DISSERTATIONES
INFORMATICAE
UNIVERSITATIS
TARTUENSIS
81

MODAR SULAIMAN

From Data to Fair Decisions: On Ensuring Fairness in Machine Learning Models

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

81

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS
81

MODAR SULAIMAN

From Data to Fair Decisions: On Ensuring Fairness in Machine Learning Models

UNIVERSITY OF TARTU
Press
1632

Institute of Computer Science, Faculty of Science and Technology, University of Tartu, Estonia.

Dissertation has been accepted for the commencement of the degree of Doctor of Philosophy (PhD) in Computer Science on March 31, 2026 by the Council of the Institute of Computer Science, University of Tartu.

Supervisor

Assist. Prof. Kallol Roy
University of Tartu, Estonia

Opponents

Prof. Catuscia Palamidessi
National Institute for Research in Digital Science and Technology, France

Assoc. Prof. Özlem Özgöbek
Norwegian University of Science and Technology, Norway

The public defense will take place on May 4, 2026 at 11:00 in Narva Rd. 18-1017.

The publication of this dissertation was financed by the Institute of Computer Science, University of Tartu.

ISSN 2613-5906 (print) ISSN 2806-2345 (pdf)
ISBN 978-9908-57-185-0 (print) ISBN 978-9908-57-186-7 (pdf)

University of Tartu Press
http://www.tyk.ee/

To all marginalized and underrepresented communities

ABSTRACT

Machine learning systems are increasingly used across finance, healthcare, commerce, and the public sector. While these systems can deliver strong predictive utility, they can also produce systematic disparities across demographic groups. A central challenge is therefore the *fairness–performance trade-off*: reducing group disparities without sacrificing too much predictive performance. This thesis addresses that trade-off in supervised learning through three complementary intervention points in the learning pipeline: *model architecture*, *data composition and optimization*, and *label quality*.

First, we introduce *The Fairness Stitch* (TFS), an in-processing architectural paradigm that inserts a lightweight, trainable 'stitch' layer into a pre-trained network and optimizes it under fairness-aware constraints while keeping the backbone fixed. By moving beyond last-layer-only adjustments, TFS shows that effective debiasing may require intervention where biased representations are formed (before the last layer), not only where decisions are made (in the last layer). It therefore provides a practical mechanism for improving fairness with limited retraining.

Second, we propose *Group-Level Cost-Sensitive Deep Learning* (GLCS), which addresses group-level class imbalance by translating imbalance into group-aware costs within a constrained risk-minimization framework that targets *Equal Opportunity*. The formulation is modular, can be applied to targeted groups, and extends naturally to the setting of group robustness, offering a principled way to improve fairness and group robustness without undue accuracy loss.

Third, we present *Graph-based Fairness-aware Label Correction* (GFLC), a graph-based method that improves learning under biased or noisy supervision by coupling graph-guided consistency with an uncertainty-aware and parity-sensitive objective under *Demographic Parity*. GFLC upgrades supervision where it matters for downstream equity by identifying and correcting label noise patterns that would otherwise distort both accuracy and fairness.

Together, these three contributions support a unified conclusion: fairness is best treated as a *pipeline-level design property*, and fairness interventions are most effective when applied at the stage where bias is introduced or reinforced: in representations (TFS), in data and optimization (GLCS), or in supervision (GFLC). The thesis provides design principles and empirical evidence toward building classifiers that are not only accurate but also equitable under realistic constraints and deployment considerations.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronyms

ABROCA Absolute Between-ROC Area. 10, 47–49
AE Accuracy Equality. 25, 48
AE-Diff Accuracy Equality Difference. 11, 47, 49, 50
AF Balanced Accuracy and Fairness metric. 47
AI Artificial Intelligence. 18
ALM Augmented Lagrangian Method. 55
AP Average Precision. 60, 74
AUCP ROC AUC Parity. 62, 68, 69, 72

BACC Balanced Accuracy. 47, 66, 67, 70, 71
BERT Bidirectional Encoder Representations from Transformers. 64
BFN Balance for Negative Class. 62
BFP Balance for Positive Class. 62
BNSP-AUC Background Negative Subgroup Positive AUC. 63
BPSN-AUC Background Positive Subgroup Negative AUC. 63

DL Deep Learning. 20
DP Demographic Parity. 86, 89

EO-Diff Equalized Odds Difference. 47

FDR Fair Deep Feature Reweighting. 10, 44–51
FNR False Negative Rate. 89
FPR False Positive Rate. 89

GLCS Group-Level Cost-Sensitive Deep Learning. 53

ML Machine Learning. 18

OBNC Ordering-Based Label Noise Correction. 88

TNR True Negative Rate. 89
TPR True Positive Rate. 89

LIST OF ORIGINAL PUBLICATIONS

Publications included in the thesis

1. **Modar Sulaiman** and Kallol Roy. "The Fairness Stitch: A Novel Approach for Neural Network Debiasing". In: *Acta Informatica Pragensia* 13.3 (Aug. 2024), pp. 359–373. DOI: 10.18267/j.aip.241.
   First author: The first author led the conceptualization and study design, implemented the methodology, conducted the experiments, and performed the analysis. Moreover, the first author produced the visualizations (figures and tables), drafted the whole manuscript, and interpreted the results in relation to the research question.
2. **Modar Sulaiman**, Nesma Talaat Abbas Mahmoud, and Kallol Roy. "Advancing Equal Opportunity Fairness and Group Robustness through Group-Level Cost-Sensitive Deep Learning". In: *Baltic Journal of Modern Computing* 13.1 (2025), pp. 96–127. DOI: 10.22364/bjmc.2025.13.1.06.
   First author: The first author conceived the idea and translated it into a concrete study design, implemented the method GLCS, and built the full evaluation framework. In addition, the first author conducted a preliminary exploration that informed the final scope of the study and completed all the experiments. Moreover, the first author cleaned the datasets, conducted the whole experiment, analyzed the results, and produced all visualizations and tables. Furthermore, the first author drafted the manuscript.
3. **Modar Sulaiman** and Kallol Roy. "GFLC: Graph-based Fairness- aware Label Correction for Fair Classification". In: *Baltic Journal of Modern Computing* 13.3 (2025), pp. 567–591. DOI: 10.22364/bjmc.2025.13.3.02.
   First author: The first author took full responsibility for this contribution from inception to dissemination. Additionally, the first author designed the methodology, implemented and tested the model, established the experimental protocols, and analyzed the outcomes. Finally, the first author generated all figures and tables, wrote and revised the manuscript.

Publications not included in the thesis

4. **Modar Sulaiman** and Kallol Roy. "Fair Classification via Transformer Neural Networks: Case Study of an Educational Domain". In: *The FATED (Fairness, Accountability, and Transparency in Educational Data) 2022 Workshop at EDM (The 15th International Conference on Educational Data Mining).*

1. INTRODUCTION

1.1. Overview

Machine learning has become fundamental to decision-making processes in finance, healthcare, commercial platforms, and the public sector. This adoption is not just experimental; organizations now depend on deployed systems in critical settings. These models support tasks such as fraud detection [Ali+22] and personalized content delivery [Ste+21], while clinical implementations have shown measurable benefits for decision support and patient outcomes [Han+24a]. The effectiveness of such systems, however, depends as much on the data and the deployment context as on the algorithms themselves.

Real-world datasets are often imperfect and may encode judgments made by prior decision makers, mirror structural inequities in society, or exhibit regularities that serve as proxies for historical patterns of exclusion [BS16]. When such artifacts are learned and amplified by the models, performance can differ systematically across demographic groups, thereby reinforcing existing disparities. Empirical studies have documented underdiagnosis risks for minority populations in clinical prediction [Sey+21] and discriminatory failures in face recognition systems trained on biased data [Bol+16; Jen+19]. Bolukbasi et al. [Bol+16] demonstrated that widely used word embeddings (e.g., Word2Vec) encode gender stereotypes, as evidenced by completing the analogy 'man is to computer programmer as woman is to X' with 'homemaker', thereby propagating such biases throughout any system that utilizes these embeddings.

The field of *fairness in machine learning* seeks to characterize, measure, and mitigate these disparities. In addition, various group fairness metrics and mitigation strategies have emerged in this field. However, at its core is the question of *intervention*: how to reduce harm caused by bias without compromising the intended utility of the system. In particular, detecting and mitigating bias in supervised learning for a given task often reveals the need to find a reasonable trade-off between fairness and performance.

In general, gains in fairness may sometimes require accepting a reduction in performance, and vice versa. This trade-off problem between fairness and predictive performance has been widely recognized and studied in the literature. For example, Zhao and Gordon [ZG22] identified an inherent trade-off between one of the commonly used fairness notions *statistical parity* and performance in the classification task. Specifically, they provided a lower bound on the sum of group-wise classification errors that any classifier satisfying statistical parity must incur. This thus formalizes the cost of enforcing fairness constraints in supervised learning. Xian et al. [XYZ23] presented a characterization of the inherent trade-off induced by enforcing demographic parity in classification tasks, formulated under a general setting that allows for multiple groups, multiple classes, and label noise.

In summary, developing more advanced methods that address the trade-off between fairness and performance is crucial for building models that are technically sound, responsible, and equitable in real-world contexts.

Accordingly, this thesis approaches the fairness–performance trade-off along three complementary angles. Despite growing interest in fair machine learning, important research gaps remain in understanding how this trade-off is shaped by decisions made at different stages of supervised learning. Prior work has shown that data can encode historical inequities and proxy signals that may later translate into disparate outcomes. However, much of the existing literature addresses fairness mitigation in a relatively isolated manner, for example, through post hoc constraints or task-specific adjustments, rather than through a more integrated analysis of the learning pipeline. More specifically, although a wide range of fairness definitions, metrics, and mitigation strategies has been proposed, the literature still provides limited guidance on how decisions related to model architecture, data composition, and labeling influence the fairness–performance trade-off in practice. Addressing these gaps is essential for moving toward learning systems that are both practically deployable and socially responsible. Against this background, the research questions of this thesis are organized around three core dimensions: model architecture, data composition, and labeling.

1.2. Research questions

This thesis addresses the fairness-performance trade-off by proposing three approaches, each through a different angle and presented as a distinct contribution that influences the balance achieved by machine-learning classifiers. We structure the thesis around the following three guiding questions that span model design, data composition, and labeling:

1. **Model Architecture.** Which architectural modifications to neural networks can effectively improve the fairness-performance balance of classifiers?
2. **Data.** How can *dataset distribution* issues, such as group-level class imbalance, be mitigated to enhance the balance between fairness and predictive utility?
3. **Labels.** Given that perfectly clean labels are rare in practice, how can supervised machine learning models learn under biased or noisy supervision to improve fairness while maintaining competitive performance?

1.3. Contributions

This thesis introduces new methods to tackle the trade-off between fairness and performance in supervised learning along three different angles, aligned with the previously defined questions and based on the three fundamental items in supervised learning (model architecture, data, and labels). In summary, the key contributions can be viewed as follows:

1. In the first contribution [SR24], we propose an innovative framework, 'The Fairness Stitch', to better mitigate bias better while preserving model performance, and empirically proving the limitation of last-layer fine-tuning in attaining an optimal balance between fairness and performance, as well as the need to consider the layers preceding the last layer.
2. In the second contribution [SMR25], we introduce GLCS (Group-Level Cost-Sensitive Deep Learning) that addresses group-level class imbalances, enabling more nuanced handling of demographic disparities in machine learning systems.
3. In the third contribution [SR25], we introduce Graph-based Fairness-aware Label Correction (GFLC), a novel method to address the problem of instance-dependent label noise. Moreover, GFLC provides high-quality corrections for each data point in the dataset and achieves a good trade-off between fairness and performance.

