Section 2.2 presents the field of Software Engineering (SE) with regards to fairness considerations. The different stages of the software development lifecycle are addressed, to position this thesis within fair SE. Section 2.3 provides further details on bias mitigation methods, which enable a repair of biased software. Afterwards, Section 2.4 presents datasets, on which bias mitigation methods can be evaluated on. This resembles a subset of the information collected in the literature review (Chapter 3). Lastly, we outline the AI Fairness 360 toolkit (Section 2.5), which provides implementations of the bias mitigation methods, fairness metrics and datasets addressed in this chapter.

2.1 Measuring Fairness

Fairness metrics are designed to define and quantitatively measure ML fairness. There are two primary types of fairness as indicated by Speicher et al. [44]: individual fairness and group fairness. *Individual fairness* is satisfied when similar individuals receive the same treatment [45]. To determine the degree of similarity between to individuals, distance metrics are used, which compare their attribute values. Frequently, individual fairness is determined by assessing whether individuals who only differ in sensitive attributes receive the same treatment (e.g., “If the gender of a person would change, does the outcome remain the same?”) [46,47].

Group fairness requires that the predictive performance of a classification model is equal across different groups [48], which are divided by the values of protected attributes (i.e., race, age, sex). Groups are either *privileged* (more likely to get an advantageous outcome), or *unprivileged* (more likely to get a disadvantageous outcome).

For proceeding experiments, we consider group fairness metrics to measure bias, as these are widely adopted in the literature [28,30,43,48]; second, most bias mitigation methods are designed to optimize for group fairness. Not only can group fairness metrics be used to quantify the bias of model predictions, but also the bias in underlying datasets that are used for training. These are called dataset metrics (Section 2.1.1) and classification metrics (Section 2.1.2).

In the following, we use $\hat{y}$ to denote the predictions of a classification model. We use D to denote a group (privileged or unprivileged). We use Pr to denote probability. Section 2.1.1 and Section 2.1.2 define the metrics considered in our experiments.

2.1.1 Dataset Metrics

Dataset metrics are used to determine bias in the instances of a dataset. Mean Difference (MD) is a dataset metric which computes differences between privileged and unprivileged group in regards to how likely it is that they receive a favourable treatment (i.e., a positive label).[1]

$$\begin{aligned} MD = Pr(y = 1 | D = unprivileged) \\ - Pr(y = 1 | D = privileged) \end{aligned} \tag{2.1}$$

[1] Mean Difference can also be called Statistical Parity Difference. We choose to call it Mean Difference to not confuse it with the classification metric which is also called Statistical Parity Difference.

2.1.2 Classification Metrics

Classification metrics are used to determine the bias of predictions made by classification models. We consider three popular classification metrics: Statistical Parity Difference (SPD) [45], Equal Opportunity Difference (EOD) [41] and Average Odds Difference (AOD) [41].

SPD is a fairness metric requiring that decisions are made independently of protected attributes [39]. Positive and negative classifications for each demographic group should be identical over the whole population [45]:

$$\begin{aligned} SPD = Pr(\hat{y} = 1 | D = unprivileged) \\ -Pr(\hat{y} = 1 | D = privileged) \end{aligned} \tag{2.2}$$

2.2 Fairness in Software Engineering

Software fairness is a growing concern of software engineers, and, as highlighted in Brun and Meliou's FSE'18 vision paper [25], novel strategies need to be conceived to achieve fairness during the software development life cycle of both traditional and ML-based software systems. Recent surveys by Soremekun et al. [49] and Chen et al. [5] address developments of fairness research in the software development life cycle, and fairness testing in particular [5]. Here we outline fairness considerations for each respective stage of the software development life cycle.

2.2.1 Requirements

Requirements in the software development life cycle describe how a software system should behave and what components should be implemented [50]. In this regard, fairness is a non-functional requirement that describes an unbiased behaviour of a software system [24].

To understand the importance of fairness requirements, it can be helpful to investigate current practice employed in industry [51, 52]. Habibullah and Horkoff [51] interviewed ten practitioners to gain an understanding of the non-functional requirements that are considered in practice and guide future practices. Among other non-functional requirements, fairness is becoming more prominent. Balasubramaniam et al. [52] analyzed the ethical guidelines employed by three companies. Their results indicate that ethical properties, such as transparency, explainability, fairness, and privacy can be critical when developing AI systems.

Finkelstein et al. [53] addressed fairness requirements form a multi-objective perspective when prioritising multiple requirements with competing interests. For example, some customers may wish to receive equal spend from the developers, while others may prefer to receive an equal number of their desired requirements compared to other customers.

2.2.2 Architecture and Design

Software architecture and design describes the components of a software system and their interaction [54].

To combat the risk of incorporating biases in software systems and potential negative impacts on in software quality, ethical values, such as fairness, should be incorporated early in the design process. For this purpose, Shu et al. [55] proposed a tool, “Fairness in Design”, to support practitioners in designing fairness-aware systems and highlight potential fairness concerns.

To include users in the design process, Stumpf et al. [56] proposed CoFAIR, a method for designing user interfaces in a co-design fashion. In particular, a small numbers of users is involved in each stage of the design process, with equal say as researchers and designers.

2.2.3 Verification

Verification is used to certify that given criteria are met. In the fairness context, this could entail verifying that anti-discrimination laws are followed [48].

To verify a fair treatment of population groups, Albarghouthi et al. [57] proposed FairSquare for automatically verifying fairness properties. While FairSquare can be used for verifying fairness properties when dealing with multiple population groups, John et al. [58] considered fairness verification when dealing with individuals.

2.2.4 Testing

Fairness testing is used to reveal fairness bugs that reside in software systems. Fairness bugs refer to conflicts between. required and existing fairness requirements. A detailed survey on fairness testing techniques and trends is provided Chen et al. [5]. Here we outline a subset of popular fairness testing works.

Burnett et al. [59] proposed *GenderMag*, an approach to identify gender bias in software interfaces and respective workflows. Several tools have been proposed to test for software fairness [24] and identify instances on which systems exhibit bias (e.g., a different behaviour for individuals only differing in sensitive attributes). For example, Themis [60, 61] and AEQUITAS [62] can be used to automatically generate test suites in order to examine the extent to which individual discrimination is present in a ML model. Whereas, Aggarwal et al. [63] propose a black box approach to generate test inputs to detect individual discrimination. FairTest [64] considers multiple fairness metrics and tests outcomes based on sensitive user attributes. A different approach to fairness testing is the adaptation and changes of user profiles to test web services. One example is the work of Datta et al. [65], which tests the behaviour of Google ads by changing profile information. Another is that of Hannak et al. [66], which evaluated pricing behaviour of e-commerce sites while simulating different user features.

2.2.5 Debugging and Repair

After detecting fairness bugs in software systems with testing, debugging tools are required to repair and remove causes of bias [25]. Such a repair can also be called "bias mitigation", as stated by Chen et al. [5].

This thesis belongs to the stage of debugging and fairness repair. Proceeding, we provide further details on approaches for repairing biased software systems and achieving fairness improvements (i.e, applying bias mitigation methods).

2.3 Bias Mitigation Methods

Bias can occur at any stage of the machine learning system development process. To repair bias, researchers have applied bias mitigation methods in three different stages: pre-processing, in-processing and post-processing [32]. Pre-processing methods aim at processing the training data to reduce bias in the data. In-processing methods aim to mitigate bias during training by directly optimising algorithms. Post-processing methods change the prediction outcomes of a model to mitigate bias after the model has been trained. Following, we present methods for each of the three categories that are used for benchmarking in our experiments.

2.3.1 Pre-Processing Bias Mitigation Methods

Reweighing (RW) is a pre-processing method that applies weights to different groups in the training data to achieve fairness [35, 67]. Instances in the training data are weighted according the frequency of their label and protected attribute (e.g., less frequent combinations receive a higher weight). *Learning Fair Representations* (LFR) encodes data into an intermediate representation with the aim of obfuscating protected attribute information, while minimising the overall information disruption [68]. Calmon et al. [33] formulated the learning of fair representations as an optimization problem as well, which is called *Optimized Pre-processing*. Labels and features are transformed with three objectives in mind: decreasing bias, reducing changes to data samples, retaining utility.

2.3.2 In-Processing Bias Mitigation Methods

Zhang et al. [36] proposed a debiasing approach based on adversarial learning. They trained a Logistic Regression model to predict the label Y while preventing an adversary from predicting the protected attribute under consideration of three fairness metrics: Demographic Parity, Equality of Odds, and Equality of Opportunity. Both, predictor and adversary, are implemented as Logistic regression models. This technique is called *Adversarial Debiasing* (AD).

Kamishima et al. [30] proposed *Prejudice Remover* (PR), a regularisation approach for learning fair classification models. While classification models are ordinarily optimized for accuracy, PR includes a fairness regularisation term in the training objective.

2.3.3 Post-Processing Bias Mitigation Methods

Kamiran et al. [14, 15] introduced the notion of reject option which modifies the prediction of individuals close to the decision boundary (*Reject Option Classification*). In particular, individuals belonging to the unprivileged group receive a positive outcome and privileged individuals an unfavourable outcome. Hardt et al. [41] proposed the modification of classifiers to achieve fairness with respect to True Positive and False Positive rates. Given an unfair classifier $\widehat{Y}$, the classifier $\widetilde{Y}$ is derived by solving an optimization problem under consideration of fairness loss terms. Similarly, this procedure has been adapted for calibrated True Positive and False Positive rates [40]. The two methods are called *Equalized Odds Post-processing* and *Calibrated Equalized Odds Post-processing*, respectively.

AOD is a group fairness metric that averages the differences in True Positive Rate (TPR) and True Negative Rate (TNR) among privileged and unprivileged groups [41]:

$$\begin{aligned} AOD = \frac{1}{2}((&FPR_{D=unprivileged} - FPR_{D=privileged}) \\ &+ (TPR_{D=unprivileged} - TPR_{D=privileged})) \end{aligned} \tag{2.3}$$

EOD corresponds to the TPR difference [41]:

$$EOD = TPR_{D=unprivileged} - TPR_{D=privileged} \tag{2.4}$$

Following previous work [2, 28], we are interested in the absolute values of these metrics, thus a minimal value of zero indicates no bias detected by the corresponding metric. Larger metric values correspond to a degree of bias.

2.4 Datasets

This section presents five popular, publicly available datasets, that are used for proceeding empirical evaluations.

The Adult Census Income (**Adult**) [69] contains financial and demographic information about individuals from the 1994 U.S. census. The privileged and unprivileged groups are distinguished by whether their income is above 50 thousand dollars a year.

Table 2.1: Dataset information.

Dataset	Size	Attributes	Favourable Label	Majority Label	Protected	Privileged - Unprivileged
Adult	48,842	14	1 (income >50k)	0 (75%)	Sex Race	Male - female White - non white
COMPAS	7,214	28	0 (No recid)	0 (54%)	Sex Race	Female - male Caucasian - not Caucasian
Bank	41,188	20	1 (yes)	0 (87%)	Age	≥ 25 - < 25
German	1,000	20	1 (good credit)	1 (70%)	Age	> 25 - ≤ 25
MEPS19	15,830	138	1 ($\geq$ 10 visits)	0 (83%)	Race	White - non-white

The Bank Marketing (**Bank**) [70] dataset contains details of a direct marketing campaign performed by a Portuguese banking institution. Predictions are made to determine whether potential clients are likely to subscribe to a term deposit after receiving a phone call. The dataset also includes information on the education and type of job of individuals.

The German Credit Data (**German**) [69] dataset contains the credit information of $1,000$ individuals. A classification is made, whether individuals have a good or bad credit risk. Among others, the dataset contains additional information about the credit purpose, credit history and employment status.

The Correctional Offender Management Profiling for Alternative Sanctions (**COMPAS**) [17] dataset contains the criminal history and demographic information of offenders in Broward County, Florida. To indicate whether a previous offender is likely to re-offend, they receive a *recidivism* label.

The Medical Expenditure Panel Survey (**MEPS19**) [71] represents a large scale survey of families and individuals, their medical providers, and employers across the United States. The favourable label is determined by “Utilization” (i.e., how frequently individuals frequented medical providers).

In Table 2.1, we provide the following information about the five datasets: number of rows and features, the favourable label and majority class. In addition, we list the protected attributes for each dataset and the respective privileged and unprivileged groups for each protected attribute.

2.5 AI Fairness 360 Toolkit

The AI Fairness 360 Toolkit (AIF360) is a popular open-source framework for fairness research, proposed by Bellamy et al. [72]. The AIF360 toolkit implements and makes available a diverse set of bias mitigation methods, datasets and fairness metrics for research and industrial settings. The design of the the AIF360 toolkit allows for an easy use and extensibility with new functionalities, such as the inclusion of additional metrics or bias mitigation methods. To this end, AIF360 is available via a Python library or online demo.[2]

In the experiments of following chapters, we use AIF360 implementations for datasets, metrics and existing bias mitigation methods for benchmarking. For example, the configuration of protected attributes and their privileged and unprivileged groups displayed in Section 2.4, are in accordance with the choices provided by the AIF360 toolkit.

[2] https://aif360.mybluemix.net/

3

Literature Review

> *Science is organized knowledge; and before knowledge can be organized, some of it must first be possessed.*
>
> – Herbert Spencer

Machine Learning (ML) has been increasingly popular in recent years, both in the diversity and importance of applications [73]. ML is used in a variety of critical decision-making applications including justice risk assessments [17, 18] and job recommendations [74].

While ML systems have the advantage to relieve humans from tedious tasks and are able to perform complex calculations at a higher speed [75], they are only as good as the data on which they are trained [76]. ML algorithms, which are never designed to intentionally incorporate bias, run the risk of replicating or even amplifying bias present in real-world data [19,76,77]. This may cause unfair treatment in which some individuals or groups of people are *privileged* (i.e., receive a favourable treatment) and others are *unprivileged* (i.e., receive an unfavourable treatment). In this context, a fair treatment of individuals constitutes that decisions are made independent of sensitive attributes such as gender or race, such that individuals are treated based on merit [14, 15, 20]. For example, one can aim for an equal probability of population groups to receive a positive treatment, or an equal treatment of individuals that only differ in sensitive attributes.

Human bias has been transferred to various real-word systems relying on ML. There are many examples of this in the literature. For instance, bias has been found in advertisement and recruitment processes [74, 78], affecting university admissions [79] and human rights [20]. Not only is such a biased behaviour undesired, but it can fall under regulatory control and risk the violation of anti-discrimination laws [19, 22, 23], as sensitive attributes such as age, disability, gender identity, race are protected by US law in the Fair Housing Act and Equal Credit Opportunity Act [80].

Another example for a biased treatment of population groups can be found in the **COMPAS** (Correctional Offender Management Profiling for Alternative Sanctions) software, used by courts in US to determine the risks of an individual to reoffend. These scores are used to motivate decisions on whether and when defendants are to be set free, in different stages of the justice system. Problematically, this software falsely labelled non-white defendants with higher risk scores than white defendants [17].

To reduce the degree of bias that such systems exhibit, practitioners use three types of bias mitigation methods [32]:

- **Pre-processing:** bias mitigation in the training data, to prevent it from reaching ML models;

- **In-processing:** bias mitigation while training ML models;

- **Post-processing:** bias mitigation on trained ML models.

There has been a growing interest in fairness research, including definitions, measurements, and improvements of ML models [2, 73, 75, 81, 82]. In particular, a variety of recent work addresses the mitigation of bias in binary classification models: given a collection of observations (training data) are labelled with a binary label (testing data) [83].

Despite the large amount of existing bias mitigation methods and surveys on fairness research, as Pessach and Shmueli [75] pointed out, there remain open challenges that practitioners face when designing new bias mitigation methods: “It is not clear how newly proposed mechanisms should be evaluated, and in particular which measures should be considered? which datasets should be used? and which mechanisms should be used for comparison?” [75]

To combat this challenge, we set out to perform a comprehensive survey of existing research on bias mitigation for ML models. We analyse 341 publications to identify practices applied in fairness research when creating bias mitigation methods. In particular, we consider the datasets to which bias mitigation methods are applied, the metrics used to determine the degree of bias, and the approaches used for benchmarking the effectiveness of bias mitigation methods. By doing so, we allow practitioners to focus their effort on creating bias mitigation methods rather than requiring a lot of time to determine their experimental setup (e.g., which datasets to test on, which benchmark to consider).

To the best of our knowledge, this is the first survey to systematically and comprehensively cover bias mitigation methods and their evaluation. To summarize, the contribution of this survey are:

1. we provide a comprehensive overview of the research on bias mitigation methods for ML classifiers;

2. we introduce the experimental design details for evaluating existing bias mitigation methods;

3. we identify challenges and opportunities for future research on bias mitigation methods.

4. we make the collected paper repository public, to allow for future replication and manual investigation of our results [84].

The rest of this chapter is structured as follows. Section 3.1 presents an overview of related surveys. The search methodology is described in Section 3.2. Sections 3.3-3.6 describe research on bias mitigation methods. Challenges that the field of fairness research and bias mitigation methods face are discussed in Section 3.7. Section 3.8 concludes this survey.

3.1 Related Surveys

In this section, we provide an overview on existing surveys in the fairness literature and their contents. This allows us to identify the knowledge gap filled by our survey.

Mehrabi et al. [20] and Pessach and Shmueli [75] provided an overview of bias and discrimination types, fairness definitions and metrics, bias mitigation methods, and existing datasets. For example, Pessach and Shmueli [75, 85] listed the datasets and metrics used by 27 bias mitigation methods. A similar focus has been pursued by Dunkelau and Leuschel [81], who provided an extensive overview on fairness notions, available frameworks, and bias mitigation methods for classification problems. They moreover provided a classification of approaches for each type (i.e., pre-, in-, and post-processing). The most exhaustive categorization of bias mitigation methods, to date, has been conducted by Caton and Haas [86], who also presented fairness metrics and fairness platforms.

A detailed collection of prominent fairness definitions for classification problems is provided by Verma and Rubin [83]. Similarly, Žliobaite [87] surveyed measures for indirect discrimination for ML.

In addition to the surveys on fairness metrics, Le Quy et al. [88] provided a survey with 15 frequently used datasets in fairness research. For each dataset, they described the available features and their relationships with sensitive attributes.

Other surveys are concerned with fairness and consider the following perspectives: learning-based sequential decision algorithms [89], criminal justice [18], graph representations [90], ML testing [24], Software Engineering [5, 49], or Natural Language Processing [91, 92].

While previous surveys focus on ML classification, and some mention bias mitigation methods, none has yet systematically covered the evaluation bias mitigation methods (e.g., how are methods benchmarked, what dataset are used). The surveys related closest to our focus are provided by Dunkelau and Leuschel [81], and Pessach and Shmueli [75, 85].

Dunkelau and Leuschel [81] provided an overview of bias mitigation methods, with a focus on their implementation and underlying algorithms. However, further evaluation details of these methods, such as dataset and metric usage, were not addressed. While Pessach and Shmueli [75, 85] listed the datasets and metrics used by 27 bias mitigation methods, they do not provide actionable insights to support developers. In addition to combining aspects of both surveys (i.e., extensive collection of bias mitigation methods like Dunkelau and Leuschel [81], and information on datasets and metrics similar to Pessach and Shmueli [75]), we aim to analyze the findings of a comprehensive literature search to devise recommendations.

3.2 Survey Methodology

The purpose of this survey is to gather and categorize research work, that mitigates bias in ML models. Given that the existing literature focuses on classification for tabular data, this survey also focuses on bias mitigation methods for such classification tasks.

3.2.1 Search Methodology

This section outlines our search procedure. We start with a preliminary search, followed by a repository search and snowballing.

Preliminary Search. Prior to systematically searching online repositories, we conduct a preliminary search. The goal of the preliminary search is to gain a deeper understanding of the field and assess whether there is a sufficient number of publications to allow for subsequent analysis. In particular, we collect bias mitigation publications from four existing surveys (see Section 3.1):

- Mehrabi et al. [20] : 24 bias mitigation methods;
- Pessach and Shmueli [75, 85]: 30 bias mitigation methods;
- Dunkelau and Leuschel [81]: 40 bias mitigation methods;
- Caton and Haas [86]: 70 bias mitigation methods.

In total, we collect 100 unique bias mitigation methods from these four surveys.

Repository Search. After the preliminary search, we conduct a search of six established online repositories (IEEE, ACM, ScienceDirect, Scopus, arXiv, and Google Scholar).

The search procedure is guided by two groups of keywords:

- Domain: machine learning, deep learning, artificial intelligence;
- Bias Mitigation: fairness-aware, discrimination-aware, bias mitigation, debias*, unbias*;

In this context, *Domain* keywords ensure that the bias discussed in the publication affects machine learning systems. *Bias Mitigation* ensures that the publication addresses bias reduction via the use of bias mitigation methods. For the six repositories, we collected publications that contain at least one *Domain* and one *Bias mitigation* keyword (i.e., we check each possible combination of keywords for the two categories).

Selection To ensure that the publications included in this survey are relevant to the context of bias mitigation for ML models, we consider the following **inclusion criteria**: 1) describe human biases; 2) address classification problems; 3) use tabular data (e.g., do not make decisions based on images or text alone).

To ensure that irrelevant publications are excluded from the search results, we manually check publications in three filtration stages [93]:

1. **Title:** Publications with irrelevant titles to the survey are excluded;
2. **Abstract:** The abstract of every publication is checked. Publications that show to be irrelevant to the survey at this step are excluded (e.g. not about ML, do not apply debiasing);
3. **Body:** For publications that passed the previous two steps, we check the entire publication to determine whether they satisfy the inclusion criteria. If not, they are excluded.

Table 3.1: Publications found at each stage of the search procedure.

Stage	Publications
Preliminary search	100
Repository search Oct'21	75
Repository search Jul'22	56
Snowballing	78
Author feedback	32
Total	341

Snowballing After conducting the repository search, we apply backward snowballing (i.e., finding new publications that are cited by publications we already selected) for each publication retained after the "Body" stage [94]. This snowballing step is repeated for every new publication found. The goal of snowballing is to find missing related work with regards to the collected publications. This is in particular useful if undiscovered bias mitigation methods are used for benchmarking.

3.2.2 Selected Publications

In total, we gathered 341 publications over the different stages of our search procedure. Table 3.2 summarizes the results of two repository searches. The first search was conducted from the 7th of October to 10th of October 2021, and the second search was conducted on the 21st of July 2022. The purpose of the second search is to collect publications from the year 2022 (i.e., we filtered search results for the publication year 2022). In October 2021, Google Scholar provided 8,738 publications that were in line with the search keywords. We restricted our search to the first 1,000 entries as prioritized by Google Scholar based on relevance. Similarly, the second search yielded 1,995 results and we focused on the first 1,000 publications.

To ensure that our survey is comprehensive and accurate, we contacted the corresponding authors of the 309 publications collected via the preliminary search, the two repository searches and snowballing. We asked them to check whether our description about their work is correct. Based on their feedback, we included additional 31 publications. The amount of publications found for each step of the search is listed in Table 3.1.

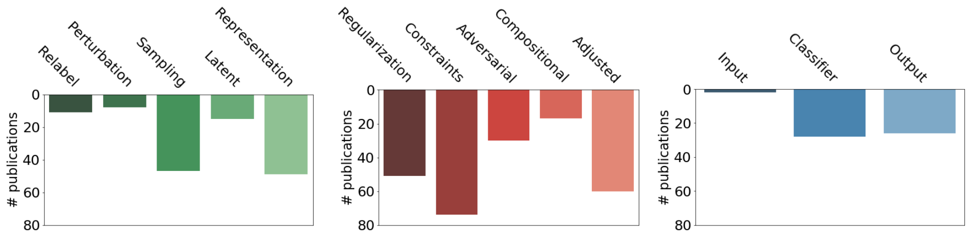
Table 3.2: Results of the repository search. For each of the six search repositories, we show the number of publications retained after each filtration stage, where the "Body" column shows the number of publications included in this survey.

Repository	Initial	Title	Abstract	Body
ACM	118	26	16	13
ScienceDirect	166	9	5	3
IEEE	401	18	9	9
arXiv	650	69	48	38
Scopus	1063	44	28	21
Google Scholar	8738	119	90	77

Search results October'21.

Repository	Initial	Title	Abstract	Body
ACM	468	17	14	8
ScienceDirect	88	6	3	2
IEEE	90	8	1	1
arXiv	465	42	23	17
Scopus	356	13	9	5
Google Scholar	1995	62	51	35

Search results July'22.



(a) Pre-processing. **(b)** In-processing. **(c)** Post-processing.

Figure 3.1: Categorization of bias mitigation methods. Categories are grouped based on their type (i.e., pre-processing, in-processing, post-processing) and the number of publications of each category is shown.

3.3 Algorithms

In this section, we present the bias mitigation methods found in our literature search. We distinguished bias mitigation methods based on their type (i.e., in which stage of the ML process are they applied): pre-processing (Section 3.3.1), in-processing (Section 3.3.2) and post-processing (Section 3.3.3) methods [32]. Moreover, we organize methods in categories (i.e., the bias mitigation approach). For this, we follow taxonomies devised by Dunkelau and Leuschel [81], as well as Caton and Haas [86]. Figure 3.1 illustrates the 13 categories we use.

A single publication may reside in multiple categories, for example if their approach applies pre-processing before adapting the training procedure during an in-processing stage. This is the case for 70 publications, for which we provide more information in Section 3.3.4.

Among the 341 publications, 123 used pre-processing (Section 3.3.1), 212 used in-processing (Section 3.3.2) and 56 used post-processing methods (Section 3.3.3).

3.3.1 Pre-processing Bias Mitigation Methods

In this section, we present bias mitigation methods that combat bias by applying changes to the training data. Table 3.3 and Table 3.4 list the 123 publications we found, according to the type of pre-processing method used.

3.3.1.1 Relabelling and Perturbation

This section presents bias mitigation methods that apply changes to the values of the training data. Changes have been applied to the ground truth labels (relabelling) or the remaining features (perturbation).

A popular approach for relabelling the dataset is “massaging”, proposed by Kamiran and Calders [95] in 2009. In the first stage, “massaging” uses a ranker to determine the best candidates for relabelling. In particular, instances close to the decision boundary are selected, to minimize the negative impact of relabelling on accuracy. Afterwards, an equal amount of instances with positive and negative labels are typically selected, according to their rank. For selected instances, their

Table 3.3: Publications on Pre-processing bias mitigation methods.

Category	Authors [Ref]	Year	Venue
Relabel	Kamiran and Calders [95]	2009	ICCCC
	Calders et al. [67]	2009	ICDMW
	Loung et al. [96]	2011	KDD
	Žliobaite et al. [97]	2011	ICDM
	Hajian et al. [98]	2012	IEEE Trans Knowl Data Eng
	Kamiran and Calders [35]	2012	KAIS
	Zhang et al. [99]	2018	IJCAI
	Iosifidis et al. [100]	2019	DEXA
	Sun et al. [101]	2022	EuroS&P
	Seker et al. [102]	2022	Stud. Health Technol. Inform
	Alabdulmohsin et al. [103]	2022	arXiv
Perturbation	Hajian et al. [98]	2012	IEEE Trans Knowl Data Eng
	Feldman et al. [34]	2015	KDD
	Lum and Johndrow [104]	2016	arXiv
	Wang et al. [105]	2018	NeurIPS
	Wang et al. [106]	2019	ICML
	Johndrow and Lum [107]	2019	Ann Appl Stat
	Li et al. [108]	2022	SSRN
	Li et al. [109]	2022	ICSE
Sampling	Calders et al. [67]	2009	ICDMW
	Kamiran and Calders [110]	2010	BNAIC
	Žliobaite et al. [97]	2011	ICDM
	Kamiran and Calders [35]	2012	KAIS
	Zhang et al. [111]	2017	IJCAI
	Krasanakits et al. [112]	2018	TheWebConf
	Xu et al. [113]	2018	Big Data
	Chen et al. [114]	2018	NeurIPS
	Iosifidis and Ntoutsi [115]	2018	report
	Salimi et al. [116]	2019	MOD
	Iosifidis et al. [100]	2019	DEXA
	Zelaya et al. [117]	2019	KDD
	Xu et al. [118]	2019	IJCAI
	Xu et al. [119]	2019	Big Data
	Iosifidis et al. [120]	2019	Big Data
	Abusitta et al. [121]	2019	arXiv
	Sharma et al. [122]	2020	AIES
	Chakraborty et al. [28]	2020	FSE
	Jiang and Nachum [123]	2020	AISTATS
	Hu et al. [124]	2020	DS
	Morano [125]	2020	Thesis
	Yan et al. [126]	2020	CIKM
	Celis et al. [127]	2020	ICML
	Abay et al. [128]	2020	arXiv
	Salazar et al. [129]	2021	IEEE Access
	Zhang et al. [130]	2021	PAKDD
	Chuang and Mroueh [131]	2021	ICLR
	Amend and Spurlock [132]	2021	JCSC
	Verma et al. [133]	2021	arXiv
	Cruz et al. [134]	2021	ICDM
	Chakraborty et al. [135]	2021	FSE
	Jang et al. [136]	2021	AAAI
	Du and Wu [137]	2021	CIKM
	Roh et al. [138]	2021	NeurIPS
	Iofinova et al. [139]	2021	arXiv
	Yu [140]	2021	arXiv
	Singh et al. [141]	2021	Mach. learn. knowl. Extr.
	Sun et al. [101]	2022	EuroS&P
	Pentyala et al. [142]	2022	arXiv
	Rajabi et al. [143]	2022	Mach. learn. knowl. Extr.
	Dablain et al. [144]	2022	arXiv
	Chen et al. [145]	2022	FSE
	Li et al. [146]	2022	PMLR
	Chakraborty et al. [147]	2022	FairWARE
	Wang et al. [148]	2022	ICML
	Almuzaini et al. [149]	2022	FAccT
	Chai and Wang [150]	2022	ICML

Table 3.4: Publications on Pre-processing bias mitigation methods - Part 2.

Category	Authors [Ref]	Year	Venue
Representation	Zemel et al. [68]	2013	ICML
	Edwards and Storkey [151]	2015	arXiv
	Louizos et al. [152]	2016	ICLR
	Xie et al. [153]	2017	NeurIPS
	Hacker and Wiedemann [154]	2017	arXiv
	McNamara et al. [155]	2017	arXiv
	Pérez-Suay et al. [156]	2017	ECML PKDD
	Calmon et al. [33]	2017	NeurIPS
	Komiyama and Shimao [157]	2017	arXiv
	Samadi et al. [158]	2018	NeurIPS
	Madras et al. [159]	2018	ICML
	du Pin Calmon et al. [160]	2018	IEEE J Sel
	Moyer et al. [161]	2018	NeurIPS
	Quadrianto et al. [162]	2018	arXiv
	Grgić-Hlača et al. [163]	2018	AAAI
	Song et al. [164]	2019	AISTATS
	Wang and Huang [165]	2019	arXiv
	Lahoti et al. [166]	2019	VLDB
	Feng et al. [167]	2019	arXiv
	Lahoti et al. [168]	2019	ICDE
	Creager et al. [169]	2019	ICML
	Gordaliza et al. [170]	2019	ICML
	Quadrianto et al. [171]	2019	CVPR
	Zhao et al. [172]	2020	ICLR
	Zehlike et al. [173]	2020	Data Min. Knowl. Discov
	Sarhan et al. [174]	2020	ECCV
	Tanu et al. [175]	2020	AISTATS
	Jaiswal et al. [176]	2020	AAAI
	Madhavan and Wadhwa [177]	2020	CIKM
	Ruoss et al. [178]	2020	NeurIPS
	Kim and Cho [179]	2020	AAAI
	Fong et al. [180]	2021	arXiv
	Salazar et al. [181]	2021	VLDB
	Gupta et al. [182]	2021	AAAI
	Grari et al. [183]	2021	ECML PKDD
	Zhu et al. [47]	2021	ICCV
	Oh et al. [184]	2022	arXiv
	Agarwal and Deshpande [185]	2022	FAccT
	Wu et al. [186]	2022	arXiv
	Shui et al. [187]	2022	arXiv
	Qi et al. [188]	2022	arXiv
	Balunović et al. [189]	2022	ICLR
	Kairouz et al. [190]	2022	IEEE Trans. Inf. Forensics Secur
	Liu et al. [191]	2022	Neural Process. Lett.
	Cerrato et al. [192]	2022	arXiv
	Kamani et al. [193]	2022	Mach. Learn.
	Rateike et al. [194]	2022	FAccT
	Galhotra et al. [195]	2022	SIGMOD
	Kim and Cho [196]	2022	Neurocomputing
Latent	Calders and Verwer [31]	2010	Data Min. Knowl. Discov
	Kilbertus et al. [197]	2017	NeurIPS
	Gupta et al. [198]	2018	arXiv
	Madras et al. [199]	2019	FAccT
	Oneto et al. [200]	2019	AIES
	Wei et al. [201]	2020	PMLR
	Kehrenberg et al. [202]	2020	Front. Artif. Intell.
	Grari et al. [203]	2021	arXiv
	Chen et al. [204]	2022	arXiv
	Liang et al. [205]	2022	arXiv
	Jung et al. [206]	2022	CVPR
	Diana et al. [207]	2022	FAccT
	Chakraborty et al. [147]	2022	FairWARE
	Wu et al. [208]	2022	CLeaR
	Suriyakumar et al. [209]	2022	arXiv

labels are switched.

Massaging has later been extended by Kamiran and Calders [35], and Calders et al. [67]. Moreover, Žliobaite et al. [97] created a related method called "local massaging". "Massaging" has also been applied by other work [99, 100].

Another relabelling approach was proposed by Loung et al. [96], who relabelled instances based on their k-nearest neighbours, such that similar individuals receive similar labels.

Feldman et al. [34] used perturbation to modify non-protected attributes, such that their values for privileged and unprivileged groups are comparable. In particular, the values are adjusted to bring their distributions closer together while preserving the respective ranks within a group (e.g., the highest values of attribute a for the privileged group remains highest after perturbation). Lum and Johndrow [104, 107] used conditional models for perturbation, which allowed for modification of multiple variables (continuous or discrete). Li et al. [108] proposed an iterative approach for perturbation. At each step, the most bias-prone attribute is selected and transformed, until the degree of bias exhibited by a classification model is below a specified threshold.

Other than perturbing the underlying data for all groups to move them closer [34, 104, 107], Wang et al. [105, 106] considered only the unprivileged group for perturbation seeking to resolve disparity by improving the performance of the unprivileged group. Hajian et al. [98] applied both relabeling and perturbation (i.e., changes to the sensitive attribute).

3.3.1.2 Sampling

Sampling methods change the training data by changing the distribution of samples (e.g., adding, removing samples) or adapting their impact on training. Similarly, the impact of training data instances can be achieved by reweighing their importance [35, 67, 100, 127, 128, 137, 140, 142, 146, 149, 150].

Reweighing was first introduced by Calders et al. [67]. Each instance receives a weight according to its label and protected attribute (e.g., instances in the unprivileged group and positive label receive a higher weight as this is less likely). In the training process of classification models, a higher instance weight causes higher losses when misclassified. Weighted instances are sampled with replacement according to their weights. If the classification model is able to process weighted instances, the dataset can be used for training without resampling [35].

Jiang and Nachum [123] and Krasanakits et al. [112] used reweighing to combat biased labels in the original training data.

Instead of assigning equal weights to data instances of the same population subgroup, Li et al. [146] assigned individual weights to instances of the training data.

Other sampling strategies include the removal of data points (downsampling) [28, 116, 130, 133, 134, 138, 139, 145, 148] or the addition of new data points (upsampling). Popular methods for upsamplig are oversampling for duplicating instances of the minority group [115, 117, 125, 132] and the use of SMOTE [210]. SMOTE does not duplicate instances but generates synthetic ones in the neighborhood of the minority group [115, 117, 125, 126, 129, 135, 141, 144, 147].

To sample datapoints, uniform [35] and preferential [35, 97, 110, 117, 124] strategies have been followed, where preferential sampling changes the distribution of instances close to the decision boundary.

Xu et al. [113, 118, 119] used a generative approach to generate discrimination-free data for training [121, 136, 143]. Zhang et al. [111] used causal networks to create a new dataset. The initial dataset is used to create a causal network, which is then modified to reduce discrimination. The debiased causal network is used to generate a new dataset.

Sharma et al. [122] created additional data for augmentation by duplicating existing datasets and swapping the protected attribute of each instance. The newly-created data is successively added to the existing dataset.

3.3.1.3 Latent variables

Latent variable describes the augmentation of the training data with additional features that are preferably unbiased. In previous work, latent variables have been used to represent labels [201, 202] and group memberships (i.e., protected or unprotected group) [147, 198, 200, 203, 204, 205, 206, 207, 209].

For instance, Calders and Verwer [31] clustered the instances to detect those that should receive a positive latent label and those that should receive a negative one. For this purpose, they used an expectation maximization algorithm.

Gupta et al. [198] tackled the problem of bias mitigation for situations where group labels are missing in the datasets. To combat this issue, they created a latent "proxy" variable for the group membership and incorporated constraints for achieving fairness for such proxy groups in the training procedure.

Frequently, latent variables are considered when dealing with causal graphs [197, 199, 203].

3.3.1.4 Representation

Representation learning aims at learning a transformation of training data such that bias is reduced while maintaining as much information as possible.

The first bias mitigation approach for learning fair representations was Learning Fair Representations (LFR), proposed by Zemel et al. [68]. LFR translates representation learning into an optimization problem with two objectives: 1) removing information about the protected attribute; 2) minimizing the information loss of non-sensitive attributes.

A popular used approach for generating fair representations is optimization [33, 154, 155, 160, 161, 164, 166, 168, 170, 173, 187]. Other used techniques are:

- adversarial learning [47, 151, 153, 159, 167, 172, 176, 178, 179, 183, 188, 190];
- variational autoencoders [152, 169, 184, 191, 194];
- adversarial variational autoencoder [186];
- normalizing flows [189, 192];
- dimensionality reduction [156, 158, 175, 193];
- residuals [157];

- contrastive learning [182];
- neural style transfer [162, 171].

Another method for improving the fairness of the data representation is the removal [163,165,177] or addition of features [180,181,195]. Grgić-Hlača et al. [163] investigated fairness while using different sets of features, thereby making training features choices. Madhavan and Wadhwa [177] removed discriminating features from the training data. Salazar et al. [181] applied feature creation techniques, which apply nonlinear transformation, and then drop biased features.

3.3.2 In-processing Bias Mitigation Methods

This section presents in-processing methods; methods that mitigate bias during the training procedure of the algorithm. Overall, we found a total of 212 publications (see Table 3.5, Table 3.6 and Table 3.7 for more details) that apply in-processing methods. For more details on in-processing methods, we refer to the survey by Wan et al. [384], which provides information on 38 in-processing approaches developed for various ML tasks.

3.3.2.1 Regularization and Constraints

Regularization and constraints are both approaches that apply changes to the learning algorithm's loss function. Regularization adds a term to the loss function. While the original loss function is based on accuracy metrics, the purpose of regularization term is to penalize discrimination (i.e., discrimination leads to a higher loss of the ML algorithm. Constraints on the other hand determine specific bias levels (according to loss functions) that cannot be breached during training.

To widen the range of fairness definitions that can be considered when applying constraints, Celis et al. [38] proposed a Meta-algorithm. This Meta-algorithm takes a fairness constraint as input.

When applied to Decision Trees, regularization can be used to modify the splitting criteria [42, 221, 224, 236, 237, 246, 249]. Traditionally, leaves are iteratively split to achieve an improvement in accuracy. To improve fairness while training,

Table 3.5: Publications on In-processing bias mitigation methods.