1.4. Thesis Outline

Across seven chapters, this thesis provides the background necessary for a coherent and incremental treatment of fairness in machine learning. The chapters are sequenced to guide the reader from foundational material to more specialized topics. The rest of the manuscript is organized as follows.

Chapter 2 provides essential background, covering machine learning background, supervised learning, supervised machine learning techniques, AI Bias, and types of bias. It also introduces key concepts of fairness in machine learning, reviews core fairness notions, and discusses the various categories of debiasing approaches.

Chapter 3 surveys state-of-the-art methods for fairness in machine learning. The discussion classifies methods by their alignment with the debiasing paradigm used in each contribution and the methodological domain from which each approach originates. We analyze the existing techniques, identifying open problems with particular attention to the fairness-performance trade-off.

Chapter 4 presents our first contribution, which focuses on modifying the neural network architecture for improving the fairness–performance trade-off. In particular, we introduce 'model stitching', which links two distinct networks to leverage complementary representations, and extend this idea within a single network (self-stitching). Building on these mechanisms, we develop *The Fairness Stitch* framework, an explicit in-processing debiasing method that integrates stitching with fairness constraints. Our results challenge the prevailing assumption that last-layer adjustments suffice for bias mitigation.

Chapter 5 presents the second key contribution, 'GLCS: ADVANCING EQUAL OPPORTUNITY FAIRNESS AND GROUP ROBUSTNESS THROUGH GROUP-LEVEL COST-SENSITIVE DEEP LEARNING'. In the proposed method GLCS, we address group-wise class imbalance defined by a specified sensitive attribute. We formulate

the method as a constrained optimization that targets the *equal opportunity* criterion while preserving predictive performance. We further extend the framework to the setting of group robustness.

In Chapter 6, we introduce 'GFLC: GRAPH-BASED FAIRNESS-AWARE LABEL CORRECTION FOR FAIR CLASSIFICATION'. Motivated by the difficulty of obtaining a clean and accurate set of features and labels in a dataset, and by the observation that noisy annotations can exacerbate group disparities and may even cause fairness constraints to backfire, we design GFLC to reconcile predictive accuracy with *demographic parity* in the presence of label noise.

Chapter 7 concludes the thesis by consolidating key results and contributions. It reflects on their impact within the field of fairness in machine learning and delineates promising directions for subsequent inquiry.

2. BACKGROUND

Fairness in machine learning is an active area of research at the intersection of statistics, computer science, and the social sciences. Its central aim is to characterize, measure, and mitigate systematic disparities in model behavior across populations. This chapter provides the foundation of fairness in machine learning. We begin with essential machine learning preliminaries, focusing on the supervised learning setup and the two principal task types: classification and regression. We then explain the bias problem in this thesis, its origins in the data, and relate these ideas to the broader discourse on bias and fairness in AI. Building on this basis, we present core fairness notions commonly used in supervised settings, along with their practical interpretation. Next, we discuss the trade-off between predictive performance and fairness in machine learning. Finally, we provide the main categories of debiasing methodologies, starting with the classical triad of pre-processing, in-processing, and post-processing, and then turning to contemporary strategies tailored to modern neural systems: distributional approaches, one-step training, multi-step training, and inferential methods.

2.1. Machine learning

Machine Learning (ML) is a foundational subfield of Artificial Intelligence (AI), enabling systems to improve with experience rather than explicit programming [Dri21]. As a branch of computer science, ML focuses on discovering patterns in data to enhance performance across tasks [JGR20]. Supervised, unsupervised, and reinforcement learning are widely recognized as the three major categories of machine learning [Dri21]. In supervised learning, a model learns to map inputs to outputs using labeled examples, enabling it to make predictions on unseen data. Unsupervised learning, on the other hand, searches for latent structure in unlabeled datasets, such as clusters or lower-dimensional representations, without predefined targets. Reinforcement learning trains an agent to make sequential decisions by interacting with an environment and optimizing cumulative reward. In this thesis, our focus is mainly on supervised learning.

2.1.1. Supervised Learning

Supervised learning is an important approach in artificial intelligence that focuses on learning from examples where the correct answer is known in advance. In particular, using labeled data, a supervised learning algorithm learns a function that maps inputs to outputs and can later apply that learned relationship to new data. Moreover, supervised learning is used mainly for two tasks: classification and regression [Dri21].

Classification Task. The goal of the classification task is to place an observation into one of several predefined categories. Classification is a supervised learning

setting in which models learn from labeled examples to assign inputs to discrete categories [SHG19] [Kad19], and then apply that learned mapping to previously unseen data [Dri21] [MY15]. A canonical example is MNIST, where training is performed on images of handwritten digits paired with their corresponding labels (0–9), and then the correct digit is predicted for new images. To rigorously evaluate performance in a classification task, various performance metrics can be employed, such as precision, recall, and the F1-score (macro, micro, and weighted). We can inspect the confusion matrix to reveal systematic errors and, in addition, the Receiver Operating Characteristic-Area Under the Curve (ROC-AUC) and the area under the precision-recall (PR) curve (PR-AUC).

Regression Task. Regression is a supervised learning task for predicting a numerical rather than a categorical target. It learns a function f that links input features to a continuous target [Dri21] [SHG19]. In its most familiar form, *simple linear regression* fits a straight line using a single feature; *multiple linear regression* extends this idea to various features, allowing the model to weigh each input [Dri21]. When relationships are not linear, we can add transformed features or use nonlinear models, while maintaining the same goal: reliable numeric prediction. To evaluate performance, standard measures include the mean squared error (MSE), which averages the squared gaps between predictions and actual values, and R^2 (also known as "R-squared"), which reports how much of the variation in the target the model explains.

Because explicit targets guide the learning process, supervised methods are used in settings that demand accuracy and consistent performance, often without consideration for fairness. This thesis focuses mainly on classification in supervised learning, with fairness considerations. Widely used methods form the backbone of classification workflows across various structured and unstructured datasets [Dri21], ranging from visual to tabular and textual data. In this thesis, we consider the supervised classification task across multiple data modalities. In particular, this setting provides the common methodological basis used in this thesis for investigating how fairness-aware interventions at the level of model architecture, data composition, and label quality influence the fairness–performance trade-off.

2.1.2. Supervised Machine Learning Techniques

In this section, we outline five major families of supervised machine learning techniques [MY15]:

Logic-Based Algorithms. Those types of algorithms include two symbolic approaches: *decision trees* and *rule-based classifiers* [MY15; KZP+07]. A decision tree classifies an instance by routing it through tests on its feature values. Each node denotes a feature of the instance, and each branch corresponds to a value that the feature can take. Classification begins at the root and proceeds by this value-based sorting [KZP+07; MY15]. For learning sets of rules, decision trees can be translated into a set of rules by creating a separate rule for each path from

the root to a leaf in the tree [Sal94; MY15].

Statistical Learning Algorithms. Statistical learning provides a principled framework for machine learning, drawing on statistics and functional analysis [MRT18; MY15]. Its central aim is to formalize inference, that is, to construct models from data to extract knowledge, generate predictions, and support decision-making [BBL03].

Instance-Based (Lazy) Learning.. Instance-based learning is often described as *lazy learning* because it defers induction or generalization until classification time [MCM86]. Compared with eager approaches such as decision trees and neural networks, these methods typically require less computation during training but more at prediction time, where the work is performed on demand. A canonical example is the nearest-neighbor algorithm, which classifies by referencing stored instances [KZP+07; MY15].

Support Vector Machines (SVMs). Support Vector Machines are supervised learning methods that can be applied to various tasks in machine learning, including classification, regression, and outlier detection [MY15]. Furthermore, SVMs perform reliably in high-dimensional feature spaces. Because the decision function depends only on a subset of training examples, the *support vectors*, both storage and evaluation can be efficient. Moreover, SVMs are also versatile: by choosing an appropriate kernel, the decision boundary can be tailored to the problem, from a simple linear separator to richly nonlinear relationships.

Deep Learning (DL). DL is a branch of machine learning (ML), widely regarded as a core technology of the Fourth Industrial Revolution (4IR, also known as Industry 4.0). Rooted in artificial neural networks (ANNs) and driven by data-driven learning, DL has become a central topic in computing, with broad applications in healthcare, visual recognition, text analytics, cybersecurity, and beyond [Sar21]. Within this paradigm, the *'Deep Networks for Supervised Learning'* family provides discriminative functions for supervised and classification tasks. Some representative models in this family include the Multi-Layer Perceptron (MLP), Convolutional Neural Networks (CNN/ConvNet), Recurrent Neural Networks (RNN), and transformer-based classifiers.

Although this overview is introductory, it is directly relevant to the contributions of this thesis. The dissertation is situated within the setting of supervised classification, and the fairness–performance trade-off examined throughout the thesis is shaped, by the underlying learning paradigm and model family. Moreover, this overview of supervised machine learning techniques provides the methodological context for understanding the major families of supervised learning methods.

2.2. AI Bias

Machine learning models are increasingly being used in real-world decision-making. This has raised critical concerns about their fairness, transparency, and

accountability. A key challenge in this context is to understand the many ways bias can emerge, from data collection to model deployment, and how it can systematically propagate through data-driven systems. Against this backdrop, this thesis focuses primarily on the bias problem.

Generally, the word *'bias'* in artificial intelligence has many different meanings. For example, Mitchell [Mit80] defined bias as *'any basis for choosing one generalization over another, other than strict consistency with the instances'*. That is an inductive bias (also known as a learning bias). However, in this thesis, we focus on another type of bias: data bias. In particular, Olteanu et al. [Olt+19] defined data bias as *'A systematic distortion in the sampled data that compromises its representativeness.'*. Besides, the No Free Lunch (NFL) theorems by Wolpert and Macready [WM02] establish that no learning algorithm is uniformly superior across all problem distributions; any observed advantage necessarily reflects inductive assumptions that align the learner with the data-generating process. Consequently, when the training data or label definitions encode historical or measurement distortions, a standard empirical risk minimization (ERM) will align with and thereby propagate those distortions through its inductive bias. Mitigating these effects requires introducing principled counter-assumptions that steer the learner toward the intended construct rather than toward artifacts of the observed data.

In many supervised learning scenarios, the data available for training does not fully or fairly reflect the real-world population. Truly clean, complete, and unbiased datasets are the exception rather than the rule, meaning that the learning process often inherits some degree of distortion, imbalance, and/or historical bias. The consequences are particularly serious in high-stakes machine learning tasks (e.g., predictive policing, credit approval, fraud detection, medical diagnosis), where trained models produce biased outcomes that can disproportionately affect specific groups. Addressing this bias is therefore essential to ensuring that machine learning systems deliver decisions that achieve a good balance between performance and fairness.