Category	Authors [Ref]	Year	Venue
Regularization	Kamiran et al. [42]	2010	ICDM
	Kamishima et al. [211]	2011	ICDMW
	Kamishima et al. [30]	2012	ECML PKDD
	Ristanoski et al. [212]	2013	CIKM
	Fish et al. [213]	2015	FATML
	Berk et al. [29]	2017	arXiv
	Pérez-Suay et al. [156]	2017	ECML PKDD
	Bechavod and Ligett [214]	2017	arXiv
	Quadrianto and Sharmanska [215]	2017	NeurIPS
	Raff et al. [216]	2018	AIES
	Goel et al. [217]	2018	AAAI
	Enni and Assent [218]	2018	ICDM
	Mary et al. [219]	2019	ICML
	Beutel et al. [220]	2019	AIES
	Zhang et al. [221]	2019	ICDMW
	Aghaei et a l. [222]	2019	AAAI
	Huang and Vishnoi [223]	2019	ICML
	Zhang and Ntoutsi [224]	2019	IJCAI
	Tavakol [225]	2020	SIGIR
	Baharlouei et al. [226]	2020	ICLR
	Di Stefano et al. [227]	2020	arXiv
	Kim et al. [228]	2020	ICML
	Jiang et al. [229]	2020	UAI
	Romano et al. [230]	2020	NeurIPS
	Ravichandran et al. [231]	2020	arXiv
	Liu et al. [232]	2020	Preprint
	Keya et al. [233]	2020	arXiv
	Hickey et al. [234]	2020	ECML PKDD
	Kamani [235]	2020	Thesis
	Abay et al. [128]	2020	arXiv
	Chuang and Mroueh [131]	2021	ICLR
	Zhang and Weiss [236]	2021	ICDM
	Ranzato et al. [237]	2021	CIKM
	Kang et al. [238]	2021	arXiv
	Grari et al. [239]	2021	IJCAI
	Wang et al. [240]	2021	SIGKDD
	Mishler and Kennedy [241]	2021	arXiv
	Lowy et al. [242]	2021	arXiv
	Zhao et al. [243]	2021	arXiv
	Yurochkin and Sun [244]	2021	ICLR
	Sun et al. [101]	2022	EuroS&P
	Zhao et al. [245]	2022	WSDM
	Wang et al. [246]	2022	CAV
	Deng et al. [247]	2022	arXiv
	Lee et al. [248]	2022	Entropy
	Zhang and Weiss [249]	2022	AAAI
	Jiang et al. [250]	2022	ICLR
	Lee et al. [251]	2022	ICASSP
	Do et al. [252]	2022	ICML
	Patil and Purcell [253]	2022	Future Internet
	Kim and Cho [196]	2022	Neurocomputing
Adversarial	Beutel et al. [254]	2017	arXiv
	Gillen et al. [255]	2018	NeurIPS
	Kearns et al. [37]	2018	ICML
	Wadsworth et al. [256]	2018	arXiv
	Agarwal et al. [257]	2018	ICML
	Raff and Sylvester [258]	2018	DSAA
	Zhang et al. [36]	2018	AIES
	Sadeghi et al. [259]	2019	ICCV
	Adel et al. [260]	2019	AAAI
	Zhao and Gordon [261]	2019	NeurIPS
	Celis and Keswani [262]	2019	nan
	Beutel et al. [220]	2019	AIES
	Grari et al. [263]	2019	ICDM
	Xu et al. [119]	2019	Big Data
	Yurochkin et al. [264]	2020	ICLR
	Garcia de Alford et al. [265]	2020	SMU DSR
	Roh et al. [266]	2020	ICML
	Delobelle et al. [267]	2020	ASE
	Rezaei et al. [268]	2020	AAAI
	Lahoti et al. [269]	2020	NeurIPS
	Amend and Spurlock [132]	2021	JCSC
	Rezaei et al. [270]	2021	AAAI
	Grari et al. [239]	2021	IJCAI
	Grari et al. [203]	2021	arXiv
	Liang et al. [205]	2022	arXiv
	Chen et al. [204]	2022	arXiv
	Tao et al. [271]	2022	FSE
	Petrović et al. [272]	2022	Neurocomputing
	Yang et al. [273]	2022	medRxiv
	Yazdani-Jahromi et al. [274]	2022	arXiv

Table 3.6: Publications on In-processing bias mitigation methods - Part 2.

Category	Authors [Ref]	Year	Venue
Constraints	Dwork et al. [45]	2012	ITCS
	Calders et al. [275]	2013	ICDM
	Fukuchi and Sakuma [276]	2015	arXiv
	Fukuchi et al. [277]	2015	IEICE Trans. Inf.& Syst.
	Goh et al. [278]	2016	NeurIPS
	Zafar et al. [279]	2017	AISTATS
	Russel et al. [280]	2017	NeurIPS
	Corbett-Davies et al. [43]	2017	KDD
	Quadrianto and Sharmanska [215]	2017	NeurIPS
	Zafar et al. [39]	2017	TheWebConf
	Komiyama and Shimao [157]	2017	arXiv
	Woodworth et al. [281]	2017	COLT
	Kilbertus et al. [197]	2017	NeurIPS
	Zafar et al. [282]	2017	NeurIPS
	Gillen et al. [255]	2018	NeurIPS
	Olfat and Aswani [283]	2018	AISTATS
	Narasimhan [284]	2018	AISTATS
	Kearns et al. [37]	2018	ICML
	Zhang and Bareinboim [285]	2018	AAAI
	Heidari et al. [286]	2018	NeurIPS
	Kim et al. [287]	2018	NeurIPS
	Gupta et al. [198]	2018	arXiv
	Agarwal et al. [257]	2018	ICML
	Farnadi et al. [288]	2018	AIES
	Goel et al. [217]	2018	AAAI
	Nabi and Shpitser [289]	2018	AAAI
	Wu et al. [290]	2018	arXiv
	Zhang and Bareinboim [291]	2018	NeurIPS
	Grgić-Hlača et al. [163]	2018	AAAI
	Komiyama et al. [292]	2018	ICML
	Donini et al. [293]	2018	NeurIPS
	Balashankar et al. [294]	2019	arXiv
	Zafar et al. [295]	2019	JMLR
	Lamy et al. [296]	2019	NeurIPS
	Cotter et al. [297]	2019	ALT
	Jung et al. [298]	2019	arXiv
	Oneto et al. [200]	2019	AIES
	Cotter et al. [299]	2019	J. Mach. Learn. Res.
	Wick et al. [300]	2019	NeurIPS
	Cotter et al. [301]	2019	ICML
	Nabi et al. [302]	2019	ICML
	Xu et al. [303]	2019	TheWebConf
	Celis et al. [38]	2019	FAccT
	Agarwal et al. [304]	2019	ICML
	Kilbertus et al. [305]	2020	AISTATS
	Lohaus et al. [306]	2020	ICML
	Ding et al. [307]	2020	AAAI
	Chzhen et al. [308]	2020	NeurIPS
	Wang et al. [309]	2020	NeurIPS
	Cho et al. [310]	2020	NeurIPS
	Oneto et al. [311]	2020	IJCNN
	Maity et al. [312]	2020	arXiv
	Chzhen and Schreuder [313]	2020	arxiv
	Manisha and Gujar [314]	2020	IJCAI
	Scutari et al. [315]	2021	arXiv
	Celis et al. [316]	2021	NeurIPS
	Celis et al. [317]	2021	PMLR
	Petrović et al. [318]	2021	Eng. Appl. Artif. Intell.
	Padh et al. [319]	2021	Uncertainty artif. intell.
	Zhao et al. [320]	2021	KDD
	Zhang et al. [321]	2021	MOD
	Li et al. [322]	2021	LAK
	Du and Wu [137]	2021	CIKM
	Perrone et al. [323]	2021	AIES
	Słowik and Bottou [324]	2021	arXiv
	Mishler and Kennedy [241]	2021	arXiv
	Lawless et al. [325]	2021	arXiv
	Choi et al. [326]	2021	AAAI
	Park et al. [327]	2022	WWW
	Wang et al. [246]	2022	CAV
	Zhao et al. [328]	2022	KDD
	Boulitsakis-Logothetis [329]	2022	arXiv
	Hu et al. [330]	2022	arXiv
	Wu et al. [208]	2022	CLeaR

Table 3.7: Publications on In-processing bias mitigation methods - Part 3.

Category	Authors [Ref]	Year	Venue
Adjusted	Luo et al. [331]	2015	DaWaK
	Joseph et al. [332]	2016	NeurIPS
	Johnson et al. [333]	2016	Stat Sci
	Kusner et al. [334]	2017	NeurIPS
	Joseph et al. [335]	2018	AIES
	Hashimoto et al. [336]	2018	ICML
	Hébert-Johnson et al. [337]	2018	ICML
	Chiappa and Isaac [338]	2018	IFIP
	Alabi et al. [339]	2018	COLT
	Madras et al. [340]	2018	NeurIPS
	Kamishima et al. [341]	2018	Data Min Knowl Discov
	Kilbertus et al. [342]	2018	ICML
	Dimitrakakis et al. [343]	2019	AAAI
	Chakraborty et al. [344]	2019	arXiv
	Noriega-Campero et al. [345]	2019	AIES
	Chiappa [346]	2019	AAAI
	Madras et al. [199]	2019	FAccT
	Iosifidis and Ntoutsi [347]	2019	CIKM
	Kilbertus et al. [305]	2020	AISTATS
	Zhang and Ramesh [348]	2020	arXiv
	Chakraborty et al. [28]	2020	FSE
	Mandal et al. [349]	2020	NeurIPS
	Hu et al. [124]	2020	DS
	Liu et al. [232]	2020	Preprint
	da Cruz [350]	2020	Thesis
	Iosifidis and Ntoutsi [351]	2020	DS
	Kamani [235]	2020	Thesis
	Martinez et al. [352]	2020	ICML
	Ignatiev et al. [353]	2020	CP
	Ezzeldin et al. [354]	2021	arXiv
	Zhang et al. [130]	2021	PAKDD
	Wang et al. [355]	2021	FAccT
	Ozdayi et al. [356]	2021	arXiv
	Islam et al. [357]	2021	AIES
	Sharma et al. [358]	2021	AIES
	Cruz et al. [134]	2021	ICDM
	Lee et al. [359]	2021	ICML
	Hort and Sarro [13]	2021	ASE
	Perrone et al. [323]	2021	AIES
	Roh et al. [360]	2021	ICLR
	Valdivia et al. [361]	2021	Int. J. Intell. Syst.
	Wang et al. [362]	2022	arXiv
	Roy and Ntoutsi [363]	2022	ECML PKDD
	Sikdar et al. [364]	2022	FAccT
	Agarwal and Deshpande [185]	2022	FAccT
	Park et al. [327]	2022	WWW
	Djebrouni [365]	2022	Eurosys
	Short and Mohler [366]	2022	Int. J. Forecast.
	Maheshwari and Perrot [367]	2022	arXiv
	Zhao et al. [328]	2022	KDD
	Tizpaz-Niari et al. [368]	2022	ICSE
	Roy et al. [369]	2022	DS
	Mohammadi et al. [370]	2022	arXiv
	Gao et al. [371]	2022	ICSE
	Huang et al. [372]	2022	Expert Syst. Appl.
	Candelieri et al. [373]	2022	arXiv
	Anahideh et al. [374]	2022	Expert Syst. Appl.
	Rateike et al. [194]	2022	FAccT
	Li et al. [375]	2022	arXiv
	Iosifidis et al. [376]	2022	KAIS
Compositional	Calders and Verwer [31]	2010	Data Min. Knowl. Discov
	Pleiss et al. [40]	2017	NeurIPS
	Dwork et al. [377]	2018	FAccT
	Ustun et al. [378]	2019	ICML
	Oneto et al. [200]	2019	AIES
	Iosifidis et al. [120]	2019	Big Data
	Monteiro and Reynoso-Meza [379]	2021	PLM
	Ranzato et al. [237]	2021	CIKM
	Mishler and Kennedy [241]	2021	arXiv
	Kobayashi and Nakao [380]	2021	DiTTEt
	Jin et al. [381]	2022	ICML
	Chen et al. [145]	2022	FSE
	Roy et al. [369]	2022	DS
	Liu and Vicente [382]	2022	CMS
	Blanzeisky and Cunningham [383]	2022	Knowl Eng Rev
	Boulitsakis-Logothetis [329]	2022	arXiv
	Suriyakumar et al. [209]	2022	arXiv

Kamiran et al. [42] considered fairness in addition to accuracy when leaf splitting. They applied three splitting strategies:

1. only allow non-discriminatory splits;

2. choose best split according to $\delta_{accuracy}/\delta_{discrimination}$;

3. choose best split according to $\delta_{accuracy} + \delta_{discrimination}$.

While constraints and regularization usually utilize group fairness definitions, they have also been applied for achieving individual fairness [45, 255, 287, 298]. Moreover, they can be applied to achieve fairness for multiple sensitive attributes and fairness definitions [37,225,238,292,319], or extend existing adjustments, such as adding fairness regularization in addition to the L2 norm, which is used to avoid overfitting [30, 211].

3.3.2.2 Adversarial Learning

Adversarial learning simultaneously trains classification models and their adversaries [385]. While the classification model is trained to predict ground truth values, the adversary is trained to exploit fairness issues. Both models then perform against each other, to improve their performance.

Zhang et al. [36] trained a Logistic Regression model to predict the label Y while preventing an adversary from predicting the protected attribute under consideration of three fairness metrics: Demographic Parity, Equality of Odds, and Equality of Opportunity. Both, predictor and adversary, are implemented as Logistic regression models.

Similarly, Beutel et al. [254] trained a neural network to predict two outputs: labels and sensitive attributes. While a high overall accuracy is desired, the adversarial setting optimizes a low ability to predict sensitive information. The network is designed to share layers between the two output, such that only one model is trained [220, 258, 259, 260, 267].

Lahoti et al. [269] proposed Adversarially Reweighted Learning (ARL) in which a learner is trained to optimize performance on a classification task while the adversary adjusts the weights of computationally-identifiable regions in the input space with high training loss. By so-doing, the learner can then improve performance in these regions.

Other than using adversaries to prevent the ability to predict sensitive attributes (e.g., for reducing bias according to population groups), it has also been used to improve robustness to data poisoning [266], to improve individual fairness [264], and to reweigh training data [272]. In particular, Petrović et al. [272] used adversarial training to learn a reweighing function for training data instances as an in-processing procedure (contrary to applying reweighing as pre-processing, see Section 3.3.1.2).

3.3.2.3 Compositional

Compositional approaches combat bias by training multiple classification models. Predictions can then be made by a specific classification model for each population group (e.g., privileged and unprivileged) [31, 40, 200, 209, 329, 378, 381] or in an ensemble fashion (i.e., a voting of multiple classification models at the same time) [120, 145, 237, 241, 369, 380, 382, 386].

While decoupled classification models for privileged and unprivileged groups can achieve improved accuracy for each group, the amount training data for each classifier is reduced. To reduce the impact of small training data sizes Dwork et al. [377] utilized transfer training. With their transfer learning approach, they trained classifiers on data for the respective group and data from the other groups with reduced weight. Ustun et al. [378] built upon the work of Dwork et al. [377] and incorporates "preference guarantees", which states that each group prefers their decoupled classifier over a classifier trained on all training data and any classifier of the other groups. Similarly, Suriyakumar et al. [209] followed the concept of "fair use", which states that if a classification uses sensitive group information, it should improve performance for every group.

Training multiple classification models with different fairness goals allows for the creation of a pareto-front of solutions [241, 361, 369, 382, 383]. Practitioners can then choose which fairness-accuracy trade-off best suits their need. For example, Liu and Vicente [382] treated bias mitigation as multi-objective optimization problem that explores fairness-accuracy trade-offs under consideration of multiple fairness metrics. Mishler and Kennedy [241] proposed an ensemble method that builds classification models based on a weighted combination of metrics chosen by users.

3.3.2.4 Adjusted Learning

Adjusted learning methods mitigate the bias via changing the learning procedure of algorithms or the creation of novel algorithms [81].

Changes have been suggested for a variety of classification models, including Bayesian models [343, 387], Markov Random Fields [348], Neural Networks [124, 258, 352], Decision Trees, bandits [332, 335, 388], boosting [337, 347, 351, 369], Logistic Regression [360]. We outline a selection of publications in the following, to provide insight on techniques applied to different classification models.

Noriega-Campero et al. [345] proposed an active learning framework for training Decision Trees. During the training, a decision maker is able to collect more information about individuals to achieve fairness in predictions. In this context, not all information about individuals is available. There is an information budget that determines how many enquiries can be performed. Similarly, Anahideh et al. [374] used an active learning framework to balance accuracy and fairness by selecting instances to be labelled.

Madras et al. [340] proposed a rejection learning approach for joint decision-making with classification models and external decision makers. In particular, the classification model learns when to defer from making prediction (i.e., when it is more useful to have predictions from external decision makers). If the coverage of classification can be reduced (i.e., the classification model abstains from making some of the predictions), selective classification approaches can be used [359].

Martinez et al. [352] proposed the algorithm Approximate Projection onto Star Sets (APStar) to train Deep Neural Networks to minimize the maximum risk among all population groups. This procedure ensures that the final classifier is part of the Pareto Front [389]. Hu et al. [124] incorporated representation learning into the training procedure of Neural Networks to learn them jointly the classifier.

Hébert-Johnson et al. [337] proposed *Multicalibration*, a learning procedure similar to boosting. A classifier is trained iteratively. At each iteration, the predictions of the most biased subgroup are corrected until the classifier is adequately calibrated.

Hashimoto et al. [336] found fairness issues with the use of empirical risk minimization and proposed the use of distributionally robust optimization (DRO) when training classifiers such as Logistic Regression. During training, DRO optimizes the worst-case risk over all groups present.

Kilbertus et al. [342] adjusted the training procedure for Logistic Regression to take privacy into account. Sensitive user information is encrypted such that it cannot be used for classification tasks while retaining the ability to verify fairness issues. By doing so, users can provide sensitive information without the fear that someone can read them.

The learning procedure of existing classification models has also been adjusted by tuning their hyper-parameters [13,28,134,323,344,350,357,361,368].

3.3.3 Post-processing Bias Mitigation Methods

Post-processing bias mitigation methods are applied once a classification model has been successfully trained. With 56 publications that apply post-processing methods (Table 3.8), post-processing methods are the least frequently applied of those covered in this survey.

3.3.3.1 Input Correction

Input correction approaches apply a modification step to the testing data. This is comparable to pre-processing approaches (Section 3.3.1) [81], which conduct modifications to training data (e.g., relabelling, perturbation and representation learning).

We found only two publications that apply input corrections to testing data, both of which use perturbations. While Adler et al. [390] used perturbation in a post-processing stage, Li et al. [109] first performed perturbation in a pre-processing stage and then applied an identical procedure for post-processing.

Table 3.8: Publications on Post-processing bias mitigation methods.

Category	Authors [Ref]	Year	Venue
Input	Adler et al. [390]	2018	KAIS
	Li et al. [109]	2022	ICSE
Classifier	Calders and Verwer [31]	2010	Data Min. Knowl. Discov
	Kamiran et al. [42]	2010	ICDM
	Hardt et al. [41]	2016	NeurIPS
	Woodworth et al. [281]	2017	COLT
	Pleiss et al. [40]	2017	NeurIPS
	Gupta et al. [198]	2018	arXiv
	Morina et al. [391]	2019	arXiv
	Noriega-Campero et al. [345]	2019	AIES
	Kim et al. [392]	2019	AIES
	Kanamori and Arimura [393]	2019	JSAI
	Kim et al. [228]	2020	ICML
	Jiang et al. [229]	2020	UAI
	Savani et al. [394]	2020	NeurIPS
	Chzhen et al. [395]	2020	NeurIPS
	Chzhen et al. [308]	2020	NeurIPS
	Awasthi et al. [396]	2020	PMLR
	Chzhen and Schreuder [313]	2020	arxiv
	Schreuder and Chzhen [397]	2021	UAI
	Kanamori and Arimura [398]	2021	JSAI
	Mishler et al. [399]	2021	FAccT
	Mishler and Kennedy [241]	2021	arXiv
	Du et al. [400]	2021	NeurIPS
	Grabowicz et al. [401]	2022	FAccT
	Zhang et al. [402]	2022	FairWARE
	Mehrabi et al. [403]	2022	TrustNLP
	Wu and He [404]	2022	FAccT
	Marcinkevics et al. [405]	2022	MLHC
	Iosifidis et al. [376]	2022	KAIS
Output	Pedreschi et al. [406]	2009	SDM
	Kamiran et al. [14]	2012	ICDM
	Fish et al. [213]	2015	FATML
	Fish et al. [407]	2016	SDM
	Kim et al. [287]	2018	NeurIPS
	Zhang et al. [99]	2018	IJCAI
	Menon and Williamson [408]	2018	FAccT
	Liu et al. [409]	2018	arXiv
	Kamiran et al. [15]	2018	J. Inf. Sci.
	Chiappa [346]	2019	AAAI
	Chzhen et al. [410]	2019	NeurIPS
	Iosifidis et al. [120]	2019	Big Data
	Lohia et al. [46]	2019	ICASSP
	Wei et al. [201]	2020	PMLR
	Alabdulmohsin [411]	2020	arXiv
	Alabdulmohsin and Lucic [412]	2021	NeurIPS
	Nguyen et al. [413]	2021	J. Inf. Sci.
	Kobayashi and Nakao [380]	2021	DiTTEt
	Lohia [414]	2021	arXiv
	Jang et al. [415]	2022	AAAI
	Pentyala et al. [142]	2022	arXiv
	Snel and van Otterloo [416]	2022	Com. Soc. Res. J.
	Alghamdi et al. [417]	2022	arXiv
	Mohammadi et al. [370]	2022	arXiv
	Zeng et al. [418]	2022	arXiv
	Zeng et al. [419]	2022	arXiv

3.3.3.2 Classifier Correction

Post-processing approaches can also directly be applied to classification models, which Savani et al. [394] called intra-processing. A successfully trained classification model is adapted to obtain a fairer one. Such modification have been applied to Naive Bayes [31], Logistic Regression [229], Decision Trees [42,398,402], Neural Networks [394,400,403,405] and Regression Models [308].

Hardt et al. [41] proposed the modification of classifiers to achieve fairness with respect to Equalized Odds and Equality of Opportunity. Given an unfair classifier $\widehat{Y}$, the classifier $\widetilde{Y}$ is derived by solving an optimization problem under consideration of fairness loss terms. This approach has been adapted and extended by further publications [198,391,396,399].

Woodworth et al. [281] showed that this kind of modification can lead to a poor accuracy, for example when the loss function is not strictly convex. In addition to constraints during training, they proposed an adaptation of the approach by Hardt et al. [41].

Pleiss et al. [40] split a classifier in two (h_0, h_1, for the privileged and unprivileged group). To balance the false positive and false negative rate of the two classifiers, h_1 is adjusted such that with a probability of α the class mean is returned rather than the actual predication.Noriega-Campero et al. [345] followed the calibration approach of Pleiss et al. [40].

Kamiran et al. [42] modified Decision Tree classifiers by relabeling leaf nodes. The goal of relabeling was to reduce bias while sacrificing as little accuracy as possible. A greedy procedure was followed which iteratively selects the best leaf to relabel (i.e., highest ratio of fairness improvement per accuracy loss). Kanamori and Arimura [398] formulated the modification of branching thresholds for Decision Trees as a mixed integer program.

Kim et al. [392] proposed *Multiaccuracy Boost*, a post-processing approach similar to boosting for training classifiers. Given a black-box classifier and a learning algorithm, *Multiaccuracy Boost* iteratively adapts the current classifier based on its predictive performance.

3.3.3.3 Output Correction

The latest stage of applying bias mitigation methods is the correction of the output. In particular, the predicted labels are modified.

Pedreschi et al. [406] considered the correction of rule based classifiers, such as CPAR [420]. For each individual, the *k* rules with highest confidence are selected to determine the probability for each output label. Given that some of the rules can be discriminatory, their confidence level is adjusted to reduce biased labels.

Menon and Williamson [408] proposed a plugin approach for thresholding predictions. To determine the thresholds to use, the class probabilities are estimated using logistic regression.

Kamiran et al. [14,15] introduced the notion of reject option which modifies the prediction of individuals close to the decision boundary. In particular, individuals belonging to the unprivileged group receive a positive outcome and privileged individuals an unfavourable outcome. Similarly, Lohia et al. [46] relabeled individuals that are likely to receive biased outcomes, but rather than considering the decision boundary, they used an “individual bias detector” to find predictions that are likely suffer from individual discrimination. This work was extended in 2021, where individuals were ranked based on their “Unfairness Quotient” (i.e., the difference between regular prediction and with perturbed protected attribute). Fish et al. [407] proposed a confidence-based approach which returns a positive label for each prediction above a given threshold. This has also been applied to AdaBoost [213]. Other than using a general threshold for all instances, group dependent thresholds can be used [120,142,380,410,411,415,418,419].

Chiappa [346] addressed the fairness of causal models under consideration of a counterfactual world in which individuals belong to a different population group. The impact of the protected attribute on the prediction outcome is corrected to ensure that it coincides with counterfactual predictions. This way, sensitive information is removed while other information remains unchanged.

3.3.4 Combined Approaches

While most publications propose the use of a single type of bias mitigation method, we found 70 that applied multiple techniques at the same time (e.g., two pre-processing methods, one in-processing and one post-processing methods). Table 3.9 summarizes these approaches.

Among these 70 publications, 86% (60 out of 70) applied in-processing, 54% (38 out of 70) applied pre-processing, and 31% (22 out of 70) applied post-processing methods.

Additionally, 26 out of 70 publications applied multiple types of bias mitigation methods but at the same stage of the development process (e.g., two pre-processing approaches). In particular, the are 7 publications which applied multiple pre-processing methods. Among these 7 publications, 5 applied sampling and relabeling [35, 67, 97, 100, 101]. The remaining 19 out of 26 publications applied multiple in-processing methods, 17 of which include regularization or constraints.

47 publications applied at least two methods at different stages of the development process for ML models (e.g. one pre-processing and one in-processing method). This illustrates that bias mitigation methods can be used in conjunction [421]. Moreover, there are three publications that addressed bias mitigation at each stage: pre-processing, in-processing and post-processing [31, 120, 198].

Calders and Verwer [31] proposed three approaches for achieving discrimination-free classification of naive bayes models. At first, a latent variable is added to represent unbiased labels. The data is then used to train a model for each possible sensitive attribute value. Lastly, the probabilities output by the model are modified to account for unfavourable treatment (i.e., increasing the probability of positive outcomes for the unprivileged group and reducing it for the privileged group).

Gupta et al. [198] tackled the problem of bias mitigation for situation where group labels are missing in the datasets. To combat this issue, they created a latent "proxy" variable for the group membership and incorporated constraints for achieving fairness for such proxy groups in the training procedure. Lastly, they followed the approach of Hardt et al. [41] to debias and existing classifier by adding an additional variable to the prediction problem (see Section 3.3.3.2).

Iosifidis et al. [120] followed an ensemble approach of multiple AdaBoost classifiers. In particular, each classifier is trained on an equal amount of instances from each population group and label by sampling. Predictions are then modified by applying group-dependent thresholds.

3.3.5 Classification Models

Here we outline the classification models on which the three types of bias mitigation methods (pre-, in-, post-processing) have been applied on. Table 3.10 shows the frequency with which each type of classification model has been applied.

Currently, the most frequently used classification model is Logistic Regression, for each method type (pre-, in-, post-processing), with a total of 140 unique publications using it for their experiments. The next most frequently used classification models are Neural Networks (NN). A total of 102 publication used NNs for their experiments, with the majority being in-processing methods. Linear Regression models have been used in 22 publications.

Decision Trees (36 publications) and Random Forests (45 publications) are also frequently used. Moreover, different Decision Tree variants have been used, such as Hoeffding trees, C4.5, J48 and Bayesian random forests.

While the range of classification models is diverse, some of them are similar to one another:

- Boosting: AdaBoost, XGBoost, SMOTEBoost, Boosting, LightGBM, OS-Boost, Gradient Tree Boosting, CatBoost;
- Rule-based: RIPPER, PART, CBA, Decision Set, Rule Sets, Decision Rules.

Figure 3.2 illustrates the number of different classification models considered during experiments. It is clear to see that the majority of publications (70%) applied their bias mitigation method to only one classification model. While in-processing methods are model specific and directly modify the training procedure, pre-processing and most post-processing bias mitigation methods can be developed independently from the classification models they are used for. Therefore, they can

Table 3.9: Publications with multiple bias mitigation methods. "X" indicates that the publication applies a bias mitigation approach of the corresponding category (i.e., pre-, in-, or post-processing).

	Processing Method		
Authors [Ref]	Pre	In	Post
Sun et al. [101]	x x	x	
Calders et al. [67]	x x		
Žliobaite et al. [97]	x x		
Hajian et al. [98]	x x		
Kamiran and Calders [35]	x x		
Iosifidis et al. [100]	x x		
Chakraborty et al. [147]	x x		
Oneto et al. [200]	x	x x	
Calders and Verwer [31]	x	x	x
Gupta et al. [198]	x	x	x
Iosifidis et al. [120]	x	x	x
Pérez-Suay et al. [156]	x	x	
Komiyama and Shimao [157]	x	x	
Kilbertus et al. [197]	x	x	
Grgić-Hlača et al. [163]	x	x	
Madras et al. [199]	x	x	
Xu et al. [119]	x	x	
Abay et al. [128]	x	x	
Hu et al. [124]	x	x	
Chakraborty et al. [28]	x	x	
Chuang and Mroueh [131]	x	x	
Zhang et al. [130]	x	x	
Grari et al. [203]	x	x	
Du and Wu [137]	x	x	
Amend and Spurlock [132]	x	x	
Cruz et al. [134]	x	x	
Chen et al. [204]	x	x	
Liang et al. [205]	x	x	
Agarwal and Deshpande [185]	x	x	
Chen et al. [145]	x	x	
Wu et al. [208]	x	x	
Rateike et al. [194]	x	x	
Kim and Cho [196]	x	x	
Suriyakumar et al. [209]	x	x	
Zhang et al. [99]	x		x
Wei et al. [201]	x		x
Pentyala et al. [142]	x		x
Li et al. [109]	x		x
Mishler and Kennedy [241]		x x x	x
Quadrianto and Sharmanska [215]		x x	
Agarwal et al. [257]		x x	
Gillen et al. [255]		x x	
Kearns et al. [37]		x x	
Goel et al. [217]		x x	
Beutel et al. [220]		x x	
Kilbertus et al. [305]		x x	
Liu et al. [232]		x x	
Kamani [235]		x x	
Perrone et al. [323]		x x	
Grari et al. [239]		x x	
Ranzato et al. [237]		x x	
Park et al. [327]		x x	
Wang et al. [246]		x x	
Zhao et al. [328]		x x	
Roy et al. [369]		x x	
Boulitsakis-Logothetis [329]		x x	
Kamiran et al. [42]		x	x
Fish et al. [213]		x	x
Woodworth et al. [281]		x	x
Pleiss et al. [40]		x	x
Kim et al. [287]		x	x
Chiappa [346]		x	x
Noriega-Campero et al. [345]		x	x
Chzhen and Schreuder [313]		x	x
Kim et al. [228]		x	x
Jiang et al. [229]		x	x
Chzhen et al. [308]		x	x
Kobayashi and Nakao [380]		x	x
Iosifidis et al. [376]		x	x
Mohammadi et al. [370]		x	x

Table 3.10: Frequency of classification model usage for evaluating bias mitigation methods. Amounts are provided for each category and as a unique measure to avoid counting publications with multiple approaches double.

		Processing Method		
Model	Unique	Pre	In	Post
Logistic Regression	140	58	80	19
NN	102	34	65	17
Random Forest	45	20	22	14
SVM	37	15	18	9
Decision Tree	36	14	16	9
Naive Bayes	24	12	11	5
Linear Regression	22	4	20	3
AdaBoost	8	1	5	4
XGBoost	8	1	6	1
Nearest Neighbour	7	3	2	3
Causal	7	2	6	1
Nearest Neighbor	6	4	0	2
LightGBM	4	2	3	0
Bandit	3	0	3	0
Boosting	3	0	2	2
J48	2	1	1	0
Bayesian	2	0	1	1
Hoeffding Tree	2	1	1	0
Gaussian Process	2	2	0	0
CPAR	1	0	0	1
RIPPER	1	1	0	0
PART	1	1	0	0
C4.5	1	1	0	0
CBA	1	0	1	0
Lattice	1	1	1	1
Lasso	1	0	1	0
PSL	1	0	1	0
BART	1	0	1	0
RTL	1	0	1	0
Tree Ensemble	1	0	1	0
AUE	1	1	0	0
CART	1	0	1	0
SMOTEBoost	1	0	1	0
Gradient boosted trees	1	1	0	1
Cox model	1	0	1	0
Decision Rules	1	0	1	0
Gradient Tree Boosting	1	0	1	0
Kmeans	1	0	1	0
OSBoost	1	0	1	0
POEM	1	0	1	0
Markov random filed	1	0	1	0
SMSGDA	1	0	1	0
Probabilistic circuits	1	0	1	0
Rule Sets	1	0	1	0
Ridge Regression	1	0	1	1
Extreme Random Forest	1	1	0	0
Factorization Machine	1	1	0	0
Discriminant analysis	1	0	1	0
Generalized Linear Model	1	0	1	0

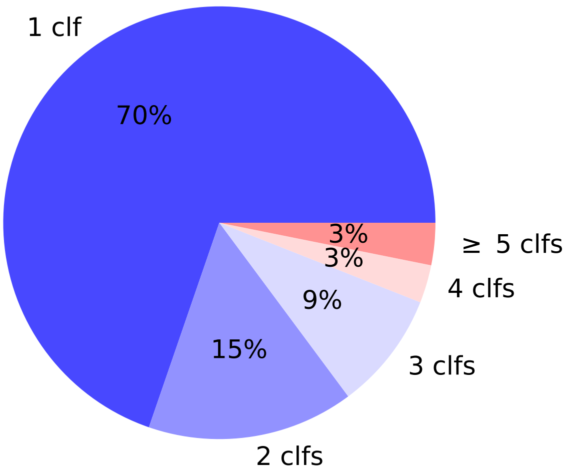


Figure 3.2: Number of classification models (clf) used for evaluation.

be devised once and applied to multiple classification models for evaluating their performance. Our observations confirm this intuition: only 24% of publications with in-processing methods consider more than one classification model, while 35% and 43% of pre- and post-processing methods consider more than one respectively.

3.4 Datasets

In this section, we investigate the use of datasets for evaluating bias mitigation methods. Among these datasets, some have been divided into multiple subsets (e.g., risk of recidivism or violent recidivism, medical data for different time periods). For clarity, we treat data from the same source as a single dataset.

Following this procedure, we gathered a total of 81 unique datasets. We discuss these datasets in Section 3.4.1 (e.g., what is the most frequently used dataset?) and Section 3.4.2 (e.g., how many datasets do experiments consider?). Additionally, 56 publications created synthetic or semi-synthetic datasets for their experiments. Section 3.4.3 provides information on the creation of such synthetic data.

For further details on datasets, we refer to Le Quy et al. [88] who surveyed 15 datasets and provided detailed information on the features and dataset characteristics. Additionally, Kuhlman et al. [80] gathered 22 datasets from publications published in the ACM Fairness, Accountability, and Transparency (FAT) Conference and 2019 AAAI/ACM conference on Articial Intelligence, Ethics and Society (AIES).

3.4.1 Dataset Usage

In this section, we investigate the frequency with which each dataset set has been used. The purpose of this analysis is to highlight the importance of each dataset and recommend the most important datasets to use for evaluating bias mitigation methods.

Among the 81 datasets, two are concerned with synthetic data (i.e., "synthetic" and "semi-synthetic") which we address in Section 3.4.3. Therefore, we are left with 81 datasets. 59% of the datasets (48 out of 81) are used by only one publication during their experiments. Another 14% of the datasets (11 out of 81) are only used twice. Thereby, 73% of the datasets (59 out of 81) are used rarely (by one or two publications).

Table 3.11 list the frequency of the remaining 22 datasets (used in three or more publications). In addition to the frequency, a percentage is provided (i.e., how many of the 324 publications use this datasets). Among all datasets, the Adult dataset is used most frequently (by 77% of the publications). While the Adult dataset contains information from the 1994 US census, Ding et al. [422] derived new datasets from the US census from 2014 to 2018.

Five other datasets are used by 10% or more of the publications (COMPAS, German Communities and Crime, Bank, Law School). This shows that in order to enable a simple comparison with existing work, one should consider at least the Adult and COMPAS dataset. A list of all datasets can be found in our online repository [84].

3.4.2 Dataset Count

In addition to detecting the most popular datasets for evaluating bias mitigation methods, we investigate the number of different datasets used, as this impacts the diversity of the performance evaluation [80].

Figure 3.3 visualizes the number of datasets used for each of the 324 publica-

Table 3.11: Frequency of widely used datasets (i.e., used in at least three publications).

Dataset Name	Frequency	Percentage
Adult [69]	249	77%
COMPAS [17]	166	51%
German [69]	97	30%
Communities and Crime [423]	42	13%
Bank [70]	38	12%
Law School [424]	33	10%
Default [425]	24	7%
Dutch Census [426]	16	5%
Health [427]	14	4%
MEPS [71]	14	4%
Drug [428]	9	3%
Student [429]	8	2%
Heart disease [69]	7	2%
National Longitudinal Survey of Youth [430]	6	2%
SQF [431]	5	2%
Arrhythmia [69]	5	2%
Wine [432]	4	1%
Ricci [433]	4	1%
University Anonymous (UNIV)	3	1%
Home credit [434]	3	1%
ACS [422]	3	1%
MIMICIII [435]	3	1%

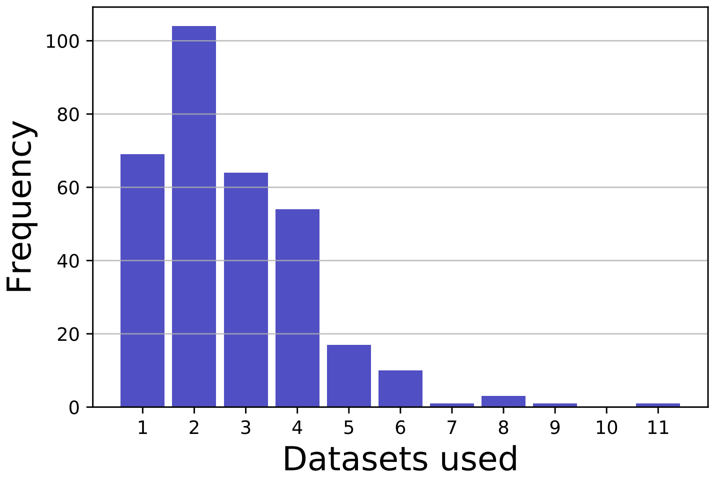


Figure 3.3: Number of datasets used per publication.

tions.

The most commonly used number of datasets considered for experiments is two, which has been observed in 104 out 324 of the publications. Over all, it can be seen that the number of considered datasets is relatively small (90% of the publications use four or fewer datasets), with an average of 2.7 datasets per publication. Two publications stand out in particular, with 9 datasets (Chakraborty et al. [135]), and 11 datasets (Do et al. [252]) respectively. In accordance with existing work, new publications should evaluate their bias mitigation methods on three datasets, and if possible more.

3.4.3 Synthetic Data

In addition to the 81 existing datasets for experiments, 54 publications created synthetic datasets to evaluate their bias mitigation method. Moreover, we found 3 publications that use semi-synthetic data (i.e., modify existing datasets to be applicable for evaluating bias mitigation methods) in their experiments [199, 305, 377].

The created datasets range from hundreds of data points [166, 283, 336, 343] to 100,000 and above [100, 173, 227, 234]. While the sampling procedures are well described, some publications do not state the dataset size used for experiments [36, 228, 239, 294, 312, 326, 401, 403].

As exemplary data creation procedure, we briefly outline the data generation approach applied by Zafar et al. [279], as it is the most frequently adapted approach by other publications [138, 228, 266, 282, 327, 342, 360, 382]. In particular, Zafar et al. [279] generated 4,000 binary class labels. These are augmented with 2-dimensional user features which are drawn from different Gaussian distributions. Lastly, the sensitive attribute is then drawn from a Bernoulli distribution.

3.4.4 Data-split

In this section we analyze whether existing publications provided information on the data splits, in particular what sizing has been chosen. Moreover, we investigate how often experiments have been repeated with such data splits, to account for training instability [32]. Our focus lies on the data-splits used when evaluating the bias mitigation methods (e.g., we are not interested in data-splits that are applied prior for hyperparameter tuning of classification models [126, 146, 166, 297, 311, 364, 372, 373, 436]).

Among the 324 publications that carry out experiments, 232 provide information on the data-split used and 143 provide information on the number of *runs* (different splits) performed. The high amount of publications that do not provide information on the data-split sizes could be explained by the fact that some of the 81 datasets provided default splits. For example. the Adult dataset has a pre-defined train-test split of 70%-30%, and Cotter et al. [301] used designated data splits for four datasets.

A widely adopted approach for addressing data-splits for applying bias mitigation methods is k-fold cross validation. Such methods divide the data in k partitions and use each part once for testing and the remaining $k-1$ partitions for training. Overall, 47 publication applied cross validation: 10-fold (23 times), 5-fold (21 times), 3-fold (twice), 20-fold (once), and once without specification of k [217].

If the data-splits are not derived from k-folds, the most popular sizes (i.e., train split size - test split size) are:

- 80%-20% (39 times);
- 70%-30% (35 times);
- 67%-33% (16 times);
- 50%-50% (11 times);
- 60%-40% (5 times);
- 75%-25% (5 times).

In addition to these regular sized datasplits, there are 23 publication which divide the data into very "specific" splits. For example, Quadrianto et al. [162] divided the Adult dataset into $28,222$ training, $15,000$ and $2,000$ validation instance. Another example are Liu and Vicente [382], who chose 5.000 training instances at random, using the remaining $40,222$ instances for testing.

Table 3.12: Popular fairness metrics. At least one metric for each category is provided.

Name	Section	#	Description
Statistical Parity Difference	3.5.2	137	Difference of positive predictions per group
Equality of Opportunity	3.5.3	90	Equal TPR per population groups
Disparate Impact, P-rule	3.5.2	59	Ratio of positive predictions per group
Equalized Odds	3.5.3	52	Equal TPR and FPR per population groups
False Positive Rate	3.5.3	38	False positive rate difference per group
Accuracy Rate Difference	3.5.3	29	Difference of prediction accuracy per group
	...		...
Causal Discrimination	3.5.5	7	Different predictions for identical individuals except for protected attribute
Mean Difference	3.5.1	6	Difference of positive labels per group in the datasets
Mutual information	3.5.6	4	Mutual information between protected attributes and predictions
	...		...
Strong Demographic Disparity	3.5.4	1	Demographic parity difference over various decision thresholds

Once the data is split in training and testing data, experiments are repeated 10 times in 54 out of 143 and 5 times in 42 out of 143 cases. The most repetitions are performed by da Cruz [350], who trained 48,000 models per dataset to evaluate different hyperparameter settings.

We have found 16 publications that use different train and test splits for experiments on multiple datasets. Reasons for that can be found in the stability of bias mitigation methods when dealing with a large amount of training data [214].

While most publications split the data in two parts (i.e., training and test split), there are 36 publication that use validation splits as well. The sizes for validation splits range from 5% to 30%, whereas the most common split uses 60% training data, 20% testing data, and 20% validation data. Furthermore, Mishler and Kennedy [241] allow for a division of the data in up to five different splits for evaluating their ensemble learning procedure.

Bias mitigation methods that process data in a streaming [100, 130, 224, 351, 366], federated learning [128, 142, 188, 330, 354], multi-source [139], sequential [149, 194, 320, 328] fashion need to be addressed differently, as they use small subsets of the training data instead of using all at once.

3.5 Fairness Metrics

Fairness metrics play an integral part in the bias mitigation process. First they are used to determine the degree of bias a classification model exhibits before applying bias mitigation methods. Afterwards, the effectiveness of bias mitigation methods can be determined by measuring the same metrics after the mitigation procedure.

Recent fairness literature has introduced a variety of different fairness metrics, that each emphasize different aspect of classification performance.

To provide a structured overview of such a large amount of metrics, we devise metric categories, and take into account the classifications by Catan and Haas [86], and Verma and Rubin [83]. Overall we categorize the metrics used in the 341 publications in six categories:

- Definitions based on labels in dataset;
- Definitions based on predicted outcome;
- Definitions based on predicted and actual outcomes;
- Definitions based on predicted probabilities and actual outcome;
- Definitions based on similarity;
- Definitions based on causal reasoning;

In the following, we provide information on how these metric types have been used. In total, we found 111 unique metrics that have been used by the 324 publications that performed experiments. Most publications consider a binary setting (i.e., two populations groups and two class labels for prediction), whereas fairness has also been measured for non-binary sensitive attributes [103,316,317,358,418], and multi-class predictions [103,417].

While some of the categories only contain few different metrics (Definitions based on labels in dataset, Definitions Based on Predicted Probabilities and Actual Outcome and Definitions Based on Similarity all have 13 or fewer different metrics); *Definitions Based on Predicted Outcome* have 22, *Definitions Based on Predicted and Actual Outcomes* have 33, and *Definitions Based on Causal Reasoning* 26 different metrics. Therefore, we outline the most frequently used metrics for *Definitions Based on Predicted and Actual Outcomes* and *Definitions Based on Causal Reasoning*.

On average, publications consider two fairness metrics when evaluating bias mitigation methods, with 45% of the publications only using one fairness metric. The most frequently used metrics are outlined in Table 3.12, while listing at least one metric per category. For detailed explanations of fairness metrics, we refer to Verma and Rubin [83].

In addition to quantifying the bias according to prediction tasks, we found metrics that determined fairness in accordance with feature usage (e.g., do users think this feature is fair [163]) and quality of representations [155, 158, 164] (see Section 3.3.1.4).

3.5.1 Definitions Based on Labels in Dataset

Fairness definition based on the dataset labels, also known as "dataset metrics", are used to determine the degree of bias in an underlying dataset [72]. One purpose of datasets metrics is determine whether there is a balanced representation of privileged and unprivileged groups in the dataset. This is in particular useful for preprocessing bias mitigation methods, as they are able to impact the data distribution of the training dataset.

Most frequently, datasets metrics are used to measure the disparity in positive labels for population groups, such as Mean Difference, slift or elift [391]. Hereby, Mean Difference is the most popular, used in 6 publications.

Another metric based on dataset labels is Balanced Error Rate (BER) [119]. Xu et al. [119] trained an SVM to compare the error rates when predicting protected attributes for both groups.

3.5.2 Definitions Based on Predicted Outcome

Definitions based on predicted outcome, or "Parity-based" metrics, are used to determine whether different population groups receive the same degree of favour. For this purpose, only the predicted outcome of the classification needs to be known.

The most popular approach for measuring fairness according to predicted outcome is the concept of *Demographic parity*, which states that privileged and unprivileged groups should receive an equal proportion of positive labels. This can be done as by computing their difference (Statistical Parity Difference) or their ratio (Disparate Impact). Similar to Disparate Impact, the p-rule compares two ratios of positive labels ($group_1/group_2$, $group_2/group_1$) and Among those two ratios, the minimum value is chosen. In addition to numeric bias scores, the disparity of group treatment can also be seen visually [41, 104, 125, 282, 302, 311].

If the direction of bias is of no interest (i.e., it is not important which group receives a favourable treatment), then the absolute bias values can be considered [258, 267, 272, 318]. While it is possible to compute fairness metrics based on differences as well as ratios between two groups, both which have been applied in the past, Žliobaite [87] advised against ratios as they are more challenging to interpret.

3.5.3 Definitions Based on Predicted and Actual Outcomes

Definitions based on predicted and actual outcomes are used to evaluate the prediction performance of privileged and unprivileged groups (e.g., is the classification model more likely to make errors when dealing with unprivileged groups?). Similar to definitions based on predicted outcomes, the rates for privileged and unprivileged groups are compared.

The most popular metric of this type is *Equality of Opportunity* (used 90 times), followed by *Equalized odds* (used 52 times). While *Equality of Opportunity* is satisfied when populations groups have equal TPR, *Equalized odds* is satisfied if population groups have equal TPR and FPR. In addition to evaluating fairness in according to the confusion matrix (FPR - 38 times, TNR - 8 times), the accuracy rate, difference in accuracy for both groups, has been used 29 times. Moreover, conditional TNR and TPR have been evaluated [116, 181].

3.5.4 Definitions Based on Predicted Probabilities and Actual Outcome

While Section 3.5.3 detailed metrics based on actual outcomes and predicted labels, this Section outlines metrics that consider predicted probabilities instead.

Jiang et al. [229] proposed strong demographic disparity (SDD) and SPDD, which are parity metrics computed over a variety of thresholds (i.e., prediction tasks apply a threshold of 0.5 by default). Chzhen et al. [308] also varied thresholds, to compute the Kolmogorov-Smirnov distance. Heidari et al. [286] measured fairness based on positive and negative residual differences. Agarwal et al. [304] computed a Bounded Group Loss (BGL) to minimize the worst loss of any group, according to least squares.

3.5.5 Definitions Based on Similarity

Definitions based on similarity are concerned with the fair treatment individuals. In particular, it is desired that individuals that exhibit a certain degree of similarity receive the same prediction outcome. For this purpose, different similarity measures have been applied. The most popular similarity metric used is *consistency* or *inconsistency* (used in 4 and 1 publications respectively) [68]. *Consistency* compares the prediction of an individual with the k-nearest-neighbors according the input space [68]. Loung et al. [96] also utilized k-nearest-neighbors, to investigate the difference in predictions for different values of k.

Similarities between individuals have been computed according to ℓ_∞-distance [178], and euclidean distance with weights for features [68]. Individuals have also been treated as similar if they have equal labels [29], are equal except for non-sensitive feature or based on predicted label [133]. If similarity of individuals is determined solely by differences in sensitive features, one is speaking of "causal discrimination" [46,47].[1]

In contrast to determining similarity computationally, Jung et al. [298] allowed stakeholders to judge whether two individuals should receive the same treatment.

Moreover, Ranzato et al. [237] considered four types of similarity relations (NOISE, CAT, NOISE-CAT, CONDITIONAL-ATTRIBUTE), when dealing with numerical and categorical features. Verma et al. [133] considered two types of similarities: input space (identical on non-sensitive features), output space (identical prediction). Lahoti et al. [166] built a similarity graph to detect similar individu-

[1] Some publications refer to this as "Counterfactual fairness' [244,264,386], but we follow the guidelines of Verma and Rubin [83] and treat counterfactual fairness as a Causal metric.

als. This graph is built based on pairwise information on individuals that should be treated equally with respect to a given task.

3.5.6 Causal Reasoning

Fairness definitions based on causal reasoning take causal graphs in account to evaluate relationships between sensitive attributes and outcomes [83].

For example, Counterfactual fairness states that a causal graph is fair, if the prediction does not depend on descendants of the protected attribute [334]. This definition has been adopted by four publications. Moreover, the impact of protected attributes on the decision has been observed in two ways: direct and indirect prejudice [111]. Direct discrimination occurs when the treatment is based on sensitive attributes. Indirect discrimination results in biased decision for population groups based on non-sensitive attributes, which might appear to be neutrals. This could occur due to statistical dependencies between protected and non-protected attributes.

Direct and indirect discrimination can be modelled based on the causal effect along paths taken in causal graphs [111]. To measure indirect discrimination, Prejudice Index (PI) or Normalized Prejudice Index (NPI) haven been applied four times [30]. NPI quantifies the mutual information between protected attributes and predictions. Mutual information has also been used to determine the fairness of representations [161,164]. Similar to determining the degree of mutual information between sensitive attributes and labels, the ability to predict sensitive information based on representations has been used in eight publications.

3.6 Benchmarking

After establishing on which datasets bias mitigation methods are applied, and which metrics are used to measure their performance (Section 3.5), we investigate how they have been benchmarked.

Benchmarking is important for ensuring the performance of bias mitigation methods. Nonetheless, we found 15 out of 324 publications that perform experiments but do not compare results with any type of benchmarking. Therefore, the remaining section addresses 308 publications which: 1) perform experiments; 2)

Table 3.13: Benchmarking against bias mitigation method types. For each bias mitigation category (i.e., pre-, in-, or post-processing), we count the type of benchmarking methods.

				Type		
		#	None	Pre	In	Post
Type	Pre	114	50	55	37	16
	In	184	66	56	108	51
	Post	52	16	17	25	27

apply benchmarking.

3.6.1 Baseline

To determine whether bias mitigation methods are able to reduce effectively, different types of baselines have been used.

The most general baseline is to compare the fairness achieved by classification models after applying a bias mitigation method with the fairness of a fairness-agnostic *Original Model*. If a method is not able to exhibit an improved fairness over a fairness agnostic classification model, then it is not applicable for bias mitigation. Given that this is the minimum requirement for bias mitigation methods, it is the most frequently used baseline (used in 254 out of 308 experiments).

Another baseline method is *suppressing*, which performs a naive attempt of mitigating bias by removing the protected attribute from the training data. However, it has been found that solely removing protected attributes does not remove unfairness [19,67], as the remaining features are often correlated with the protected attribute. To combat this risk, Kamiran et al. [42] suppressed not only the sensitive feature but also the k-most correlated ones. *Suppressing* has been used in 30 out of 308 experiments.

Random baselines constitute more competitive baselines than solely suppressing the protected attribute. Bias mitigation methods that outperform random baselines show that they are not only able to improve fairness but also able to perform better than naive methods. Random baselines have been used in 13 out of 308 experiments.

Moreover, we found four publications that considered a constant classifier for benchmarking (i.e., a classifier that returns the same label for every instance) [161, 228, 309, 370]. This serves as a fairness-aware baseline, as every individual and population group receive the same treatment.

3.6.2 Benchmarking Against Bias Mitigation Methods

In addition to baselines, we investigate how methods are benchmarked against other, existing bias mitigation methods. In particular, we are interested in which methods are popular, how many bias mitigation methods are used for benchmarking, and to what category these methods belong.

At first, we investigate what type of bias mitigation method are considered for benchmarking (e.g., are pre-processing methods more likely to benchmark against other pre-processing methods or in-/post-processing methods). Table 3.13 illustrates the results. In particular, # shows how many unique publications propose a given type of bias mitigation method (i.e., there are 114 publications with pre-processing methods). For each of these methods we determine whether they benchmark against pre-, in- or post-processing methods. If no benchmarking against other bias mitigation methods is performed, we count this as “None”.

We find that pre-processing methods are the most likely to not benchmark against other bias mitigation methods at 44% (50 out of 114). 36% (66 out of 184) of in-processing methods and 31% (16 out of 52) of post-processing methods do not benchmark against other bias mitigation methods. Furthermore, we can see that each bias mitigation type is more likely to benchmark against methods of the same type.

In addition to detecting the type of bias mitigation methods for benchmarking, we are interested in what approaches in particular are used for benchmarking. Therefore, we count how often each of the 341 bias mitigation methods we gathered have been used for benchmarking.

Overall, 137 bias mitigation methods have been used as a benchmark by at least one other publication. Figure 3.4 illustrates the most frequently used bias mitigation methods for benchmarking. Among the 18 listed methods, all of which are used for benchmarking by at least eight other publications, eight are pre-

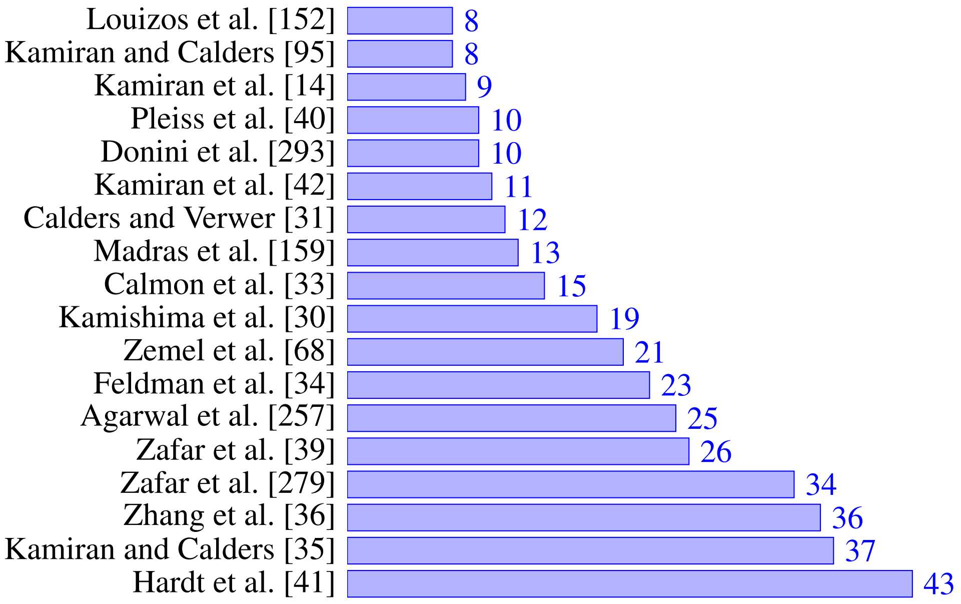


Figure 3.4: Most frequently benchmarked publications. For each publication, the number of times it has been used for benchmarking is shown.

processing, nine in-processing, and four post-processing. Notably, the five most-frequently used methods include each of the three types: sampling and relabelling for pre-processing [35], constraints [39, 279] and adversarial learning [36] for in-processing, and classifier modification for post-processing [41].

3.6.3 Benchmarking Against Fairness-Unaware Methods

In addition to benchmarking against existing bias mitigation methods, practitioners can use other methods for benchmarking, which are not designed for taking fairness into consideration. Overall, we found 51 publications that use fairness-unaware methods for benchmarking (i.e., using a general data augmentation method to benchmarking fairness-aware resampling).

Table 3.14 shows the publications that benchmark their proposed method against at least one fairness-unaware methods, according to the type of approach applied. Among the 13 types of approaches, as shown in Section 3.3.1 - 3.3.3, seven can be found to benchmark against fairness-unaware methods. This occurs

Table 3.14: Publications that benchmark against at least one fairness-unaware method.

Type	Category	Section	References
Pre	Sampling	3.3.1.2	[118, 119, 121, 126, 127, 130, 134, 137, 138, 142, 144]
	Representation	3.3.1.4	[152, 168, 169, 174, 176, 181, 182, 184, 187, 188, 189, 195]
In	Regularization	3.3.2.1	[232, 236, 246, 249, 250]
	Constraints	3.3.2.1	[137, 246, 277, 284, 307, 320, 321, 328]
	Adversarial	3.3.2.2	[119, 259, 266, 269, 270, 274]
	Adjusted	3.3.2.4	[130, 134, 232, 328, 331, 351, 358, 359, 362, 367, 373]
Post	Input	3.3.3.1	[390]
	Classifier	3.3.3.2	[403, 404]
	Output	3.3.3.3	[15, 142, 412]

rarely for post-processing methods, six publications in total, with at least one per approach type. 23 and 27 publications for pre-processing and in-processing methods respectively, benchmark against fairness-unaware methods.

3.7 Challenges

Research on bias mitigation is fairly young and does therefore enable challenges and opportunities for future research. In this section, we highlight five challenges that we extracted from the collected publications, that call for future action or extension of current work.

3.7.1 Fairness Definitions

A variety of different metrics have been proposed and used in practice (see Section 3.5), which can be applied to different use cases. However, with such a variety of metrics it is difficult to evaluate bias mitigation on all and ensure their applicability. Synthesizing or selecting a fixed set of metrics to use is still an open challenge [20, 144, 265], as can be seen by the 111 different fairness metrics obtained in Section 3.5.

While synthesising existing fairness notions is one problem, it is also relevant to ensure that the used metrics are representative for the problem at hand. Often, this means evaluating fairness in a binary classification problem for two population groups. While this can be the correct way to model fairness scenarios, it is not sufficient to handle all cases, such that future work should focus on multi-class problems [35, 262, 372, 386, 391] and non-binary sensitive attributes, which was mentioned by 15 publications.

Other challenges regarding metrics include the trade-offs when dealing with accuracy and/or multiple fairness metrics [75,86,257,437], as well as the allowance of some degree of discrimination as long it as explainable (e.g., enforcing a fairness criteria completely could lead to unfairness in another) [31,35,68,110].

3.7.2 Fairness Guarantees

Guarantees are of particular importance when dealing with domains that fall under legislation and regulatory controls [30,34]. Therefore, it is not always sufficient to establish the effectiveness of a bias mitigation method based on the performance on the test set without any guarantees.

In particular, Dunkelau and Leuschel [81] pointed out that most bias mitigation methods are evaluated on test sets and their applicability to real-world tasks depends on whether the test set reliably represents reality. If that is not the case, fairness guarantees could ensure that bias mitigation methods are able to perform well with regards to unknown data distributions. Therefore, eight publications considered fairness guarantees as a relevant avenue of future work. Similarly, allowing for interpretable and explainable methods can aid in this regard [30,107,162,281].

3.7.3 Datasets

Another challenge that arises when applying bias mitigation methods is the availability and use of datasets. The most pressing concern is the reliability and access to protected attributes, which was mentioned in nine publications, as this information is often not available in practice [438].

Moreover, it is not guaranteed that the annotation process of the training data is bias free [41]. If possible an unbiased data collection should be enforced [215]. Other options are the debiasing of ground truth labels [47, 140] or use of expert opinions to annotate data [400]. If feasible, more data can be collected [107,114], which is difficult from a research perspective, as commonly, existing and public datasets are used without the chance to manually collect new samples.

Furthermore, the variety of protected attributes addressed in experiments, as found by Kuhlman et al. [80], is lacking diversity, with the majority of cases considering race and gender only. In practice, "collecting more training data" is the most common approach for debiasing, according to interviews conducted by Holstein et al. [438].

3.7.4 Real-world Applications

While the experiments are conducted on existing, public datasets, it is not clear whether they can be transferred to real-world applications without any adjustments. For example, Hacker and Wiedemann [154] see the challenge of data distributions changing over time, which would require continuous implementations of bias mitigation methods.

Moreover, developers might struggle to detect the relevant population groups to consider when measuring and mitigating bias [438], whereas the datasets investigated in Section 3.4 often simplify the problem and already provide binarized protected attributes (e.g., in the COMPAS, six "demographic" categories are transformed to "Caucasian" and "not Caucasian" [72]). Therefore, Martinez et al. [352] stated that automatically identifying sub-populations with high-risk during the learning procedure as a field of future work.

Given the multitude of fairness metrics (as seen in Section 3.5), real world applications could even suffer further unfairness after applying bias mitigation methods due to choosing incorrect criteria [248]. Similarly, showing low bias scores does not necessarily lead to a fair application, as the choice of metrics could be used for "Fairwashing" (i.e., using fake explanations to justify unfair decisions) [403, 439]. Nonetheless, Sylvester and Raff [440] argue that considering fairness criteria while developing ML models is better than considering none, even if the metric is not optimal.

Sharma et al. [122] show the potential of user studies to not only provide bias mitigation methods that work well in a theoretical setting, but to make sure practitioners are willing to use them. In particular, the are interesting in finding how comfortable developers and policy makers are with regards to training data augmentation.

To facilitate the use and implementation of existing bias mitigation methods, metrics and datasets, popular toolkits such as AIF360 [72] and Fairlearn [441] can be used.

3.7.5 Extension of Experiments

Lastly, a challenge and field of future research is the extension of conducted experiments to allow for more meaningful results.

The most frequently discussed aspect of extending experiments is the consideration of further metrics (in 40 publications). Moreover, the usefulness of bias mitigation methods can be investigated when applied to additional classification models. This was pointed out by 12 publications. Given the 81 datasets that were used at least once, and on average 2.7 datasets used per publication, only eight publications see the consideration of further datasets as a useful consideration for extending their experiments [13,28,112,126,344,350,355,373].

While the consideration of additional metrics, classification models and datasets does not lead to changes in the training procedure and experimental design, there are also intentions to apply bias mitigation methods to other tasks and contexts, such as recommendations [238,279], ranking [30,223,279] and clustering [30].

3.8 Conclusion

In this literature survey, we have focused on the adoption of bias mitigation methods to achieve fairness in classification problems and provided an overview of 341 publications. Our survey first categories bias mitigation methods according to their type (i.e., pre-processing, in-processing, post-processing) and illustrates their procedures. We found 123 pre-processing, 212 in-processing, and 56 post-processing methods, showing that in-processing methods are the most commonly used. We devised 13 categories for the three method types, based on their approach (e.g., pre-processing methods can perform sampling). The most frequently applied approaches perform changes to the loss function in an in-processing stage (51 publications applying regularization and 74 applying constraints). Other approaches are less frequently used, with input correction in a post-processing stage only being used twice.

We further provided insights on the evaluation of bias mitigation methods according to three aspects: datasets, metrics, and benchmarking. We found a total of 81 datasets that have been used at least once by one of the 341 publications, among which the Adult dataset is the most popular (used by 77% of publications). Even though 81 datasets are available for evaluating bias mitigation methods, only 2.7 datasets are considered on average.

Similarly, we found a large number of fairness metrics that have been used at least once (111 unique metrics), which we divide in six categories. The most frequently used metrics belong to two categories: 1) Definitions based on predicted outcome; 2) Definitions based on predicted and actual outcomes.

When it comes to benchmarking bias mitigation methods, they can be compared against baselines, other bias mitigation methods, or non-bias mitigation approaches. Among the three baselines we found (original model, suppressing, random), the 82% of bias mitigation methods consider the original model (i.e., the classification model without any bias mitigation applied) as a baseline. However, the three baselines are not competitive and it can be expected for bias mitigation methods to outperform them. Moreover, benchmarking increases in complexity when multiple metrics are considered (i.e., fairness and accuracy metrics). Therefore, we set out to propose a competitive benchmarking approach in the following Chapter, for evaluating the quality of achieved fairness-accuracy trade-offs.

4

Benchmarking Bias Mitigation Methods with Fairea

If you can not measure it, you can not improve it.

– Lord Kelvin

The previous literature survey showed that there exist a multitude of methods that seek to improve the fairness of ML software. While these bias mitigation methods are able to reduce bias in light of a given fairness metric, the improvement in fairness often comes at the cost of a lower prediction accuracy [29]. In other words, there is a *software engineering trade-off* between accuracy and fairness for ML software, as revealed by many previous theoretical and empirical studies [34, 35, 43].

The existence of such trade-offs brings challenges for judging the effectiveness of bias mitigation methods. Previous work presented the trade-offs in a qualitative manner. They either report and analyse the bias mitigation effectiveness by plotting the accuracy and fairness for a visual comparison [15, 35, 40], or display accuracy and fairness separately [28, 30, 68, 97] (in tables or bar charts). As far as we know, there is no trade-off **baseline**, nor is there any **quantitative** approach that can automatically evaluate and compare the fairness-accuracy trade-offs of software bias mitigation methods.

This chapter introduces *Fairea*. *Fairea* is a novel **model behaviour mutation approach** to automatically **benchmarking and quantifying** the fairness-accuracy trade-off achieved by bias mitigation methods for ML software. With *Fairea*, we conduct a large-scale empirical study to benchmark and compare the effectiveness of 8 widely-studied bias mitigation methods that are publicly available in the popular IBM AI Fairness 360 library (AIF360) [72]. *Fairea* is the first quantitative approach to benchmarking the fairness-accuracy trade-off for bias mitigation methods. Our empirical study is also the first large-scale systematic study to evaluate the effectiveness of existing bias mitigation methods.

Our results reveal that, surprisingly, in 49% of the cases, bias mitigation methods have a poor bias mitigation effectiveness. In particular, 15% which reduce bias exhibit worse trade-offs than the baseline provided by *Fairea*, while 34% lead to a decrease in accuracy and an increase in bias. Furthermore, our observations reveal the following **limitations** among the existing bias mitigation methods: 1) it is challenging to achieve a good trade-off between fairness and accuracy; 2) methods designed to optimize one fairness metric often decrease the values of other fairness metrics; 3) the effectiveness of a method is often dataset- and model-dependent. Only rarely does an approach work well on all datasets and ML models.

To conclude, this chapter makes the following primary contributions:

- A **baseline** approach that enables evaluating the fairness-accuracy trade-off of ML bias mitigation methods through model behaviour mutation.
- A **quantitative measurement** for comparing different ML bias mitigation methods and trade-off parameters.
- A **large-scale study** on widely-studied bias mitigation methods in regards to their bias mitigation effectiveness as well as their achieved fairness-accuracy trade-offs.
- An open-source implementation of *Fairea* that has been made publicly available [442] for ML software developers and researchers to evaluating their bias mitigation methods.

The rest of the chapter is organized as follows. Section 4.1 provides information on the “fairness-accuracy” trade-off and existing practices for benchmarking bias mitigation performance. Section 4.2 introduces our approach. The experimental design is described in Section 4.3. Experiments and results are presented in Section 4.4. Section 4.5 concludes.

4.1 Fairness-Accuracy Trade-off

There have been numerous works studying the fairness-accuracy trade-off of bias mitigation methods [95]. Kamishima et al. [30] proposed a regularisation approach that adjusts the fairness-accuracy trade-off based on parameter η. Larger values of η improve fairness, but also cause a higher loss in accuracy. Berk et al. [29] normalized the loss of accuracy to study the severity of the fairness-accuracy trade-off. They call the decrease of accuracy brought by bias mitigation “Price of Fairness”. Corbett-Davies et al. [43] analysed the trade-off of public safety and racial disparities. Similar to Berk et al. [29], they showed that trade-offs can be very common in practice. Kamiran and Calders [35] gave a theoretical analysis of the trade-off. A classifier achieves an optimal trade-off if it is not dominated by another classifier (i.e., with larger accuracy and less bias).

To compare the fairness-accuracy trade-off achieved by bias mitigation methods, practitioners either observe the fairness and accuracy changes in separate graphs, or visualize them in a 2-dimensional graph (one dimension is accuracy, the other dimension is fairness) [15,31,33,34,35,37,38,40,42,67,279,443]. The proposed mitigation methods are often compared with previous methods [14,15,28,30,37,38,40,42,68,97], different configurations [14,15,30,31,34,67], the original non-optimized classifier [33,35,36,67,68], or a classifier trained without using protected attributes [33,35,67,97].

In all of these works, the loss of accuracy and improvement of fairness are measured and visualized separately. It is unclear whether the improved fairness is simply the consequence of the loss in accuracy. There is no unified baseline or quantitative measurement to evaluate and compare the fairness-accuracy trade-off throughout different studies.

Fairea aims to provide a unified standard to evaluate bias mitigation methods.

The baseline *Fairea* provides enables developers to classify the fairness-accuracy trade-offs of a bias mitigation method into good or poor. The quantitative measurement *Fairea* provides enables developers to compare different mitigation methods in a more fine-grained way, and help tune fairness penalty parameters.

4.2 The Fairea Approach

There are three primary steps in *Fairea* to benchmarking and quantitatively evaluating bias mitigation methods.

Step1: Baseline Creation with Model Behaviour Mutation. First, *Fairea* builds the baseline by simulating the behaviours of a series of naive bias mitigation models. *Fairea* does this via model behaviour mutation. The accuracy and fairness of these simulated models, together with the original classification model, are adopted to construct the fairness-accuracy trade-off baseline.

Step2: Bias mitigation effectiveness region division. Second, *Fairea* maps the effectiveness of a bias mitigation method into five mitigation regions with the *Fairea* baseline constructed in the first step. The division of such regions helps to classify bias mitigation effectiveness into different levels, providing an intuitive overview of the changes in accuracy and fairness of a mitigation method.

Step3: Quantitative Evaluation of Trade-off Effectiveness. Third, *Fairea* quantifies the effectiveness of fairness-accuracy trade-off by measuring the gap between its effectiveness and the *Fairea* baseline. This step focuses on the bias mitigation methods that improve fairness but decrease accuracy, and enables the quantitative comparison among their trade-offs.

The details for each step are explained below.

4.2.1 Baseline Creation

When presenting the fairness and accuracy of a bias mitigation method in a two-dimensional coordinate system, the baseline that *Fairea* provides can be viewed as a line, as shown by Figure 4.1. The line is constructed by connecting the fairness-accuracy points of the original model (i.e., the model obtained by using the original classifier without applying any mitigation method) and a series of naive mitigation

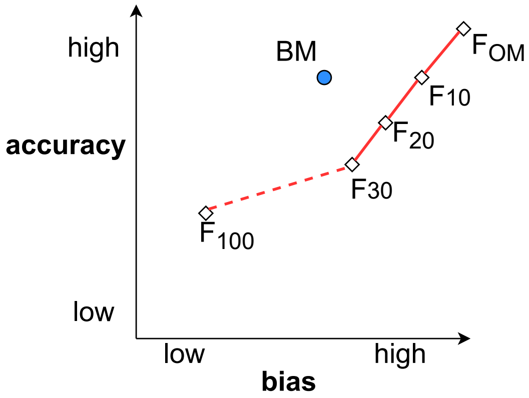


Figure 4.1: The *Fairea* fairness-accuracy trade-off baseline is represented by the F_{OM} trade-off point and the $F_{10}...F_{100}$ points obtained by model behaviour mutation. A bias mitigation method *BM* is effective if it exhibits a better trade-off than the *Fairea* baseline (i.e., if it is above the red line).

models constructed by model behaviour mutation. In the following, we explain how we obtain these points.

Trade-off points Collection: The starting trade-off point is based on the accuracy and fairness of the original model (i.e., the model without applying any bias mitigation method), as shown by point F_{OM} in Figure 4.1. The remaining points are based on the accuracy and fairness of a series of pseudo models whose behaviours are mutated from the original model. The hypothesis is that these models could improve the fairness of the original model in a naive way: by "blindly" sacrificing its accuracy with model behaviour mutation. For example, when *Fairea* mutates the original model into a random guessing model, the fairness will be greatly improved (because the predictive performance are equally worse among different protected groups), yet the accuracy is largely sacrificed. The fairness-accuracy trade-offs of such mutated models are expected to be surpassed by any reasonable bias mitigation methods. This hypothesis holds unless the original model performs even worse than a random guess model. Moreover, the bias measured by fairness metrics should monotonically decrease with an increased mutation degree. As far as we know, widely-adopted fairness metrics such as SPD, AOD, and EOD all satisfy this condition.

Mutation Degree: To obtain mutated model behaviours, we copy the original model predictions, then mutate the predictions made by this model (i.e., instead of returning the original predicted label, a random subset of the predictions is replaced by other labels). We consider different mutation degrees (i.e., the fraction of predictions to mutate) from 10% to 100%, with a step-size of 10%. For example, when the mutation degree is 10%, we randomly choose 10% of the predictions made by the original model to mutate.

Mutation Strategy: There are different mutation strategies we can choose to mutate the prediction behaviours, such as random mutation or mutating all the chosen predictions into the same label. In this chapter, we choose the second strategy following the zero-normalisation principle introduced by Speicher et al. [44], which states that fairness metrics are minimized when each individual receives the same label. For an n-class classification problem, there are n labels that one can choose to conduct mutation, therefore n mutation strategies are possible, one for each label. We choose the label that will yield the highest accuracy when 100% of the predictions are mutated, in order to provide a tighter trade-off baseline. We explore the influence of different mutation strategies in RQ4 (see more details in Section 4.4.4).

Example: Table 4.1 illustrates an example of the mutation process and its corresponding fairness-accuracy trade-off for binary classification. There are 10 instances in this example (ID from 1 to 10) belonging to two groups ($g1$ and $g2$). The column "Bias" shows the absolute False Positive Rate (FPR) difference between group g_1 and g_2. A larger absolute FPR difference indicates more bias in the model towards the two groups. The original model achieves an accuracy of 0.80, with a bias of 0.5. When the mutation degree is 40%[1], the accuracy is reduced to 0.6, the fairness is improved, with a bias of 0.17. Finally, mutating 100% of the labels achieves the best fairness with a bias of 0.0, but also leads to a low accuracy of 0.50.

Baseline Construction: As shown by Table 4.1, each mutation degree corresponds to one mutated model, whose accuracy and fairness will form a point for constructing the baseline of *Fairea*. For example, in Figure 4.1, F_{10}, F_{20}, F_{30}, ..., F_{100} illustrate the fairness and accuracy of mutated models with mutation degree of 10%, 20%, ..., 100%, respectively. These points, together with the initial fairness and

[1] In this example, mutating the predictions to label 1 and 0 have equal effects on the baseline strictness. We thus demonstrate only the results of mutating the predictions into 1.

Table 4.1: An example of the mutation procedure in *Fairea*. Bias is represented by the absolute False Positive Rate difference (*Bias*). From the table, bias can be reduced by simply "sacrificising" accuracy through mutating model predictions.

ID Group True label	1 g1 0	2 g1 1	3 g1 0	4 g1 1	5 g1 0	6 g1 1	7 g2 1	8 g2 1	9 g2 0	10 g2 0	Accuracy	Bias
Original model	0	1	0	1	0	0	1	1	1	0	0.80	0.50
mutation degree: 40%	**1**	**1**	**1**	**1**	0	0	1	1	1	0	0.60	0.17
mutation degree: 100%	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	0.50	0.00

accuracy of the original model, are connected to form the baseline of *Fairea*. The shape of the baseline is not necessarily linear. Different fairness metrics may have different baseline shapes. Both accuracy and bias values are re-scaled to a range between 0 and 1[2] for ease of presentation, which does not affect the relative comparison results among different bias mitigation methods.

4.2.2 Bias Mitigation Outcome Categorisation

After obtaining a baseline, *Fairea* categorizes the bias mitigation method's effectiveness into several regions, with different regions representing different categories of bias mitigation effectiveness.

As shown by Figure 4.2, there are five mitigation regions. If a bias mitigation method improves the accuracy and reduces the bias of the original model, it belongs to the *win-win* region. This win-win region is challenging to achieve, but is still possible [300]. A bias mitigation method falls in the *lose-lose* region if it reduces the accuracy but at the same time increases the bias of the original model (i.e., it produces worse results for both measures). If a bias mitigation improves accuracy but introduces more bias it falls in the *inverted* trade-off region. The *trade-off* region means that a bias mitigation method reduces bias but decreases accuracy. There are two types of trade-off regions: the *good trade-off* region indicates that the bias mitigation method achieves better trade-off than the baseline of *Fairea*; otherwise, it belongs to the *poor trade-off* region.

This five-region categorisation of *Fairea* helps provide an overview of the over-

[2] Given a list of values x, each element $x_i \in x$ is re-scaled given the minimum (x_{min}) and maximum (x_{max}) in x: $x_i = \frac{x_i - x_{min}}{x_{max} - x_{min}}$.