2.2.1. Types of Bias

The problem of bias in artificial intelligence is complex and extends far beyond simple technical errors. It may involve systemic patterns that can unintentionally disadvantage specific individuals or groups, thereby weakening the fairness, trustworthiness, and overall integrity of AI-enabled decisions. Bias can be introduced at any stage of the machine learning lifecycle. For example, Suresh and Guttag [SG21] provided a framework that includes seven distinct sources of potential downstream harm (including bias) across data collection, model development, and system deployment. These issues frequently stem from entrenched historical inequalities, imbalanced or unrepresentative datasets, and design or modeling decisions that fail to account for diverse real-world contexts. Recognizing how and where these biases emerge is essential for developing AI systems that are

transparent, accountable, and equitable in practice. Understanding the different types of bias in AI is therefore essential not only for identifying where and how unfairness is introduced, but also for implementing robust mitigation strategies that promote transparency, accountability, and equitable outcomes across downstream applications. In this section, we present the main types of bias in AI, following the categorization proposed by Mehrabi et al. [Meh+21]. Their framework structures these definitions around three central components of the AI pipeline: the data, the learning algorithm, and the user interaction loop.

Data to Algorithm. Bias can arise from the dataset used to train machine learning models, creating distortions that are later amplified by the learning process. As noted by Mehrabi et al. [Meh+21], this may include measurement or reporting bias, omitted-variable bias, and representation or sampling bias, all of which arise from how data are collected, selected, or proxied. Aggregation-related problems (e.g., Simpson's paradox or spatial aggregation effects) and temporal or network-related artifacts can further misrepresent subpopulations, leading the algorithm to learn patterns that statistically align with the data but are misaligned with the real-world population.

Algorithm to User. Even when input data is relatively sound, algorithmic design choices and model behavior can introduce new forms of bias that ultimately shape user experience. According to Mehrabi et al. [Meh+21], algorithms can produce biased outcomes through modeling assumptions, optimization objectives, or evaluation on unbalanced benchmarks, and these biases can subsequently influence users via presentation, ranking, or popularity effects. Over time, such feedback can give rise to emergent and evaluation biases, reinforcing visibility for specific items or groups while systematically sidelining others.

User to Data. Because many AI systems rely on user-generated or user-mediated data, pre-existing social, historical, and behavioral biases in users can flow back into the data ecosystem.Mehrabi et al. [Meh+21] identified several factors, including historical bias, population or self-selection bias, social influence, temporal drift, and differences in content production, that collectively shape data production and the context in which it occurs. This, in turn, affects future training data and closes the feedback loop between users, algorithms, and data, making it necessary to consider these biases jointly rather than in isolation.

For this thesis, the preceding categorization of bias types is not merely descriptive, but also structurally important. The thesis investigates the fairness–performance trade-off through three intervention points in supervised learning. The first contribution focuses on model architecture, which is most closely related to the *Data-to-Algorithm* type of bias. The second contribution addresses data composition, particularly group-level imbalance and its implications for both fairness and predictive utility, and can likewise be situated within the *Data-to-Algorithm* category. The third contribution examines label quality, which is closely connected to the *User-to-Data* type of bias, since biased labels may also propagate unfair-

ness into downstream decisions. Taken together, this bias taxonomy provides an important conceptual foundation for the thesis by showing why fairness must be examined across the learning pipeline rather than at only a single stage.

2.3. Fairness in Machine Learning

Fairness in machine learning has become a central concern as predictive models are increasingly used to support, inform, or even automate decisions that affect people's lives. Fairness differs fundamentally from traditional performance metrics, such as accuracy or precision. Rather than measuring overall model performance, fairness concerns how outcomes are distributed across different populations. In particular, fairness examines whether certain groups defined by sensitive attributes (gender, ethnicity, age, socioeconomic status) receive systematically different treatment from the algorithm. Ensuring fairness, therefore, requires not only technical adjustments to models and data but also a clear understanding of the social, legal, and ethical contexts in which these systems operate.

2.3.1. Bias & Fairness in AI

Fairness and bias are closely related concepts, but they differ in several respects [Fer24]. According to Žliobaitė [Žli17], bias in AI refers to the systematic and consistent deviation of a model's predictions from the actual or intended values. Thus, it reflects errors arising from the data, the model, or the learning process. In contrast, fairness focuses on the absence of unjustified discrimination or preferential treatment toward individuals or groups based on protected attributes such as race, gender, age, or religion [Dwo+12]. However, despite the distinctions between bias and fairness, the two remain deeply interdependent. In practice, reducing systematic bias is a necessary step toward achieving fairer and more accountable AI systems [Fje+20].

2.3.2. Types of Fairness in Supervised Learning

Different notions of fairness in AI recognize that 'fair' is not a single, universal standard, but a context-dependent goal shaped by the application, the people affected, and the surrounding legal or ethical framework. Broadly, fairness in machine learning is discussed along three main lines: group fairness, subgroup fairness, and individual fairness [Meh+21]. Selecting the right notion of fairness is therefore not merely a technical choice; it requires aligning the fairness objective with the system's purpose, the social environment in which it operates, and the harms it is meant to prevent. However, this thesis primarily focuses on group fairness. We briefly describe these three types of fairness below.

Group Fairness. Group fairness aims to ensure that outcomes of a model are comparable across protected groups. Thus, the primary goal in this type of fairness is to prevent situations in which automated decisions consistently benefit one

segment of the population while burdening others. This ensures that all groups, particularly those characterized by sensitive attributes such as gender, race, or age, are treated non-discriminatorily. In practical settings, this requires checking whether a model produces outcomes that are applied consistently across relevant demographic groups. To achieve this, researchers usually select one or more fairness criteria or statistical tests relevant to the task, recognizing that different definitions of fairness highlight distinct aspects of equity and may yield different conclusions.

Individual Fairness. Individual fairness, on the other hand, asks that similar individuals be treated similarly, emphasizing consistency at the level of the person rather than the group.

Subgroup Fairness. Subgroup fairness aims to bridge the gap between group-level and individual-level notions of fairness by leveraging the advantages of both. Instead of treating them as competing approaches, it builds on a chosen group-fairness criterion, such as requiring equal false-positive rates, and then tests whether that requirement is satisfied not just for broad protected groups but also for their meaningful subpopulations.

Counterfactual Fairness. Beyond group, individual, and subgroup-based perspectives, the fairness literature also includes causal notions such as counterfactual fairness, where a decision is considered fair if it remains unchanged in a counterfactual world in which the individual had belonged to a different sensitive group [Kus+17].

The distinction among group, individual, subgroup, and counterfactual fairness is particularly relevant for positioning the contributions of this thesis. Although these perspectives offer complementary views of fairness, the dissertation mainly adopts a group fairness perspective, since all three contributions study disparities across demographic groups defined by sensitive attributes.

2.3.3. Fairness: Core Definitions in Machine Learning

In this section, we lay the groundwork for this thesis by introducing a set of notations and group fairness definitions that serve as fundamental prerequisites for comprehending and interpreting the content presented here. We begin by formalizing our classification setting as a triplet dataset $T = \{(x_i, a_i, y_i)\}_{i=1}^{N}$, where x_i is the feature drawn from a distribution over the alphabet X, $a_i \in A = \{0,1\}$ is sensitive attribute (race, gender. etc.), and $y_i \in Y = \{0,1\}$ is the output label. We define H as the hypothesis class of predictors (U) mapping from the input space X to the output space Y. In what follows, we present the definitions of group fairness that serve as the basis for this thesis.

Definition 2.3.1 (Equalized Odds 'EO')**.** EO is a fairness criterion for classification tasks that ensures equitable false-positive and false-negative rates across different groups. It is defined in terms of the conditional probability distributions associated

with distinct groups identified by sensitive attributes $a_i \in \{0,1\}$.

A classifier U satisfies equalized odds if the following condition holds [HPS16]:

$$P(\hat{y}_i = 1 \mid a_i = 0, y_i = y) \approx P(\hat{y}_i = 1 \mid a_i = 1, y_i = y)$$

where $y \in \{0,1\}$, $\hat{y}_i$ is the predicted label for the i-th sample, a_i is the sensitive attribute value for the i-th sample and y_i is the target label for the i-th sample. This ensures the elimination of disparities in different groups that are affected by both false positive (FPR) and false negative (FNR) rates in predictive models.

To operationalize the pursuit of equalized odds, practitioners often employ a training strategy that involves minimizing specific objectives as follows [Che+21; PG20]:

$$\min_{U_w} \left\{ \hat{L}(U_w) + \rho(FPR + FNR) \right\} \tag{2.1}$$

where ρ is a predetermined weight, $\hat{L}$ signifies the cross-entropy loss, and FPR and FNR are derived as:

$$FPR = \left| \frac{\sum_i p_i(1-y_i)a_i}{\sum_i a_i} - \frac{\sum_i p_i(1-y_i)(1-a_i)}{\sum_i (1-a_i)} \right|$$

$$FNR = \left| \frac{\sum_i (1-p_i)y_i a_i}{\sum_i a_i} - \frac{\sum_i (1-p_i)y_i(1-a_i)}{\sum_i (1-a_i)} \right|$$

In this scenario, p_i signifies the softmax output of model U_w in a binary prediction task, considering the sensitive attribute $a_i \in [0,1]$ and the true label $y_i \in Y = \{0,1\}$ corresponding to each feature vector $x_i \in X$ in the dataset.

Definition 2.3.2 (Equal Opportunity)**.** It requires that a model achieve the same true positive rate (TPR) for different subgroups when considering only instances with a positive label [HPS16]. Formally, it is defined as:

$$P(\hat{y}_i = 1 \mid a_i = 0, y_i = 1) \approx P(\hat{y}_i = 1 \mid a_i = 1, y_i = 1)$$

This condition ensures that individuals from different groups who are actually positive (i.e., have a positive true label) have an equal probability of being classified as positive by the model.

Definition 2.3.3 (Accuracy Equality)**.** Accuracy Equality (AE) evaluates whether the subjects in protected and unprotected groups experience similar false positive rate (FPR) and false negative rate (FNR). The objective of AE is to ensure that misclassification rates are approximately equal across these sensitive groups, such as different demographic categories, to prevent discriminatory or biased outcomes in predictive models [Zaf+17]. AE is defined as:

$$P(\hat{y}_i \neq y \mid a_i = 0) = P(\hat{y}_i \neq y \mid a_i = 1)$$

To enforce accuracy equality in practice, one way to implement this during training is to minimize the following objective:

$$\min_{U_w}\{\hat{L}(U_w)+\rho \mid \hat{L}^{-a}(U_w)-\hat{L}^{+a}(U_w) \mid \}$$

for a given predefined weight ρ, where $\hat{L}^{+a}(U_w)$ is the cross-entropy loss of samples with $a=1$ and $\hat{L}^{-a}(U_w)$ is the cross-entropy loss of samples with $a=0$.

Definition 2.3.4 (Max-Min Fairness)**.** Max-Min Fairness (MMF) is a fairness principle that prioritizes and maximizes the performance of the least advantaged or worse-off group within a given context [Raw01]. MMF optimizes the performance for the group with the lowest utility while still meeting overall performance goals, and is used in the area of fairness-aware machine learning [Lah+20; Che+21]. Max-min fairness is defined as:

$$\arg\max_{\hat{y}\in Y}\ \min_{a\in A} P(\hat{y}=y \mid a)$$

Max-min fairness is satisfied by minimizing the following objective [Che+21]:

$$\min_{w}\ \max\left\{\hat{L}^{(y^+,a^+)}(U_w),\ \hat{L}^{(y^+,a^-)}(U_w),\ \hat{L}^{(y^-,a^+)}(U_w),\ \hat{L}^{(y^-,a^-)}(U_w)\right\}$$

where $\hat{L}^{(y',a')}(U_w)$ denotes the cross-entropy loss on the training samples where $y=y'$ and $a=a'$.