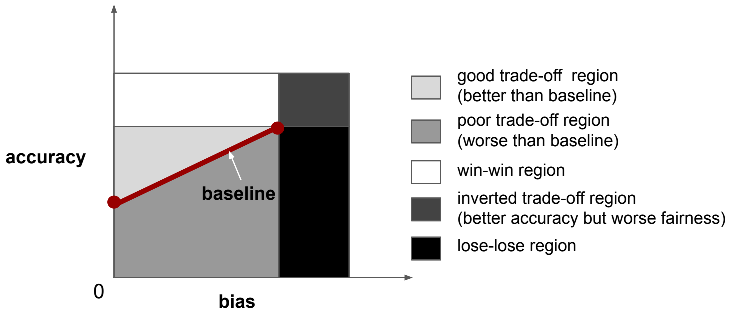

accuracy
baseline
0
bias
good trade-off region (better than baseline)
poor trade-off region (worse than baseline)
win-win region
inverted trade-off region (better accuracy but worse fairness)
lose-lose region

Figure 4.2: Mitigation regions of bias mitigation methods based on changes in accuracy and fairness. The baseline is created following the procedure we introduced in Section 4.2.1

all effectiveness of a bias mitigation method. In the following, we introduce how *Fairea* quantitatively measures the goodness of fairness-accuracy trade-off.

4.2.3 Trade-off Quantitative Evaluation

The *win-win*, *lose-lose*, and *poor trade-off* regions provide sufficiently clear signals on the effectiveness of the bias mitigation method. Thus, in this section, we focus on providing a quantitative measurement on the trade-off goodness of bias mitigation methods that fall into the *good trade-off* region, to facilitate a more fine-grained comparison for different bias mitigation methods.

Fairea measures the goodness of such a trade-off by calculating the area encompassed by a mitigation method and the *Fairea* baseline. Figure 4.3 illustrates the area obtained by connecting the bias mitigation trade-off point to the *Fairea* baseline, vertically and horizontally. The vertical line and horizontal line, together with the *Fairea* baseline, form a closed area. For example, for the case in Figure 4.3, the closed area is shown by the filled blue area, which is formed by five points: BM, BM', BM'', F_{10}, and F_{20}.

When comparing the area of two bias mitigation methods, the method with a larger area is regarded to have a better fairness-accuracy trade-off. Using the area as a trade-off measurement, instead of other criterion such as the distance to the baseline, ensures a reasonable comparison when the baseline is curved.

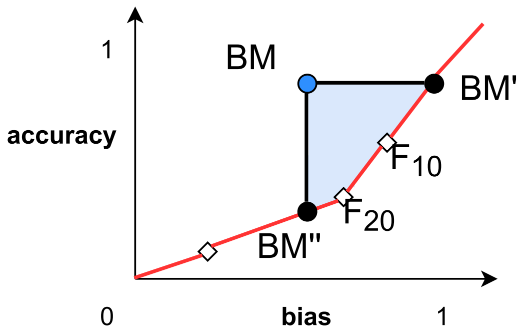


Figure 4.3: Quantifying the fairness-accuracy trade-off of a given bias mitigation method by measuring the area between *Fairea* baseline and the mitigation method. *BM* represents the accuracy and bias of the mitigation method; the red line represents *Fairea*'s baseline; the area is constructed by connecting *BM* horizontally (BM') and vertically (BM'') to the *Fairea* baseline.

4.3 Experimental Setup

In this section, we describe the design of the experiments we carry out to evaluate *Fairea*. We first introduce the research questions, then introduce the subjects and the experimental procedure. The implementation code and the results are available at our homepage [442] to support reproducibility and future studies.

4.3.1 Research Questions

Our evaluation answers the following research questions:

RQ1: Which mitigation regions do the existing bias mitigation methods fall into according to *Fairea*?

This research question evaluates the overall performance of state-of-the-art bias mitigation methods by checking how they are matched into the five mitigation regions shown by Figure 4.2, according to *Fairea*. To answer this question, we analyse the effectiveness of 8 popular state-of-the-art bias mitigation methods when used with three classification models, by mapping their accuracy-bias trade-off into mitigation regions as illustrated in Figure 4.2. We show the proportion of bias mitigation cases that fall into each mitigation region.

RQ2. What fairness-accuracy trade-off do state-of-the-art bias mitigation methods achieve based on *Fairea*?

This research question compares the methods that fall into the *good trade-off* region with the quantitative measurement *Fairea* provides. To answer this question, we calculate the area for the target method under each mitigation task (with different ML models, datasets, and fairness metrics). This allows us to quantitatively compare the methods and determine which bias mitigation method achieves the best fairness-accuracy trade-off under each task.

RQ3. Can *Fairea* be used to tune trade-off parameters for in-processing bias mitigation methods?

For in-processing methods, there are usually trade-off parameters for controlling the degree of bias mitigation. A larger trade-off parameter mitigates more bias, thus may sacrifice more accuracy. The quantitatively measurement of *Fairea* naturally enables automatic tuning of such parameters for the purpose of achieving the best trade-off. To answer the question, we investigate the in-processing methods (Prejudice Remover [30] with fairness trade-off parameter η, and Adversarial Debiasing [36] with the *adversary_loss_weight*), then check whether our measurement helps to easily spot parameters that yield good fairness-accuracy trade-off.

RQ4. How does the mutation strategy influence *Fairea*?

As explained in Section 4.2, different mutation strategies can be used to build *Fairea*. This question evaluates the difference among mutation strategies in providing the baseline. To answer this question, we compare the baselines created by the different strategies, to motivate the choice of the most suitable mutation strategy.

4.3.2 Datasets

We perform our experiments on the three[3] mostly widely-studied, real-world datasets in the fairness literature: the Adult, German, and COMPAS datasets (see Section 2.4 for more details).

These datasets are the most widely-explored in the fairness literature. For example, Galhotra et al. [60] used two datasets: Adult and German; Chakraborty et al. [344] used the same three datasets.

Table 4.2 provides more information about these three datasets. This includes

[3] The number of datasets we used align with the fairness literature. According to our collection, 90% of fairness papers use no more than three datasets in their evaluation.

Table 4.2: Dataset information.

Dataset	Size	Attri.	Favour Label	Majority Label	Prot.Attrib	Privileged
Adult	48,842	14	1 (income >50k)	0 (75%)	Sex Race	male white
COMPAS	7,214	28	0 (no recidivism)	0 (54%)	Sex Race	female caucasian
German	1,000	20	1 (good credit)	1 (70%)	Sex	male

the size of the dataset (Column “Size”), the number of attributes (Column “Attri.”), the favourable label, and the majority label. For each dataset, we present the protected attributes that are present in the dataset (Column “Prot.Attrib”). Privileged groups are outlined for protected attributes (Column “Priviledged”).

4.3.3 Bias Mitigation Methods

We explore all the three types of bias mitigation methods during our evaluation (see more details in Section 2.3). Under each type, we choose widely-studied methods, which have been implemented in the IBM AIF360 library:

- **Pre-processing**: Optimized Pre-processing (OP), Learning Fair Representations (LFR), Reweighing (RW);
- **In-processing**: Prejudice Remover (PR), Adversarial Debiasing (AD);
- **Post-processing**: Reject Option Classification (ROC), Calibrated Equalized Odds (CO), Equalized odds (EO).

In AIF360, ROC and CO are implemented with three different fairness metrics to guide the bias mitigation process. ROC offers a choice between SPD, AOD, and EOD; CO offers a choice between False Negative rate (FNR), False Positive Rate (FPR), and a weighted metric to combine both. We implemented and evaluated every of the three methods for ROC and CO. All together, we study 8 bias mitigation methods.

4.3.4 Experimental Configuration

Pre-processing and post-processing methods are model independent. We implement them using three traditional classification models, which have been widely adopted in previous works that study fairness: Logistic Regression (LR) [14, 15, 28, 30, 34, 279], Decision Tree (DT) [14,15], and Support Vector Machine (SVM) [15,34,279]. As in previous work [14, 15, 28], we use the default configuration for each classifier, as provided by Scipy.[4]

The two in-processing methods studied in this chapter have their own model with different trade-off parameters. In this case, to build *Fairea*, when getting the original model, we turn off the trade-off parameters (so that such a model does not use any bias mitigation function); when evaluating the effectiveness of a in-process method in RQ1 and RQ2, we use its default trade-off parameter. In RQ3, we explore the trade-off performance of different parameters and investigate whether *Fairea*'s quantitative measurement helps to tune the parameters to get the best trade-off.

We perform our experiments on the three most widely-studied, real-world datasets in the fairness literature: the Adult, German, and COMPAS datasets. We apply each of the bias mitigation methods to the three datasets and their protected attributes, with three ML models and two fairness metrics. Thus, for each mitigation method, it will be evaluated per *(dataset, protected attribute, ML model, fairness metric)* combination. We call such as a combination a mitigation **task**.

Each optimisation process is repeated 50 times, each time with a random respilt of the data based on a fixed train-test split ratio 7:3. We use the mean value of these multiple runs to represent the method's average performance, as a common practice in the fairness literature [38, 444]. We treat each single run as an individual **mitigation case**, and present the proportion of cases that fall into each bias mitigation region for a bias mitigation method (to answer RQ1). The baseline is also obtained by repeating the label model behaviour mutation procedure 50 times for each mutation degree (10%, 20%, ..., 100%).

The source code containing the implementation of *Fairea* and the implementation/configuration of each bias mitigation method, as well as the results, are available in our project repository [442].

[4] https://www.scipy.org/

4.3.5 Threats to Validity

The primary threat to internal validity lies in the implementation of *Fairea*. To reduce this threat, the authors independently reviewed the implementation code. The adoption of IBM AIF360 framework [72], a widely adopted fairness tool in software fairness [28, 444], also reduces such threat. The threats to external validity lie primarily with the subjects investigated. To reduce this threat, we use the three most widely adopted datasets in fairness research. We study 8 bias mitigation methods, with different classification models, to obtain more generalized conclusions. Moreover, we make our scripts and data publicly available, to allow for reproductions, replications and its adoption in future bias mitigation studies [442].

4.4 Empirical Study Results

This section presents the results of our experiments to answer the research questions explained in Section 4.3.1.

4.4.1 RQ1: Mitigation Region Distribution

The first research question checks the mitigation region distribution of the existing bias mitigation methods. We apply bias mitigation methods to the three datasets to evaluate their region distribution (Section 4.2) according to the baseline provided by *Fairea*.

We apply each pre- and post-processing bias mitigation method on three classification models (LR, DT, SVM) used for five bias mitigation tasks (i.e., *Adult-sex*, *Adult-race*, *COMPAS-sex*, *COMPAS-race*, *German-sex*). Each task is repeated for 50 times with different training-test splits. DT achieves a prediction accuracy below the majority class for the German dataset. Therefore, it does not meet our baseline requirement (as introduced in Section 4.2.1) and is disregarded in the subsequent experiments. Thus, for each bias mitigation method, there are 5*3*50-50 = 700 evaluations.

For each in-processing method, as we introduced in Section 4.3.4, we build the baseline upon an original model without applying bias mitigation (with the trade-off parameter set to 0). For Prejudice Remover, its accuracy on the COMPAS/German

Table 4.3: RQ1: Proportion of mitigation cases that fall into each mitigation region. We observe that half of the existing bias mitigation methods either decrease accuracy and increase bias (*lose-lose*) of the original model, or have a worse trade-off than the *Fairea* baseline (*poor trade-off*).

Bias mitigation method		Statistical Parity Difference (SPD)					Average Odds Difference (AOD)				
		Lose-Lose	Poor	Inverted	Good	Win-Win	Lose-Lose	Poor	Inverted	Good	Win-Win
Pre	LFR	19%	48%	0%	20%	13%	33%	38%	0%	17%	13%
	OP	11%	16%	14%	40%	18%	20%	11%	13%	36%	20%
	RW	5%	14%	4%	54%	23%	12%	12%	3%	49%	24%
In	PR	1%	6%	0%	85%	8%	11%	0%	1%	81%	7%
	AD	29%	5%	12%	44%	10%	55%	5%	15%	17%	8%
Post	CO_{fnr}	52%	2%	15%	30%	2%	52%	5%	14%	26%	2%
	CO_{fpr}	58%	20%	7%	7%	8%	66%	13%	7%	6%	8%
	$CO_{weighed}$	64%	3%	21%	6%	7%	64%	2%	20%	6%	8%
	ROC_{SPD}	19%	26%	0%	45%	9%	28%	25%	0%	37%	9%
	ROC_{AOD}	45%	16%	4%	26%	9%	26%	28%	3%	34%	9%
	ROC_{EOD}	47%	15%	4%	26%	9%	43%	14%	3%	31%	9%
	EO	11%	6%	6%	69%	8%	14%	4%	7%	67%	8%
	Mean	33%	16%	7%	33%	10%	36%	15%	7%	31%	11%

dataset is too low to be reduced by mutation, we thus only present its results on the Adult dataset. Therefore, our experiment conducts 50 evaluates on Prejudice Remover, and 250 evaluations on Adversarial Debiasing.

We then calculate the percentage of evaluations that fall into each region. We use the proportion as a high-level indication of the bias mitigation performance of each method.

4.4.1.1 Overall Results

Table 4.3 shows the results of the region classification of bias mitigation methods. Each row represents a bias mitigation method. Each cell contains a percentage of scenarios that fall into corresponding regions for a mitigation method. The last row shows the overall ratios for each mitigation region.

We make the following primary observations from Table 4.3. **First**, to our surprise, a large proportion of bias mitigation performance falls into the *lose-lose* trade-off region. For example, for the CO_{fnr} post-processing method, the proportion is as high as 52% for AOD. The mean value of the *lose-lose* proportion is 33% for SPD and 36% for AOD, which means that those bias mitigation methods perform worse than the original model. For SPD, 49% of the bias mitigation methods perform worse than *Fairea* while 43% perform better. Similarly, 51% of the bias mitigation methods achieve worse trade-offs than *Fairea* for AOD, while being better among 42% of the evaluations.

One possible reason for this is that mitigation methods are often designed to optimize one fairness metric, but such kind of one-target optimisation usually affects other fairness metrics [445,446]. For example, CO_{fnr} and CO_{fpr} are designed to optimize the difference of false negative/positive rate between privileged and unprivileged groups. Their *lose-lose* percentages measured by SPD and AOD are over 50%. Nevertheless, we observe that when using the same metric to optimize and measure mitigation performance, the *lose-lose* percentages are still high (i.e., 19% for ROC_{SPD} measured by SPD, and 26% for ROC_{AOD} measured by AOD).

Second, a notable proportion of evaluations fall into the poor trade-off region (16% for SPD and 15% for AOD). While this means that they achieve more fairness than the original model, their fairness-accuracy trade-off is worse than the baseline of *Fairea*.

We also observe a small ratio of evaluations falling into the *win-win* region (10%) or *inverted* trade-off region (7%). A larger proportion of pre-processing methods belong to the *win-win* region, in comparison to in- and post-processing methods. This may indicate that optimising training data has more promises in providing solutions to optimize both accuracy and fairness.

Third, pre-processing methods are more likely to fall into the *win-win* region with both accuracy and fairness being improved. For example, for SPD, the average proportion of pre-processing methods that fall into the *win-win* region is 18%, which is only 7% for post-processing methods. This suggests that, if one pursues improving both accuracy and fairness, it might be favourable to pre-process the training data and prevent the bias from reaching the model, than to mitigate the bias after the model has learned the bias from the data.

4.4.1.2 Comparison among Different Models and Datasets

We further analyse the region distribution based on ML models (for pre- and post-processing methods) and datasets. The purpose is to investigate whether the performance of different bias mitigation methods are influenced by ML models or datasets.

Table 4.4: RQ1: Averaged proportion of mitigation cases that fall into each mitigation region organized by different ML models (top three rows) and datasets (bottom five rows). The differences across models and datasets indicate that the effectiveness of the methods we studied are model and dataset dependent.

	Lose-Lose	Poor	Inverted	Good	Win-Win
LR	30%	20%	3%	41%	6%
DT	43%	8%	12%	24%	12%
SVM	28%	18%	6%	36%	13%
Adult - Sex	49%	17%	1%	32%	1%
Adult - Race	43%	15%	3%	37%	2%
COMPAS - Sex	18%	13%	10%	35%	23%
COMPAS - Race	26%	8%	15%	34%	16%
German - Sex	34%	27%	9%	19%	12%

Table 4.5: RQ2: Trade-off assessment results for pre-processings and post-processing methods. For each method in the *good trade-off* region, a trade-off measurement value provided by *Fairea* is given; for other regions the region type is displayed. The values in bold indicate the best mitigation method for each mitigation task. From this table, we observe that *Fairea* provides distinguishable measurements for trade-off comparison, and helps to detect the best mitigation method under each bias mitigation task. We abbreviate "win-win" with "WW", and "lose-lose" with "LL".

			Logistic Regression (LR)					Decision Tree				SVM				
			Adult		COMPAS		German	Adult		COMPAS		Adult		COMPAS		German
			Sex	Race	Sex	Race	Sex	Sex	Race	Sex	Race	Sex	Race	Sex	Race	Sex
Statistical Parity Difference	Pre	LFR	poor	poor	poor	poor	poor	poor	poor	poor	poor	poor	poor	poor	poor	poor
		OP	poor	0.002	0.076	0.011	LL	0.008	0.111	**WW**	inverted	0.000	0.002	**WW**	inverted	LL
		RW	0.001	0.007	0.195	0.138	poor	**0.029**	**0.176**	**WW**	**WW**	0.001	0.029	**WW**	**WW**	LL
	Post	CO_{fnr}	0.014	0.019	LL	LL	LL	LL	LL	LL	LL	0.011	0.012	LL	LL	LL
		CO_{fpr}	LL	LL	poor	LL	**0.115**	LL	LL	0.000	LL	LL	LL	poor	LL	**0.063**
		$CO_{weighed}$	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL
		ROC_{SPD}	0.006	poor	**0.274**	**0.273**	poor	LL	LL	0.112	0.043	poor	poor	0.264	0.258	poor
		ROC_{AOD}	LL	LL	0.185	0.185	poor	LL	LL	LL	LL	LL	poor	0.172	0.180	poor
		ROC_{EOD}	LL	LL	0.149	0.093	poor	LL	LL	LL	LL	LL	poor	0.126	0.108	poor
		EO	**0.024**	**0.067**	0.104	0.159	0.038	poor	LL	0.002	0.000	**0.021**	**0.054**	0.118	0.166	0.018
Average Odds Difference	Pre	LFR	LL	LL	poor	poor	poor	LL	LL	poor	poor	poor	poor	poor	poor	poor
		OP	poor	0.028	0.108	0.027	LL	poor	LL	**WW**	inverted	0.028	0.041	**WW**	inverted	LL
		RW	0.041	0.039	0.213	0.153	poor	**0.016**	LL	**WW**	**WW**	0.009	0.026	**WW**	**WW**	LL
	Post	CO_{fnr}	0.000	0.066	LL	LL	LL	LL	LL	LL	LL	0.037	0.087	LL	LL	LL
		CO_{fpr}	LL	LL	poor	LL	**0.054**	LL	LL	0.000	LL	LL	LL	LL	LL	**0.038**
		$CO_{weighed}$	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL	LL
		ROC_{SPD}	LL	poor	**0.281**	**0.201**	poor	LL	LL	0.140	0.040	poor	poor	0.240	0.215	LL
		ROC_{AOD}	poor	poor	0.229	0.187	poor	LL	LL	LL	LL	poor	0.001	0.204	0.201	LL
		ROC_{EOD}	LL	LL	0.197	0.112	poor	LL	LL	LL	LL	LL	0.003	0.154	0.141	LL
		EO	**0.169**	**0.198**	0.111	0.159	0.029	LL	LL	0.003	0.000	**0.158**	**0.087**	0.120	0.160	0.010

Table 4.4 shows the results. Among the three classification models, we observe that different models have different results, which indicates that the effectiveness of pre- and post-processing methods are model dependent. Overall, LR and SVM have a better effectiveness (higher percentage of *good trade-offs*) than DT.

Among different datasets and protected attributes, the differences are also notable. We observe that for the COMPAS dataset, there are more scenarios in the win-win region and fewer scenarios in the lose-lose region. We suspect that this is because COMPAS dataset is more balanced than Adult and German (54% of majority labels v.s. 75% and 70% majority labels according to Table 4.2).

To conclude, for RQ1, we have the following answer:

Answer to RQ1: Surprisingly, approximately 50% of the bias mitigation scenarios have a poor mitigation effectiveness, with 34% of them decreasing accuracy and increasing bias (*lose-lose*), and 15% of them exhibiting a *poor trade-off* according to *Fairea*.

4.4.2 RQ2: Quantitative Measurement for Fairness-Accuracy Trade-off

To answer RQ2, we present the quantitative measurement results of the fairness-accuracy trade-off achieved by different bias mitigation methods with *Fairea*. We quantify results that fall into the *good trade-off* region, as the other regions are either strictly dominating the original model (*win-win*), dominated by the *Fairea* baseline (*lose-lose* and *poor trade-off*, or do not improve fairness (*inverted*). We use the arithmetic mean results of the 50 runs to indicate the average level of mitigation effectiveness.

Table 4.5 shows the results for pre- and post-processing bias mitigation methods. The values in bold indicate the best mitigation method for each mitigation task (i.e., the combination of dataset, protected attribute, ML model, and fairness metric). From the table, the quantitative trade-off measurement *Fairea* provides helps to compare different the trade-offs among different mitigation method, and to choose the best one under each mitigation task.

The same as RQ1, we observe that the trade-offs of bias mitigation methods are highly dataset dependent. For example, the best trade-off on the Adult dataset is achieved by EO (highest scores for both AOD and SPD). The best trade-off on German is achieved by CO_{fpr}.

We also explore whether the protected attribute considered under each dataset impacts the performance of bias mitigation methods. From Table 4.5, for the same dataset, different protected attributes have very similar patterns. Specifically, in 85% (102/120) of the cases, bias mitigation methods are classified into the same mitigation region with different protected attributes. This suggests, that the protected attribute has a limited impact on the trade-off performance of bias mitigation methods.

Table 4.6: RQ2: Trade-off assessment results for in-processing methods.

		Adult		COMPAS		German
		Sex	Race	Sex	Race	Sex
Statistical Parity Difference	PR	0.042	0.003	NA	NA	NA
	AD	0.176	0.042	lose-lose	lose-lose	lose-lose
Average Odds Difference	PR	0.090	0.011	NA	NA	NA
	AD	lose-lose	lose-lose	lose-lose	lose-lose	lose-lose

Due to the different characteristics of in-processing methods, we provide their quantitative results separately in Table 4.6. Prejudice Remover is not applicable to the COMPAS and German dataset (see Section 4.4.1.1 for more details) so we mark the results as “NA”. Adversarial Debiasing is applicable for all three datasets, however only achieves *good trade-offs* on the Adult dataset for SPD. All the other trade-offs are in the *lose-lose* region. However, when comparing the two in-processing methods on Adult dataset measured by SPD, Adversarial Debiasing has a better trade-off than Prejudice Remover.

These observations lead to the following answer to RQ2:

Answer to RQ2: The quantitative measurement of *Fairea* allows us to determine and compare fairness-accuracy trade-offs achieved by different bias mitigation methods. For example, *Fairea* measures that the EO method achieves a 71.4% better trade-off than CO_{fnr} (i.e., 0.024 vs. 0.014) for the case LR-Adult-Sex under Statistical Parity Difference. Different datasets have different bias mitigation methods that achieve the best trade-off (i.e., EO for Adult; CO_{fpr} for German).

4.4.3 RQ3: Parameter Tuning

In RQ3, we investigate the effectiveness of *Fairea* in evaluating the parameter tuning for in-processing methods. For this purpose, we apply *Fairea* on the original model of Prejudice Remover with $\eta = 0$, and Adversarial Debiasing with an $adversary\_loss\_weight = 0$.

As in previous experiments, we perform 50 train-test splits for all numerical values of η between 1-100 for PR, with a step size of 1. We evaluate *adversary_loss_weights* in a range of 0.05-1, with a step size of 0.05. Due to limited space, we choose the Adult dataset as an example to illustrate experiments on parameter tuning. Full results are available in our project repository [442].

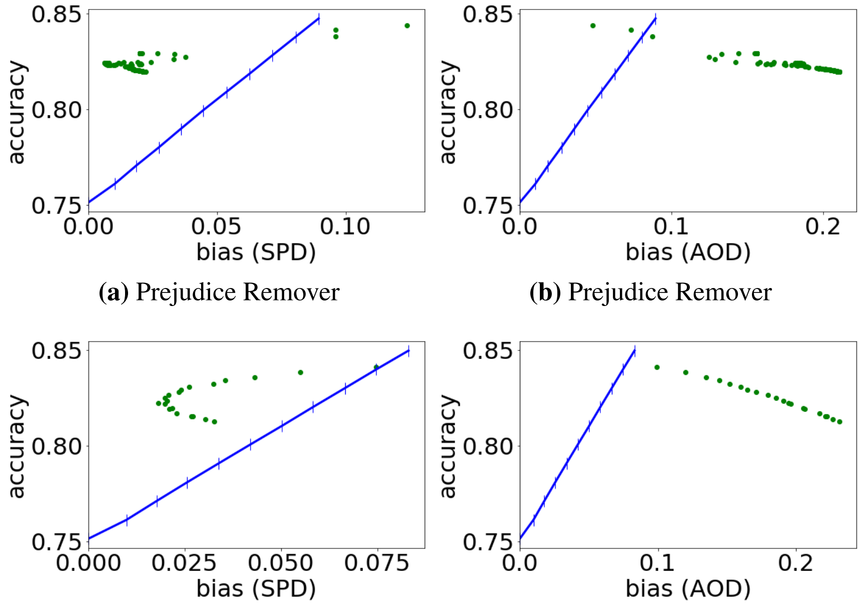
We first plot the accuracy and fairness achieved by each parameter setting, shown in Figure 4.4. For both methods, all parameter settings for SPD achieves better trade-off than the original model. However, the bias mitigation effectiveness for AOD is much worse.

Although the different parameters all belong to the good trade-off region, it is difficult to determine which parameter setting achieves the best fairness-accuracy trade-off. We therefore investigate whether our quantitative measurement in *Fairea* helps spot the parameter that achieves the best trade-off.

Figure 4.5 shows these results. Sub-Figures 4.5 (a) and (b) show the trade-off measurement results provided by *Fairea* with different trade-off parameters. The remaining sub-figures show the accuracy and fairness changes separately without *Fairea*.

From sub-Figure 4.5.(a) and sub-Figure 4.5.(b), we observe that, when the trade-off parameter changes, our trade-off measurement first increases, then decreases, with a turning point indicating the parameter with the best trade-off. However, from the remaining sub-figures, without the support from *Fairea*, it is difficult to choose a parameter with accuracy and fairness changing at the same time.

Of course, in practice, the desired trade-offs may depend on the application scenario and the specific requirement. Some applications may demand a higher degree of fairness, with the capability of enduring more accuracy loss. However,



(a) Prejudice Remover

(b) Prejudice Remover

(c) Adversarial Debiasing

(d) Adversarial Debiasing

Figure 4.4: RQ3: Accuracy and fairness achieved by Prejudice Remover (sub-figure a and b), and Adversarial Debiasing (sub-figure c and d) with different parameters on Adult-sex. Each green point represents a trade-off parameter, the blue line represents the *Fairea* baseline.

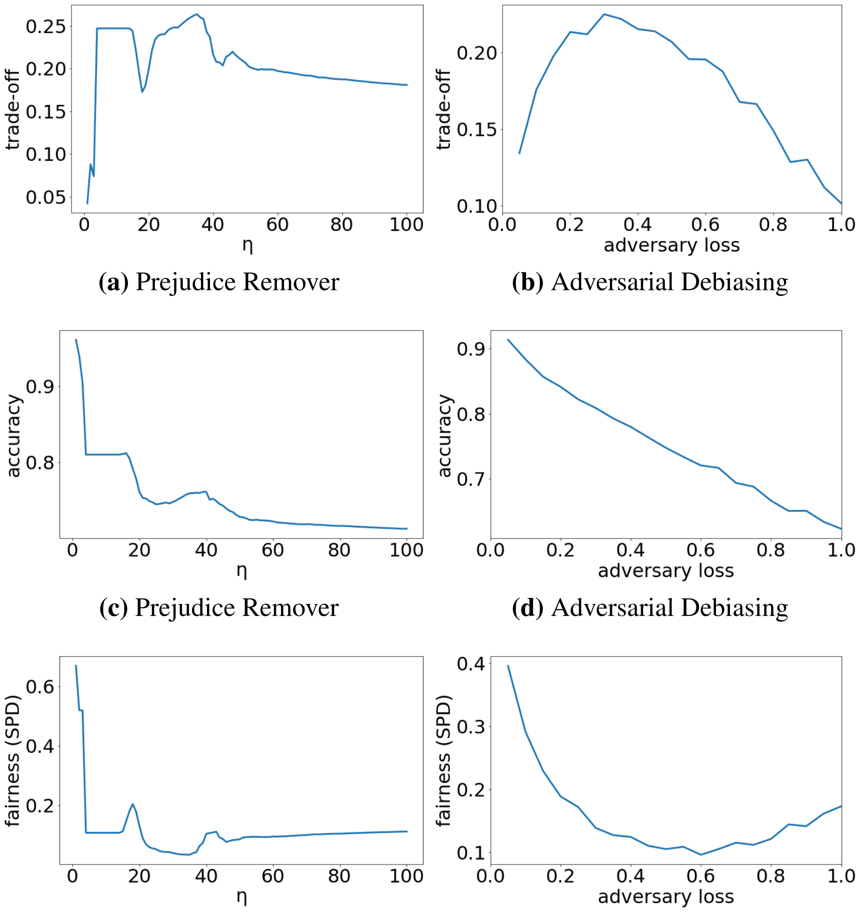
the quantitative measurement in *Fairea* provides an engineering solution for finding the best trade-off as a reference for developers.

Answer to RQ3: Our trade-off measurement helps to quantify the fairness-accuracy trade-offs achieved by in-processing methods with different trade-off parameter settings, and to identify parameters that achieve the best fairness-accuracy trade-offs.

4.4.4 RQ4: Influence of Mutation Strategies

This research question is designed to investigate how the mutation strategy for simulating naive mitigation methods affects the construction of the *Fairea* baseline. We show and compare three different mutation strategies: replace labels with “0”, replace labels with “1”, and replace labels at random.

Figure 4.6 shows the accuracy and fairness (SPD, AOD) of the three mutation strategies. We analysed all three datasets, the conclusions are identical, so we only present results for the Adult-sex task (full results are available in our project



(a) Prejudice Remover **(b)** Adversarial Debiasing

(c) Prejudice Remover **(d)** Adversarial Debiasing

(e) Prejudice Remover **(f)** Adversarial Debiasing

Figure 4.5: RQ3: In-processing trade-off parameter tuning with *Fairea*. The horizontal axis in each sub-figure shows different parameter values. Figure (a) and (b) show the trade-off measurement changes provided by *Fairea*. Figure (c), (d), (e), (f) show the changes of accuracy and fairness separately.

repository [442]).

As can be seen, when we mutate the labels with the majority class label (0 in the case for Adult-sex), its baseline is on top of the other two strategies. This means that overwriting with the *majority* label provides a more strict baseline than the other two strategies. Mutation with the *minority* class label (1 in the case for Adult-sex) instead leads to a baseline with lower accuracy on the same level of fairness. Using such a baseline would provide weaker conditions when checking the trade-off of bias mitigation methods. Replacement with random labels leads to a baseline in-between the other two strategies, but with 100% labels replaced, the fairness values are not minimized at zero because of the imbalanced data distribution.

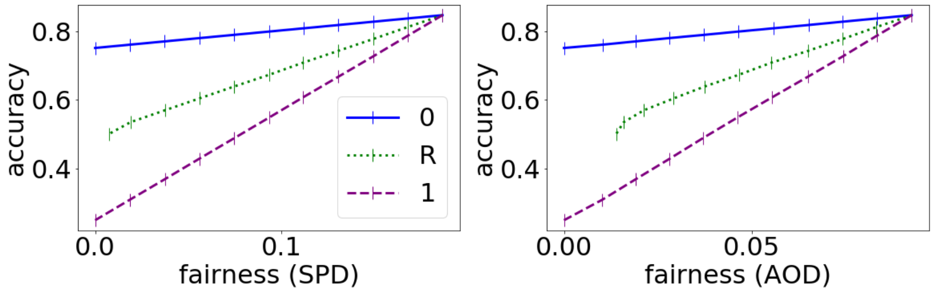


Figure 4.6: RQ4: Comparison of three mutation strategies (mutate the original prediction into 0, 1, or randomly (*R*)) on the Adult dataset with the protected attribute *sex*.

In this chapter, we adopted the strategy of mutating predictions with the majority class label in the training data. Although this is the most strict among the three strategies, it is still a naive bias mitigation method achieved simply by label overwriting, which we expect that a reasonably effective bias mitigation method should outperform.

Answer to RQ4: Among the different mutation strategies we explored, replacing labels with the majority class label for a dataset leads to the strictest baseline.

4.5 Conclusions

In this chapter, we proposed *Fairea*, a novel approach to evaluating and quantitatively measuring the fairness-accuracy trade-off. There are three primary questions that previous work could not answer without *Fairea*: 1) The *Fairea* baseline tells whether a bias mitigation method trades accuracy for fairness (or even worse than that). The qualitative approach used by previous work is not able to differentiate “good trade-off” and “poor trade-off” like *Fairea* does; 2) *Fairea* provides extra information for developers by telling whether bias mitigation method A outperforms method B when they both achieve a “good trade-off”; 3) *Fairea* helps to tune the fairness mitigation parameter for in-processing methods.

We performed a large scale empirical study to evaluate our baseline *Fairea* on three widely used datasets and 8 bias mitigation methods. We found that half of the bias mitigation methods are not able to achieve a reasonable bias mitigation effectiveness (either achieving a worse trade-off than our baseline, or decreasing accuracy and increasing bias). In addition, few methods perform well on all datasets and all models. These results show the limitations and challenges of the existing bias mitigation methods, suggesting the need for further research effort on improving ML software fairness.

5

Fairness and Accuracy Improvements with Post-Processing

> *Knowing that one may be subject to bias is one thing; being able to correct it is another.*
>
> – Jon Elster

A naive approach for repairing fairness issues in ML software is the removal of sensitive attributes (i.e., attributes that constitute discriminative decisions, such as age, gender, or race) from the training data. However, this has shown to not be able to combat unfairness and discriminative classification, owing to correlation of other attributes with sensitive attributes [19, 67, 95]. Therefore, more advanced methods have been proposed in the literature, which apply bias mitigation at different stages of the software development process (see Chapter 3). However, there are limitations for the applicability of these methods and it has been shown that they often reduce bias at the cost of accuracy [14, 15], known as the *price of fairness* [29].

In this chapter, we introduce a **search-based** procedure, which can be applied to **mutate** classification models in a post-processing stage, in order to automatically **repair software fairness and accuracy issues**. We conduct a thorough empirical study to evaluate the feasibility and effectiveness of our search-based approach. We apply our method on two widely-studied binary classification models in ML software fairness research, namely Logistic Regression [14, 15, 28, 30, 34, 279] and Decision Trees [14, 15, 42, 97], which belong to two different families of classifiers. These two models are also widely adopted in practice on fairness-critical scenarios, mainly due to their advantages in explainability.[1] We investigate the performance on four widely adopted datasets, and measure the fairness with three widely-adopted fairness metrics. Furthermore, we benchmark our method with all existing post-processing methods publicly available from the popular IBM AIF360 framework [72].

The results show that our approach is able to improve both accuracy and fairness of Logistic Regression and Decision Tree classifiers in 61% of the cases. The three existing bias mitigation methods we studied conform to the fairness-accuracy trade-off and therefore decrease accuracy when attempting to mitigate bias. Among all investigated bias repair methods, our approach achieves the highest accuracy in 100% of the cases, while also achieving the lowest bias in 33% of these. With our approach, engineers are able to develop fairer classification models without the need to sacrifice accuracy.

In summary, we make the following contributions:

- We propose a novel search-based post-processing approach for mutating classification models to repair both, fairness and accuracy issues. This approach is applied to Logistic Regression and Decision Trees.

- We carry out a thorough empirical study to evaluate the applicability and effectiveness of our search-based post-processing approach to two different classification models on four datasets, three fairness metrics, and three state-of-the-art post-processing methods used as a benchmark.

Additionally, we make our scripts and experimental results publicly available to allow for replication and extension of our work [447].

The rest of this chapter is organized as follows. Section 5.1 provides the background and related work on fairness research, including fairness metrics and bias mitigation methods. Section 5.2 introduces our approach that is used to adapt

[1] Decision-making scenarios that highly demand fairness often require high explainability, while low explainability is a big disadvantage of big complex models such as Deep Neural Networks.

trained classification models. The experimental design is described in Section 5.3. Experiments and results are presented in Section 5.4. Section 5.5 concludes.

5.1 Background and Related Work

Our approach targets the design of fair ML software systems. While software systems can be designed to reduce discrimination, previous work has observed that this frequently is accompanied by a reduction of the accuracy or correctness of said models [34, 35, 43]. In their FSE'2020 work, Chakraborty et al. [28] integrated bias mitigation into the design of ML software. In particular, they applied multi-objective optimization to the hyperparameter tuning of a Logistic Regression model. Similarly, our approach integrates bias mitigation into the software development process, however at a different stage. While Chakraborty et al. [28] considered pre- and in-processing approach for bias mitigation, we propose a post-processing approach. Moreover, our approach is not focused on a single classification model, but can be transferred to multiple ones, as we show by using it to improve Logistic Regression and Decision Tree models. Lastly, while their multi-objective optimization does not prevent the improvement of accuracy and fairness at the same time, our approach demands the improvement of both.

Herein we propose a novel post-processing method, therefore in the following we discuss the most common post-processing methods, which are also used as a benchmark in our experiments (Section 5.4), and the main difference with the approach we propose herein.

Kamiran et al. [14, 15] proposed Reject Option based Classification (ROC), which exploits predictions with high uncertainty. This follows the intuition that discriminatory decisions are made close to the decision boundary and therefore with uncertainty. Given a region with low confidence (e.g., labels close to 0.5 in binary classification), instances belonging to the unprivileged group receive a favorable label, and instances of the privileged group an unfavorable label. Instances outside the low confidence region remain unchanged.

Other than modifying predictions in a post-processing stage, trained classifiers can be addressed as well. Savani et al. [394] called the post-processing of trained classification models “intra-processing” and proposed an approach for modifying the weights of Neural Networks.

Kamiran et al. [42] applied leaf relabeling, as a post-processing method on already trained Decision Trees. Usually, labels of leaves are determined by the majority class of the training data which is classified by this particular leaf node. In their debiasing method, leaves are relabeled to reduce discrimination (e.g., a leaf that is returning “false” is changed to return “true”), while also keeping the loss in accuracy minimal. In particular, each leaf node is investigated to select and relabel the leaf with the highest ratio of discrimination reduction and accuracy loss. Their approach assumes that, in order to lower discrimination of DTs, one has to lower accuracy.

Hardt et al. [41] proposed a post-processing method based on equalized odds. A classifier is said to satisfy equalized odds when it is independent of protected attribute and true label (i.e., true positive and false positive rates across privileged and unprivileged group are equal). Given a trained classification model, they used linear programming to derive an unbiased one. Another variant of the equalized odds bias mitigation method has been proposed by Pleiss et al. [40]. In contrast to the original equalized odds method, they used calibrated probability estimates of the classification model (e.g., if 100 instances receive $p = 0.6$, then 60% of them should belong to the favorable label 1).

Our herein proposed post-processing approach differs from the leaf relabeling approach proposed by Kamiran et al. [42], as we do apply changes to the classification model only if they do not decrease its accuracy. In other words, our approach is the first to deliberately optimize classification models for accuracy and fairness at the same time, unlike existing methods that are willing to reduce bias at the cost of accuracy [29]. Overall, we apply a search procedure rather than deterministic approaches [14,15,40,41,42] and we do not assume that bias reduction has to come with a decrease in accuracy. To the best of our knowledge our proposal is the first to improve classification models according to *both* fairness and accuracy by mutating the classification model itself, rather than manipulating the training data or the predictions.

5.2 Proposed Approach

This section introduces the search-based procedure we propose for mutating classification models to simultaneously improve both accuracy and fairness. In addition, we describe implementation details for two classification models (Logistic Regression, Decision Trees) to perform such a procedure.

5.2.1 Procedure

Our search-based post-processing procedure aims to iteratively mutate a trained classification model in order to improve both accuracy and fairness at the same time. For this purpose, we require a representation of the classification model that allows changes ("mutation") to the prediction function. To simplify the mutation process, we apply mutation incrementally (i.e., repeatedly changing small aspects of the classifier). Such a procedure is comparable to the local optimisation algorithm Hill Climbing. Based on an original solution, Hill Climbing evaluates neighboring solutions and selects them only if it improves the original fitness [448]. We mutate a trained classification model clf with the goal to achieve improvements in accuracy and fairness. In this context, the fitness function measures the accuracy and fairness of clf on a validation dataset (i.e., a dataset that has not been used during the initial training of clf). "Accuracy" (acc) refers to the standard accuracy in machine learning, which is the number of correct predictions against the total number of predictions. To measure fairness, we use the three fairness metrics introduced in Section 2.1 (SPD, AOD, EOD).

Algorithm 1 outlines our procedure to improve accuracy and fairness of a trained classification model clf, where $\succ$ denotes the domination operator (i.e., $a \succ b$ shows that a dominates b) [449]. In a multi-objective optimization environment, a solution a dominates another solution b, if for none of the objectives a is worse than b, and a is better than b in at least one objective. In line 4, $fitness(clf)$ determines the fitness of the modified classification model in terms of accuracy

(acc') and a fairness metric ($fair'$). In our empirical study we experiment with three different fairness metrics (see Section 2.1), one at a time. If desired, $fitness(clf)$ can also be modified to take multiple fairness metrics into account simultaneously.

We only apply a mutation if the accuracy and fairness of the mutated model ($acc', fair'$) dominates the accuracy and fairness of the previous classification model ($acc, fair$) (Line 5). If that is not the case, the mutation is reverted ($undo\_mutation$) and the procedure continues until the terminal condition is met (e.g., the search procedure was repeated for a predefined number of iterations). A mutation of the trained model at each iteration of the search process that leads to an improvement in one objective (either accuracy or fairness) will almost certainly change the other objective at the same time. If the other objective is not worsened, the change is kept; otherwise, the change is reverted. This effect is accumulated over each iteration.

To show the generalizability of the approach, and in line with previous work [14, 15, 28], we use the default configuration, as provided by scikit [450] to train the classification models before applying our post-processing procedure.

Algorithm 1 Post-processing Procedure of a trained classification model clf

```
1:  $acc, fair \Leftarrow fitness(clf)$
2:  while terminal condition not met do
3:      $clf \Leftarrow mutate(clf)$
4:      $acc', fair' \Leftarrow fitness(clf)$
5:      if $(acc', fair') \succ (acc, fair)$ then
6:          $acc \Leftarrow acc'$
7:          $fair \Leftarrow fair'$
8:      else
9:          $clf \Leftarrow undo\_mutation(clf)$
10:     end if
11: end while
```

5.2.2 Logistic Regression

Representation. Logistic regression (LR) is a linear classifier that can be used for binary classification. Given training data, LR determines the best weights for its coefficients. Below, we illustrate the computation of the LR prediction with four tuneable weights (b_0, b_1, b_2, b_3). At first, Equation 5.1 presents the computation of predictions with a regular linear regression classifier. To make a prediction, LR uses this the *Linear* prediction in a sigmoid function (Equation 5.2):

$$Linear(x_1,x_2,x_3) = b_0 + b_1x_1 + b_2x_2 + b_3x_3 \tag{5.1}$$

$$P(Y) = \frac{1}{1+e^{-Y}} \tag{5.2}$$

This *prediction* function determines the binary label of a 3-dimensional (x_1,x_2,x_3) input. In a binary classification scenario, we treat predictions ≥ 0.5 as labels 1, and 0 otherwise.

This shows that the binary classification is determined by n variables $(b_0 \dots b_{n-1})$. To represent an LR model, we store the n coefficients in an n-dimensional vector.

Mutation. Given that an LR classification model can be represented by one-dimensional vector, we mutate single vector elements to create mutated variants of the model. In particular, we pick an element at random and change its value within a range of $\{-10\%, 10\%\}$.

5.2.3 Decision Tree

Representation. Decision Trees (DT) are classification models that solve the classification process by creating tree-like solutions, which create leaves and branches based on features of the training data. We are interested in binary DTs. In binary DTs, every interior node (i.e., all nodes except for leaves) have exactly two child nodes (left and right).

Mutation. We use pruning as a means to mutate DTs. The pruning process deletes all the children of an interior node, transforming it into a leaf node, and has shown to improve the accuracy of DT classification in previous work [451, 452, 453]. In particular, we pick an interior node i at random and treat it as a leaf node by removing all subjacent child nodes. We choose to use pruning, instead of leaf relabeling, because preliminary experiments showed that pruning outperforms leaf relabeling (i.e., Kamiran et al. [42] used leaf relabeling in combination with an in-processing method but not in isolation).

5.3 Experimental Setup

In this section, we describe the experimental design we carry out to assess our search-based bias repair method for binary classification models (i.e., Logistic Regression and Decision Trees). We first introduce the research questions, followed by the subjects and the experimental procedure used to answer these questions.

5.3.1 Research Questions

Our evaluation aims to answer the following research questions:

RQ1: To what extent can the proposed search-based approach be used to improve both, accuracy and fairness, of classification models?

To answer this question, we apply our post-processing approach to LR and DTs (Section 5.2) on four datasets with a total of six protected attributes (Section 5.3.2).

The search procedure is guided by accuracy and each of the three fairness metrics (SPD, AOD, EOD) separately. Therefore, for each classification model, we perform 3 (fairness metrics) x 6 (datasets) = 18 experiments. For each of the fairness metrics, we mutate the classification models and measure changes in accuracy and the particular fairness metric used to guide the search (e.g., we post-process LR based on accuracy and SPD). We then determine whether the improvement in accuracy and fairness (as explained in Section 5.2) achieved by mutating the classification models are statistically significant, in comparison to the performance of the default classification model.

Furthermore, we compare optimization results from post-processing with existing bias mitigation methods:

RQ2: How does the proposed search-based approach compare to existing bias mitigation methods?

To answer this question, we benchmark our approach against three existing and widely-used post-processing methods (Section 5.3.3), provided by the AIF360 framework [72]. In particular, we applied the existing post-processing methods to LR and DTs on the same set of problems (four datasets) to compare their fairness-accuracy trade-off with the one achieved by our proposed approach.

While the objectives considered during the optimization procedure are im-

Table 5.1: Datasets used in our empirical study

Dataset	Size	Attributes	Favourable Label	Majority Label	Protected	Privileged - Unprivileged
Adult	48,842	14	1 (income >50k)	0 (75%)	Sex Race	Male - female White - non white
COMPAS	7,214	28	0 (No recid)	0 (54%)	Sex Race	Female - male Caucasian - not Caucasian
Bank	41,188	20	1 (yes)	0 (87%)	Age	≥ 25 - < 25
MEPS19	15,830	138	1 ($\geq$ 10 visits)	0 (83%)	Race	White - non-white

proved, this has shown to carry detrimental effects on other objectives [28, 454]. Therefore, we determine the impact optimization for one fairness metric has on the other two fairness metrics, which have not been considered during the optimization procedure:

RQ3: What is the impact of post-processing guided by a single fairness metric on other fairness metrics?

To answer this question, we apply our post-processing method on LR and DTs. While optimizing for each of the three fairness metrics, we measure changes of the other two. We are then able to compare the fairness metrics before and after the optimization process, and visualize changes using boxplots. Moreover, we can determine whether there are statistically significant changes to “untouched” fairness metrics, which are not optimized for.

5.3.2 Datasets

We perform our experiments on four real-world datasets used in previous software fairness work [26, 28] with a total of six protected attributes: Adult, COMPAS, Bank, MEPS19. A detailed explanation for each datasets can be found in Section 2.4.

In Table 5.1, we provide the following information about the four datasets: number of rows and features, the favourable label and majority class. In addition, we list the protected attributes for each dataset (as provided by the AIF360 framework [72]), which are investigated in our experiments, and the respective privileged and unprivileged groups for each protected attribute.

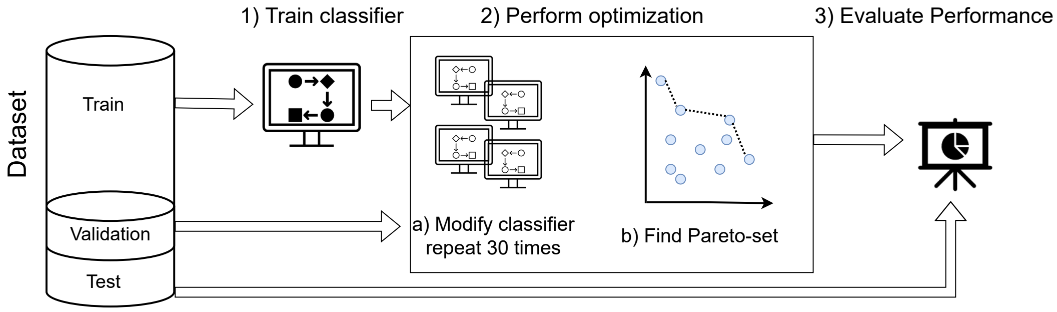


Figure 5.1: Empirical evaluation of a single data split.

5.3.3 Benchmark Bias Mitigation Methods

As our proposed method belongs to the category of post-processing methods, we compare it with all the state-of-art post-processing bias mitigation methods made publicly available in the AIF360 framework [72]:

- Reject Option Classification (ROC) [14, 15];
- Equalized odds (EO) [41];
- Calibrated Equalized Odds (CO) [40].

AIF360 [72] provides ROC and CO with the choice of three different fairness metrics to guide the bias mitigation procedure. ROC can be applied with SPD, AOD, and EOD. CO can be applied with False Negative rate (FNR), False Positive Rate (FPR), and a “weighed” combination of both. We apply both, ROC and CO, with each of the available fairness metrics. EO does not provide choices for fairness metrics to users.

5.3.4 Validation and Evaluation Criteria

To validate the effectiveness of our post-processing approach to improve accuracy and fairness of classification models, we apply it to LR and DT. Since our optimization approach applies random mutations, we expect variation in the results. Figure 5.1 illustrates the empirical evaluation procedure of our method for a single datasplit. At first, we split the data in three sets: training (70%), validation (15%), test (15%). To mitigate variation, we apply each bias mitigation method, including our newly proposed approach on 50 different data splits.

The training data is used to create a classifier which we can post-process. Once a classifier is trained (i.e., Logistic Regression or Decision Tree), we apply our optimization approach 30 times (Step 2). To then determine the performance (accuracy and fairness) of our approach on a single data split, we compute the Pareto-optimal set[2] based on the performance on the validation set. Once we obtain the Pareto-set of optimized classification models based on their performance on the validation set, we average their performance on the test set. Performance on the test set (i.e., accuracy and fairness) is used to compare different bias mitigation methods and determine their effectiveness. Each run of our optimization approach is limited to $2,500$ iterations (terminal condition, Algorithm 1). The existing post-processing methods are deterministic, and therefore applied only once for each data split.

In order to assess the effectiveness of our approach (RQ1) and compare it with existing bias mitigation methods (RQ2), we consider both summary statistics (i.e., average accuracy and fairness), statistical significance tests and effect size measures, and Pareto-optimality. Furthermore, we use boxplots to visualize the impact of optimizing accuracy and one fairness metric on the other two fairness metrics (RQ3).

Pareto-optimality states that a solution a is not worse in all objectives than another solution b and better in at least one [448]. We use Pareto-optimality to both measure how often our approach dominates the default classification model or is Pareto-optimal, and to plot the set of solutions found to be non-dominated (and therefore equally viable) with respect to the state-of-the-art (RQs 1-2). In the case where there are two objectives, such as ours, this leads to a two dimensional Pareto surface.

To determine whether the differences in the results achieved by all approaches are statistical significant, we use the Wilcoxon Signed-Rank test, which is a non-parametric test that makes no assumptions about underlying data distribution [455]. We set the confidence limit, α, at 0.05 and applied the Bonferroni correction for

[2] This is the set of solutions that are non-dominated to each other but are superior to the rest of solutions in the search space. In other words each solution of the Pareto-set includes at least one objective inferior to another solution in that Pareto-set, although both solutions are superior to others in the rest of the search space with respect to all objectives.

multiple hypotheses testing (α/K, where K is the number of hypotheses). This correction is the most conservative of all corrections and its usage allows us to avoid the risk of Type I errors (i.e., incorrectly rejecting the Null Hypothesis and claiming predictability without strong evidence). In particular, depending on the RQ, we test the following null hypothesis:

(RQ1) H_0: *The fairness and accuracy achieved by approach$_x$ is not improved with respect to the default classification model.* The alternative hypothesis is as follows: H_1: *The fairness and accuracy achieved by approach$_x$ improves with respect to the default classification model.* In this context, "improved" means that the accuracy is increased and fairness metric values are decreased (e.g., a SPD of 0 indicates that there is no unequal treatment of privileged and unprivileged groups).

(RQ3) H_0: *Optimizing for accuracy and fairness metric m_1 does not improve fairness metric m_2 with respect to the default classification model.* The alternative hypothesis is as follows: H_1: *Optimizing for accuracy and fairness metric m_1 improves fairness metric m_2 with respect to the default classification model.* For this RQ, we summarize the results of the Wilcoxon tests by counting the number of win-tie-loss as follows: p–value<0.01 (win), p–value>0.99 (loss), and 0.01≤ p–value ≥0.99 (tie), as done in previous work [456,457,458,459].

In addition to evaluating statistical significance, we measure the effect size based on the Vargha and Delaney's $\hat{A}_{12}$ non-parametric measure [460], which does not require that the data is normally distributed [461]. The $\hat{A}_{12}$ measure compares an algorithm A with another algorithm B, to determine the probability that A performs better than B with respect to a performance measure M:

$$\hat{A}_{12} = (R_1/m - (m+1)/2)/n \tag{5.3}$$

In this formula, m and n represent the number of observations made with algorithm A and B respectively; R_1 denotes the rank sum of observations made with A. If A performs better than B, $\hat{A}_{12}$ can display one of the following effect sizes: $\hat{A}_{12} \geq 0.72$ (large), $0.64 < \hat{A}_{12} < 0.72$ (medium), $0.56 < \hat{A}_{12} < 0.64$ (small), although these thresholds are not definitive [462].

5.3.5 Threats to Validity

The threats to *internal* validity lie in the confidence that our experimental results are trustworthy and correct. To alleviate this threat, we applied our post-processing method and existing bias mitigation methods 50 times, under different train/validation/test splits. This allowed us to apply statistical significance tests to verify our results and findings. We have used traditional measures used in the software fairness literature to assess ML accuracy, while we recognize alternative measures could be used to take into account data imbalance [463].

Threats to *external* validity: the generalizability of our results, are primarily concerned with the investigated datasets, approaches and metrics. While we believe using more data will increase the generalizability of our results, the majority of publications we surveyed investigated less than four datasets, and we have included herein all datasets publicly available which have been previously used in the literature to solve the same problem addressed in this chapter. Furthermore, we have successfully applied our post-processing method on two inherently different classification models (Logistic Regression, Decision Trees), which strengthens the confidence that our approach could be applied to other classifiers. We have also explored all state-of-the-art fairness metrics and post-processing debiasing methods available from the AIF360 framework [72] (version 0.3.0), which is publicly available, to strength the generalizability and reproducibility of our work.

Threats to *construct* validity: to further support the applicability and generalizability of our approach and the replication and extension of our work, we make our scripts and results publicly available [447].

5.4 Results

This section presents the results of our experiments to answer the research questions explained in Section 5.3.1.

5.4.1 RQ1: Fairness-Accuracy Improvement

In the first research question, we investigate whether our post-processing approach is able to improve both fairness and accuracy when applied to binary classification

Table 5.2: RQ1-Logistic Regression: Average accuracy and fairness of non-dominated solutions over 50 different data splits (i.e., for each data split, we select the non-dominated solutions and average their performance on the test set). **Bold** values indicate improvements over the default classification model. The p-value of the Wilcoxon Signed-Rank test comparing each approach with the default Logistic Regression model, is given in brackets for each metric. Colors are used to show the effect size (large , medium , small).

		Adult		Compas		Bank	Meps19
		Sex	Race	Sex	Race	Age	Race
Accuracy	LR$_{default}$	0.833	0.833	0.677	0.677	0.899	0.838
	LR$_{SPD}$	**0.845 (0.00)**	**0.845 (0.00)**	0.676 (0.22)	0.675 (0.31)	**0.900 (0.01)**	0.835 (0.00)
	LR$_{AOD}$	**0.846 (0.00)**	**0.845 (0.00)**	0.675 (0.29)	0.675 (0.31)	**0.900 (0.06)**	0.834 (0.00)
	LR$_{EOD}$	**0.846 (0.00)**	**0.845 (0.00)**	0.675 (0.20)	0.676 (0.72)	**0.900 (0.05)**	0.834 (0.00)
SPD	LR$_{default}$	0.191	0.034	0.279	0.173	0.074	0.123
	LR$_{SPD}$	**0.171 (0.00)**	0.086 (0.00)	**0.199 (0.00)**	**0.157 (0.00)**	**0.074 (0.59)**	**0.107 (0.00)**
AOD	LR$_{default}$	0.120	0.044	0.254	0.150	0.051	0.125
	LR$_{AOD}$	**0.083 (0.00)**	**0.041 (0.42)**	**0.178 (0.00)**	**0.133 (0.00)**	0.054 (0.20)	**0.111 (0.00)**
EOD	LR$_{default}$	0.150	0.078	0.194	0.094	0.076	0.205
	LR$_{EOD}$	**0.088 (0.00)**	**0.049 (0.01)**	**0.115 (0.00)**	**0.079 (0.00)**	0.082 (0.33)	**0.175 (0.00)**

Table 5.3: RQ1-Decision Tree: Average accuracy and fairness of non-dominated solutions over 50 different data splits (i.e., for each data split, we select the non-dominated solutions and average their performance on the test set). **Bold** values indicate improvements over the default classification model. The p-value of the Wilcoxon Signed-Rank test comparing each approach with the default Decision Tree model, is given in brackets for each metric. Colors are used to show the effect size (large , medium , small).

		Adult		Compas		Bank	Meps19
		Sex	Race	Sex	Race	Age	Race
Accuracy	DT$_{default}$	0.817	0.817	0.622	0.622	0.877	0.760
	DT$_{SPD}$	**0.836 (0.00)**	**0.841 (0.00)**	**0.645 (0.00)**	**0.638 (0.00)**	**0.892 (0.00)**	**0.798 (0.00)**
	DT$_{AOD}$	**0.838 (0.00)**	**0.838 (0.00)**	**0.648 (0.00)**	**0.640 (0.00)**	**0.889 (0.00)**	**0.798 (0.00)**
	DT$_{EOD}$	**0.832 (0.00)**	**0.831 (0.00)**	**0.646 (0.00)**	**0.642 (0.00)**	**0.887 (0.00)**	**0.791 (0.00)**
SPD	DT$_{default}$	0.180	0.085	0.129	0.114	0.107	0.128
	DT$_{SPD}$	**0.110 (0.00)**	**0.060 (0.00)**	**0.083 (0.00)**	**0.091 (0.00)**	**0.088 (0.00)**	**0.047 (0.00)**
AOD	DT$_{default}$	0.073	0.035	0.107	0.098	0.068	0.091
	DT$_{AOD}$	**0.032 (0.00)**	**0.028 (0.00)**	**0.075 (0.00)**	**0.081 (0.00)**	**0.057 (0.00)**	**0.036 (0.00)**
EOD	DT$_{default}$	0.056	0.034	0.089	0.064	0.077	0.093
	DT$_{EOD}$	**0.041 (0.00)**	0.034 (0.70)	**0.057 (0.00)**	**0.062 (0.81)**	0.081 (0.61)	**0.022 (0.00)**

models (namely LR and DT). The baseline considered is the default classification model. We apply our approach on four datasets, as outlined in Section 5.3.4. In total, we apply post-processing with three different configurations, to optimize for accuracy and one of the three fairness metric at a time. We will call those configurations DT_{SPD}, DT_{AOD}, DT_{EOD}, LR_{SPD}, LR_{SPD}, LR_{SPD} to determine the classification model and the fairness metric considered during optimization. These configurations are applied to four datasets on 50 train/validation/test splits and repeated 30 times. Table 5.2 and Table 5.3 show these results for Logistic Regression and Decision Trees respectively. These tables show the results of the default classification model and the three optimization configurations.

We can see that our post-processing approach is able to improve the accuracy of the two classification models (LR and DT) in 27 out of 36 cases. In the majority of the cases the accuracy of LR is statistically significant better (9 out of 18 cases) or comparable (2 out of 18 cases) with respect to the default model, while in the remaining cases (6 out of 18) it is reduced although no statistical significant difference is observed. All the 18 out of 18 cases improve the accuracy of DT, all of which are statistically significant with large effect sizes.

When investigating the impact of our post-processing approach on each of the three fairness metrics (i.e., mutation is applied if the particular fairness metric and accuracy are improved), we compare the fairness of the default classification model with the configuration to optimize for that particular metric (e.g., we compare the SPD of the default LR with the SPD achieved by LR_{SPD}). Therefore, instead of 18 cases for LR and DT, we have six comparisons for each metric.

For each of the three fairness metrics (SPD, AOD, EOD) our post-processing approach is able to improve fairness on 5 out of 6 datasets on LR. LR_{SPD} is not able to achieve SPD improvements on the Adult dataset (protected attribute = "race"), LR_{AOD} and LR_{EOD} are not able to achieve fairness improvements on the Bank dataset. Among the 15 out of 18 cases that improve fairness on LR, 12 are statistically significant, with six of those having large effect sizes. Furthermore, it can be noted that the instances where our approach is not able to improve fairness, already have a low bias score. According to the online tool of the AIF360 framework [72], values ≤ 0.1 can be seen as fair, when investigating SPD, AOD and EOD.[3] Applied to DTs, our post-processing approach improves fairness for 16 out of 18 cases. In particular, in 6 out of 6 cases DT_{SPD} and DT_{AOD} achieve statistically significant fairness improvements on their respective fairness metric. In 3 out of 6 cases, DT_{EOD} achieves statistically significant improvements. In the remaining two cases (i.e., EOD on the Adult-race and Bank-Age datasets), our approach is not able to significantly improve fairness, likely because the default model already shows a low bias (≤ 0.1).

Overall, the three post-processing configurations achieve improvements in both accuracy and fairness in 22 out of 36 cases, and improvements in at least one of the two (i.e., either accuracy and fairness) in the remaining 14 out of 36 cases. Notably, our post-processing approach improves accuracy and fairness of DTs in 16 out of 18 cases.

In addition to comparing the average performance of our optimization approach for each data-split (i.e., we average accuracy and fairness of all solutions in the Pareto-front), we perform a comparison of each solution in the Pareto-front with the default classification model. Table 5.4 shows the results. For each combination of datasets and metric optimized by our approach, we compute the percentage of solutions that: dominate the default model, are Pareto-optimal, are dominated by the default model. This comparison (e.g., do solutions in the Pareto-front dominate the default classification model?) is performed for each data-split and weighted accordingly, such that each data-split has the same contribution to the results (e.g., a data-split with 10 solutions in the Pareto-front is treated equally as a data-split with 2 solutions in the Pareto-front). Our post-processing methods applied on Logistic Regression achieves comparable or better performance than the default model in 91% of the cases across all datasets studied, and, specifically, it dominates the default model in 38% of the cases and is dominated in only 9% of the cases. This shows that our approach is a useful tool for optimizing LR models (i.e., develop-

[3] https://aif360.mybluemix.net/

Table 5.4: RQ1: Comparison of each individual run of our approach (30 runs over 50 datasplits) against the default classification model. For each dataset and metric, we measure the percentage of runs that: dominate the default model - are Pareto-optimal - are dominated by the default model.

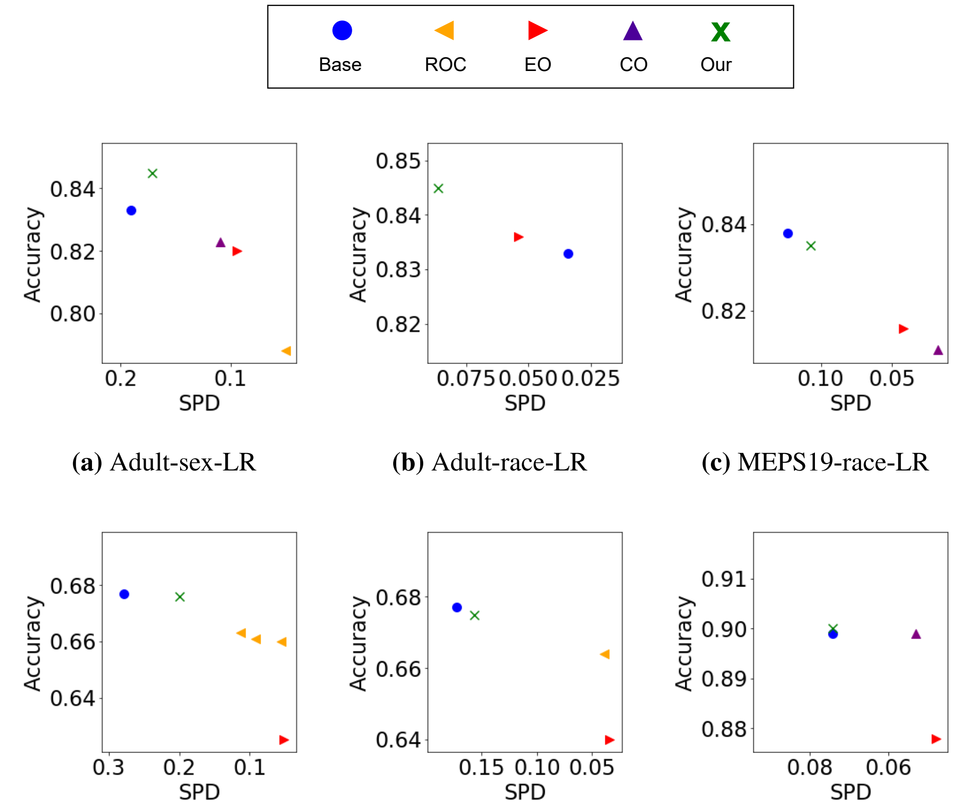
		Adult		Compas		Bank	Meps19	
		Sex	Race	Sex	Race	Age	Race	Total
LR	SPD	59-41-0	0-98-2	36-57-7	38-47-16	37-50-14	25-68-8	32-60-8
	AOD	65-34-1	50-50-0	36-54-10	37-48-16	26-50-24	15-65-19	38-50-12
	EOD	71-29-0	61-39-0	37-58-6	41-44-15	31-49-19	17-72-11	43-48-8
Total		65-35-0	37-62-1	36-56-7	39-46-16	31-50-19	19-68-13	38-53-9
DT	SPD	100-0-0	100-0-0	91-9-0	76-23-2	69-31-0	99-1-0	89-11-0
	AOD	100-1-0	71-29-0	85-14-1	69-31-1	63-37-0	95-5-0	80-19-0
	EOD	78-22-0	54-46-0	78-20-2	47-52-1	43-57-0	89-11-0	65-35-0
Total		92-8-0	75-25-0	85-15-1	64-35-1	58-42-0	94-6-0	78-22-0

Table 5.5: RQ2: Frequency of bias mitigation methods in the Pareto-front. Each combination of bias mitigation method and fairness metric is evaluated on six datasets.

	Logistic Regression				Decision Tree			
	Our	CO	ROC	EO	Our	CO	ROC	EO
SPD	6	3	3	6	6	0	2	0
AOD	6	2	2	6	6	0	2	0
EOD	6	2	4	5	6	0	2	1
Total	18/18	7/18	9/18	17/18	18/18	0/18	1/18	1/18

ers are either able to choose a strictly better model, or models with competitive fairness-accuracy trade-offs). When we apply our approach to DTs, we observe an even higher performance improvement: It dominates the default DT models in 78% of the cases and not dominated in the remaining cases.

Answer to RQ1: In 22 out of 36 cases (61%), our search-based approach is able to improve both, fairness and accuracy of Logistic Regression and Decision Trees with respect to the default model when considering all datasets and fairness metrics. Notably, this happens in 16 out of 18 cases when applying our optimization approach to Decision Trees, with 15 of these cases achieving statistically significant improvements with large and medium effect sizes in the vast majority of case (14 out of 15).



(a) Adult-sex-LR

(b) Adult-race-LR

(c) MEPS19-race-LR

(d) COMPAS-sex-LR

(e) COMPAS-race-LR

(f) BANK-age-LR

Figure 5.2: RQ2: Comparison of our proposed approach against existing bias mitigation methods and default classification models based on Pareto-optimality. The figure shows six exemplary comparisons for LR and SPD.

5.4.2 RQ2: Comparison to Existing Bias Mitigation Methods

To answer RQ2, we compare our post-processing method against three existing post-processing bias mitigation methods (Section 5.3.3) applied to LR and DT on the same datasets (Adult, COMPAS, Bank, MEPS19) by using identical train/validation/test splits, as described in Section 5.3. The mean performance of these methods over 50 data splits, and of our post-processing method, are shown in Figure 5.2. While this Figure only includes six cases for LR and measuring SPD, the remaining results for other metrics and DTs are available in our online appendix [447]. In each sub-figure, we show the performance of every non-dominated bias mitigation method on the respective dataset and fairness metric. A summary on how often each bias mitigation method is part of the Pareto-front is provided in Table 5.5.

When comparing the accuracy of classification models achieved after applying our post-processing method against the existing bias mitigation methods, we observe that all of the existing bias mitigation methods have a lower accuracy. Moreover, all of the existing bias mitigation methods reduce the accuracy of the default classification model, thereby conforming to the fairness-accuracy trade-off. On the other end, our approach, which takes into account accuracy in the bias mitigation process, is always able to generate a widely applicable solution (i.e., our approach always produces at least a solution belonging to each of 36 Pareto-fronts, and therefore is never dominated by any of the existing methods).

We can observe a difference in performance of our approach when applied to LR and DT. While our approach, applied to LR, is able to outperform some of the existing bias mitigation methods on the three fairness metrics (CO and ROC), it is only able to dominate EO in 1 out of 18 cases (Bank-age EOD). In the remaining 17 cases, EO has a lower accuracy than our approach while improving fairness to a higher degree. On the other end, when applying our post-processing approach to DTs, it not only produces solutions that dominate the default classification model (as seen in RQ1), but also all investigated bias mitigation methods in 12 out of 18 cases. Furthermore, for DT, our approach outperforms existing bias mitigation methods on the three fairness metrics, in addition to achieving the highest accuracy. In particular, our approach achieves the lowest bias on all three fairness metrics for the Adult, Bank and MEPS19 datasets. Only ROC is able to achieve a lower level of bias for the COMPAS dataset in 6 out of 6 cases, and EO in 1 out of 6 cases. This may be due to the fact that COMPAS is the smallest of the datasets we investigate herein.

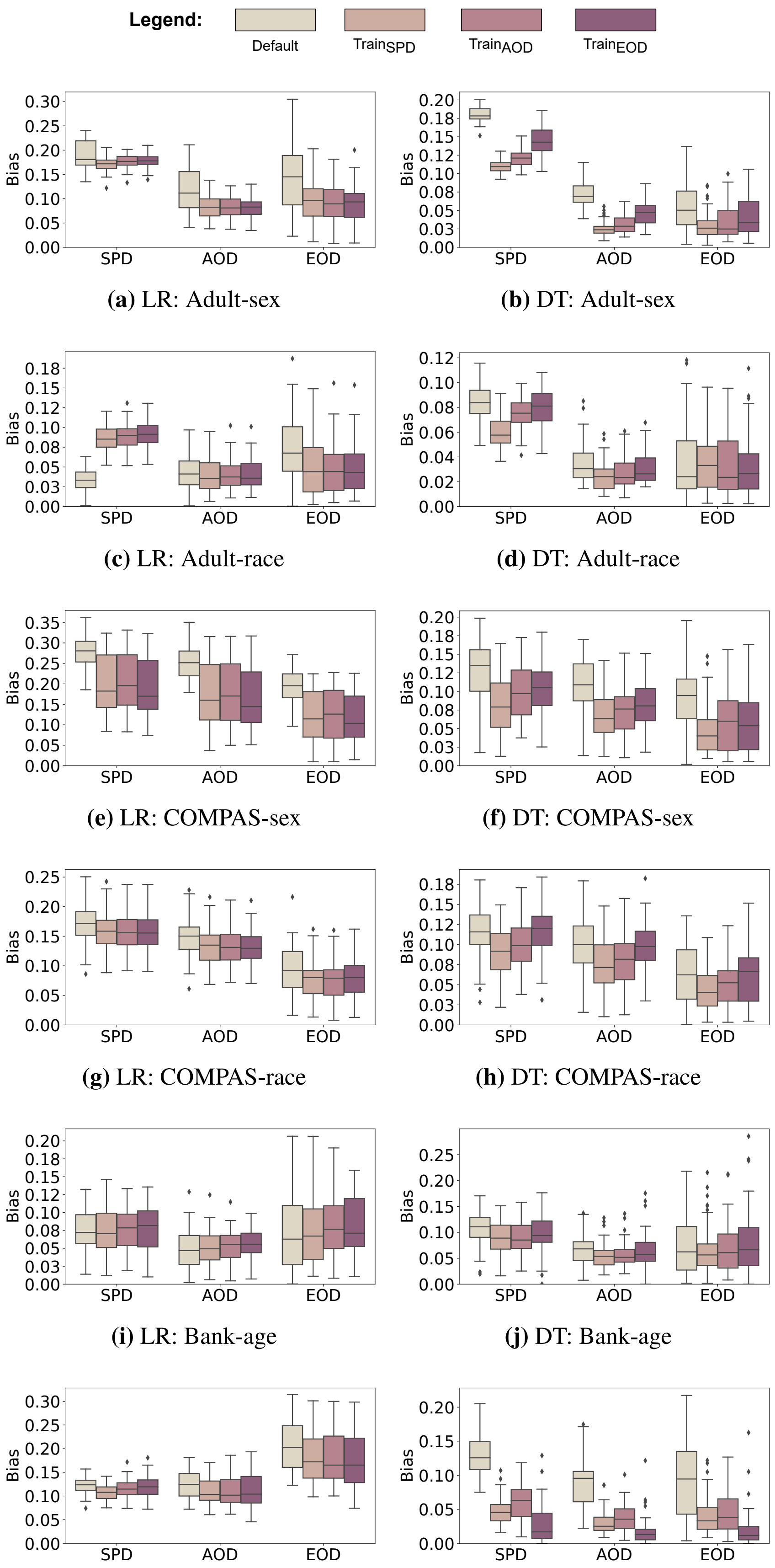
Answer to RQ2: Our approach provides Pareto-optimal solutions when applied to both Decision Trees and Logistic Regression for each of the datasets investigated in our study. In particular, it achieves the highest accuracy with respect to the existing bias mitigation methods in 100% of the cases and the highest fairness in 33% of the cases. Notably, our approach provides the best performance when applied to Decision Trees, as in this case it generates solutions that strictly dominate those provided by the existing bias mitigation methods in 12 out of 18 cases (i.e., it achieves both higher accuracy and lower bias), and achieves a higher accuracy in the remaining 6 out of 18 cases.

5.4.3 RQ3: Impact on Fairness Metrics

In RQ3, we investigate the impact of optimizing for one fairness metric on the other two (e.g., if we optimize for accuracy and AOD, how do SPD and EOD change?). Therefore, we apply the three configurations of our post-processing approach on the four datasets and measure every kind of fairness metric at the end of the optimization procedure. In accordance with RQ1 and RQ2, we investigate the performance over 50 different train/validation/test splits.

Figure 5.3 shows the results of the optimization results. For each dataset, we use boxplots to show the default performance of the classification model, as well as the performance after optimization with each of the three configurations. Thereby, three colors represent optimization with one of the fairness metrics, and one color represents the fairness of the default classification model.

Given the results, we can see that the fairness achieved by an optimized, post-processed classification model behaves similarly, independent of the fairness metric used for optimization. For example, this can be seen on the Adult-sex dataset for LR and DT. Regardless of the fairness metric considered during optimization, the average AOD of all three configurations is better than the default classification model. Such a behaviour (all three optimization configurations achieve improvements on a fairness metric) happens in 28 out of 36 cases. There is one case (Adult-race for LR) in which none of the three search configurations achieve improvements on SPD



(a) LR: Adult-sex

(b) DT: Adult-sex

(c) LR: Adult-race

(d) DT: Adult-race

(e) LR: COMPAS-sex

(f) DT: COMPAS-sex

(g) LR: COMPAS-race

(h) DT: COMPAS-race

(i) LR: Bank-age

(j) DT: Bank-age

(k) LR: MEPS19-race

(l) DT: MEPS19-race

Figure 5.3: RQ3: Summary of bias values (the lower the better) achieved by the three different post-processing settings (SPD, AOD, EOD) and the default classification models. Boxplots are grouped based on the fairness metric they measure.

Table 5.6: RQ3: Win-tie-loss summary of the Wilcoxon tests when optimizing for one fairness metric and measuring the other two (e.g., use SPD during optimization and test on EOD) in comparison to the default classification model.

Method	SPD		AOD		EOD		Total
Test	AOD	EOD	SPD	EOD	SPD	AOD	
LR	4-2-0	4-1-1	4-2-0	5-1-0	3-3-0	4-2-0	24-11-1
DT	6-0-0	3-2-1	6-0-0	3-3-0	3-2-1	3-3-0	24-10-2

(neither LR_{SPD}, LR_{AOD} nor LR_{EOD}).

In the remaining 7 out of 36 cases, there are differences when using different optimization configurations. One example for this is the Bank-age datasets for LR. Only LR_{SPD} achieves improvements over the default LR model in SPD, AOD and EOD. LR_{AOD} and LR_{EOD} are not able to improve any fairness metric (neither SPD, AOD or EOD).

To evaluate the overall level of bias mitigation achieved by optimization on a different fairness metric, we summarize the statistical significance differences we found over the four datasets in Table 5.6. In particular, we investigate whether significant improvements over the default classification models are achieved (*win*), whether no significant differences can be found (*tie*), or whether the default classification model has a statistically significant lower bias than the optimized model (*loss*). Combining the results for LR and DT, there are 48 wins, 21 ties and 3 losses. This indicates, that while our post-processing approach optimizes for one fairness metric, it can positively effect other metrics as well.

Answer to RQ3: Based on the three investigated fairness metrics (SPD, AOD, EOD), fairness improvements are achieved independently of the metric used during optimization. In 78% of the cases (28 out of 36), fairness metrics are improved by all three configurations (e.g., SPD on the Adult-sex dataset is improved by LR_{SPD}, LR_{AOD}, LR_{EOD}). We do not observe any dramatic detrimental effect, as in 96% of the cases (69 out of 72) there is no performance deterioration in "untouched" fairness metrics, which are not optimized for. Among those, in 67% of the cases (48 out of 72), our approach even leads to statistically significant improvements.