2.4. Classical Categories of Debiasing Approaches in Machine Learning

In Section 2.3.3, we reviewed the landscape of fairness notions for machine learning systems. In this section, we outline the principal strategies for promoting fairness in machine learning models. Because bias can arise at any point in the ML pipeline, including during data collection, labeling, modeling, or evaluation, solutions are commonly grouped into three families: *pre-processing*, *in-processing*, and *post-processing* methods [CH24; Meh+21].

Generally, debiasing is closely connected to fairness in machine learning because it provides the practical mechanisms through which fairness objectives can be pursued. While fairness notions specify the type of group-level parity or protection that is desired, debiasing methods determine how the learning pipeline can be modified to move the model toward those objectives. In this sense, pre-processing, in-processing, and post-processing approaches should be understood as intervention points for reducing unfair disparities. At the same time, the relation is not one-to-one: a debiasing method is usually designed to improve a fairness criteria, and its

effect depends on the dataset, the task, and the performance constraints. Therefore, debiasing is best viewed as the operational side of fairness, that is, the set of methodological choices used to translate some fairness principles into concrete changes in model behavior.

2.4.1. Pre-Processing Bias Mitigation Methods

Pre-processing methods aim to mitigate bias by transforming the training data before training the model using the initial dataset. A widely used approach is *Reweighing (RW)*, which assigns instance weights across groups to counteract imbalances and reduce disparate treatment or impact [KC12; CKP09]. In practice, weights are adjusted according to the frequency of the class label and the protected attribute, with rarer label-group combinations receiving relatively higher weights, so that the effective training distribution better reflects a fairness-aware target. Another prominent family is *Learning Fair Representations (LFR)*, which maps inputs into an intermediate latent space designed to obfuscate information about protected attributes while preserving the task signal as much as possible [Zem+13]. By constraining the representation to carry minimal protected-attribute information, LFR aims to curb downstream discrimination without sacrificing predictive utility entirely. Moreover, Calmon et al. [Cal+17] consider the pre-processing approach for fairness is considered as a constrained optimization problem, *Optimized Pre-processing*, for three goals: (i) control discrimination, (ii) limit distortion to individual samples, and (iii) preserve utility. Their analysis also characterizes the effects of limited sample sizes on these trade-offs. Empirically, they report that discrimination can be substantially reduced at only a modest cost in classification accuracy.

2.4.2. In-Processing Bias Mitigation Methods

In-processing methods (also known as *algorithmic interventions*) mitigate bias by modifying the model's objective or update rule during training [Wan+23]. A common strategy augments the empirical risk with a fairness-aware penalty term, thereby trading off task loss against criteria such as demographic parity or equalized odds, much like regularization controls model complexity and overfitting. Moreover, a complementary line of work uses *adversarial* training. Here, a predictor is jointly optimized with an adversary that attempts to recover the protected attribute from intermediate representations or outputs; the predictor learns to minimize prediction error while simultaneously hindering the adversary, thereby reducing protected attribute leakage and, in turn, algorithmic bias [ZLM18]. Furthermore, more recently, *multi-objective* formulations explicitly optimize accuracy and fairness as competing objectives. For example, La Cava [La 23] proposed *FOMO*, an evolutionary framework that employs *NSGA-II* [Deb+00] to find a good trade-off between fairness and performance. Rather than relying solely on gradient-based regularization, FOMO introduces a meta-model that maps protected attributes to

sample weights and optimizes these weights to achieve improved fairness–accuracy trade-offs.

2.4.3. Post-Processing Bias Mitigation Methods

Post-processing methods intervene after model training, adjusting scores or decisions to mitigate disparate outcomes. A straightforward way is to apply *group-specific decision thresholds*, selecting different threshold values for protected and unprotected groups to better align with a chosen fairness criterion.

Kamiran et al. [Kam+18] introduced the *reject option* framework, which revises predictions for individuals whose scores lie within an uncertainty band around the decision boundary, typically favoring members of the disadvantaged group within that band, to reduce discrimination while preserving accuracy away from the boundary; this is often referred to as *Reject Option Classification* [Kam+18].

The interaction between calibration and error-rate parity has been examined in depth by Pleiss et al. [Ple+17]. Calibration requires predicted probabilities to match empirical outcomes within each group, whereas error-rate criteria (e.g., equal opportunity via equal false-negative rates) target parity of mistakes across groups. Their analysis shows these goals are generally incompatible unless base rates coincide or the predictor is perfect; attempting to satisfy both can lead to degenerate, low-utility solutions. Consequently, post-processing must navigate an explicit trade-off between calibration and error-rate constraints rather than expecting them to be simultaneously achievable.

2.4.4. From Classical Categories to Contemporary Debiasing Strategies

Debiasing methods are traditionally grouped into the canonical triad of pre-processing, in-processing, and post-processing [CH24; Meh+21]. In light of rapid advances in neural architectures for vision and language, however, Parraga et al. [Par+25] propose a complementary, network-oriented taxonomy comprising *distributional* debiasing, *one-step training* debiasing, *multi-step training* debiasing, and *inferential* debiasing. They argue that the conventional three-way scheme is too coarse to organize many contemporary techniques and that the proposed categories better align with how debiasing is implemented in modern neural pipelines.
According to Parraga et al. [Par+25], *distributional* strategies reshape the dataset *prior* to training, for example, through resampling or augmentation that increases the number of examples in targeted subpopulations. They further divide training-time optimization approaches into two classes: (i) *one-step training*, in which a fair model for a given task is produced via a single optimization run; and (ii) *multi-step training*, in which an additional training phase is introduced to adjust an already trained, biased model to improve its fairness. Finally, *inferential* debiasing methods consider the model's outputs, addressing the problem of fairness by detecting and mitigating biases without requiring further weight updates or additional dataset

manipulation [Par+25].

Although this chapter introduces the main background concepts, several more specialized notions are required for the contributions developed later in the thesis. In particular, Contribution I in Chapter 4 builds on fairness-aware in-processing and neural network fine-tuning strategies; Contribution II in Chapter 5 relies on cost-sensitive learning, threshold optimization, and group robustness; and Contribution III in Chapter 6 draws on graph-based concepts such as k-nearest-neighbor graphs, the graph Laplacian, and discrete Ricci curvature. To support this progression, Chapter 3 provides a more focused state-of-the-art discussion of the methodological domains most relevant to the thesis, while Chapters 4, 5, and 6 introduce the technical preliminaries required for each contribution in greater detail. Accordingly, the purpose of the present chapter is to establish the common vocabulary of fairness, bias, and debiasing, whereas the more specialized concepts are introduced progressively where they become methodologically relevant.

3. STATE OF THE ART

The previous chapter established the conceptual and technical background for this thesis. In this chapter, we survey state-of-the-art methods aligned with each of our contributions. For each line of work, we situate our approach within its source domain, summarize the core ideas and typical assumptions, and highlight how these strands inform the methods developed in this thesis.

3.1. Bias Mitigating Techniques

Debiasing techniques are mainly categorized as (i) pre-preprocessing, (ii) in-processing, and (iii) post-processing. These fairness techniques, pioneered by seminal works from Dwork et al. [Dwo+12], Hashimoto et al. [Has+18], and Kearns et al. [Kea+18], represent a paradigm shift towards algorithmically engineered fairness.

In-processing debiasing techniques in machine learning aimed at mitigating disparity were studied by Wan et al. [Wan+23]. They proposed adding a regularizer to reduce the correlation between sensitive attributes and prediction outcomes. Zafar et al. [Zaf+19; Zaf+17] proposed a novel measure of decision boundary (un)fairness. They used the covariance between the sensitive attributes and the (signed) distance between the subjects' feature vectors and the decision boundary as a metric for the classifier.

Another line of research was introduced by Kamishima et al. [KAS11], who proposed the prejudice index (PI) as a regularizer to quantify the level of dependence between a sensitive variable and a target variable. Jiang et al. [Jia+20], on the other hand, used information geometry metric Wasserstein-1 distances between classifier outputs and sensitive information as a regularization in the optimization process. In a more practical setting, Beutel et al. [Beu+19a] proposed a new metric of conditional equality while implementing equality of opportunity. In the context of recommender systems, Beutel et al. [Beu+19b] proposed pairwise comparisons from randomized experiments as a tractable way to measure fairness.

3.2. Overfitting Problem & Fairness

Cherepanova et al. [Che+21] highlighted that in-processing techniques are less effective for over-parameterized large neural networks, as these models can easily overfit fairness objectives during training, especially when the training data is imbalanced. This fairness-overfitting issue, raised by Cherepanova et al. [Che+21], remains an open challenge. Over-parameterization of neural networks leads to highly flexible decision boundaries, and attempting to meet fairness criteria for one attribute can negatively impact fairness with respect to another sensitive attribute.

However, over-parameterization has been essential for achieving high prediction accuracy, particularly in neural networks designed for challenging tasks.

To address the problem of fairness criteria overfitting identified by Cherepanova et al. [Che+21], a novel framework proposed by Mao et al. [Mao+23] called last-layer fairness fine-tuning. Mao et al. [Mao+23] took a different route and avoided adding a fairness constraint as a regularization for deep neural network models. This is because large, deep neural models have the tendency to overfit fairness criteria. Instead, they used a pre-trained model, then fine-tuning its last layer to improve the fairness-performance trade-off. In particular, the method proposed by Mao et al. [Mao+23] involves training a model using empirical risk minimization and subsequently fine-tuning only the last layer with various fairness constraints as an in-processing debiasing technique. In extensive experiments on benchmark image datasets with different fairness notions, the authors demonstrated the efficacy of last-layer fine-tuning in enhancing fairness.

Other important works along this direction are Kirichenko et al. [KIW22] and Lee et al. [Lee+22]. The authors showed that only fine-tuning the last layer(s) while keeping other layers frozen during the gradient descent achieves better performance.

However, in a recent counterclaim study by Kumar et al. [Kum+22], it was observed that fine-tuning can actually achieve worse accuracy than linear probing out-of-distribution (OOD). This is particularly in cases where the pretrained features are of high quality, and the distribution shift is substantial. The authors conducted experiments across ten distinct distribution shift datasets, revealing that while fine-tuning tends to achieve, on average, a 2% higher accuracy in-distribution (ID), it results in a 7% lower accuracy on out-of-distribution (OOD) data compared to linear probing. To address this disparity, they propose a two-step strategy known as LP-FT, which begins with linear probing and is followed by full fine-tuning. This approach capitalizes on the strengths of both fine-tuning and linear probing and consistently leads to superior performance. In their empirical evaluations, Kumar et al. [Kum+22] demonstrated that LP-FT outperforms both fine-tuning and linear probing across various datasets, achieving 1% higher accuracy in-distribution and an impressive 10% higher accuracy out-of-distribution.

3.2.1. Model Stitching

Very few works have examined fairness from the vantage point of model *architecture* within the machine learning literature. For example, Sheng et al. [She+24] demonstrated that the architecture of neural networks influences downstream fairness outcomes. Sheng et al. [She+24] showed that addressing fairness effectively requires a holistic approach, with joint attention to data, learning algorithms, and architectural design. Leveraging AUTOML, specifically neural architecture search (NAS), they propose *BiaslessNAS* to promote fairer predictions on skin/lesion datasets. While promising, this line of work faces practical constraints: NAS is

computationally intensive and challenging to reproduce across different hardware and software stacks.