5.5 Conclusions

We proposed a novel search-based approach to mutate classification models in a post-processing stage, in order to simultaneously repair fairness and accuracy issues. This approach differentiates itself from existing bias mitigation methods, which conform to the fairness-accuracy trade-off (i.e., repair fairness issues come at a cost of a reduced accuracy). We performed a large scale empirical study to evaluate our approach with two popular classifiers (Logistic Regression and Decision Trees) on four widely used datasets and three fairness metrics, publicly available in the popular IBM AIF360 framework [72]).

We found that our approach is able to simultaneously improve accuracy and fairness of both classification models in 61% of the cases. Our approach is particularly effective for Decision Trees, where we achieve statistically significant improvement on both accuracy and fairness in 81.1% of the cases. Moreover, we achieved improvements without detrimental effect on other fairness metrics that are not considered during optimization.

The comparison with three existing post-processing bias mitigation methods showed that none of these methods is able to achieve an accuracy as high as our method in any of the datasets. Furthermore, our approach is able to outperform existing post-processing methods in both accuracy and fairness in 12/18 cases for

Decision Trees.

These findings show not only the feasibility but also the effectiveness of our approach with respect to existing bias mitigation methods. Software engineers would benefit to have this tool at their disposal when developing fair software, as it allows them to find good trade-offs between competing objectives rather than proposing a solution which often sacrifices accuracy, as done in previous work. According to their needs, engineers can choose the solution that better conforms to their fairness and accuracy constraints.

6

Age Attribute and Fairness

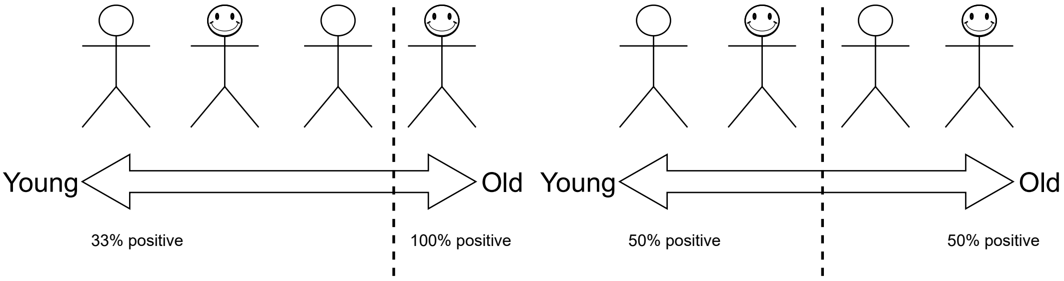
Forty is the old age of youth; fifty is the youth of old age.

– Victor Hugo

Among the sensitive attributes studied in software fairness literature, race and gender are categorical features, that are used to divide the population into privileged and unprivileged groups (e.g., male - female, white - non-white) [17]. The protected attribute “age” is continuous and needs to be addressed differently.[1] While there exist methods for dealing with continuous attributes (e.g., pairwise comparisons [465] and correlations [219, 239]), we focus on treating protected attributes as binary attributes, in accordance with prior works [28, 95, 135]. To divide the population into two groups (i.e., young - old) one needs to select an age threshold which divides the population as follows: everyone older than the threshold is “old”; everyone of the same age as threshold or younger is “young”.

Figure 6.1 illustrates the impact of different age thresholds on fairness, when treating different populations groups, and the risks of selecting unsuitable thresholds. In Figure 6.1 (a) a high threshold is applied, dividing the population in three young individuals and one old individual. By doing so, the “old” population group, on average, receives a more favourable treatment (represented by smiling faces) than the “young” group. Using instead the age threshold shown in Figure 6.1 (b), we can observe that an equal treatment of the two groups is possible, when an ad-

[1] As pointed out by Jacobs and Wallach [464], protected attributes, such as race and gender, are contested constructs.



(a) High age threshold.

(b) Medium age threshold.

Figure 6.1: Example of the fairness of two different age thresholds. Smiling faces represent a favourable treatment.

equate age threshold is chosen. While this is a simplified example, it signifies the importance of selecting sensible age thresholds when investigating the fairness of ML software.

Currently, there exists no systematic study focusing on the problem on how to approach the choice of sensible age thresholds when faced with new datasets, and what the impact of age thresholds has on: 1) the dataset; 2) the classification models that are trained on the dataset. Therefore, we aim to provide guidelines on how to approach the protected attribute "age" from a computational point of view. Nonetheless, if regulations exist, the final choice of acceptable age thresholds is to be done by law makers and domain experts [97, 466].

In summary, the main contributions of this chapter are:

- a general approach on how to choose age thresholds;
- an empirical evaluation on bias in classification models with respect to age thresholds on two datasets.

The rest of this chapter is organized as follows. Section 6.1 presents related work on software fairness research, including types of bias and an overview of methods to combat bias in classification models. The experimental design, fairness metrics and datasets are outlined in Section 6.2. Experiments and results are presented in Section 6.3 while Section 6.4 concludes.

6.1 Related Work

To foster a better understanding of fairness issues and increase the usability of fairness techniques, frameworks, such as AIF Fairness 360 (AIF360) [72] and Fairlearn [441], have been created. Among others, these provide bias mitigation methods, fairness metrics, datasets, and have been frequently used by the research community [2, 28, 135].

Investigations on the effect of datasets on fairness have been carried out by Zhang and Harman [26], and Kamiran and Calders [95]. Zhang and Harman [26] investigated the influence of training data on the fairness of classification models. Particularly, rich feature sets have the ability to improve the fairness of ML models. Kamiran and Calders [95] proposed a pre-processing method called “massaging” with the goal to create an unbiased datasets with the least intrusive modifications before training classification models. Their investigation covered the German dataset (see Section 6.2.3), for which they chose an age threshold of 25, as a high degree of bias was observed. This age threshold is incorporated in the AIF360 framework [72]. Nonetheless, other thresholds have been used as well, such as 30 and 45 for other datasets [98, 414], and 50 for the German dataset [98].

While Kamiran and Calders [95] focus lay on proposing a novel bias mitigation method on the German dataset with the protected attribute “age”, we focus our investigation entirely on the choice of age thresholds for multiple datasets. In particular, we do not only consider the German dataset, but a second dataset (Bank), which uses the same age threshold of 25 (according to the AIF360 framework [72]). In addition to measuring the bias and comparing the usability of different age thresholds (e.g., is an age threshold of 25 suitable for the Bank dataset?), we measure the impact of age thresholds on the proceeding bias of three classification models (Logistic Regression, Decision Tree, Support Vector Machine).

6.2 Empirical Study Design

In this section, we describe the design of the analysis we carry out to investigate the impact age thresholds have on the fairness of datasets and classification models. We first introduce the research questions, followed by the subjects and the experimental procedure.

6.2.1 Research Questions

To determine the relation of the protected attribute “age” and the resulting bias in classification problems, we first investigate the bias present in datasets:

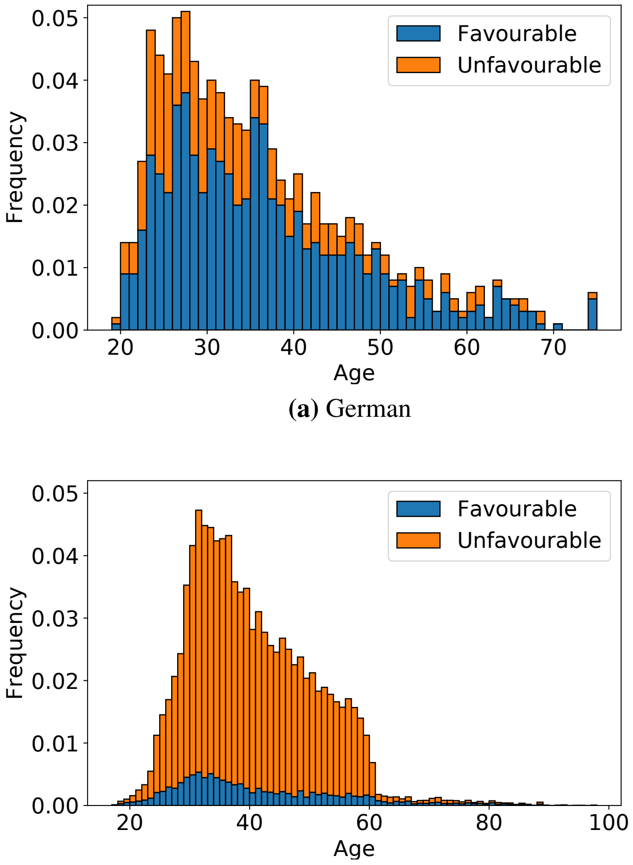
RQ1: What is the impact of age thresholds on the bias in datasets?
To answer this research question, we investigate the dataset fairness of two datasets (German [69] and Bank [70]) according to the dataset fairness metric *Mean Difference* (see Section 2.1). In particular, we evaluate Mean Difference for each possible age threshold for the respective dataset (i.e., the ages present in the dataset). Not only does this allow us to detect the degree of bias that the datasets exhibit, when following different rules to divide the population in to “young” and “old”, but also the direction of bias (i.e., which population group receives a favourable treatment).

After determining the degree of bias with respect to the age threshold within a dataset, we investigate the impact of age thresholds on the bias in classification models:

RQ2. What is the impact of age thresholds on the bias in classification models?
For this purpose, we train three different classification models (Logistic Regression, Decision Trees, Support Vector Machine) on two datasets (German [69] and Bank [70]). According to the experiments in RQ1, we train the classification models for every possible age threshold to measure resulting biases. This allows us to determine the relation of dataset bias and classification bias in two aspects:

- **RQ2.1** What is the impact of dataset bias on the direction of classification bias (e.g., if the dataset bias favours privileged groups, do classification models as well)?

- **RQ2.2** What is the impact of dataset bias on the degree of classification bias (e.g., does a high dataset bias lead to a high classification bias)?



(a) German

(b) Bank

Figure 6.2: Distribution of favourable and unfavourable labels in the German and Bank dataset by age.

6.2.2 Fairness Metrics

For our investigation, we are concerned with the disparate treatment of population groups (privileged and unprivileged). Therefore, we use group fairness metrics [28, 30, 43, 48], to determine the "age" bias in datasets. We investigate four group fairness metrics in total (one dataset metric and three classification metrics). We use Mean Difference to measure dataset bias. SPD, EOD and AOD are used to measure classification bias. See Section 2.1 for definitions of the four metrics.

6.2.3 Datasets

We perform our experiments on two publicly available, real-world datasets, widely studied in the fairness literature [28, 34, 38, 39, 67, 68]: the German, and Bank dataset. While there exist other datasets that have been used for fairness research,

Table 6.1: Dataset Information

Dataset	Size	Features	Favour Label	Majority Label	Priv. - Unpriv.
German	$1,000$	20	1 (good credit)	1 (70%)	> 25 - ≤ 25
Bank	$41,188$	20	1 (yes)	0 (87%)	≥ 25 - < 25

such as the Adult [69] and COMPAS [17] datasets, we only focus on those datasets that are publicly available in the AIF360 framework [72], have *age* as a protected attribute, and use a default threshold to divide the privileged and unprivileged groups.

Table 6.1 provides more information about the two datasets. This includes the size of the dataset, the number of features, the favourable label, and the majority label. The default criteria to form privileged and unprivileged groups from the protected attribute "age" are given.[2] At the time of performing our experiments, individuals with an age > 25 are part of the privileged group in the German datasets, whereas the individuals with an age ≥ 25 are part of the privileged group in the Bank dataset, according to the default settings of the AIF 360 framework [72].

For the two datasets, Figure 6.2 provides histograms to show how many individuals receive favourable and unfavourable outcomes. When comparing the two datasets, we can see that the average age of the Bank dataset is higher than on the German dataset (40 vs. 35.5). Furthermore, the age range within the dataset is larger on the Bank dataset (17-98) in contrast to the German dataset (19-75).

6.2.4 Experimental Configuration

To carry out our experiments, we use the dataset and fairness metric implementations provided by the AIF360 framework [72]. When loading the datasets, the AIF360 framework allows the definition of rules to determine the age threshold which we use to modify the datasets in RQ1 and RQ2.

For RQ2, we use the data investigated in RQ1 to train classification models. In particular, we consider three classification models that have previously been used in fairness research: Logistic Regression (LR) [14, 15, 28, 30, 34, 39, 135], Decision

[2] We use the default parameter from version 0.4.0 or the AIF360 framework, last updated on the fourth of March 2021.

Trees (DT) [14, 15], and Support Vector Machines (SVM) [15, 34, 39, 135]. We implemented each classification model with scikit-learn [450], according to their default configuration.

When training classification models (RQ2), we use random data-splits with a train-test split of 70%-30%. For each age threshold, we adjust the "age" label of the underlying dataset to "young" and "old" before training classification models. To measure the classification bias, we repeat experiments 50 times, with different train-test splits, and average the results [38, 444].

As RQ2.2 considers the degree of bias and not the direction of bias, we compute the absolute bias values. Thereby, bias is minimized at 0 and maximized at 1. Afterwards, we use the Pearson correlation coefficient [467] to determine the correlation between dataset bias and classification bias.

6.3 Empirical Study Results

This section presents the results of our experiments to answer the research questions explained in Section 6.2.1.

6.3.1 RQ1: Dataset Fairness

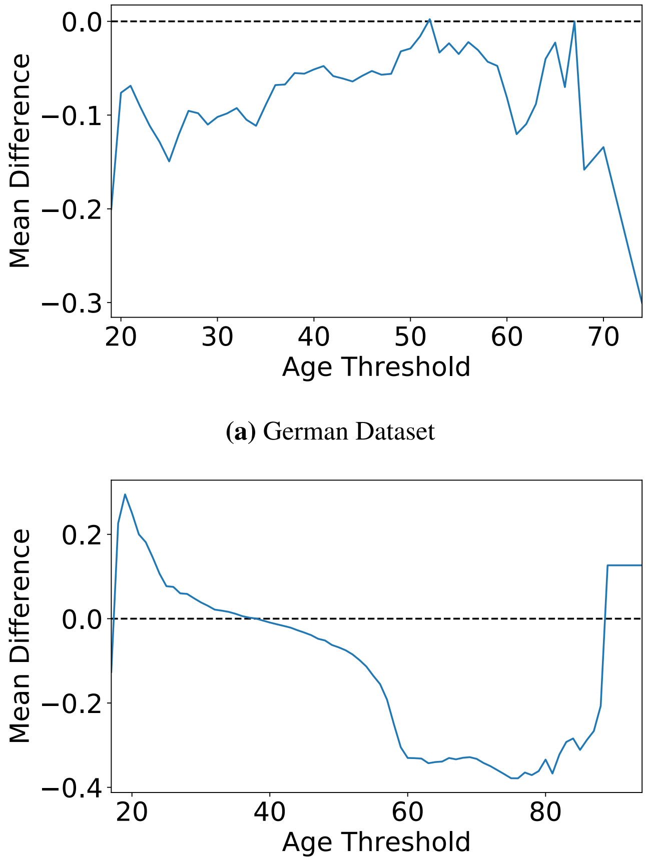
The first research question investigates the fairness of the two datasets (German, Bank) according to the dataset fairness metric *Mean Difference* (see Section 2.1).

To evaluate datasets based on Mean Difference (probability that unprivileged group receives a favourable label - probability that privileged group receives a favourable label), we compute the Mean Difference for every possible age threshold to create privileged and unprivileged groups. In particular, we gather a list of unique ages that are present in the two datasets (i.e., there are 53 unique age values in the German dataset and 78 unique ages in the Bank dataset) and use each value to separate privileged and unprivileged groups. Given an age threshold a_t, the privileged group consists of all instances of a dataset for which $age > a_t$, the remaining instances are part of the unprivileged group. This is due to the fact that for both, the German and Bank dataset, "young" individuals are deemed (according to the default configuration of the AIF360 framework [72]). We perform this for each age that is present in the dataset except for the maximum (oldest) age, to ensure that both groups (privileged and unprivileged) are not empty. Therefore, we collect 52 measures of Mean Difference for the German datasets and 77 measures for the Bank dataset. Figure 6.3 illustrates the results.

When analyzing the Mean Difference of the different age thresholds, we can see that general notion of bias and privilege holds for the German dataset: **Privileged groups are more likely to receive a favourable outcome**. This is indicated by a negative Mean Difference. Only at the thresholds 52 and 67 are non-negative Mean Difference values reached (0.002 and 0). Furthermore, we observe that the default setting ($age > 25$ is old) provided by the AIF360 framework [72] which was chosen according to Kamiran and Calders [95], is a logical choice. The Mean Difference with an age threshold of 25 is -0.15, which is a local minimum. This divides the dataset into an unprivileged group which contains 19% of the instances, the remaining 81% are part of the privileged group. A balanced division of groups, according to the age median of 33, achieves a Mean Difference of -0.1 while 48% of the instances belong to the privileged and 52% to the unprivileged group. Given the purpose of the protected attribute (e.g., causing the highest disparity between privileged or unprivileged, or having groups of balanced sizes) an age threshold between 25 and 33 inclusive is reasonable.

Using age thresholds of 19, 68 and 74 achieves an even higher Mean Difference than 25, however they cause imbalanced sizes of privileged and unprivileged groups (with the smaller of the two being of size 2%, 7%, 2% respectively).

The Mean Difference of age thresholds for the Bank dataset shows a different situation: **There are thresholds at which the unprivileged group is more likely to receive a favourable outcome than the privileged group**. In particular, the Mean Difference is positive within the intervals 18-38 (38 being the median age of the Bank dataset) and 89-94. We disregard the latter interval, because the size of the privileged group at an age threshold of 89 is only 10%, given a dataset size of $41,188$ (Table 6.1). Choosing an age threshold within 18-38 would violate our conception of bias, as it does not favour the privileged group. Therefore, using a



(a) German Dataset

(b) Bank Dataset

Figure 6.3: RQ1: Mean Difference of the German and Bank dataset for each possible age threshold to divide privileged and unprivileged groups.

default threshold of 25, which is motivated based on the German datasets' threshold, does not represent the dataset correctly (given that the privileged group is "old"). Either the notion of privileged and unprivileged groups ought to be adjusted (i.e., "young" is a privileged group given an age threshold of 25) or the threshold value should be increased. Potential values, for which our notion of bias holds, are 47 (the 75%-percentile with a Mean Difference of -0.05) or 59 (mean difference of -0.31), which is the first threshold followed by a sharp decrease in Mean Difference as seen in Figure 6.3. The size of the privileged group at a threshold of 59 is 3%, opposed to 24% at a threshold of 47.

To conclude: We showed that the age threshold does not only impact the degree of bias, but also the bias direction. While an age threshold of 25, to distinguish privileged and unprivileged groups, is reasonable for the German datasets, it violates our notion of fairness on the Bank dataset, by favouring the unprivileged group. In addition to determining the bias between privileged and unprivileged group, age thresholds also impact the balance between the group sizes.

6.3.2 RQ2: Classification Fairness

Following, we carry out experiments to determine the bias of classification models, when being trained on the German and Bank datasets under different age thresholds. In particular, we investigate the relation of the bias present in the dataset and its impact when using it to train classification models.

6.3.2.1 RQ2.1: Bias direction.

To answer RQ2.1, we consider the same pair of datasets as used for RQ1 as well as the same procedure to determine age thresholds. For each age threshold, the datasets are adjusted (i.e., setting the protected attribute age to "young" or "old" depending on the age of an individual and the age threshold). We then train three classification models (LR, DT, SVM) for each dataset and age threshold. Afterwards, we determine the bias degree of the classification models according to three classification metrics (SPD, AOD, EOD).

If the bias measures are < 0 it signifies that the privileged group is favoured, whereas if bias measures > 0 it shows that the unprivileged group receives a favourable treatment. If there is no bias present in the prediction made by a classification model the metric is equal to 0.

Table 6.2 shows the results. For the two datasets and three classification metrics, we perform experiments for every age threshold and measure the proportion of bias directions which agree with the Mean Difference. On the Bank dataset, we can observe that for every of the nine pairs (three classification values and three classification metrics) the direction of bias agrees with MD in at least 77% of the cases. For each classification model, the bias direction of SPD agrees with MD in at least 96% of the cases. This value is higher than for the two confusion matrix metrics AOD and EOD, as the computation of SPD and MD are similar.

Table 6.2: RQ2.1: Percentage of age thresholds for which Mean Difference and classification metrics are in the same direction (favour the same population group.

	German			Bank		
	SPD	AOD	EOD	SPD	AOD	EOD
Logistic Regression	100%	80%	97%	96%	80%	77%
Decision Tree	89%	49%	89%	99%	91%	87%
Support Vector Machine	6%	6%	14%	97%	87%	81%

While for 15 out of 18 evaluations, the direction of MD and classification metrics are alike (at least 77% of measures have the same direction), we are not able to confirm that an underlying dataset bias leads to classification models that are biased in the same way (e.g., a dataset that is biased towards the privileged group does not always lead to classification models that do the same). In particular, there are cases on the German dataset which on the dataset level favour the privileged group, but when trained on SVMs are more likely to favour the unprivileged group. Reasons for this disparity can be seen in the small size of the German dataset ($1,000$ instances) or the high degree of imbalance (87% of the instances receive an unfavourable outcome).

6.3.2.2 RQ2.2: Bias intensity.

In addition to investigating the relation of bias direction in regards to dataset and classification bias, we are interested to see whether a highly biased dataset (e.g., high Mean Difference) leads to highly biased classification model, or vice versa (i.e., low dataset bias leads to fair classification models). Since we are only interested in the bias intensity and not the direction of bias, we continue our investigation with absolute bias values.

Figure 6.4 illustrates the relation of dataset and classification bias for the Bank dataset and Figure 6.5 for the German dataset. Each age threshold is represented as a point in the graphs, with the intensity of dataset bias (Mean Difference) on the x-axis and intensity of one of the classification metrics on the y-axis. In addition to the dataset-classification bias pairs, each graph displays a regression line, with the corresponding Pearson correlation coefficient [467] shown in Table 6.3. We fol-

Table 6.3: RQ2.2: Pearson correlation coefficient and the corresponding p-value for Mean Difference (MD) and classification metrics (SPD, AOD, EOD).

Correlation	Bank			German		
(p-value)	SPD	AOD	EOD	SPD	AOD	EOD
Logistic Regression	0.95 (0.00)	0.83 (0.00)	0.63 (0.00)	0.98 (0.00)	0.91 (0.00)	0.82 (0.00)
Decision Tree	1.00 (0.00)	0.99 (0.00)	0.72 (0.00)	0.91 (0.00)	0.46 (0.01)	0.88 (0.00)
Support Vector Machine	0.98 (0.00)	0.92 (0.00)	0.75 (0.00)	-0.69 (0.00)	-0.69 (0.00)	0.28 (0.11)

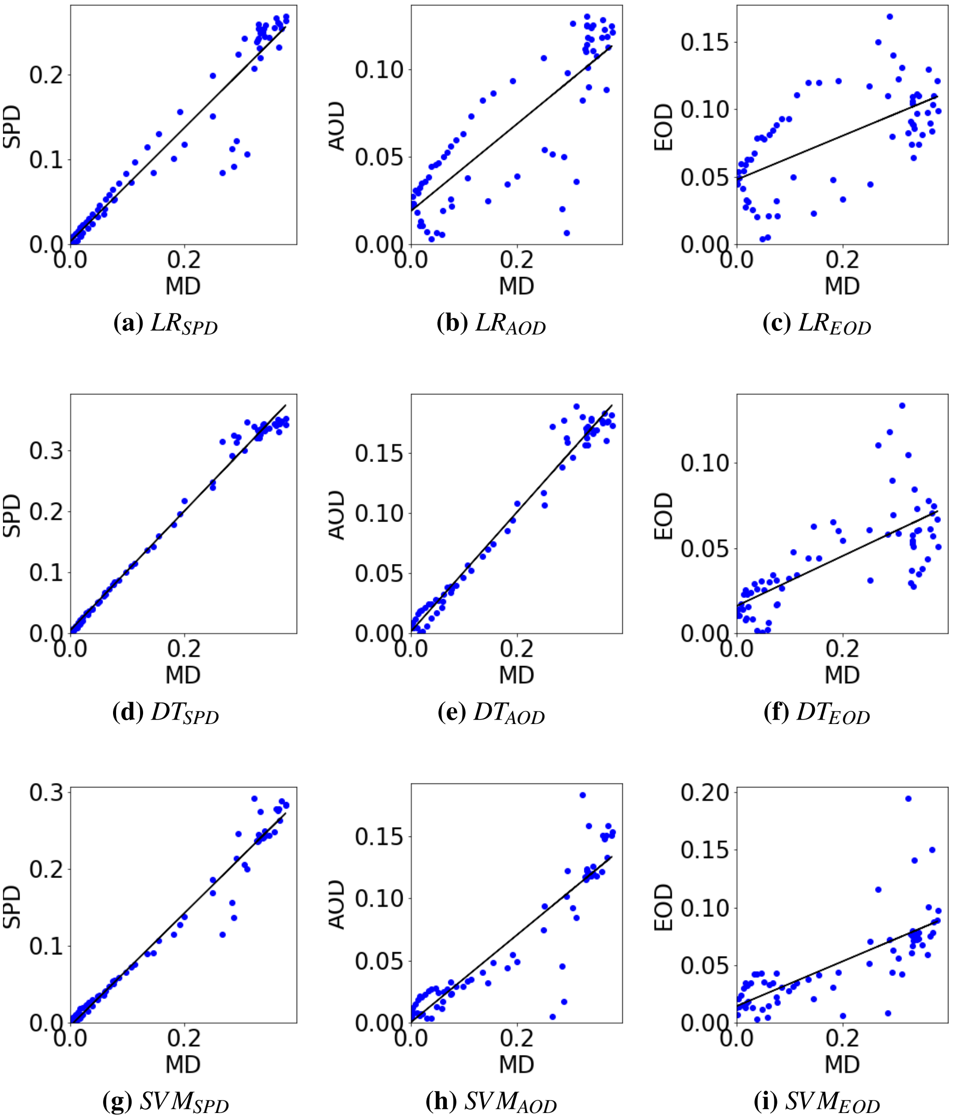
low the guidelines proposed by Evans [468], who described correlation strength as: very weak (±0.00 ±0.19), weak (±0.20 ±0.39), moderate (±0.40 ±0.59), strong (±0.60 ±0.79) and very strong (±0.80 ±1.00).

When looking at the Bank dataset, we can observe that the correlation between dataset and classification metrics are either very strong (for SPD and AOD) or strong (EOD) for all classification models. The Bank dataset confirms the intuition that a high bias in the dataset (according to Mean Difference) leads to a high bias in classification models that are trained on this data. A similar conclusion can be drawn for the German dataset when only considering LR and DT classification models. However, the results on SVMs do not comply with this intuition. Differently from all the other evaluations, dataset bias and classification bias are inverse-correlated for SVMs on the German dataset (i.e., a large dataset bias leads to classification models with little bias). Reasons for such observations could be the small dataset size or properties of the classification model.

6.4 Conclusions

Recent advances on the investigation of software fairness are conducted by dividing the population in two groups (privileged and unprivileged) based on protected attributes. Protected attributes come in the form of categorical, and continuous attributes for which thresholds need to be chosen. Our work provides choices on thresholds when dealing with continuous protected attributes (i.e., “age”), which has a direct impact on the perceived bias of software systems. We performed a detailed study on age thresholds and their impact on fairness for two frequently used datasets in fairness research.

Critically, the choice of age bias impacts the degree of bias measured in

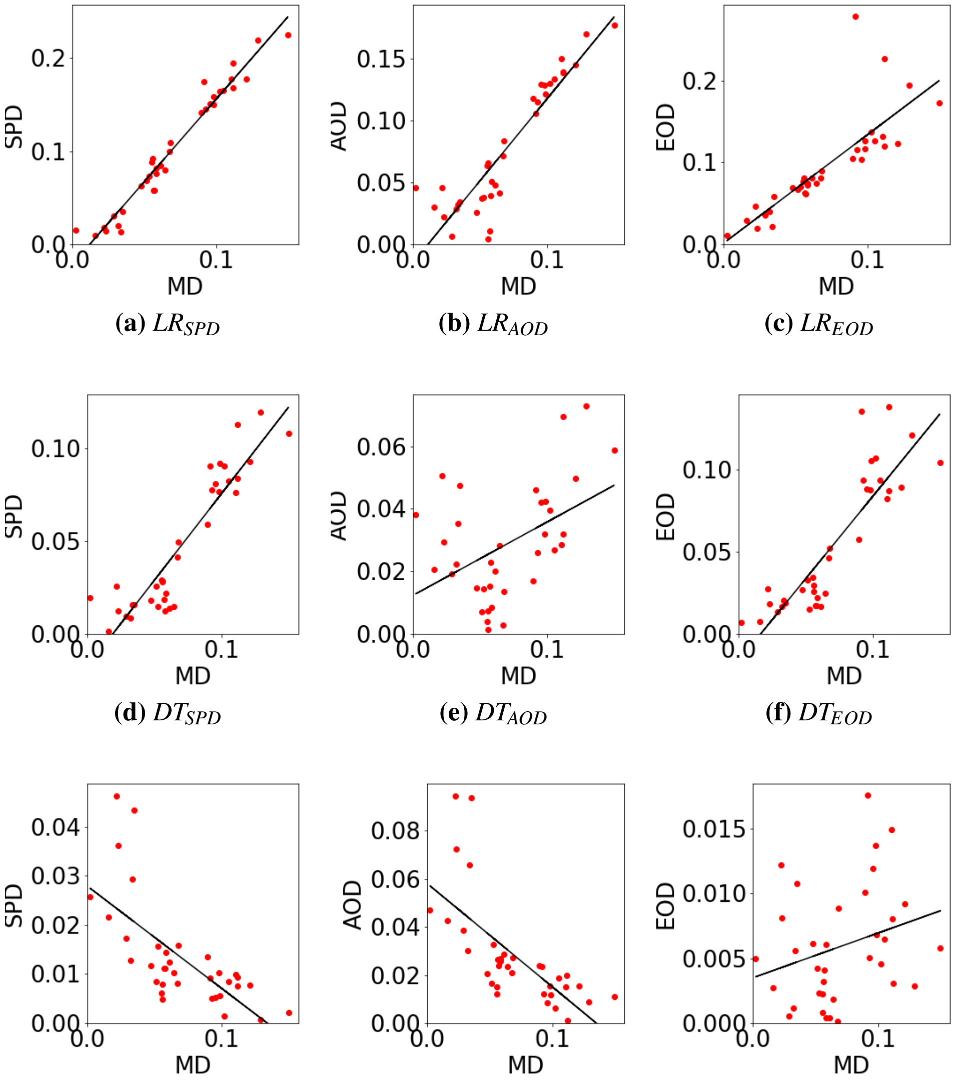


(a) LR_{SPD} **(b)** LR_{AOD} **(c)** LR_{EOD}

(d) DT_{SPD} **(e)** DT_{AOD} **(f)** DT_{EOD}

(g) SVM_{SPD} **(h)** SVM_{AOD} **(i)** SVM_{EOD}

Figure 6.4: RQ2.2 Bank: Relation of Mean Difference (MD) and classification metrics (SPD, AOD, EOD). Each point represents the bias of an age threshold (dataset bias before training and classification bias after training the given classification model). A regression line is shown in black.



(a) LR_{SPD} **(b)** LR_{AOD} **(c)** LR_{EOD}

(d) DT_{SPD} **(e)** DT_{AOD} **(f)** DT_{EOD}

(g) SVM_{SPD} **(h)** SVM_{AOD} **(i)** SVM_{EOD}

Figure 6.5: RQ2.2 German: Relation of Mean Difference (MD) and classification metrics (SPD, AOD, EOD). Each point represents the bias of an age threshold (dataset bias before training and classification bias after training the given classification model). A regression line is shown in black.

datasets and classification models trained on this data. Thereby, practitioners run the risk of fairness measurements to incorrectly represent the real world when selecting inaccurate thresholds. For example, practitioners could deliberately choose an inaccurate threshold to "fairwash" their system [403, 439]. In this context, "fairwashing" means that instead of debiasing the ML model, an age threshold is chosen that leads to low bias measurements according to a fairness metrics (e.g., an age threshold of 67 shows no dataset bias on the German dataset).

Moreover, our findings show that age thresholds that are sufficient for one dataset (e.g., 25 for the German dataset) can not be transferred to other datasets without further considerations. Furthermore, even though the dataset bias is correlated to the bias in subsequently trained classification models (e.g., high bias in datasets leads to a high bias in classification models), we also found examples for which this is not true. Therefore, we cannot confirm the notion that a high bias in datasets corresponds with a high bias in classification models.

While we provided potential age thresholds for the German and Bank datasets, and support the decision making process when dealing with continuous protected attributes, we note that the ultimate choice of age thresholds is up to practitioners and lawmakers.

7

Conclusions

> *The end of a melody is not its goal; but nonetheless, if the melody had not reached its end it would not have reached its goal either. A parable.*
>
> – Friedrich Nietzsche

This thesis addresses the problem of trade-offs when applying bias mitigation methods for improving the fairness of ML systems and associated performance deterioration with respect to other objectives, such as accuracy.

In particular, the thesis set out to achieve the following objectives:

1. Analyze state-of-the-art methods for bias reduction.

2. Measure and compare the quality of bias mitigation methods.

3. Improve the accuracy and fairness of Machine Learning software.

At first, we performed a literature review of 341 bias mitigation method. These are divided in 13 categories and analyzed further. Among others, we investigated popular datasets (i.e., the Adult dataset is the most popular dataset for fairness research), benchmarking practices, and the metrics used to measure fairness.

Secondly, we proposed *Fairea*, a model behaviour mutation approach to benchmarking ML bias mitigation methods. The usefulness of *Fairea* as a baseline and benchmarking approach is evaluated in an empirical study of 8 bias mitigation methods. In 49% of the cases, the baseline of *Fairea* exhibits a better bias mitigation ability than the studied bias mitigation methods, reinforcing its competitiveness.

34% of the cases accord to the fairness-accuracy trade-off, which allows *Fairea* to quantify their trade-off goodness.

We also found 10% of bias mitigation methods that reduced bias in hand with an improvement in fairness, which we categorize as a "win-win" scenario. This "win-win" scenario motivates our novel post-processing bias mitigation method (Chapter 5). Our post-processing methods provides a general search procedure that can be applied to any classification model as long as a modification operator exists. In our case, we considered two classification models of different families: Logistic Regression and Decision Trees. In comparison with three existing post-processing methods, our approach achieved the highest accuracy in all cases, while also exhibiting the lowest degree of bias in 33% of the cases.

Moreover, this thesis addresses the challenge of dealing with the fairness of ML software and the impact of different age thresholds (i.e., dividing the population into young and old) on bias mitigation (Chapter 6). We analyzed two datasets (Bank and German) with pre-defined age thresholds, to show that age thresholds do not only impact the intensity of bias in these datasets, but also the direction (e.g., which population group receives favourable treatments).

Our literature review (Chapter 3) collected a multitude of metrics and datasets to carry out experiments, as well as bias mitigation methods for benchmarking, which allow for large-scale empirical studies and an extension of our experiments in future work. For instance, *Fairea* can be applied for a larger set of bias mitigation methods, datasets and fairness metrics. Also, the availability of additional metrics may lead to an increase in the required trade-offs to consider when developing fair ML systems. Investigating other mutation operators could lead to further improvements in our results. For example, we carried out preliminary experiments with additional mutation operators for Decision Trees (i.e., leaf relabeling and swapping of subtrees) which were outperformed by pruning, but no systematic study was conducted.

Moreover, our post-processing approach (Chapter 5) can be applied to one of the 49 classification models that have been used for evaluating bias mitigation, two of which have already been addressed (i.e., Logistic Regression and Decision Trees). This includes the creation of novel representations for a diverse set of classification models and new modification operators.

While choosing a unified fairness metric is challenging and still an open challenge [20, 144, 265], other aspects of the empirical evaluation for bias mitigation methods could be consolidated. For example, open-source frameworks, such as the AIF360 framework, support this by making selected datasets and bias mitigation methods available. However, considerations on which and how many datasets to use for evaluation, the size of training and test splits, as well as number of repetitions for experiments are up to practitioners and exhibit a high variation. In addition to evaluating bias mitigation methods on a common set of real-world datasets, a suite of synthetic datasets with designated fairness properties and degrees of bias could support the benchmarking of bias mitigation methods further.

Lastly, while bias mitigation methods have been extensively studied from a research standpoint, as shown by the 341 publications gathered in the literature review (Chapter 3), the underlying purpose is to achieve fairness in real world systems. One aspect that can support the applicability of bias mitigation methods by practitioners is the prevention of fairness-accuracy trade-offs, as shown by our post-processing approach. In this way, existing systems would not require performance deterioration for fairness improvements.

Bibliography

[1] Max Hort, Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. Bias Mitigation for Machine Learning Classifiers: A Comprehensive Survey. *arXiv preprint arXiv:2207.07068*, 2022.

[2] Max Hort, Jie M. Zhang, Federica Sarro, and Mark Harman. Fairea: A Model Behaviour Mutation Approach to Benchmarking Bias Mitigation Methods. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2021.

[3] Max Hort, Jie M Zhang, Federica Sarro, and Mark Harman. Search-based Automatic Repair for Fairness and Accuracy in Decision-making Software. *Under review*, 2022.

[4] Max Hort and Federica Sarro. Privileged and Unprivileged Groups: An Empirical Study on the Impact of the Age Attribute on Fairness. In *International Workshop on Equitable Data and Technology (FairWare '22)*. ACM, 2022.

[5] Zhenpeng Chen, Jie M Zhang, Max Hort, Federica Sarro, and Mark Harman. Fairness testing: A comprehensive survey and analysis of trends. *arXiv preprint arXiv:2207.10223*, 2022.

[6] Max Hort and Federica Sarro. The Effect of Offspring Population Size on NSGA-II: A Preliminary Study. In *Proceedings of the 2021 Genetic and Evolutionary Computation Conference Companion*, 2021.

[7] Max Hort, Maria Kechagia, Federica Sarro, and Mark Harman. A Survey of Performance Optimization for Mobile Applications. *IEEE Transactions on Software Engineering (TSE)*, 2021.

[8] Max Hort and Federica Sarro. Optimising Word Embeddings With Search-Based Approaches. In *Proceedings of the 2020 Genetic and Evolutionary Computation Conference Companion*, pages 269–270, 2020.

[9] Emeralda Sesari, Max Hort, and Federica Sarro. An Empirical Study on the Fairness of Pre-trained Word Embeddings. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing*, 2022.

[10] James Zhong, Max Hort, and Federica Sarro. Py2Cy: A Genetic Improvement Tool To Speed Up Python. In *Genetic and Evolutionary Computation Conference Companion (GECCO '22 Companion)*, 2022.

[11] Max Hort, Rebecca Moussa, and Federica Sarro. Multi-objective search for gender-fair and semantically correct word embeddings. *Applied Soft Computing*, 133:109916, 2023.

[12] Minghua Ma, Zhao Tian, Max Hort, Federica Sarro, Hongyu Zhang, Qingwei Lin, and Dongmei Zhang. Enhanced fairness testing via generating effective initial individual discriminatory instances. *arXiv preprint arXiv:2209.08321*, 2022.

[13] Max Hort and Federica Sarro. Did You Do Your Homework? Raising Awareness on Software Fairness and Discrimination. In *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 1322–1326. IEEE, 2021.

[14] Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929. IEEE, 2012.

[15] Faisal Kamiran, Sameen Mansha, Asim Karim, and Xiangliang Zhang. Exploiting reject option in classification for social discrimination control. *Information Sciences*, 425:18–33, 2018.

[16] Amitabha Mukerjee, Rita Biswas, Kalyanmoy Deb, and Amrit P Mathur. Multi–objective evolutionary algorithms for the risk–return trade–off in bank loan management. *International Transactions in operational research*, 9(5):583–597, 2002.

[17] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. propublica. *See https://www. propublica. org/article/machine-bias-risk-assessments-in-criminal-sentencing*, 2016.

[18] Richard Berk, Hoda Heidari, Shahin Jabbari, Michael Kearns, and Aaron Roth. Fairness in criminal justice risk assessments: The state of the art. *Sociological Methods & Research*, page 0049124118782533, 2018.

[19] Dino Pedreshi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 560–568, 2008.

[20] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR)*, 54(6):1–35, 2021.

[21] Jakub Mikians, László Gyarmati, Vijay Erramilli, and Nikolaos Laoutaris. Detecting price and search discrimination on the internet. In *Proceedings of the 11th ACM workshop on hot topics in networks*, pages 79–84, 2012.

[22] Jiahao Chen, Nathan Kallus, Xiaojie Mao, Geoffry Svacha, and Madeleine Udell. Fairness under unawareness: Assessing disparity when protected class is unobserved. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 339–348, 2019.

[23] Andrea Romei and Salvatore Ruggieri. A multidisciplinary survey on discrimination analysis, 2011.

[24] Jie M Zhang, Mark Harman, Lei Ma, and Yang Liu. Machine learning testing: Survey, landscapes and horizons. *IEEE Transactions on Software Engineering*, 2020.

[25] Yuriy Brun and Alexandra Meliou. Software fairness. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 754–759, 2018.

[26] Jie Zhang and Mark Harman. ”ignorance and prejudice” in software fairness. In *2021 IEEE/ACM 43th International Conference on Software Engineering (ICSE)*. IEEE, 2021.

[27] Jennifer Horkoff. Non-functional requirements for machine learning: Challenges and new directions. In *2019 IEEE 27th International Requirements Engineering Conference (RE)*, pages 386–391. IEEE, 2019.

[28] Joymallya Chakraborty, Suvodeep Majumder, Zhe Yu, and Tim Menzies. *Fairway: A Way to Build Fair ML Software*, page 654–665. Association for Computing Machinery, New York, NY, USA, 2020.

[29] Richard Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

[30] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[31] Toon Calders and Sicco Verwer. Three naive bayes approaches for discrimination-free classification. *Data Mining and Knowledge Discovery*, 21(2):277–292, 2010.

[32] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 329–338. ACM, 2019.

[33] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*, pages 3992–4001, 2017.

[34] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[35] Faisal Kamiran and Toon Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems*, 33(1):1–33, 2012.

[36] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340. ACM, 2018.

[37] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. volume 80 of *Proceedings of Machine Learning Research*, pages 2564–2572, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR.

[38] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Classification with fairness constraints: A meta-algorithm with provable guarantees. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 319–328, 2019.

[39] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th international conference on world wide web*, pages 1171–1180, 2017.

[40] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems*, pages 5680–5689, 2017.

[41] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[42] Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. Discrimination aware decision tree learning. In *2010 IEEE International Conference on Data Mining*, pages 869–874. IEEE, 2010.

[43] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806, 2017.

[44] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. A unified approach to quantifying algorithmic unfairness: Measuring individual &group unfairness via inequality indices. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2239–2248, 2018.

[45] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[46] Pranay K. Lohia, Karthikeyan Natesan Ramamurthy, Manish Bhide, Diptikalyan Saha, Kush R. Varshney, and Ruchir Puri. Bias mitigation post-processing for individual and group fairness. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2847–2851, 2019.

[47] Wei Zhu, Haitian Zheng, Haofu Liao, Weijian Li, and Jiebo Luo. Learning bias-invariant representation by cross-sample mutual information minimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15002–15012, 2021.

[48] Sam Corbett-Davies and Sharad Goel. The measure and mismeasure of fairness: A critical review of fair machine learning. *arXiv preprint arXiv:1808.00023*, 2018.

[49] Ezekiel Soremekun, Mike Papadakis, Maxime Cordy, and Yves Le Traon. Software fairness: An analysis and survey. *arXiv preprint arXiv:2205.08809*, 2022.

[50] Karl Wiegers and Joy Beatty. *Software requirements*. Pearson Education, 2013.

[51] Khan Mohammad Habibullah and Jennifer Horkoff. Non-functional requirements for machine learning: understanding current use and challenges in industry. In *2021 IEEE 29th International Requirements Engineering Conference (RE)*, pages 13–23. IEEE, 2021.

[52] Nagadivya Balasubramaniam, Marjo Kauppinen, Sari Kujala, and Kari Hiekkanen. Ethical guidelines for solving ethical issues and developing ai systems. In *International Conference on Product-Focused Software Process Improvement*, pages 331–346. Springer, 2020.

[53] Anthony Finkelstein, Mark Harman, S Afshin Mansouri, Jian Ren, and Yuanyuan Zhang. A search based approach to fairness analysis in requirement assignments to aid negotiation, mediation and decision making. *Requirements engineering*, 14(4):231–245, 2009.

[54] Len Bass, Paul Clements, and Rick Kazman. *Software architecture in practice*. Addison-Wesley Professional, 2003.

[55] Ying Shu, Jiehuang Zhang, and Han Yu. Fairness in design: A tool for guidance in ethical artificial intelligence design. In *International Conference on Human-Computer Interaction*, pages 500–510. Springer, 2021.

[56] S Stumpf, L Strappelli, S Ahmed, Y Nakao, A Naseer, GD Gamba, and D Regoli. Design methods for artificial intelligence fairness and transparency. In *CEUR Workshop Proceedings*, volume 2903, pages 1613–0073, 2021.

[57] Aws Albarghouthi, Loris D'Antoni, Samuel Drews, and Aditya V Nori. Fairsquare: probabilistic verification of program fairness. *Proceedings of the ACM on Programming Languages*, 1(OOPSLA):1–30, 2017.

[58] Philips George John, Deepak Vijaykeerthy, and Diptikalyan Saha. Verifying individual fairness in machine learning models. In *Conference on Uncertainty in Artificial Intelligence*, pages 749–758. PMLR, 2020.

[59] Margaret Burnett, Simone Stumpf, Jamie Macbeth, Stephann Makri, Laura Beckwith, Irwin Kwan, Anicia Peters, and William Jernigan. Gendermag: A method for evaluating software's gender inclusiveness. *Interacting with Computers*, 28(6):760–787, 2016.

[60] Sainyam Galhotra, Yuriy Brun, and Alexandra Meliou. Fairness testing: testing software for discrimination. In *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, pages 498–510. ACM, 2017.

[61] Rico Angell, Brittany Johnson, Yuriy Brun, and Alexandra Meliou. Themis: Automatically testing software for discrimination. In *Proceedings of the*

2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering, pages 871–875, 2018.

[62] Sakshi Udeshi, Pryanshu Arora, and Sudipta Chattopadhyay. Automated directed fairness testing. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, pages 98–108, 2018.

[63] Aniya Aggarwal, Pranay Lohia, Seema Nagar, Kuntal Dey, and Diptikalyan Saha. Black box fairness testing of machine learning models. In *Proceedings of the 2019 27th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, pages 625–635, 2019.

[64] Florian Tramer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, Jean-Pierre Hubaux, Mathias Humbert, Ari Juels, and Huang Lin. Fairtest: Discovering unwarranted associations in data-driven applications. In *2017 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 401–416. IEEE, 2017.

[65] Amit Datta, Michael Carl Tschantz, and Anupam Datta. Automated experiments on ad privacy settings: A tale of opacity, choice, and discrimination. *Proceedings on privacy enhancing technologies*, 2015(1):92–112, 2015.

[66] Aniko Hannak, Gary Soeller, David Lazer, Alan Mislove, and Christo Wilson. Measuring price discrimination and steering on e-commerce web sites. In *Proceedings of the 2014 conference on internet measurement conference*, pages 305–318, 2014.

[67] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *2009 IEEE International Conference on Data Mining Workshops*, pages 13–18. IEEE, 2009.

[68] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.

[69] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[70] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

[71] Medical expenditure panel survey dataset. `https://meps.ahrq.gov/mepsweb/data_stats/download_data_files_detail.jsp?cboPufNumber=HC-192`, 2016. Retrieved on June 12, 2022.

[72] Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, et al. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018.

[73] Alexandra Chouldechova and Aaron Roth. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*, 2018.

[74] Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. Learning gender-neutral word embeddings. *EMNLP*, pages 4847–4853, 2018.

[75] Dana Pessach and Erez Shmueli. Algorithmic fairness. *arXiv preprint arXiv:2001.09784*, 2020.

[76] Solon Barocas and Andrew D Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

[77] Benjamin van Giffen, Dennis Herhausen, and Tobias Fahse. Overcoming the pitfalls and perils of algorithms: A classification of machine learning biases and mitigation methods. *Journal of Business Research*, 144:93–106, 2022.

[78] Jeffrey Dastin. Amazon scraps secret ai recruiting tool that showed bias against women. `https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G`, oct 2018.

[79] Peter J Bickel, Eugene A Hammel, and J William O'Connell. Sex bias in graduate admissions: Data from berkeley. *Science*, 187(4175):398–404, 1975.

[80] Caitlin Kuhlman, Latifa Jackson, and Rumi Chunara. No computation without representation: Avoiding data and algorithm biases through diversity. *arXiv preprint arXiv:2002.11836*, 2020.

[81] Jannik Dunkelau and Michael Leuschel. Fairness-aware machine learning. 2019.

[82] Zhenpeng Chen, Jie M Zhang, Federica Sarro, and Mark Harman. A comprehensive empirical study of bias mitigation methods for software fairness. *arXiv preprint arXiv:2207.03277*, 2022.

[83] Sahil Verma and Julia Rubin. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7. IEEE, 2018.

[84] On-line appendix: Survey results, 2022.

[85] Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)*, 55(3):1–44, 2022.

[86] Simon Caton and Christian Haas. Fairness in machine learning: A survey. *arXiv preprint arXiv:2010.04053*, 2020.

[87] Indre Žliobaite. A survey on measuring indirect discrimination in machine learning. *arXiv preprint arXiv:1511.00148*, 2015.

[88] Tai Le Quy, Arjun Roy, Vasileios Iosifidis, Wenbin Zhang, and Eirini Ntoutsi. A survey on datasets for fairness-aware machine learning. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, page e1452, 2022.

[89] Xueru Zhang and Mingyan Liu. Fairness in learning-based sequential decision algorithms: A survey. In *Handbook of Reinforcement Learning and Control*, pages 525–555. Springer, 2021.

[90] Wenbin Zhang, Jeremy C Weiss, Shuigeng Zhou, and Toby Walsh. Fairness amidst non-iid graph data: A literature review. *arXiv preprint arXiv:2202.07170*, 2022.

[91] Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. Mitigating gender bias in natural language processing: Literature review. *arXiv preprint arXiv:1906.08976*, 2019.

[92] Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (technology) is power: A critical survey of" bias" in nlp. *arXiv preprint arXiv:2005.14050*, 2020.

[93] William Martin, Federica Sarro, Yue Jia, Yuanyuan Zhang, and Mark Harman. A survey of app store analysis for software engineering. *IEEE transactions on software engineering*, 43(9):817–847, 2016.

[94] Claes Wohlin. Guidelines for snowballing in systematic literature studies and a replication in software engineering. In *Proceedings of the 18th international conference on evaluation and assessment in software engineering*, pages 1–10, 2014.

[95] Faisal Kamiran and Toon Calders. Classifying without discriminating. In *2009 2nd international conference on computer, control and communication*, pages 1–6. IEEE, 2009.

[96] Binh Thanh Luong, Salvatore Ruggieri, and Franco Turini. k-nn as an implementation of situation testing for discrimination discovery and prevention. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 502–510, 2011.

[97] Indre Žliobaite, Faisal Kamiran, and Toon Calders. Handling conditional discrimination. In *2011 IEEE 11th International Conference on Data Mining*, pages 992–1001. IEEE, 2011.

[98] Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *IEEE transactions on knowledge and data engineering*, 25(7):1445–1459, 2012.

[99] Lu Zhang, Yongkai Wu, and Xintao Wu. Achieving non-discrimination in prediction. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 3097–3103. AAAI Press, 2018.

[100] Vasileios Iosifidis, Thi Ngoc Han Tran, and Eirini Ntoutsi. Fairness-enhancing interventions in stream classification. In *International Conference on Database and Expert Systems Applications*, pages 261–276. Springer, 2019.

[101] Haipei Sun, Kun Wu, Ting Wang, and Wendy Hui Wang. Towards fair and robust classification. In *2022 IEEE 7th European Symposium on Security and Privacy (EuroS&P)*, pages 356–376. IEEE, 2022.

[102] Emel Seker, John R Talburt, and Melody L Greer. Preprocessing to address bias in healthcare data. *Studies in Health Technology and Informatics*, 294:327–331, 2022.

[103] Ibrahim Alabdulmohsin, Jessica Schrouff, and Oluwasanmi Koyejo. A reduction to binary approach for debiasing multiclass datasets. *arXiv preprint arXiv:2205.15860*, 2022.

[104] Kristian Lum and James Johndrow. A statistical framework for fair predictive algorithms. *arXiv preprint arXiv:1610.08077*, 2016.

[105] Hao Wang, Berk Ustun, Flavio P Calmon, and SEAS Harvard. Avoiding disparate impact with counterfactual distributions. In *NeurIPS Workshop on Ethical, Social and Governance Issues in AI*, 2018.

[106] Hao Wang, Berk Ustun, and Flavio Calmon. Repairing without retraining: Avoiding disparate impact with counterfactual distributions. In *International Conference on Machine Learning*, pages 6618–6627. PMLR, 2019.

[107] James E Johndrow and Kristian Lum. An algorithm for removing sensitive information: application to race-independent recidivism prediction. *The Annals of Applied Statistics*, 13(1):189–220, 2019.

[108] Tianyi Li, Zhoufei Tang, Tao Lu, and Xiaoquan Michael Zhang. 'propose and review': Interactive bias mitigation for machine classifiers. *Available at SSRN 4139244*, 2022.

[109] Yanhui Li, Linghan Meng, Lin Chen, Li Yu, Di Wu, Yuming Zhou, and Baowen Xu. Training data debugging for the fairness of machine learning software. In *2022 IEEE/ACM 44th International Conference on Software Engineering (ICSE)*, pages 2215–2227, 2022.

[110] Faisal Kamiran and Toon Calders. Classification with no discrimination by preferential sampling. In *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*, pages 1–6. Citeseer, 2010.

[111] Lu Zhang, Yongkai Wu, and Xintao Wu. A causal framework for discovering and removing direct and indirect discrimination. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3929–3935, 2017.

[112] Emmanouil Krasanakis, Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, and Yiannis Kompatsiaris. Adaptive sensitive reweighting to

mitigate bias in fairness-aware classification. In *Proceedings of the 2018 World Wide Web Conference*, pages 853–862, 2018.

[113] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Fairgan: Fairness-aware generative adversarial networks. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 570–575. IEEE, 2018.

[114] Irene Chen, Fredrik D Johansson, and David Sontag. Why is my classifier discriminatory? *Advances in Neural Information Processing Systems*, 31, 2018.

[115] Vasileios Iosifidis and Eirini Ntoutsi. Dealing with bias via data augmentation in supervised learning scenarios. *Jo Bates Paul D. Clough Robert Jäschke*, 24, 2018.

[116] Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data*, pages 793–810, 2019.

[117] Vladimiro Zelaya, Paolo Missier, and Dennis Prangle. Parametrised data sampling for fairness optimisation. *KDD XAI*, 2019.

[118] Depeng Xu, Yongkai Wu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Achieving causal fairness through generative adversarial networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019.

[119] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Fairgan+: Achieving fair data generation and classification through generative adversarial nets. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 1401–1406. IEEE, 2019.

[120] Vasileios Iosifidis, Besnik Fetahu, and Eirini Ntoutsi. Fae: A fairness-aware ensemble framework. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 1375–1380. IEEE, 2019.

[121] Adel Abusitta, Esma Aïmeur, and Omar Abdel Wahab. Generative adversarial networks for mitigating biases in machine learning systems. *arXiv preprint arXiv:1905.09972*, 2019.

[122] Shubham Sharma, Yunfeng Zhang, Jesús M Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R Varshney. Data augmentation for discrimination prevention and bias disambiguation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 358–364, 2020.

[123] Heinrich Jiang and Ofir Nachum. Identifying and correcting label bias in machine learning. In *International Conference on Artificial Intelligence and Statistics*, pages 702–712. PMLR, 2020.

[124] Tongxin Hu, Vasileios Iosifidis, Wentong Liao, Hang Zhang, Michael Ying Yang, Eirini Ntoutsi, and Bodo Rosenhahn. Fairnn-conjoint learning of fair representations for fair decisions. In *International Conference on Discovery Science*, pages 581–595. Springer, 2020.

[125] Alice Morano. *Bias mitigation for automated decision making systems*. PhD thesis, Politecnico di Torino, 2020.

[126] Shen Yan, Hsien-te Kao, and Emilio Ferrara. Fair class balancing: enhancing model fairness without observing sensitive attributes. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 1715–1724, 2020.

[127] L Elisa Celis, Vijay Keswani, and Nisheeth Vishnoi. Data preprocessing to mitigate bias: A maximum entropy based approach. In *International Conference on Machine Learning*, pages 1349–1359. PMLR, 2020.

[128] Annie Abay, Yi Zhou, Nathalie Baracaldo, Shashank Rajamoni, Ebube Chuba, and Heiko Ludwig. Mitigating bias in federated learning. *arXiv preprint arXiv:2012.02447*, 2020.

[129] Teresa Salazar, Miriam Seoane Santos, Helder Araújo, and Pedro Henriques Abreu. Fawos: Fairness-aware oversampling algorithm based on distributions of sensitive attributes. *IEEE Access*, 2021.

[130] Wenbin Zhang, Albert Bifet, Xiangliang Zhang, Jeremy C Weiss, and Wolfgang Nejdl. Farf: A fair and adaptive random forests classifier. In *Advances in Knowledge Discovery and Data Mining: 25th Pacific-Asia Conference, PAKDD 2021, Virtual Event, May 11–14, 2021, Proceedings, Part II*, pages 245–256, 2021.

[131] Ching-Yao Chuang and Youssef Mroueh. Fair mixup: Fairness via interpolation. In *International Conference on Learning Representations*, 2021.

[132] Jack J Amend and Scott Spurlock. Improving machine learning fairness with sampling and adversarial learning. *Journal of Computing Sciences in Colleges*, 36(5):14–23, 2021.

[133] Sahil Verma, Michael Ernst, and Rene Just. Removing biased data to improve fairness and accuracy. *arXiv preprint arXiv:2102.03054*, 2021.

[134] André F Cruz, Pedro Saleiro, Catarina Belém, Carlos Soares, and Pedro Bizarro. Promoting fairness through hyperparameter optimization. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 1036–1041, 2021.

[135] Joymallya Chakraborty, Suvodeep Majumder, and Tim Menzies. Bias in machine learning software: Why? how? what to do? In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, ESEC/FSE 2021, page 429–440, New York, NY, USA, 2021. Association for Computing Machinery.

[136] Taeuk Jang, Feng Zheng, and Xiaoqian Wang. Constructing a fair classifier with generated fair data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7908–7916, 2021.

[137] Wei Du and Xintao Wu. Fair and robust classification under sample selection bias. In *Proceedings of the 30th ACM International Conference on Information amp; Knowledge Management*, CIKM '21, page 2999–3003, New York, NY, USA, 2021. Association for Computing Machinery.

[138] Yuji Roh, Kangwook Lee, Steven Euijong Whang, and Changho Suh. Sample selection for fair and robust training. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

[139] Eugenia Iofinova, Nikola Konstantinov, and Christoph H Lampert. Flea: Provably fair multisource learning from unreliable training data. *arXiv preprint arXiv:2106.11732*, 2021.

[140] Zhe Yu. Fair balance: Mitigating machine learning bias against multiple protected attributes with data balancing. *CoRR*, abs/2107.08310, 2021.

[141] Arashdeep Singh, Jashandeep Singh, Ariba Khan, and Amar Gupta. Developing a novel fair-loan classifier through a multi-sensitive debiasing pipeline: Dualfair. *Machine Learning and Knowledge Extraction*, 4(1):240–253, 2022.

[142] Sikha Pentyala, Nicola Neophytou, Anderson Nascimento, Martine De Cock, and Golnoosh Farnadi. Privfairfl: Privacy-preserving group fairness in federated learning. *arXiv preprint arXiv:2205.11584*, 2022.

[143] Amirarsalan Rajabi and Ozlem Ozmen Garibay. Tabfairgan: Fair tabular data generation with generative adversarial networks. *Machine Learning and Knowledge Extraction*, 4(2):488–501, 2022.

[144] Damien Dablain, Bartosz Krawczyk, and Nitesh Chawla. Towards a holistic view of bias in machine learning: Bridging algorithmic fairness and imbalanced learning. *arXiv preprint arXiv:2207.06084*, 2022.

[145] Zhenpeng Chen, Jie M. Zhang, Federica Sarro, and Mark Harman. Maat: A novel ensemble approach to addressing fairness and performance bugs for machine learning software. In *Proceedings of the 2022 ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ESEC/FSE'22*, 2022.

[146] Peizhao Li and Hongfu Liu. Achieving fairness at no utility cost via data reweighing with influence. In *International Conference on Machine Learning*, pages 12917–12930. PMLR, 2022.

[147] Joymallya Chakraborty, Suvodeep Majumder, and Huy Tu. Fair-ssl: Building fair ml software with less data. In *International Workshop on Equitable Data and Technology (FairWare '22)*, 2022.

[148] Jialu Wang, Xin Eric Wang, and Yang Liu. Understanding instance-level impact of fairness constraints. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23114–23130. PMLR, 17–23 Jul 2022.

[149] Abdulaziz A. Almuzaini, Chidansh A. Bhatt, David M. Pennock, and Vivek K. Singh. Abcinml: Anticipatory bias correction in machine learning applications. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1552–1560, New York, NY, USA, 2022. Association for Computing Machinery.

[150] Junyi Chai and Xiaoqian Wang. Fairness with adaptive weights. In *International Conference on Machine Learning*, pages 2853–2866. PMLR, 2022.

[151] Harrison Edwards and Amos Storkey. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897*, 2015.

[152] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. The variational fair autoencoder. In Yoshua Bengio and Yann LeCun,

editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[153] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. Controllable invariance through adversarial feature learning. *Advances in neural information processing systems*, 30, 2017.

[154] Philipp Hacker and Emil Wiedemann. A continuous framework for fairness. *arXiv preprint arXiv:1712.07924*, 2017.

[155] Daniel McNamara, Cheng Soon Ong, and Robert C Williamson. Provably fair representations. *arXiv preprint arXiv:1710.04394*, 2017.

[156] Adrián Pérez-Suay, Valero Laparra, Gonzalo Mateo-García, Jordi Muñoz-Marí, Luis Gómez-Chova, and Gustau Camps-Valls. Fair kernel learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 339–355. Springer, 2017.

[157] Junpei Komiyama and Hajime Shimao. Two-stage algorithm for fairness-aware machine learning. *arXiv preprint arXiv:1710.04924*, 2017.

[158] Samira Samadi, Uthaipon Tantipongpipat, Jamie H Morgenstern, Mohit Singh, and Santosh Vempala. The price of fair pca: One extra dimension. In *Advances in Neural Information Processing Systems*, pages 10976–10987, 2018.

[159] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, pages 3384–3393. PMLR, 2018.

[160] Flavio du Pin Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. Data pre-processing for discrimination prevention: Information-theoretic optimiza-

tion and analysis. *IEEE Journal of Selected Topics in Signal Processing*, 12(5):1106–1119, 2018.

[161] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Greg Ver Steeg, and Aram Galstyan. Invariant representations without adversarial training. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 9102–9111, 2018.

[162] Novi Quadrianto, Viktoriia Sharmanska, and Oliver Thomas. Neural styling for interpretable fair representations. *arXiv preprint arXiv:1810.06755*, 2018.

[163] Nina Grgić-Hlača, Muhammad Bilal Zafar, Krishna P Gummadi, and Adrian Weller. Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[164] Jiaming Song, Pratyusha Kalluri, Aditya Grover, Shengjia Zhao, and Stefano Ermon. Learning controllable fair representations. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2164–2173. PMLR, 2019.

[165] Xiaoqian Wang and Heng Huang. Approaching machine learning fairness through adversarial network. *arXiv preprint arXiv:1909.03013*, 2019.

[166] Preethi Lahoti, Krishna P. Gummadi, and Gerhard Weikum. Operationalizing individual fairness with pairwise fair representations. *Proc. VLDB Endow.*, 13(4):506–518, dec 2019.

[167] Rui Feng, Yang Yang, Yuehan Lyu, Chenhao Tan, Yizhou Sun, and Chunping Wang. Learning fair representations via an adversarial framework. *arXiv preprint arXiv:1904.13341*, 2019.

[168] Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. ifair: Learning individually fair data representations for algorithmic decision making. In

2019 ieee 35th international conference on data engineering (icde), pages 1334–1345. IEEE, 2019.

[169] Elliot Creager, David Madras, Jörn-Henrik Jacobsen, Marissa Weis, Kevin Swersky, Toniann Pitassi, and Richard Zemel. Flexibly fair representation learning by disentanglement. In *International conference on machine learning*, pages 1436–1445. PMLR, 2019.

[170] Paula Gordaliza, Eustasio Del Barrio, Gamboa Fabrice, and Jean-Michel Loubes. Obtaining fairness using optimal transport theory. In *International Conference on Machine Learning*, pages 2357–2365. PMLR, 2019.

[171] Novi Quadrianto, Viktoriia Sharmanska, and Oliver Thomas. Discovering fair representations in the data domain. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8227–8236, 2019.

[172] Han Zhao, Amanda Coston, Tameem Adel, and Geoffrey J. Gordon. Conditional learning of fair representations. In *International Conference on Learning Representations*, 2020.

[173] Meike Zehlike, Philipp Hacker, and Emil Wiedemann. Matching code and law: achieving algorithmic fairness with optimal transport. *Data Mining and Knowledge Discovery*, 34(1):163–200, 2020.

[174] Mhd Hasan Sarhan, Nassir Navab, Abouzar Eslami, and Shadi Albarqouni. Fairness by learning orthogonal disentangled representations. In *European Conference on Computer Vision*, pages 746–761. Springer, 2020.

[175] Zilong Tan, Samuel Yeom, Matt Fredrikson, and Ameet Talwalkar. Learning fair representations for kernel models. In *International Conference on Artificial Intelligence and Statistics*, pages 155–166. PMLR, 2020.

[176] Ayush Jaiswal, Daniel Moyer, Greg Ver Steeg, Wael AbdAlmageed, and Premkumar Natarajan. Invariant representations through adversarial forget-

ting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4272–4279, 2020.

[177] Ramanujam Madhavan and Mohit Wadhwa. Fairness-aware learning with prejudice free representations. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pages 2137–2140, 2020.

[178] Anian Ruoss, Mislav Balunovic, Marc Fischer, and Martin Vechev. Learning certified individually fair representations. *Advances in Neural Information Processing Systems*, 33:7584–7596, 2020.

[179] Jin-Young Kim and Sung-Bae Cho. Fair representation for safe artificial intelligence via adversarial learning of unbiased information bottleneck. In *SafeAI@ AAAI*, pages 105–112, 2020.

[180] Hortense Fong, Vineet Kumar, Anay Mehrotra, and Nisheeth K Vishnoi. Fairness for auc via feature augmentation. *arXiv preprint arXiv:2111.12823*, 2021.

[181] Ricardo Salazar, Felix Neutatz, and Ziawasch Abedjan. Automated feature engineering for algorithmic fairness. *Proceedings of the VLDB Endowment*, 14(9):1694–1702, 2021.

[182] Umang Gupta, Aaron Ferber, Bistra Dilkina, and Greg Ver Steeg. Controllable guarantees for fair outcomes via contrastive information estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7610–7619, 2021.

[183] Vincent Grari, Oualid El Hajouji, Sylvain Lamprier, and Marcin Detyniecki. Learning unbiased representations via rényi minimization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 749–764. Springer, 2021.

[184] Changdae Oh, Heeji Won, Junhyuk So, Taero Kim, Yewon Kim, Hosik Choi, and Kyungwoo Song. Learning fair representation via distributional contrastive disentanglement. *arXiv preprint arXiv:2206.08743*, 2022.

[185] Sushant Agarwal and Amit Deshpande. On the power of randomization in fair classification and representation. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1542–1551, New York, NY, USA, 2022. Association for Computing Machinery.

[186] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. Semi-fairvae: Semi-supervised fair representation learning with adversarial variational autoencoder. *arXiv preprint arXiv:2204.00536*, 2022.

[187] Changjian Shui, Qi Chen, Jiaqi Li, Boyu Wang, and Christian Gagné. Fair representation learning through implicit path alignment. *arXiv preprint arXiv:2205.13316*, 2022.

[188] Tao Qi, Fangzhao Wu, Chuhan Wu, Lingjuan Lyu, Tong Xu, Zhongliang Yang, Yongfeng Huang, and Xing Xie. Fairvfl: A fair vertical federated learning framework with contrastive adversarial learning. *arXiv preprint arXiv:2206.03200*, 2022.

[189] Mislav Balunović, Anian Ruoss, and Martin Vechev. Fair normalizing flows. In *International Conference on Learning Representations*, 2022.

[190] Peter Kairouz, Jiachun Liao, Chong Huang, Maunil Vyas, Monica Welfert, and Lalitha Sankar. Generating fair universal representations using adversarial models. *IEEE Transactions on Information Forensics and Security*, 17:1970–1985, 2022.

[191] Shaofan Liu, Shiliang Sun, and Jing Zhao. Fair transfer learning with factor variational auto-encoder. *Neural Processing Letters*, pages 1–13, 2022.

[192] Mattia Cerrato, Alesia Vallenas Coronel, Marius Köppel, Alexander Segner, Roberto Esposito, and Stefan Kramer. Fair interpretable representation learning with correction vectors. *arXiv preprint arXiv:2202.03078*, 2022.

[193] Mohammad Mahdi Kamani, Farzin Haddadpour, Rana Forsati, and Mehrdad Mahdavi. Efficient fair principal component analysis. *Machine Learning*, pages 1–32, 2022.

[194] Miriam Rateike, Ayan Majumdar, Olga Mineeva, Krishna P Gummadi, and Isabel Valera. Don't throw it away! the utility of unlabeled data in fair decision making. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 1421–1433, 2022.

[195] Sainyam Galhotra, Karthikeyan Shanmugam, Prasanna Sattigeri, and Kush R. Varshney. Causal feature selection for algorithmic fairness. In *Proceedings of the 2022 International Conference on Management of Data*, SIGMOD '22, page 276–285, New York, NY, USA, 2022. Association for Computing Machinery.

[196] Jin-Young Kim and Sung-Bae Cho. An information theoretic approach to reducing algorithmic bias for machine learning. *Neurocomputing*, 500:26–38, 2022.

[197] Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 656–666, Red Hook, NY, USA, 2017. Curran Associates Inc.

[198] Maya Gupta, Andrew Cotter, Mahdi Milani Fard, and Serena Wang. Proxy fairness. *arXiv preprint arXiv:1806.11212*, 2018.

[199] David Madras, Elliot Creager, Toniann Pitassi, and Richard Zemel. Fairness through causal awareness: Learning causal latent-variable models for biased

data. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 349–358, 2019.

[200] Luca Oneto, Michele Doninini, Amon Elders, and Massimiliano Pontil. Taking advantage of multitask learning for fair classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 227–237, 2019.

[201] Dennis Wei, Karthikeyan Natesan Ramamurthy, and Flavio du Pin Calmon. Optimized score transformation for fair classification. *Proceedings of Machine Learning Research*, 108, 2020.

[202] Thomas Kehrenberg, Zexun Chen, and Novi Quadrianto. Tuning fairness by balancing target labels. *Frontiers in artificial intelligence*, 3:33, 2020.

[203] Vincent Grari, Sylvain Lamprier, and Marcin Detyniecki. Fairness without the sensitive attribute via causal variational autoencoder. *arXiv preprint arXiv:2109.04999*, 2021.

[204] Canyu Chen, Yueqing Liang, Xiongxiao Xu, Shangyu Xie, Yuan Hong, and Kai Shu. On fair classification with mostly private sensitive attributes. *arXiv preprint arXiv:2207.08336*, 2022.

[205] Yueqing Liang, Canyu Chen, Tian Tian, and Kai Shu. Joint adversarial learning for cross-domain fair classification. *arXiv preprint arXiv:2206.03656*, 2022.

[206] Sangwon Jung, Sanghyuk Chun, and Taesup Moon. Learning fair classifiers with partially annotated group labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10348–10357, 2022.

[207] Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, Aaron Roth, and Saeed Sharifi-Malvajerdi. Multiaccurate proxies for downstream

fairness. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1207–1239, New York, NY, USA, 2022. Association for Computing Machinery.

[208] Songhua Wu, Mingming Gong, Bo Han, Yang Liu, and Tongliang Liu. Fair classification with instance-dependent label noise. In *Conference on Causal Learning and Reasoning*, pages 927–943. PMLR, 2022.

[209] Vinith M Suriyakumar, Marzyeh Ghassemi, and Berk Ustun. When personalization harms: Reconsidering the use of group attributes in prediction. *arXiv preprint arXiv:2206.02058*, 2022.

[210] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[211] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. Fairness-aware learning through regularization approach. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 643–650. IEEE, 2011.

[212] Goce Ristanoski, Wei Liu, and James Bailey. Discrimination aware classification for imbalanced datasets. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 1529–1532, 2013.

[213] Benjamin Fish, Jeremy Kun, and Adám D Lelkes. Fair boosting: a case study. In *Workshop on Fairness, Accountability, and Transparency in Machine Learning*. Citeseer, 2015.

[214] Yahav Bechavod and Katrina Ligett. Penalizing unfairness in binary classification. *arXiv preprint arXiv:1707.00044*, 2017.

[215] Novi Quadrianto and Viktoriia Sharmanska. Recycling privileged learning and distribution matching for fairness. *Advances in Neural Information Processing Systems*, 30:677–688, 2017.

[216] Edward Raff, Jared Sylvester, and Steven Mills. Fair forests: Regularized tree induction to minimize model bias. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 243–250, 2018.

[217] Naman Goel, Mohammad Yaghini, and Boi Faltings. Non-discriminatory machine learning through convex fairness criteria. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[218] Simon Aagaard Enni and Ira Assent. Using balancing terms to avoid discrimination in classification. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 947–952. IEEE, 2018.

[219] Jérémie Mary, Clément Calauzenes, and Noureddine El Karoui. Fairness-aware learning for continuous attributes and treatments. In *International Conference on Machine Learning*, pages 4382–4391. PMLR, 2019.

[220] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Allison Woodruff, Christine Luu, Pierre Kreitmann, Jonathan Bischof, and Ed H Chi. Putting fairness principles into practice: Challenges, metrics, and improvements. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 453–459, 2019.

[221] Wenbin Zhang, Xuejiao Tang, and Jianwu Wang. On fairness-aware learning for non-discriminative decision-making. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 1072–1079. IEEE, 2019.

[222] Sina Aghaei, Mohammad Javad Azizi, and Phebe Vayanos. Learning optimal and fair decision trees for non-discriminative decision-making. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1418–1426, 2019.

[223] Lingxiao Huang and Nisheeth Vishnoi. Stable and fair classification. In *International Conference on Machine Learning*, pages 2879–2890. PMLR, 2019.

[224] Wenbin Zhang and Eirini Ntoutsi. Faht: an adaptive fairness-aware decision tree classifier. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, pages 1480–1486, 2019.

[225] Maryam Tavakol. Fair classification with counterfactual learning. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2073–2076, 2020.

[226] Sina Baharlouei, Maher Nouiehed, Ahmad Beirami, and Meisam Razaviyayn. R\'enyi fair inference. In *8th International Conference on Learning Representations, ICLR 2020*, 2020.

[227] Pietro G Di Stefano, James M Hickey, and Vlasios Vasileiou. Counterfactual fairness: removing direct effects through regularization. *arXiv preprint arXiv:2002.10774*, 2020.

[228] Joon Sik Kim, Jiahao Chen, and Ameet Talwalkar. Fact: A diagnostic for group fairness trade-offs. In *International Conference on Machine Learning*, pages 5264–5274. PMLR, 2020.

[229] Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pages 862–872. PMLR, 2020.

[230] Yaniv Romano, Stephen Bates, and Emmanuel Candes. Achieving equalized odds by resampling sensitive attributes. *Advances in Neural Information Processing Systems*, 33:361–371, 2020.

[231] Srinivasan Ravichandran, Drona Khurana, Bharath Venkatesh, and Narayanan Unny Edakunni. Fairxgboost: Fairness-aware classification in xgboost. *arXiv preprint arXiv:2009.01442*, 2020.

[232] Wenyan Liu, Xiangfeng Wang, Xingjian Lu, Junhong Cheng, Bo Jin, Xiaoling Wang, and Hongyuan Zha. Fair differential privacy can mitigate the disparate impact on model accuracy, 2021.

[233] Kamrun Naher Keya, Rashidul Islam, Shimei Pan, Ian Stockwell, and James R Foulds. Equitable allocation of healthcare resources with fair cox models. *arXiv preprint arXiv:2010.06820*, 2020.

[234] James M Hickey, Pietro G Di Stefano, and Vlasios Vasileiou. Fairness by explicability and adversarial shap learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 174–190. Springer, 2020.

[235] Mohammad Mahdi Kamani. Multiobjective optimization approaches for bias mitigation in machine learning. 2020.

[236] Wenbin Zhang and Jeremy C Weiss. Fair decision-making under uncertainty. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 886–895. IEEE, 2021.

[237] Francesco Ranzato, Caterina Urban, and Marco Zanella. Fairness-aware training of decision trees by abstract interpretation. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 1508–1517, 2021.

[238] Jian Kang, Tiankai Xie, Xintao Wu, Ross Maciejewski, and Hanghang Tong. Multifair: Multi-group fairness in machine learning. *arXiv preprint arXiv:2105.11069*, 2021.

[239] Vincent Grari, Sylvain Lamprier, and Marcin Detyniecki. Fairness-aware neural rényi minimization for continuous features. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 2262–2268, 2021.

[240] Yuyan Wang, Xuezhi Wang, Alex Beutel, Flavien Prost, Jilin Chen, and Ed H Chi. Understanding and improving fairness-accuracy trade-offs in multi-task learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 1748–1757, 2021.

[241] Alan Mishler and Edward Kennedy. Fade: Fair double ensemble learning for observable and counterfactual outcomes. *arXiv preprint arXiv:2109.00173*, 2021.

[242] Andrew Lowy, Rakesh Pavan, Sina Baharlouei, Meisam Razaviyayn, and Ahmad Beirami. Fermi: Fair empirical risk minimization via exponential r\'enyi mutual information. *arXiv preprint arXiv:2102.12586*, 2021.

[243] Tianxiang Zhao, Enyan Dai, Kai Shu, and Suhang Wang. You can still achieve fairness without sensitive attributes: Exploring biases in non-sensitive features. *arXiv preprint arXiv:2104.14537*, 2021.

[244] Mikhail Yurochkin and Yuekai Sun. Sensei: Sensitive set invariance for enforcing individual fairness. In *International Conference on Learning Representations*, 2021.

[245] Tianxiang Zhao, Enyan Dai, Kai Shu, and Suhang Wang. Towards fair classifiers without sensitive attributes: Exploring biases in related features. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pages 1433–1442, 2022.

[246] Jingbo Wang, Yannan Li, and Chao Wang. Synthesizing fair decision trees via iterative constraint solving. In *International Conference on Computer Aided Verification*, pages 364–385. Springer, 2022.