The proposed framework, 'The Fairness Stitch', in the first contribution [SR24], addresses the issues in Section 3.2 and the limitations of last-layer fine-tuning in attaining an optimal balance between fairness and performance. In particular, it considers making architectural modifications to a pre-trained neural network to improve the balance between fairness and performance, as explained in Section 4.3. Our proposed method draws inspiration from prior work on model stitching by Lenc and Vedaldi [LV15] and Bansal et al. [BNB21]. Model stitching is a tool for studying the internal representations of deep neural networks. Model stitching combines the bottom layers of one pre-trained and frozen model (referred to as Model A) with the top layers of another model (referred to as Model B) using a trainable layer positioned in between, resulting in the creation of what is termed a "stitched model". Our work leveraged this model stitching approach with the aim of mitigating bias.

This line of work is directly relevant to Contribution I in Chapter 4. More specifically, the first contribution builds on the idea that fairness can be improved not only through the loss function but also through architectural intervention within a pretrained neural network. In this sense, the literature on overfitting, last-layer fine-tuning, and model stitching provides the immediate methodological foundation for the proposed The Fairness Stitch framework developed later in the thesis.

3.3. Cost-Sensitive Learning & Fairness

In the second contribution [SMR25], we introduce GLCS (Group-Level Cost-Sensitive Deep Learning), which is based on concepts from the domain of cost-sensitive learning. Cost-sensitive learning adaptively weighs the importance of different classes during the training process. This is typically achieved through the modification of the loss function in neural networks. This approach is effective in real-world applications such as medical diagnosis, fraud detection, and rare-event prediction, where misclassification costs are inherently asymmetric. Recent developments in this area have introduced several innovative methodologies. Zhou and Zhang [ZZ16] employed cost-sensitive learning to mitigate the problem of misclassifications of minority or critical classes. The class-balanced loss function [Cui+19] addresses the long-tailed distribution problem by introducing a weighting factor that is inversely proportional to the effective number of samples. Margin-based approaches [Cao+19] focus on enhancing the decision boundary's quality by incorporating cost-sensitivity into the margin requirements. Additionally, Sangalli et al. [San+21] uses constrained optimization to train neural networks to improve neural network performance on critical and underrepresented classes.

3.3.1. Fairness and Class Imbalance

The intricate relationship between fairness and class imbalance has emerged as a critical research domain in machine learning, with scholars developing sophisticated methodologies to address simultaneous challenges of bias mitigation and distributional disparities. Dablain et al. [DKC22] introduced Fair OverSampling (FOS), a pioneering approach that simultaneously addresses class imbalance and protected feature bias by generating synthetic minority class instances while encouraging classifiers to minimize reliance on sensitive attributes. Complementing this work, Hirzel and Ram [HR] developed Orbis, an adaptable oversampling algorithm capable of fine-tuned optimization across fairness and accuracy dimensions. Yan et al. [YKF20] critically demonstrated that conventional balancing techniques can inadvertently exacerbate unfairness, and introduced a novel fair class balancing method that enhances model fairness without explicit manipulation of sensitive attributes. Tarzanagh et al. [Tar+23] advanced this discourse through a tri-level optimization framework incorporating local, fair, and class-balanced predictors, theoretically demonstrating improved classification and fairness generalization. Subramanian et al. [Sub+21] further expanded these investigations by evaluating long-tail learning methods across sentiment and occupation classification domains, empirically validating the effectiveness of fairness enforcement techniques in mitigating demographic biases and class imbalance. Shui et al. [Shu+22] contributed a principled bilevel objective approach, demonstrating an innovative method for developing fair predictors that simultaneously manage group sufficiency and generalization error.

3.3.2. Group Robustness

Recent machine learning research has developed sophisticated methods to address the problem of group robustness. Sagawa et al. [Sag+19] introduced Group Distributionally Robust Optimization (Group-DRO), which optimizes a soft version of the worst-group loss. Liu et al. [Liu+21] proposed Just Train Twice (JTT), a method that employs a two-stage training strategy: Initially, a standard ERM model is trained for several epochs. In the subsequent stage, a refined model is trained by upweighting the training examples that the initial ERM model misclassified. Complementing these approaches, Kirichenko et al. [KIW22] demonstrated through Deep Feature Reweighting (DFR) that simple last-layer retraining can match or surpass state-of-the-art methods on spurious correlation benchmarks with significantly reduced computational complexity. Building upon this insight, Qiu et al. [Qiu+23] developed Automatic Feature Reweighting (AFR), which retrains the last layer of the ERM-trained model with a weighted loss that upweights minority group examples by emphasizing instances where the ERM model performs poorly.

3.3.3. Threshold Optimization

In the second contribution [SMR25], we considered the threshold optimization in our proposed method GLCS. Please refer to Section 5.6.2 to find more details about the threshold optimization used in our experiment in the second contribution.

Generally, threshold optimization in classification represents a sophisticated computational domain, with seminal works [LEN14; Koy+14; San16] systematically exploring methodological approaches for determining optimal decision boundaries. Research has advanced through receiver operating characteristic (ROC) curve analysis for identifying optimal operating points [FM08], cost-sensitive threshold adjustment techniques that explicitly incorporate domain-specific loss functions and contextual constraints into the threshold selection process [Rob+20], unified theoretical frameworks [HFF12] offering comprehensive computational strategies for threshold optimization that transcend traditional binary classification paradigms, and probabilistic methodologies for adaptive threshold determination, which significantly enhance the precision and reliability of predictive models across diverse computational domains [KKS23], thereby providing a comprehensive approach to optimizing classification thresholds with nuanced consideration of performance, constraints, and contextual requirements.

Taken together, the literature reviewed in Sections 3.3.1, 3.3.2, and 3.3.3 provides the main methodological context for Contribution II in Chapter 5. In particular, the interaction between fairness and class imbalance motivates the need for group-aware cost-sensitive learning, the literature on group robustness clarifies why minority-group performance must be considered explicitly, and threshold optimization provides a mechanism for optimizing classification thresholds in machine learning models. These strands therefore motivate the design choices underlying the proposed GLCS framework.

3.4. Fairness in Machine Learning and Label Correction

In this section, we present relevant literature on fairness in machine learning for group-dependent noisy labels, Ricci curvature in network science, learning with noisy labels, the graph Laplacian, and Semi-Supervised Learning (SSL). These domains are related to the third contribution [SR25], Graph-based Fairness-aware Label Correction (GFLC).

Generally, in the third contribution [SR25], we primarily focus on addressing fairness issues in machine learning in the presence of biased and group-dependent noisy training labels. Wang et al. [WLL21] presented a method for training fair classifiers in scenarios where training labels are subject to random noise, where the error rates of corruption are influenced by both the label class and the protected attribute. Furthermore, Wang et al. [WLL21] presented analytical results demonstrating that simply imposing parity constraints on demographic

disparity measures, without accounting for varying error rates across groups, can degrade both performance and fairness of the resulting classifier. They address such issues by employing empirical risk minimization with addition to a surrogate loss function and constraints. This approach helps to mitigate the challenges posed by heterogeneous label noise.

Wu et al. [Wu+22] introduced introduced general frameworks for developing fair classifiers that consider instance-dependent label noise. In their study, they addressed statistical fairness by reformulating both classification risk and fairness metrics in the context of noisy data, enabling the creation of more robust classifiers. For their causality-based approach to fairness, they leveraged the data's internal causal structure to simultaneously model label noise and ensure counterfactual fairness. In addition, [TW23] proposed an approach to address instance-dependent label noise without making assumptions about the noise distribution, utilizing all available data during training. Their focus is on a scenario commonly encountered in healthcare, in which researchers are provided with observed labels for a condition of interest, such as cardiovascular disease. They assume that a clinical expert can evaluate the accuracy of these observed labels for a small subset of the data, for instance, through a manual chart review. Using this specific subset, called the alignment set, Tjandra and Wiens [TW23] identify the underlying patterns of label noise. They then minimize a weighted cross-entropy across all the data. The authors note that their alignment set is a particular instance of anchor points [LT16], with the additional requirement that it includes instances where the ground truth and observed labels either match or do not match.

Canalli et al. [Can+24] explored the intersection of fair machine learning techniques and fundamental sources of bias, such as noisy data, emphasizing similarities and differences between fairness and noise in the context of machine learning. In addition, they developed the Fair Transition Loss, a new method for fair classification inspired by robust label noise learning techniques. Furthermore, the study presented by Silvaa et al. [Sil+24] introduces Fair-OBNC, a novel method for correcting label noise that considers fairness. This approach builds upon the Ordering-Based Label Noise Correction (OBNC) algorithm [FB15] by incorporating fairness factors. Fair-OBNC specifically aims to enhance demographic parity in training datasets, making it a distinctive method at the intersection of fairness and noise correction. Additionally, Fair-OBNC [Sil+24] is the most closely related work to the third contribution, as it serves as the baseline and uses the same dataset for evaluation.

3.4.1. Ricci Curvature in Network Science

In classical mathematics, Ricci curvature and Ricci flow are among the most important tools for analyzing manifolds according to their geometric and topological properties. Sreejith et al. [Sre+16] introduced a new discretization of the classical Ricci curvature, which was originally proposed by Forman [For03], specifically

for the domain of complex networks. Sreejith et al. [Sre+16] explored the correlation between Forman curvature and common network measures, such as degree, clustering coefficient, and betweenness centrality, in both model and real networks. Weber et al. [WJS16] built upon and expanded the work of Sreejith et al. [Sre+16]. Their research focused on the relationship between the Forman-Ricci curvature and other geometric properties of networks, such as the node degree distribution and connectivity structure. Weber et al. [WJS16] introduced an innovative change-detection method for complex dynamic networks that leverages Ricci flow on edges with respect to Forman curvature. In the work by Weber et al. [WSJ17], they applied Forman's Ricci curvature and the Bochner Laplacian [For03] to analyze networks and explore their effectiveness as network characteristics. Additionally, they introduced a method for detecting changes in evolving networks and proposed the Laplacian flow as a tool for denoising networks constructed from empirical data. Weber et al. [WSJ17] noted in their study that denoising could be a practical application resulting from their theoretical findings. Besides, in an empirical investigation, Samal et al. [Sam+18] examined two discretizations of Ricci curvature: Ollivier's Ricci curvature and Forman's Ricci curvature. They analyzed these measures across various models and real-world networks and found a strong correlation between those two types of Ricci curvature in many cases. Moreover, a significant advantage of Forman-Ricci curvature, as highlighted by Samal et al. [Sam+18], is that it captures essential geometric properties of networks while being much simpler to compute on large networks compared to Ollivier-Ricci curvature. Additionally, they noted that Forman-Ricci curvature can serve as a faster alternative to Ollivier-Ricci curvature for coarse analyses of larger real-world networks, when less precision is acceptable.

Ni et al. [Ni+19] introduced an innovative geometric approach that enables the use of powerful classical geometric methods and properties. By conceptualizing networks as geometric objects and considering communities within a network as geometric decompositions, they apply concepts such as curvature and discrete Ricci flow methods. These techniques have demonstrated significant effectiveness in analyzing and breaking down communities in networks.

Topping et al. [Top+21] examined the concept of graph bottlenecks and the issue of over-squashing, which limits the performance of message passing in graph neural networks. They introduced the Balanced Forman curvature, a new type of edge-based Ricci curvature that connects to the classical Ollivier curvature. The authors demonstrated that negatively curved edges contribute to over-squashing, suggesting that curvature-based graph rewiring could improve the graph's bottleneck characteristics. The work by Nguyen et al. [Ngu+23] established a new connection between Ollivier-Ricci curvature on graphs and the challenges of over-smoothing and over-squashing. They demonstrated that positive graph curvature is associated with over-smoothing, while negative graph curvature is linked to over-squashing. To address these issues, Nguyen et al. [Ngu+23] proposed an inno-

vative curvature-based rewiring method called Batch Ollivier-Ricci Flow (BORF), which effectively improves the performance of graph neural networks (GNNs) by tackling both over-smoothing and over-squashing simultaneously.