[247] Zhun Deng, Jiayao Zhang, Linjun Zhang, Ting Ye, Yates Coley, Weijie J Su, and James Zou. Fifa: Making fairness more generalizable in classifiers trained on imbalanced data. *arXiv preprint arXiv:2206.02792*, 2022.

[248] Joshua Lee, Yuheng Bu, Prasanna Sattigeri, Rameswar Panda, Gregory W Wornell, Leonid Karlinsky, and Rogerio Schmidt Feris. A maximal correlation framework for fair machine learning. *Entropy*, 24(4):461, 2022.

[249] Wenbin Zhang and Jeremy C Weiss. Longitudinal fairness with censorship. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 12235–12243, 2022.

[250] Zhimeng Jiang, Xiaotian Han, Chao Fan, Fan Yang, Ali Mostafavi, and Xia Hu. Generalized demographic parity for group fairness. In *International Conference on Learning Representations*, 2022.

[251] Joshua Lee, Yuheng Bu, Prasanna Sattigeri, Rameswar Panda, Gregory Wornell, Leonid Karlinsky, and Rogerio Feris. A maximal correlation approach to imposing fairness in machine learning. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3523–3527. IEEE, 2022.

[252] Hyungrok Do, Preston Putzel, Axel S Martin, Padhraic Smyth, and Judy Zhong. Fair generalized linear models with a convex penalty. In *International Conference on Machine Learning*, pages 5286–5308. PMLR, 2022.

[253] Pranita Patil and Kevin Purcell. Decorrelation-based deep learning for bias mitigation. *Future Internet*, 14(4):110, 2022.

[254] Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. *arXiv preprint arXiv:1707.00075*, 2017.

[255] Stephen Gillen, Christopher Jung, Michael Kearns, and Aaron Roth. Online learning with an unknown fairness metric. *Advances in neural information processing systems*, 31, 2018.

[256] Christina Wadsworth, Francesca Vera, and Chris Piech. Achieving fairness through adversarial learning: an application to recidivism prediction. *arXiv preprint arXiv:1807.00199*, 2018.

[257] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2018.

[258] Edward Raff and Jared Sylvester. Gradient reversal against discrimination: A fair neural network learning approach. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 189–198. IEEE, 2018.

[259] Bashir Sadeghi, Runyi Yu, and Vishnu Boddeti. On the global optima of kernelized adversarial representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7971–7979, 2019.

[260] Tameem Adel, Isabel Valera, Zoubin Ghahramani, and Adrian Weller. One-network adversarial fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2412–2420, 2019.

[261] Han Zhao and Geoff Gordon. Inherent tradeoffs in learning fair representations. *Advances in neural information processing systems*, 32, 2019.

[262] L Elisa Celis and Vijay Keswani. Improved adversarial learning for fair classification. *arXiv preprint arXiv:1901.10443*, 2019.

[263] Vincent Grari, Boris Ruf, Sylvain Lamprier, and Marcin Detyniecki. Fair adversarial gradient tree boosting. In *2019 IEEE International Conference on Data Mining (ICDM)*, pages 1060–1065. IEEE, 2019.

[264] Mikhail Yurochkin, Amanda Bower, and Yuekai Sun. Training individually fair ml models with sensitive subspace robustness. In *International Conference on Learning Representations*, 2020.

[265] Adriana Solange Garcia de Alford, Steven K Hayden, Nicole Wittlin, and Amy Atwood. Reducing age bias in machine learning: An algorithmic approach. *SMU Data Science Review*, 3(2):11, 2020.

[266] Yuji Roh, Kangwook Lee, Steven Whang, and Changho Suh. Fr-train: A mutual information-based approach to fair and robust training. In *International Conference on Machine Learning*, pages 8147–8157. PMLR, 2020.

[267] Pieter Delobelle, Paul Temple, Gilles Perrouin, Benoît Frénay, Patrick Heymans, and Bettina Berendt. Ethical adversaries: Towards mitigating unfairness with adversarial machine learning. In *Bias and Fairness in AI (BIAS 2020)*, 2020.

[268] Ashkan Rezaei, Rizal Fathony, Omid Memarrast, and Brian Ziebart. Fairness for robust log loss classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5511–5518, 2020.

[269] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.

[270] Ashkan Rezaei, Anqi Liu, Omid Memarrast, and Brian D Ziebart. Robust fairness under covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9419–9427, 2021.

[271] Guanhong Tao, Weisong Sun, Tingxu Han, Chunrong Fang, and Xiangyu Zhang. Ruler: Discriminative and iterative adversarial training for deep neural network fairness. In *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering, ESEC/FSE 2022*, 2022.

[272] Andrija Petrović, Mladen Nikolić, Sandro Radovanović, Boris Delibašić, and Miloš Jovanović. Fair: Fair adversarial instance re-weighting. *Neurocomputing*, 476:14–37, 2022.

[273] Jenny Yang, Andrew AS Soltan, Yang Yang, and David A Clifton. Algorithmic fairness and bias mitigation for clinical machine learning: Insights from rapid covid-19 diagnosis by adversarial learning. *medRxiv*, 2022.

[274] Mehdi Yazdani-Jahromi, AmirArsalan Rajabi, Aida Tayebi, and Ozlem Ozmen Garibay. Distraction is all you need for fairness. *arXiv preprint arXiv:2203.07593*, 2022.

[275] Toon Calders, Asim Karim, Faisal Kamiran, Wasif Ali, and Xiangliang Zhang. Controlling attribute effect in linear regression. In *2013 IEEE 13th international conference on data mining*, pages 71–80. IEEE, 2013.

[276] Kazuto Fukuchi and Jun Sakuma. Fairness-aware learning with restriction of universal dependency using f-divergences. *arXiv preprint arXiv:1506.07721*, 2015.

[277] Kazuto Fukuchi, Toshihiro Kamishima, and Jun Sakuma. Prediction with model-based neutrality. *IEICE TRANSACTIONS on Information and Systems*, 98(8):1503–1516, 2015.

[278] Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. In *Advances in Neural Information Processing Systems*, pages 2415–2423, 2016.

[279] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial Intelligence and Statistics*, pages 962–970, 2017.

[280] Chris Russell, M Kusner, C Loftus, and Ricardo Silva. When worlds collide: integrating different counterfactual assumptions in fairness. In *Advances in neural information processing systems*, volume 30. NIPS Proceedings, 2017.

[281] Blake Woodworth, Suriya Gunasekar, Mesrob I Ohannessian, and Nathan Srebro. Learning non-discriminatory predictors. In *Conference on Learning Theory*, pages 1920–1953. PMLR, 2017.

[282] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, Krishna P. Gummadi, and Adrian Weller. From parity to preference-based notions of fairness in classification. In *Proceedings of the 31st International Conference*

on Neural Information Processing Systems, NIPS'17, page 228–238, Red Hook, NY, USA, 2017. Curran Associates Inc.

[283] Mahbod Olfat and Anil Aswani. Spectral algorithms for computing fair support vector machines. In *International Conference on Artificial Intelligence and Statistics*, pages 1933–1942. PMLR, 2018.

[284] Harikrishna Narasimhan. Learning with complex loss functions and constraints. In *International Conference on Artificial Intelligence and Statistics*, pages 1646–1654. PMLR, 2018.

[285] Junzhe Zhang and Elias Bareinboim. Fairness in decision-making—the causal explanation formula. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[286] Hoda Heidari, Claudio Ferrari, Krishna Gummadi, and Andreas Krause. Fairness behind a veil of ignorance: A welfare analysis for automated decision making. *Advances in Neural Information Processing Systems*, 31, 2018.

[287] Michael Kim, Omer Reingold, and Guy Rothblum. Fairness through computationally-bounded awareness. *Advances in Neural Information Processing Systems*, 31, 2018.

[288] Golnoosh Farnadi, Behrouz Babaki, and Lise Getoor. Fairness in relational domains. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 108–114, 2018.

[289] Razieh Nabi and Ilya Shpitser. Fair inference on outcomes. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[290] Yongkai Wu, Lu Zhang, and Xintao Wu. Fairness-aware classification: Criterion, convexity, and bounds. *arXiv preprint arXiv:1809.04737*, 2018.

[291] Junzhe Zhang and Elias Bareinboim. Equality of opportunity in classification: A causal approach. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 3675–3685, 2018.

[292] Junpei Komiyama, Akiko Takeda, Junya Honda, and Hajime Shimao. Nonconvex optimization for regression with fairness constraints. In *International conference on machine learning*, pages 2737–2746. PMLR, 2018.

[293] Michele Donini, Luca Oneto, Shai Ben-David, John Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 2796–2806, 2018.

[294] Ananth Balashankar, Alyssa Lees, Chris Welty, and Lakshminarayanan Subramanian. What is fair? exploring pareto-efficiency for fairness constrained classifiers. *arXiv preprint arXiv:1910.14120*, 2019.

[295] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. Fairness constraints: A flexible approach for fair classification. *The Journal of Machine Learning Research*, 20(1):2737–2778, 2019.

[296] Alex Lamy, Ziyuan Zhong, Aditya K Menon, and Nakul Verma. Noise-tolerant fair classification. *Advances in Neural Information Processing Systems*, 32, 2019.

[297] Andrew Cotter, Heinrich Jiang, and Karthik Sridharan. Two-player games for efficient non-convex constrained optimization. In *Algorithmic Learning Theory*, pages 300–332. PMLR, 2019.

[298] Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, Logan Stapleton, and Zhiwei Steven Wu. An algorithmic framework for fairness elicitation. *arXiv preprint arXiv:1905.10660*, 2019.

[299] Andrew Cotter, Heinrich Jiang, Maya R Gupta, Serena Wang, Taman Narayan, Seungil You, and Karthik Sridharan. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *J. Mach. Learn. Res.*, 20(172):1–59, 2019.

[300] Michael Wick, Swetasudha Panda, and Jean-Baptiste Tristan. Unlocking fairness: a trade-off revisited. In *NeurIPS*, 2019.

[301] Andrew Cotter, Maya Gupta, Heinrich Jiang, Nathan Srebro, Karthik Sridharan, Serena Wang, Blake Woodworth, and Seungil You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *International Conference on Machine Learning*, pages 1397–1405. PMLR, 2019.

[302] Razieh Nabi, Daniel Malinsky, and Ilya Shpitser. Learning optimal fair policies. In *International Conference on Machine Learning*, pages 4674–4682. PMLR, 2019.

[303] Depeng Xu, Shuhan Yuan, and Xintao Wu. Achieving differential privacy and fairness in logistic regression. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 594–599, 2019.

[304] Alekh Agarwal, Miroslav Dudík, and Zhiwei Steven Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pages 120–129. PMLR, 2019.

[305] Niki Kilbertus, Manuel Gomez Rodriguez, Bernhard Schölkopf, Krikamol Muandet, and Isabel Valera. Fair decisions despite imperfect predictions. In *International Conference on Artificial Intelligence and Statistics*, pages 277–287. PMLR, 2020.

[306] Michael Lohaus, Michaël Perrot, and Ulrike Von Luxburg. Too relaxed to be fair. In *International Conference on Machine Learning*, pages 6360–6369. PMLR, 2020.

[307] Jiahao Ding, Xinyue Zhang, Xiaohuan Li, Junyi Wang, Rong Yu, and Miao Pan. Differentially private and fair classification via calibrated functional mechanism. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 622–629, 2020.

[308] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression with wasserstein barycenters. *Advances in Neural Information Processing Systems*, 33:7321–7331, 2020.

[309] Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Michael Jordan. Robust optimization for fairness with noisy protected groups. *Advances in Neural Information Processing Systems*, 33:5190–5203, 2020.

[310] Jaewoong Cho, Gyeongjo Hwang, and Changho Suh. A fair classifier using kernel density estimation. *Advances in Neural Information Processing Systems*, 33:15088–15099, 2020.

[311] Luca Oneto, Michele Donini, and Massimiliano Pontil. General fair empirical risk minimization. In *2020 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2020.

[312] Subha Maity, Debarghya Mukherjee, Mikhail Yurochkin, and Yuekai Sun. There is no trade-off: enforcing fairness can improve accuracy. *arXiv preprint arXiv:2011.03173*, 2020.

[313] Evgenii Chzhen and Nicolas Schreuder. A minimax framework for quantifying risk-fairness trade-off in regression. *arXiv preprint arXiv:2007.14265*, 2020.

[314] Manisha Padala and Sujit Gujar. Fnnc: Achieving fairness through neural networks. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence,{IJCAI-20}, International Joint Conferences on Artificial Intelligence Organization*, 2020.

[315] Marco Scutari, Francesca Panero, and Manuel Proissl. Achieving fairness with a simple ridge penalty. *arXiv preprint arXiv:2105.13817*, 2021.

[316] L Elisa Celis, Anay Mehrotra, and Nisheeth Vishnoi. Fair classification with adversarial perturbations. *Advances in Neural Information Processing Systems*, 34:8158–8171, 2021.

[317] L Elisa Celis, Lingxiao Huang, Vijay Keswani, and Nisheeth K Vishnoi. Fair classification with noisy protected attributes: A framework with provable guarantees. In *International Conference on Machine Learning*, pages 1349–1361. PMLR, 2021.

[318] Andrija Petrović, Mladen Nikolić, Miloš Jovanović, Miloš Bijanić, and Boris Delibašić. Fair classification via monte carlo policy gradient method. *Engineering Applications of Artificial Intelligence*, 104:104398, 2021.

[319] Kirtan Padh, Diego Antognini, Emma Lejal-Glaude, Boi Faltings, and Claudiu Musat. Addressing fairness in classification with a model-agnostic multi-objective algorithm. In *Uncertainty in Artificial Intelligence*, pages 600–609. PMLR, 2021.

[320] Chen Zhao, Feng Chen, and Bhavani Thuraisingham. Fairness-aware online meta-learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 2294–2304, 2021.

[321] Hantian Zhang, Xu Chu, Abolfazl Asudeh, and Shamkant B Navathe. Omnifair: A declarative system for model-agnostic group fairness in machine learning. In *Proceedings of the 2021 International Conference on Management of Data*, pages 2076–2088, 2021.

[322] Chenglu Li, Wanli Xing, and Walter Leite. Yet another predictive model? fair predictions of students' learning outcomes in an online math learning platform. In *LAK21: 11th International Learning Analytics and Knowledge Conference*, pages 572–578, 2021.

[323] Valerio Perrone, Michele Donini, Muhammad Bilal Zafar, Robin Schmucker, Krishnaram Kenthapadi, and Cédric Archambeau. Fair bayesian optimiza-

tion. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 854–863, 2021.

[324] Agnieszka Słowik and Léon Bottou. Algorithmic bias and data bias: Understanding the relation between distributionally robust optimization and data curation. *arXiv preprint arXiv:2106.09467*, 2021.

[325] Connor Lawless, Sanjeeb Dash, Oktay Gunluk, and Dennis Wei. Interpretable and fair boolean rule sets via column generation. *arXiv preprint arXiv:2111.08466*, 2021.

[326] YooJung Choi, Meihua Dang, and Guy Van den Broeck. Group fairness by probabilistic modeling with latent fair decisions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12051–12059, 2021.

[327] Saerom Park, Junyoung Byun, and Joohee Lee. Privacy-preserving fair learning of support vector machine with homomorphic encryption. In *Proceedings of the ACM Web Conference 2022*, WWW '22, page 3572–3583, New York, NY, USA, 2022. Association for Computing Machinery.

[328] Chen Zhao, Feng Mi, Xintao Wu, Kai Jiang, Latifur Khan, and Feng Chen. Adaptive fairness-aware online meta-learning for changing environments. *arXiv preprint arXiv:2205.11264*, 2022.

[329] Stelios Boulitsakis-Logothetis. Fairness-aware naive bayes classifier for data with multiple sensitive features. *arXiv preprint arXiv:2202.11499*, 2022.

[330] Shengyuan Hu, Zhiwei Steven Wu, and Virginia Smith. Provably fair federated learning via bounded group loss. *arXiv preprint arXiv:2203.10190*, 2022.

[331] Ling Luo, Wei Liu, Irena Koprinska, and Fang Chen. Discrimination-aware association rule mining for unbiased data analytics. In *International Conference on Big Data Analytics and Knowledge Discovery*, pages 108–120. Springer, 2015.

[332] Matthew Joseph, Michael Kearns, Jamie H Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. *Advances in neural information processing systems*, 29, 2016.

[333] Kory D Johnson, Dean P Foster, and Robert A Stine. Impartial predictive modeling: Ensuring fairness in arbitrary models. *Statistical Science*, page 1, 2016.

[334] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pages 4066–4076, 2017.

[335] Matthew Joseph, Michael Kearns, Jamie Morgenstern, Seth Neel, and Aaron Roth. Meritocratic fairness for infinite and contextual bandits. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 158–163, 2018.

[336] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

[337] Ursula Hébert-Johnson, Michael Kim, Omer Reingold, and Guy Rothblum. Multicalibration: Calibration for the (computationally-identifiable) masses. In *International Conference on Machine Learning*, pages 1939–1948. PMLR, 2018.

[338] Silvia Chiappa and William S Isaac. A causal bayesian networks viewpoint on fairness. In *IFIP International Summer School on Privacy and Identity Management*, pages 3–20. Springer, 2018.

[339] Daniel Alabi, Nicole Immorlica, and Adam Kalai. Unleashing linear optimizers for group-fair learning and optimization. In *Conference On Learning Theory*, pages 2043–2066. PMLR, 2018.

[340] David Madras, Toni Pitassi, and Richard Zemel. Predict responsibly: improving fairness and accuracy by learning to defer. *Advances in Neural Information Processing Systems*, 31, 2018.

[341] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Model-based and actual independence for fairness-aware classification. *Data Mining and Knowledge Discovery*, 32(1):258–286, 2018.

[342] Niki Kilbertus, Adrià Gascón, Matt Kusner, Michael Veale, Krishna Gummadi, and Adrian Weller. Blind justice: Fairness with encrypted sensitive attributes. In *International Conference on Machine Learning*, pages 2630–2639. PMLR, 2018.

[343] Christos Dimitrakakis, Yang Liu, David C Parkes, and Goran Radanovic. Bayesian fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 509–516, 2019.

[344] Joymallya Chakraborty, Tianpei Xia, Fahmid M Fahid, and Tim Menzies. Software engineering for fairness: A case study with hyperparameter optimization. *arXiv preprint arXiv:1905.05786*, 2019.

[345] Alejandro Noriega-Campero, Michiel A Bakker, Bernardo Garcia-Bulle, and Alex'Sandy' Pentland. Active fairness in algorithmic decision making. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 77–83, 2019.

[346] Silvia Chiappa. Path-specific counterfactual fairness. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7801–7808, 2019.

[347] Vasileios Iosifidis and Eirini Ntoutsi. Adafair: Cumulative fairness adaptive boosting. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 781–790, 2019.

[348] Yue Zhang and Arti Ramesh. Learning fairness-aware relational structures. *arXiv preprint arXiv:2002.09471*, 2020.

[349] Debmalya Mandal, Samuel Deng, Suman Jana, Jeannette Wing, and Daniel J Hsu. Ensuring fairness beyond the training data. *Advances in neural information processing systems*, 33:18445–18456, 2020.

[350] André Miguel Ferreira da Cruz. Fairness-aware hyperparameter optimization: An application to fraud detection. 2020.

[351] Vasileios Iosifidis and Eirini Ntoutsi. Fabboo-online fairness-aware learning under class imbalance. In *International Conference on Discovery Science*, pages 159–174. Springer, 2020.

[352] Natalia Martinez, Martin Bertran, and Guillermo Sapiro. Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*, pages 6755–6764. PMLR, 2020.

[353] Alexey Ignatiev, Martin C Cooper, Mohamed Siala, Emmanuel Hebrard, and Joao Marques-Silva. Towards formal fairness in machine learning. In *International Conference on Principles and Practice of Constraint Programming*, pages 846–867. Springer, 2020.

[354] Yahya H Ezzeldin, Shen Yan, Chaoyang He, Emilio Ferrara, and Salman Avestimehr. Fairfed: Enabling group fairness in federated learning. *arXiv preprint arXiv:2110.00857*, 2021.

[355] Jialu Wang, Yang Liu, and Caleb Levy. Fair classification with group-dependent label noise. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 526–536, 2021.

[356] Mustafa Safa Ozdayi, Murat Kantarcioglu, and Rishabh Iyer. Bifair: Training fair models with bilevel optimization. *arXiv preprint arXiv:2106.04757*, 2021.

[357] Rashidul Islam, Shimei Pan, and James R Foulds. Can we obtain fairness for free? In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 586–596, 2021.

[358] Shubham Sharma, Alan H. Gee, David Paydarfar, and Joydeep Ghosh. Fair-n: Fair and robust neural networks for structured data. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '21, page 946–955, New York, NY, USA, 2021. Association for Computing Machinery.

[359] Joshua K Lee, Yuheng Bu, Deepta Rajan, Prasanna Sattigeri, Rameswar Panda, Subhro Das, and Gregory W Wornell. Fair selective classification via sufficiency. In *International Conference on Machine Learning*, pages 6076–6086. PMLR, 2021.

[360] Yuji Roh, Kangwook Lee, Steven Euijong Whang, and Changho Suh. Fair-batch: Batch selection for model fairness. In *9th International Conference on Learning Representations, ICLR 2021*, 2021.

[361] Ana Valdivia, Javier Sánchez-Monedero, and Jorge Casillas. How fair can we go in machine learning? assessing the boundaries of accuracy and fairness. *International Journal of Intelligent Systems*, 36(4):1619–1643, 2021.

[362] Guanchu Wang, Mengnan Du, Ninghao Liu, Na Zou, and Xia Hu. Mitigating algorithmic bias with limited annotations. *arXiv preprint arXiv:2207.10018*, 2022.

[363] Arjun Roy and Eirini Ntoutsi. Learning to teach fairness-aware deep multi-task learning. In *European Conference on Machine Learning and Knowledge Discovery in Databases*.

[364] Sandipan Sikdar, Florian Lemmerich, and Markus Strohmaier. Getfair: Generalized fairness tuning of classification models. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 289–299, 2022.

[365] Yasmine Djebrouni. Towards bias mitigation in federated learning. In *16th EuroSys Doctoral Workshop*, 2022.

[366] Martin B. Short and George O. Mohler. A fully bayesian tracking algorithm for mitigating disparate prediction misclassification. *International Journal of Forecasting*, 2022.

[367] Gaurav Maheshwari and Michaël Perrot. Fairgrad: Fairness aware gradient descent. *arXiv preprint arXiv:2206.10923*, 2022.

[368] Saeid Tizpaz-Niari, Ashish Kumar, Gang Tan, and Ashutosh Trivedi. Fairness-aware configuration of machine learning libraries. In *Proceedings of the 44th International Conference on Software Engineering*, ICSE '22, page 909–920. Association for Computing Machinery, 2022.

[369] Arjun Roy, Vasileios Iosifidis, and Eirini Ntoutsi. Multi-fair pareto boosting. In *International Conference on Discovery Science*.

[370] Kiarash Mohammadi, Aishwarya Sivaraman, and Golnoosh Farnadi. Feta: Fairness enforced verifying, training, and predicting algorithms for neural networks. *arXiv preprint arXiv:2206.00553*, 2022.

[371] Xuanqi Gao, Juan Zhai, Shiqing Ma, Chao Shen, Yufei Chen, and Qian Wang. Fairneuron: Improving deep neural network fairness with adversary games on selective neurons. In *Proceedings of the 44th International Conference on Software Engineering*, ICSE '22, page 921–933, New York, NY, USA, 2022. Association for Computing Machinery.

[372] Xiaoling Huang, Zhenghui Li, Yilun Jin, and Wenyu Zhang. Fair-adaboost: Extending adaboost method to achieve fair classification. *Expert Systems with Applications*, 202:117240, 2022.

[373] Antonio Candelieri, Andrea Ponti, and Francesco Archetti. Fair and green hyperparameter optimization via multi-objective and multiple information source bayesian optimization. *arXiv preprint arXiv:2205.08835*, 2022.

[374] Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. Fair active learning. *Expert Systems with Applications*, 199:116981, 2022.

[375] Xuran Li, Peng Wu, and Jing Su. Accurate fairness: Improving individual fairness without trading accuracy. *arXiv preprint arXiv:2205.08704*, 2022.

[376] Vasileios Iosifidis, Arjun Roy, and Eirini Ntoutsi. Parity-based cumulative fairness-aware boosting. *Knowledge and Information Systems*, Jul 2022.

[377] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Max Leiserson. Decoupled classifiers for group-fair and efficient machine learning. In *Conference on fairness, accountability and transparency*, pages 119–133. PMLR, 2018.

[378] Berk Ustun, Yang Liu, and David Parkes. Fairness without harm: Decoupled classifiers with preference guarantees. In *International Conference on Machine Learning*, pages 6373–6382. PMLR, 2019.

[379] Wellington Rodrigo Monteiro and Gilberto Reynoso-Meza. Proposal of a fair voting classifier using multi-objective optimization, jul 2021.

[380] Kenji Kobayashi and Yuri Nakao. One-vs.-one mitigation of intersectional bias: A general method for extending fairness-aware binary classification. In *International Conference on Disruptive Technologies, Tech Ethics and Artificial Intelligence*, pages 43–54. Springer, 2021.

[381] Jiayin Jin, Zeru Zhang, Yang Zhou, and Lingfei Wu. Input-agnostic certified group fairness via gaussian parameter smoothing. In *International Conference on Machine Learning*, pages 10340–10361. PMLR, 2022.

[382] Suyun Liu and Luis Nunes Vicente. Accuracy and fairness trade-offs in machine learning: A stochastic multi-objective approach. *Computational Management Science*, pages 1–25, 2022.

[383] William Blanzeisky and Pádraig Cunningham. Using pareto simulated annealing to address algorithmic bias in machine learning. *The Knowledge Engineering Review*, 37:e5, 2022.

[384] Mingyang Wan, Daochen Zha, Ninghao Liu, and Na Zou. In-processing modeling techniques for machine learning fairness: A survey. *ACM Trans. Knowl. Discov. Data*, jul 2022. Just Accepted.

[385] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, and Deepak Verma. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 99–108, 2004.

[386] Wellington Rodrigo Monteiro and Gilberto Reynoso-Meza. Proposal of a fair voting classifier using multi-objective optimization.

[387] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. The independence of fairness-aware classifiers. In *2013 IEEE 13th International Conference on Data Mining Workshops*, pages 849–858. IEEE, 2013.

[388] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

[389] Michael Emmerich and André H Deutz. A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural computing*, 17(3):585–609, 2018.

[390] Philip Adler, Casey Falk, Sorelle A Friedler, Tionney Nix, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. Auditing black-box models for indirect influence. *Knowledge and Information Systems*, 54(1):95–122, 2018.

[391] Giulio Morina, Viktoriia Oliinyk, Julian Waton, Ines Marusic, and Konstantinos Georgatzis. Auditing and achieving intersectional fairness in classification problems. *arXiv preprint arXiv:1911.01468*, 2019.

[392] Michael P Kim, Amirata Ghorbani, and James Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 247–254, 2019.

[393] Kentaro Kanamori and Hiroki Arimura. Fairness-aware edit of thresholds in a learned decision tree using a mixed integer programming formulation. In *The 33rd Annual Conference of the Japanese Society for Artificial Intelligence (2019)*, pages 3Rin211–3Rin211. The Japanese Society for Artificial Intelligence, 2019.

[394] Yash Savani, Colin White, and Naveen Sundar Govindarajulu. Intra-processing methods for debiasing neural networks. *Advances in Neural Information Processing Systems*, 33:2798–2810, 2020.

[395] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Fair regression via plug-in estimator and recalibration with statistical guarantees. *Advances in Neural Information Processing Systems*, 33:19137–19148, 2020.

[396] Pranjal Awasthi, Matthäus Kleindessner, and Jamie Morgenstern. Equalized odds postprocessing under imperfect group information. In *International Conference on Artificial Intelligence and Statistics*, pages 1770–1780. PMLR, 2020.

[397] Nicolas Schreuder and Evgenii Chzhen. Classification with abstention but without disparities. In *Uncertainty in Artificial Intelligence*, pages 1227–1236. PMLR, 2021.

[398] Kentaro Kanamori and Hiroki Arimura. Fairness-aware decision tree editing based on mixed-integer linear optimization. *Transactions of the Japanese Society for Artificial Intelligence*, 36(4):B–L13_1, 2021.

[399] Alan Mishler, Edward H. Kennedy, and Alexandra Chouldechova. Fairness in risk assessment instruments: Post-processing to achieve counterfactual

equalized odds. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 386–400, New York, NY, USA, 2021. Association for Computing Machinery.

[400] Mengnan Du, Subhabrata Mukherjee, Guanchu Wang, Ruixiang Tang, Ahmed Awadallah, and Xia Hu. Fairness via representation neutralization. *Advances in Neural Information Processing Systems*, 34:12091–12103, 2021.

[401] Przemyslaw A Grabowicz, Nicholas Perello, and Aarshee Mishra. Marrying fairness and explainability in supervised learning. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 1905–1916, 2022.

[402] Jiang Zhang, Ivan Beschastnikh, Sergey Mechtaev, and Abhik Roychoudhury. Fair decision making via automated repair of decision trees. In *International Workshop on Equitable Data and Technology (FairWare '22)*, 2022.

[403] Ninareh Mehrabi, Umang Gupta, Fred Morstatter, Greg Ver Steeg, and Aram Galstyan. Attributing fair decisions with attention interventions. In *Proceedings of the 2nd Workshop on Trustworthy Natural Language Processing (TrustNLP 2022)*, pages 12–25, Seattle, U.S.A., July 2022. Association for Computational Linguistics.

[404] Ziwei Wu and Jingrui He. Fairness-aware model-agnostic positive and unlabeled learning. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 1698–1708, New York, NY, USA, 2022. Association for Computing Machinery.

[405] Ricards Marcinkevics, Ece Ozkan, and Julia E Vogt. Debiasing deep chest x-ray classifiers using intra- and post-processing methods. In *Machine Learning for Healthcare Conference*.

[406] Dino Pedreschi, Salvatore Ruggieri, and Franco Turini. Measuring discrimination in socially-sensitive decision records. In *Proceedings of the 2009 SIAM international conference on data mining*, pages 581–592. SIAM, 2009.

[407] Benjamin Fish, Jeremy Kun, and Ádám D Lelkes. A confidence-based approach for balancing fairness and accuracy. In *Proceedings of the 2016 SIAM International Conference on Data Mining*, pages 144–152. SIAM, 2016.

[408] Aditya Krishna Menon and Robert C Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pages 107–118. PMLR, 2018.

[409] Jixue Liu, Jiuyong Li, Lin Liu, Thuc Duy Le, Feiyue Ye, and Gefei Li. Fairmod-making predictive models discrimination aware. *arXiv preprint arXiv:1811.01480*, 2018.

[410] Evgenii Chzhen, Christophe Denis, Mohamed Hebiri, Luca Oneto, and Massimiliano Pontil. Leveraging labeled and unlabeled data for consistent fair binary classification. *Advances in Neural Information Processing Systems*, 32, 2019.

[411] Ibrahim Alabdulmohsin. Fair classification via unconstrained optimization. *arXiv preprint arXiv:2005.14621*, 2020.

[412] Ibrahim M Alabdulmohsin and Mario Lucic. A near-optimal algorithm for debiasing trained machine learning models. *Advances in Neural Information Processing Systems*, 34:8072–8084, 2021.

[413] Dang Nguyen, Sunil Gupta, Santu Rana, Alistair Shilton, and Svetha Venkatesh. Fairness improvement for black-box classifiers with gaussian process. *Information Sciences*, 576:542–556, 2021.

[414] Pranay Lohia. Priority-based post-processing bias mitigation for individual and group fairness. *arXiv preprint arXiv:2102.00417*, 2021.

[415] Taeuk Jang, Pengyi Shi, and Xiaoqian Wang. Group-aware threshold adaptation for fair classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6988–6995, 2022.

[416] P. Snel and S. van Otterloo. Practical bias correction in neural networks: a credit default prediction case study. *Computers and Society Research Journal*, (3), 2022.

[417] Wael Alghamdi, Hsiang Hsu, Haewon Jeong, Hao Wang, P Winston Michalak, Shahab Asoodeh, and Flavio P Calmon. Beyond adult and compas: Fairness in multi-class prediction. *arXiv preprint arXiv:2206.07801*, 2022.

[418] Xianli Zeng, Edgar Dobriban, and Guang Cheng. Fair bayes-optimal classifiers under predictive parity. *arXiv preprint arXiv:2205.07182*, 2022.

[419] Xianli Zeng, Edgar Dobriban, and Guang Cheng. Bayes-optimal classifiers under group fairness. *arXiv preprint arXiv:2202.09724*, 2022.

[420] Xiaoxin Yin and Jiawei Han. Cpar: Classification based on predictive association rules. In *Proceedings of the 2003 SIAM international conference on data mining*, pages 331–335. SIAM, 2003.

[421] Bhavya Ghai, Mihir Mishra, and Klaus Mueller. Cascaded debiasing: Studying the cumulative effect of multiple fairness-enhancing interventions. *arXiv preprint arXiv:2202.03734*, 2022.

[422] Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in Neural Information Processing Systems*, 34:6478–6490, 2021.

[423] Michael Redmond and Alok Baveja. A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141(3):660–678, 2002.

[424] Linda F Wightman. Lsac national longitudinal bar passage study. lsac research report series. 1998.

[425] I-Cheng Yeh and Che-hui Lien. The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients. *Expert systems with applications*, 36(2):2473–2480, 2009.

[426] Dutch central bureau for statistics volkstelling. `http://easy.dans.knaw.nl/dms`, 2001. Retrieved on June 12, 2022.

[427] The heritage health prize dataset. `https://www.kaggle.com/c/hhp`, 2017. Retrieved on June 12, 2022.

[428] Elaine Fehrman, Awaz K Muhammad, Evgeny M Mirkes, Vincent Egan, and Alexander N Gorban. The five factor model of personality and evaluation of drug consumption risk. In *Data science*, pages 231–242. Springer, 2017.

[429] Paulo Cortez and Alice Maria Gonçalves Silva. Using data mining to predict secondary school student performance. 2008.

[430] National longitudinal surveys of youth data set. `www.bls.gov/nls/`, 2019. Retrieved on June 12, 2022.

[431] Stop, question and frisk dataset. `http://www1.nyc.gov/site/nypd/stats/reports-analysis/stopfrisk.page`, 2017. Retrieved on June 12, 2022.

[432] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision support systems*, 47(4):547–553, 2009.

[433] Supreme Court of the United States. *Ricci v. DeStefanoo*, volume 557. 2009.

[434] Home credit default risk. `https://www.kaggle.com/c/home-credit-default-risk`, 2018. Retrieved on June 12, 2022.

[435] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[436] Thomas Kehrenberg, Zexun Chen, and Novi Quadrianto. Tuning fairness by marginalizing latent target labels. *stat*, 1050:10, 2019.

[437] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464):447–453, 2019.

[438] Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miro Dudik, and Hanna Wallach. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pages 1–16, 2019.

[439] Christopher Anders, Plamen Pasliev, Ann-Kathrin Dombrowski, Klaus-Robert Müller, and Pan Kessel. Fairwashing explanations with off-manifold detergent. In *International Conference on Machine Learning*, pages 314–323. PMLR, 2020.

[440] Jared Sylvester and Edward Raff. Trimming the thorns of ai fairness research. *IEEE Data Eng. Bull.*, 43(4):74–84, 2020.

[441] Sarah Bird, Miro Dudík, Richard Edgar, Brandon Horn, Roman Lutz, Vanessa Milan, Mehrnoosh Sameki, Hanna Wallach, and Kathleen Walker. Fairlearn: A toolkit for assessing and improving fairness in ai. *Microsoft, Tech. Rep. MSR-TR-2020-32*, 2020.

[442] Max Hort, Jie Zhang, Federica Sarro, and Mark Harman. On-line appendix, fairea, 2021.

[443] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. An empirical study of rich subgroup fairness for machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 100–109, 2019.

[444] Sumon Biswas and Rajan Hridesh. Do the machine learning models on a crowd sourced platform exhibit bias? an empirical study on model fairness. *arXiv preprint arXiv:2005.12379*, 2020.

[445] Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

[446] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

[447] On-line appendix, post-processing, 2022.

[448] Mark Harman, Phil McMinn, Jerffeson Teixeira De Souza, and Shin Yoo. Search based software engineering: Techniques, taxonomy, tutorial. In *Empirical software engineering and verification*, pages 1–59. Springer, 2010.

[449] Hisao Ishibuchi, Hiroyuki Masuda, Yuki Tanigaki, and Yusuke Nojima. Modified distance calculation in generational distance and inverted generational distance. In *International conference on evolutionary multi-criterion optimization*, pages 110–125. Springer, 2015.

[450] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[451] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

[452] J. Ross Quinlan. Simplifying decision trees. *International journal of man-machine studies*, 27(3):221–234, 1987.

[453] Leonard A Breslow and David W Aha. Simplifying decision trees: A survey. *Knowledge engineering review*, 12(1):1–40, 1997.

[454] F. Ferrucci, C. Gravino, R. Oliveto, and F. Sarro. Genetic programming for effort estimation: An analysis of the impact of different fitness functions. In *2nd International Symposium on Search Based Software Engineering*, pages 89–98, 2010.

[455] Frank Wilcoxon. Individual comparisons by ranking methods. In *Breakthroughs in statistics*, pages 196–202. Springer, 1992.

[456] F. Sarro, F. Ferrucci, M. Harman, A. Manna, and J. Ren. Adaptive multi-objective evolutionary algorithms for overtime planning in software projects. *IEEE TSE*, 43(10):898–917, 2017.

[457] Ekrem Kocaguneli, Tim Menzies, and Jacky W Keung. On the value of ensemble effort estimation. *IEEE TSE*, 38(6):1403–1416, 2011.

[458] F. Sarro, M. Harman, Y. Jia, and Y. Zhang. Customer rating reactions can be predicted purely using app features. In *IEEE International Requirements Engineering Conference*, pages 76–87, 2018.

[459] F. Sarro and A. Petrozziello. Linear programming as a baseline for software effort estimation. *ACM TOSEM*, 27(3):12:1–12:28, 2018.

[460] András Vargha and Harold D Delaney. A critique and improvement of the cl common language effect size statistics of mcgraw and wong. *Journal of Educational and Behavioral Statistics*, 25(2):101–132, 2000.

[461] A. Arcuri and L. Briand. A hitchhiker's guide to statistical tests for assessing randomized algorithms in software engineering. *STVR*, 24(3):219–250, 2014.

[462] Federica Sarro, Alessio Petrozziello, and Mark Harman. Multi-objective software effort estimation. In *Procs. of the International Conference on Software Engineering (ICSE)*, pages 619–630. IEEE, 2016.

[463] Tim Menzies, Alex Dekhtyar, Justin Distefano, and Jeremy Greenwald. Problems with precision: A response to ”comments on ’data mining static

code attributes to learn defect predictors'". *IEEE Transactions on Software Engineering*, 33(9):637–640, 2007.

[464] Abigail Z. Jacobs and Hanna Wallach. Measurement and fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 375–385, New York, NY, USA, 2021. Association for Computing Machinery.

[465] Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Serena Wang. Pairwise fairness for ranking and regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5248–5255, 2020.

[466] Faisal Kamiran and Indrė Žliobaitė. Explainable and non-explainable discrimination in classification. In *Discrimination and Privacy in the Information Society*, pages 155–170. Springer, 2013.

[467] Joseph Lee Rodgers and W Alan Nicewander. Thirteen ways to look at the correlation coefficient. *The American Statistician*, 42(1):59–66, 1988.

[468] James D Evans. *Straightforward statistics for the behavioral sciences.* Thomson Brooks/Cole Publishing Co, 1996.