For the present thesis, these ideas are particularly relevant to Contribution III in Chapter 6. More specifically, GFLC uses a simplified version of discrete Forman–Ricci curvature together with a Ricci-flow-based update to refine the local structure of a k-nearest-neighbor graph before label correction. In this way, the role of curvature in this thesis is not merely descriptive: it provides a geometric signal for adjusting edge weights and neighborhood structure so that subsequent graph-based label correction can better respect the local data manifold.

3.4.2. Graph Laplacian

Learning on graphs with Laplacian regularization was explored by Ando and Zhang [AZ06], who derived generalization bounds for this approach by leveraging graph properties. Ando and Zhang [AZ06] emphasized the significance of Laplacian normalization and dimensionality reduction in graph learning.

Streicher and Gilboa [SG23] proposed a new definition of the graph Laplacian to enhance performance in Semi-Supervised Learning (SSL) tasks. Furthermore, their proposed operator facilitates smooth interpolation between the unsupervised and semi-supervised scenarios. In addition, Aquino Afonso and Berton [AB20] conducted a comprehensive empirical evaluation of different graph-based semi-supervised learning (SSL) algorithms for label noise. They demonstrated that detecting the noisiest instances is possible when the dataset satisfies the assumptions of semi-supervised learning (SSL). Furthermore, their findings revealed that in high-dimensional clusters, the Laplacian eigenmaps (LE) algorithm outperformed label propagation.

Chen et al. [Che+25] introduced LaplaceConfidence, a method that obtains label confidence (clean probabilities) using Laplacian energy. They demonstrate that their approach outperforms other classification-loss-based estimation methods, achieving state-of-the-art results on standard benchmarks for Learning with Noisy Labels (LNL). Compared with previous work, our study addresses fairness in machine learning and examines how noisy labels affect fairness in classification tasks.

The relevance of the graph Laplacian to this thesis becomes most explicit in Contribution III, presented in Chapter 6. There, the Laplacian term is used to quantify local label disagreement on the graph and is incorporated into the combined correction score that prioritizes which labels should be flipped. Accordingly, the present section provides the methodological basis for using neighborhood smoothness as a signal for label correction.

3.5. Conclusion

This chapter presents a comprehensive review of the state-of-the-art techniques for improving the trade-off between fairness and performance in machine learning. We presented related work from the domains used to develop the methods in each contribution. Specifically, we considered three distinct approaches to address this problem. First, improving the trade-off by modifying neural network architectures. Second, addressing the trade-off through the imbalanced-class problem. Third, addressing the trade-off in the presence of biased labels in the dataset.

In particular, by exploring different approaches to overfitting and fairness, we highlighted the strengths and limitations of various strategies for navigating this trade-off. Moreover, we provide an overview of the related work on the concept of 'Model Stitching' that we used to adjust the neural network architectures and develop our first contribution. Similarly, this chapter surveys related work on the second contribution, which addresses the group-level class imbalance problem in the presence of dataset bias. We also review related work on 'Group Robustness' and on threshold optimization. Finally, we presented methods for *fair classification* under *label bias*, i.e., systematic mislabeling that arises from historical inequities, annotation noise, or group-dependent measurement error. In addition, we review related concepts from network science that inform our methodology: discrete *Ricci curvature* (e.g., the Ollivier and Forman variants), which captures local geometric structure and graph robustness, and the *graph Laplacian*.

More specifically, Sections 3.2 and 3.2.1 primarily motivate Contribution I presented in Chapter 4; Sections 3.3.1, 3.3.2, and 3.3.3 inform Contribution II developed in Chapter 5; and Sections 3.4, 3.4.1, and 3.4.2 provide the main methodological foundation for Contribution III in Chapter 6. These cross-chapter links help clarify how the different strands of related work connect to the three core contributions of the thesis.

Together, these domains served as important sources of inspiration for developing each contribution in this thesis.

4. THE FAIRNESS STITCH: A NOVEL APPROACH FOR NEURAL NETWORK DEBIASING (CONTRIBUTION I)

In the previous chapters, we introduced the background for the domain of fairness in machine learning, formalized the central challenge of balancing predictive utility with algorithmic fairness in classification, and summarized our contributions toward improving that fairness-performance trade-off. Because this balance is shaped by multiple factors, including, but not limited to, data properties, optimization objectives, and model architecture, this chapter turns to the architectural dimension. It examines how design choices within neural networks influence attainable fairness without compromising accuracy unduly. In particular, this chapter (4) is based on *Publication I* [SR24], titled 'The Fairness Stitch: A Novel Approach for Neural Network Debiasing'.

4.1. Background and motivation of the study

Modern classifiers for image datasets are often deep neural networks, either large by design or built on pretrained backbones. In such settings, it is common to adapt a model to new requirements (including fairness constraints) by fine-tuning only a small part of the network, often the final layer or prediction head, because this is computationally efficient and reduces the risk of overfitting. However, this practice implicitly assumes that fairness violations can be corrected solely at the output layer. In practice, group-dependent information can be encoded throughout the representation hierarchy, and restricting the adaptation to the last layer may therefore limit the attainable fairness–performance balance.

Motivated by these considerations, *Publication I* [SR24] introduces *The Fairness Stitch (TFS)*: a lightweight, trainable 'stitch' layer inserted between two existing layers of a neural network, while keeping the remaining parameters frozen. The stitch is optimized under fairness-aware constraints, aiming to reduce group disparities with minimal additional capacity and minimal changes to task-relevant representations. This design is inspired by the broader idea of model stitching (reviewed in Chapter 3 in Section 3.2.1) and adapts it to the debiasing setting within a single network.

The remainder of the chapter is organized as follows. Section 4.2 introduces the notation and the information-theoretic viewpoint used to motivate the method. Section 4.3 presents the proposed TFS framework and its training objective. Sections 4.4 and 4.5 describe the datasets and experimental setup, and the subsequent sections report the empirical results and discussion.

Some material in this chapter is based on *Publication I* [SR24], of which I am the primary author. Portions of the text, figures, and tables are reproduced or adapted from that work.

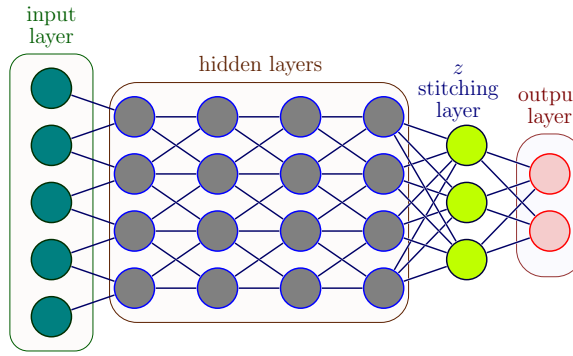


Figure 1: Illustration of 'The Fairness Stitch' framework for our stitched model. Our pre-trained model $\mathscr{M}$ includes only the input, hidden, and output layers, excluding the stitching layer z. The weights of the stitched model are kept frozen except for the weights associated with the stitching layer z.

4.2. Preliminaries

This section establishes the formal definitions and notation used throughout the chapter. We begin with the central concept of *model stitching*, originally introduced by Lenc and Vedaldi [LV15] and revisited in greater depth in Section 3.2.1. Next, Section 4.2.1 introduces the relevant information-theoretic concepts pertaining to neural network, laying the groundwork for the inspiration behind the first contribution, developed in Section 4.3.1.

Definition 4.2.1 (Model Stitching)**.** More formally, model stitching can be understood as a compositional procedure. To express this more precisely, consider a neural network architecture A and let $r\colon X \to \mathbb{R}^d$ denote a candidate representation. Following Bansal et al. [BNB21], the associated loss function is defined as:

$$L_l(r;A) \;=\; L(A_{>l} \circ s \circ r)\,, \tag{4.1}$$

where $s \in S$ denotes a stitching layer drawn from a family of stitching layers S, and $A_{>l}$ denotes the mapping from the activations at the l-th layer of A through to the model's final output. The operator $\circ$ denotes standard function composition, and l indexes the layer of A at which the stitching is applied.

4.2.1. Deep Neural Networks via Information

In the context of deep neural networks, Shwartz-Ziv and Tishby [ST17] introduced the concept of the "information plane" to provide provable guarantees for network optimization through Stochastic Gradient Descent (SGD). This framework relies on the mutual information I between two random variables Ψ_1 and Ψ_2, which is formally defined as:

$$I(\Psi_1;\Psi_2) = H(\Psi_1) - H(\Psi_1 \mid \Psi_2), \tag{4.2}$$

where $H(\Psi_1)$ denotes the marginal entropy of Ψ_1, and $H(\Psi_1 \mid \Psi_2)$ denotes the conditional entropy of Ψ_1 given Ψ_2. Furthermore, within the information plane, a representation variable denoted as T is used to map input data X and output labels Y, characterized by joint distributions $P(T|X)$ and $P(Y|T)$ [ST17]. Moreover, T_i, representing the i^{th} hidden layer as a single multivariate variable, forms Directed Path Independence (DPI) chains [ST17].

Additionally, Tishby et al. [TPB99] suggested that the layers of optimized Deep Neural Networks (DNNs) should converge toward the Information Bottleneck (IB) bound, which represents the theoretically optimal achievable compression of the input X while preserving relevant information.

As originally proposed by Shwartz-Ziv and Tishby [ST17], the examination of deep neural networks (DNNs) within the context of the "information plane" offers valuable insights. This visualization technique comes into play when the underlying distribution, denoted as $P(X;Y)$, is known, and when it's possible to calculate the encoder and decoder distributions, namely, $P(T|X)$, and $P(Y|T)$. Within this framework, two key order parameters, namely, $I(T;X)$ and $I(T;Y)$, serve as the means to visually compare different network architectures. Shwartz-Ziv and Tishby [ST17] also identified two key optimization phases through Stochastic Gradient Descent (SGD): the fast empirical error minimization (ERM) phase and the subsequent representation compression phase.

During the ERM phase, spanning a few hundred epochs, layers notably increased their information on output labels $I_Y = I(T_i;Y)$, $i \in [1,...,k]$, all while maintaining the Directed Path Independence (DPI) order, wherein lower layers retained higher label-related information. In the representation compression phase, the layers gradually reduced their information regarding input data $I_X = I(X;T_i)$, shedding irrelevant information until convergence.

Shwartz-Ziv and Tishby [ST17] underscores that their analytical insights maintain broad applicability. They anticipate even more significant dynamic phase transitions in larger networks, a conjecture grounded in the statistical ensemble properties of such networks. Taking inspiration from their confidence in the generalizability of their findings to "large real-world" problems and larger networks, we have reevaluated the concept of last-layer fairness fine-tuning, as discussed in Section 4.3.1. Additionally, in Section 4.6, we present empirical evidence supporting our reevaluation.

Binary Label Assumption. Throughout this section and the related Section 4.3.1, we assume a *binary classification* setting. In particular, the target variable Y denotes a binary label, i.e., $Y \in \{0,1\}$, where $Y = 1$ represents the positive class and $Y = 0$ represents the negative class. All mutual information quantities, conditional distributions $P(Y \mid T)$, and subsequent discussion in these sections are defined with respect to this binary outcome. This assumption is consistent with the setting and information-plane analysis of Shwartz-Ziv and Tishby [ST17].

4.3. TFS

This section presents the proposed method in the first contribution. We begin in Section 4.3.1 by outlining the main inspiration that motivates the proposed approach, before introducing the method itself and describing its formulation in detail.

4.3.1. Rethinking Last-Layer Fairness Fine-Tuning

This contribution (*Publication I*) [SR24] is inspired by the generalization drawn from the findings discussed in the Section 4.2.1. Subsequently, when training neural networks on biased datasets, we should observe that during the representation compression phase, the final layer tends to contain significantly less information about the input data (which, in our case, is biased). To elaborate further, as per the research conducted by [ST17], during the representation compression phase, it is a common observation to find that $I(X;T_j) = \varepsilon$ where ε is a very small positive number and j refers to the last layers of the neural network (e.g. $j \in [n-3,n]$). This indicates that the last layer of the model typically possesses minimal bias-related information from the input X. Consequently, using in-processing debiasing methods, such as applying fairness constraints during fine-tuning of the pretrained last layer, should be approached cautiously and may be reconsidered.

The last layer of a pretrained neural network is inherently designed for yielding high-performance predictions. However, any misguided attempt to intervene in the training process, such as solely fine-tuning the final layer, may result in underperforming on test examples sampled from out-of-distribution (OOD) data, especially when there is a significant distribution shift [Kum+22]. Furthermore, this approach may disrupt the delicate balance between model performance and fairness considerations. Typically, in the representation compression phase, we aim for an optimal scenario where $0.8 \leq I(Y;T_j) \leq 1$ [ST17]. Focusing solely on debiasing the last layer is akin to addressing its output, resembling post-processing debiasing methods.

In fairness-aware machine learning, it becomes essential to incorporate fairness constraints in updating the weights of the earlier layers throughout the two optimization phases since the value of $I(X;T_i)$ is height in the representation compression phase, where T_i indicates the earlier layers of the neural network. Regrettably, training the earlier layers with fairness constraints frequently yields unsatisfactory results [Che+21]. To surmount this challenge, we propose a novel approach, "The Fairness Stitch," that combines an in-processing debiasing technique with the concept of 'model stitching'. In particular, our approach introduces a trainable stitching layer between the frozen layers of a trained deep learning model, thereby enhancing the trade-off between fairness and performance after fine-tuning by incorporating a fairness constraint into the main objective function. By avoiding fine-tuning the last layer, we hold the potential to enhance the trade-off between fairness and performance, a point underscored by the comparison between our

findings in Section 4.6 and baseline work [Mao+23].

4.3.2. TFS: Trainable Stitching Layer for Fairness

This section presents the proposed method in full, building upon the model stitching foundations established earlier to develop a more constrained and specialised instantiation of the concept. Rather than combining two entirely distinct pre-trained models, the framework considered here operates in a setting where only a single pre-trained model M is available. This specialised variant is referred to as *self-stitching*, a term chosen to reflect the fact that the stitching operation is applied within a single model rather than across two separate ones. In this setting, the conceptual distinction between the "source" model and the "target" model collapses. A trainable stitching layer is inserted between two portions of the same network, as illustrated in Figure 1.

The Fairness Stitch 'TFS' framework includes a specialized layer that aims to ensure equal opportunities for different groups during inference (based on sensitive attributes). By integrating TFS in a neural network, the machine learning model aims to achieve a more equitable representation of features, classes, and data points, ultimately enhancing the trade-off between fairness and performance and minimizing disparities in the learning process. In our work, we challenge the conventional notion of achieving fairness through last-layer fine-tuning [Mao+23]. We instead show that freezing the last layer is necessary and sufficient to strike a better balance between fairness and performance.

In our framework, the deep learning architecture is formalized as a composition of two distinct blocks: (i) the last layer, and (ii) preceding layers, as shown in Figure 1. TFS works as follows. In the first phase, we train the model without the stitching layer denoted as M. In the second phase, we add a trainable stitching layer between the two frozen layer blocks. In this contribution, we denote M_i as the i-th layer of the pre-trained model. The cost of adding a stitching layer is given by:

$$L^*(E;z;r) = \inf_{z \in Z} (L(E \circ z \circ r) + \text{fairness constraint}) \tag{10}$$

where $E = \{M_i\}_{i=0}^{n-1}$ is the preceding layers, $r = \{M_n\}$ is the last layer of pre-trained model M and z is the stitching layer.

4.4. Datasets

This contribution draws on two open-source image datasets, CelebA and UTKFace, for the experimental evaluation. Full details of both datasets are provided in *Publication I* [SR24]. Here we summarise only the key characteristics relevant to the thesis discussion.

CelebA dataset. The CelebA dataset consists of more than 200,000 celebrity face images annotated with 40 distinct attributes [Liu+15]. For the purposes of our experiments, hair color (blonde or non-blonde) is treated as the target label (y), whereas gender (male or non-male) is used as the sensitive attribute (a). Following the dataset partitioning protocol adopted by Liu et al. [Liu+15] and Mao et al. [Mao+23], we construct Table 1. In order to obtain a balanced sub-dataset, we sample from the original training and validation splits according to the image count of the minority subgroup. More specifically, we retain 1,569 images for each (y,a) grouping, which results in a balanced dataset containing 6,276 images in total.

UTKFace dataset. The UTKFace dataset is a publicly available facial-image dataset [ZSQ17; Mao+23] that covers a broad age range, from newborns to 116-year-old individuals. In this contribution, age is used as the target variable and gender as the sensitive attribute. Following Park et al. [Par+20], age is divided into two categories: "young" (≤ 35) and "others" (> 35) Table 2. In addition, and following the same balancing strategy described earlier, a balanced sub-dataset was constructed by sampling an equal number of images from the original training and validation splits for each (y,a) group [Mao+23]. This produced 2,477 images per balanced group, for a total of 9,908 images, with females older than 35 years forming the minority category.

	Blonde Hair	Non-blonde Hair	Total
Male	1,387/182/180	66,874/8,276/7,535	68,261/8,458/7,715
Female	22,880/2,874/2,480	71,629/8,535/9,767	94,509/11,409/12,247
Total	24,267/3,056/2,660	138,503/16,811/17,302	162,770/19,867/19,962

Table 1: Overview of the **(train/val/test)** CelebA Dataset.

	Young (≤ 35)	Old (> 35)	Total
Male	4,133/1,378/1,378	3,301/1,101/1,100	7,434/2,479/2,478
Female	4,931/1,643/1,644	1,858/619/619	6,789/2,262/2,263
Total	9,064/3,021/3,022	5,159/1,720/1,719	14,223/4,741/4,741

Table 2: Overview of the **(train/val/test)** UTKFace Dataset.

4.5. Experimental Settings

This section presents the experimental configuration used to examine how effectively the TFS framework improves fairness. To situate the evaluation, we first outline the baseline method FDR, which is included as part of the systematic comparison with TFS. In the taxonomy of Parraga et al. [Par+25], both methods are categorized as two-step training-based debiasing approaches. We then describe the architectural setting of the deep learning model employed in both FDR and TFS. Finally, in Section 4.5.3, we present the fairness and performance metrics used to quantitatively evaluate each method and to examine the trade-off between

predictive performance and fairness.

4.5.1. Baseline Method

This section focuses on the FDR baseline method used in our experimental evaluations, which serves as the point of reference for comparisons with our proposed framework (TFS). The FDR method introduced by Mao et al. [Mao+23] uses empirical risk minimization (ERM), fine-tuning the last layer, and a balanced dataset concerning both class and sensitive attributes. For a comprehensive understanding of the FDR method, we recommend referring to the original paper authored by Mao et al. [Mao+23] and *Publication I* [SR24].

Baseline selection and rationale. The primary objective of Contribution I is to study fairness *from the vantage point of model architecture*. As discussed in Section 3.2.1 (*Model Stitching*), relatively few works examine fairness through explicit architectural interventions. Accordingly, we focus on the most directly comparable baseline family, the FDR approach [Mao+23].[1]

This baseline shares the same controlled setting as TFS: a fixed pretrained backbone with targeted modifications at the output under a fixed fairness constraint, enabling a more controlled and fair comparison for the architectural question studied in Contribution I.

While other in-processing approaches (e.g., adversarial debiasing or direct equal-opportunity regularization applied throughout training) are important and complementary, they address fairness from a different vantage point (optimization-level objectives applied broadly across the network), rather than a localized architectural insertion. Moreover, several practical limitations of these broad in-processing approaches are discussed in Section 3.2 (*Overfitting Problem & Fairness*). In this sense, these approaches are not replacements for one another; rather, they are complementary tools that can potentially be combined. We therefore keep Contribution I tightly scoped to architecture-centric intervention and a baseline structurally aligned with that scope.

4.5.2. Model Architecture

In this contribution, we train two separate ResNet-18 models [Che+21], one on the CelebA dataset and the other on the UTKFace dataset, and use them within both the FDR and TFS frameworks. The ResNet-18 architecture is composed of six blocks, including the input layer, with each of the four main basic blocks containing two convolutional layers. The final block corresponds to the last layer.

Justification of The Fairness Stitch insertion point. The choice of where to insert the fairness stitch is motivated by three complementary lines of work. First, the model-stitching perspective [BNB21]. Second, prior work on *last-layer fairness fine-tuning* [Mao+23] argues that the final layer is often the most effective and stable

[1] To the best of our knowledge, this was among the first works to explicitly connect last-layer fine-tuning with the *overfitting problem & fairness* discussed in Section 3.2

location for improving the fairness–performance trade-off. Third, the information-theoretic view of deep networks [ST17] emphasizes that representations closer to the output typically retain *less information about the input* and *more information about the output/labels*. In this sense, the information-theoretic perspective can be seen as being *in tension* with a purely last-layer-centric intervention: if the final layers have already discarded much of the input-specific information (including potentially sensitive cues), then modifying only the last layer may be insufficient in some settings, motivating an intervention that reshapes the final representation *just before* the classifier.

The distinction by Mao et al. [Mao+23] between (Part 1) all layers before the last layer and (Part 2) the last layer became a key step in refining the approach, and it also informed the interpretation through the lens of Shwartz-Ziv and Tishby [ST17]. Taken together, these points motivate placing the fairness stitch *after* (Part 1) and *before* (Part 2), i.e., as an explicit transformation between the representation learned by the backbone and the final decision layer.

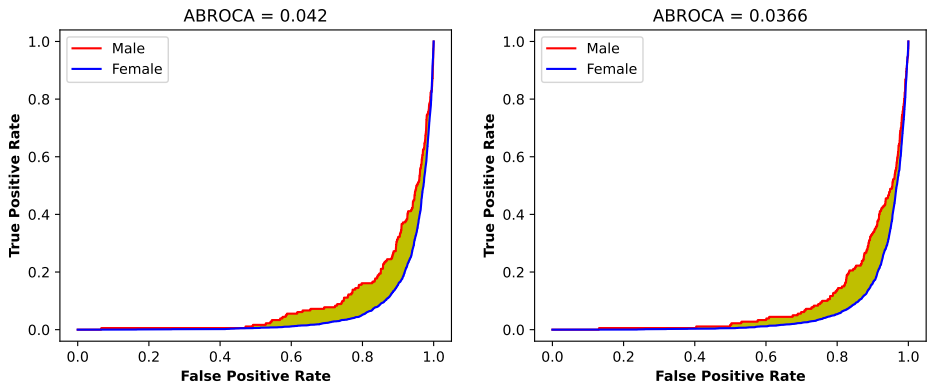
Generalization beyond CNNs (portability guidance). Conceptually, TFS requires (i) identifying an intermediate representation where biased information is present or emerges, and (ii) inserting a lightweight trainable mapping (the 'stitch') while freezing the remainder of the backbone. This recipe should transfer beyond CNNs, especially if we could know where the model retains more information about the input and less information about the output/labels. In particular:

- **MLPs:** insert the stitch between two layers (often near the penultimate layer [ST17]) as a small linear or bottleneck module, trained under the fairness objective while earlier layers remain frozen.
- **Transformers:** insert the stitch between encoder blocks (e.g., after block k) or immediately before the task head. Candidate positions include later blocks where task-relevant abstractions concentrate, or earlier blocks if sensitive-attribute leakage is detected early.

We further note that a principled way to select k is to analyze intermediate representations using information-theoretic or predictability-based probes (e.g., how well sensitive attributes can be predicted from representations at different depths, or how mutual information with sensitive attributes changes across layers), in order to find a better location to insert 'TFS' for improving the trade-off between fairness and performance. This analysis can guide stitch placement in architectures where 'blocks' are not convolutional (e.g., transformers). We explicitly connect this to the planned information-theoretic analysis that should be done for such cases as future work (Chapter 7 *Conclusion*).

Stitching layer design and training setup. This stitching layer, z, is a fully connected layer with an input dimension matching the output of the hidden layers. Thus, it produces an output dimension that matches the final layer's input dimension. Notably, the final models in both the TFS and FDR frameworks employ Stochastic Gradient Descent (SGD) with identical hyperparameters: a momentum of 0.9

and a weight decay of 5×10^{-4}. Furthermore, the stitching layer z is initialised with random weights and trained by minimising Equation 10, with its parameters remaining the only trainable component of the stitched model and all other weights are frozen throughout training. The family of stitching layers Z is drawn from the set of linear layers. In essence, the optimisation procedure trains z subject to fairness constraints on a balanced dataset of class and sensitive attribute pairs.



(a) FDR [Mao+23]

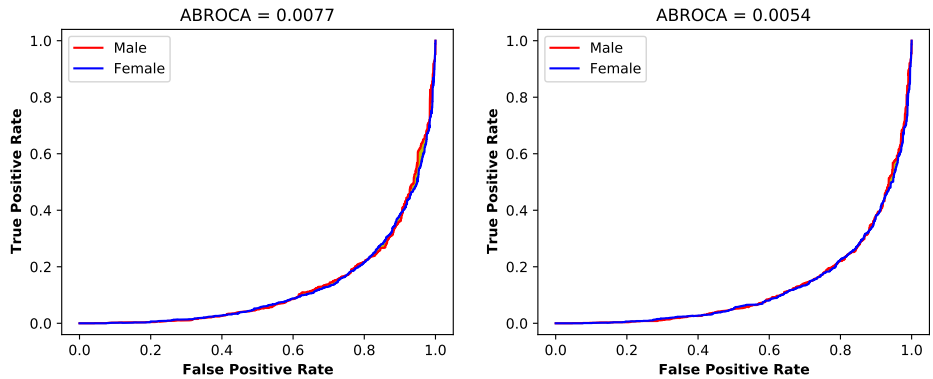
(b) TFS

Figure 2: ABROCA results on the CelebA. Figures 2a and 2b showcase ABROCA outcomes for both the FDR method [Mao+23] and our TFS framework. Specifically, they showcase the use of equalized odds as the fairness constraint with $\alpha = 20$ during fine-tuning (for FDR) and training (for TFS), respectively.

4.5.3. Performance and Fairness Metrics

To evaluate the effectiveness of TFS, we use a set of complementary performance and fairness metrics. On the performance side, we report Balanced Accuracy (BACC) [Bro+10], which accounts for class imbalance, and the Area Under the ROC Curve (AUC) [Faw04], which measures the model's ability to discriminate between positive and negative classes. On the fairness side, we report (Equalized Odds Difference (EO-Diff)) [HPS16], Accuracy Equality Difference (AE-Diff) (Section 2.3.3), and the Worst Accuracy (WA) [Mao+23], which respectively capture disparities in error rates, misclassification rates, and minimum group accuracy. Finally, the combined metric Balanced Accuracy and Fairness metric (AF) [Mao+23] integrates BACC with a fairness measure into a single score, computed as $\text{AF} = \text{BACC} - \text{EO-Diff}$ (Equalized Odds), $\text{AF} = \text{BACC} - \text{AE-Diff}$ (Accuracy Equality), or $\text{AF} = \text{BACC} + \text{WA}$ (Max-Min Fairness). We additionally report the Absolute Between-ROC Area (ABROCA) [GBB19], which quantifies the discrepancy between subgroup ROC curves across all thresholds by summing their absolute differences, with smaller values indicating greater similarity between subgroups. In all cases, smaller values of EO-Diff, AE-Diff, and ABROCA indicate better fairness, while larger values of BACC, AUC, WA, and AF indicate better

overall performance. For full formal definitions of each metric, we refer the reader to the respective original references and to *Publication I* [SR24].



(a) FDR [Mao+23]

(b) TFS

Figure 3: ABROCA results on the UTKFace. Figures 3a and 3b showcase ABROCA outcomes for both the FDR method [Mao+23] and our TFS framework. Specifically, they showcase the use of equalized odds as the fairness constraint with $\alpha = 2$ during fine-tuning (for FDR) and training (for TFS), respectively.

4.6. Main findings

This section presents the comparative evaluation of TFS against the FDR baseline across both datasets and all three fairness notions. Both methods were trained for 1000 epochs on a (class and sensitive attribute) balanced dataset, with model selection based on validation performance. Full numerical results are reported in Tables 3 and 4, and ABROCA curves are shown in Figures 2 and 3.

4.6.1. Baseline Bias Levels

Before applying any debiasing technique, the two datasets exhibit markedly different bias profiles, as shown in Tables 3 and 4. CelebA presents substantially higher initial bias (EO-Diff = 0.586), which we attribute primarily to its severe gender imbalance. UTKFace, being more balanced across sensitive groups, starts from a considerably lower bias level (EO-Diff = 0.143). This contrast motivates the use of dataset-specific hyperparameters: we set $\alpha = 20$ for CelebA and $\alpha = 2$ for UTKFace when enforcing the EO and AE constraints. These differences confirm that initial dataset characteristics play a decisive role in shaping the debiasing challenge.

4.6.2. TFS versus FDR: Overall Interpretation

Across all three fairness notions and both datasets, TFS consistently achieves a better fairness–performance trade-off than FDR, as summarised in Tables 3 and 4.

Without Applying Any Fairness Constraint (ResNet18)

Metric	Split	Value
BACC	Train	0.903
BACC	Balanced	0.792
BACC	Test	0.853
AUC	Train	0.987
AUC	Balanced	0.966
AUC	Test	0.971
EO-Diff	Test	0.586
AE-Diff	Test	0.213
WA	Test	0.183

Fairness Notion 1: Equalized Odds

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.898	0.887
BACC	Balanced	0.896	0.884
BACC	Test	0.876	0.874
AUC	Train	0.959	0.953
AUC	Balanced	0.956	0.947
AUC	Test	0.942	0.940
EO-Diff	Train	0.014	0.032
EO-Diff	Balanced	0.015	0.026
EO-Diff	Test	0.110	0.081
AF	Test	0.766	0.793

Fairness Notion 2: AE

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.906	0.900
BACC	Balanced	0.905	0.886
BACC	Test	0.883	0.881
AUC	Train	0.967	0.960
AUC	Balanced	0.964	0.951
AUC	Test	0.949	0.945
AE-Diff	Train	0.009	0.0002
AE-Diff	Balanced	0.001	0.011
AE-Diff	Test	0.008	0.0005
AF	Test	0.875	0.880

Fairness Notion 3: MMF

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.915	0.916
BACC	Balanced	0.913	0.906
BACC	Test	0.875	0.877
AUC	Train	0.978	0.979
AUC	Balanced	0.972	0.968
AUC	Test	0.960	0.960
WA	Train	0.864	0.865
WA	Balanced	0.880	0.867
WA	Test	0.800	0.811
AF	Test	1.675	1.688

Table 3: Results of our approach 'The Fairness Stitch' (TFS) with different fairness notions on the CelebA dataset. Higher is better for AUC, BACC, WA and AF; lower is better for EO-Diff and AE-Diff.

The key pattern is consistent: TFS reduces fairness gap metrics (EO-Diff, AE-Diff) and improves worst-case accuracy (WA) relative to FDR, while incurring only negligible losses in BACC and AUC. This confirms the central hypothesis of this contribution, that inserting a trainable stitching layer between frozen backbone layers, rather than fine-tuning the last layer alone, leads to a more effective and generalisable debiasing outcome.

The advantage of TFS is most pronounced under the equalized odds constraint, where it reduces EO-Diff from 0.110 to 0.081 on CelebA and from 0.062 to 0.058 on UTKFace, with a correspondingly lower ABROCA (0.0366 vs. 0.042 on CelebA). Under accuracy equality, TFS achieves a particularly striking reduction in AE-Diff on CelebA (from 0.008 to 0.0005), demonstrating strong capability to equalise misclassification rates across groups. Under the max-min fairness constraint, TFS improves worst-group accuracy on both datasets (0.811 vs. 0.800 on CelebA; 0.744 vs. 0.739 on UTKFace), reflecting its ability to lift the performance of the least-advantaged subgroup without sacrificing overall accuracy.

Taken together, these findings support the thesis argument that architectural

Without Applying Any Fairness Constraint (ResNet18)

Metric	Split	Value
BACC	Train	0.998
BACC	Balanced	0.947
BACC	Test	0.803
AUC	Train	1.000
AUC	Balanced	0.983
AUC	Test	0.885
EO-Diff	Test	0.143
AE-Diff	Test	0.023
WA	Test	0.675

Fairness Notion 1: Equalized Odds

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.996	0.996
BACC	Balanced	0.951	0.951
BACC	Test	0.796	0.793
AUC	Train	1.000	1.000
AUC	Balanced	0.986	0.985
AUC	Test	0.876	0.876
EO-Diff	Train	0.005	0.003
EO-Diff	Balanced	0.010	0.012
EO-Diff	Test	0.062	0.058
AF	Test	0.734	0.735

Fairness Notion 2: AE

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.998	0.992
BACC	Balanced	0.948	0.946
BACC	Test	0.798	0.796
AUC	Train	1.000	1.000
AUC	Balanced	0.985	0.984
AUC	Test	0.884	0.879
AE-Diff	Train	0.0008	0.010
AE-Diff	Balanced	0.003	0.001
AE-Diff	Test	0.016	0.0096
AF	Test	0.782	0.7864

Fairness Notion 3: MMF

Metric	Split	FDR [Mao+23]	TFS
BACC	Train	0.998	0.997
BACC	Balanced	0.949	0.949
BACC	Test	0.797	0.797
AUC	Train	1.000	1.000
AUC	Balanced	0.984	0.984
AUC	Test	0.881	0.880
WA	Train	0.997	0.995
WA	Balanced	0.933	0.934
WA	Test	0.739	0.744
AF	Test	1.536	1.541

Table 4: Results of our approach 'The Fairness Stitch' (TFS) with different fairness notions on the UTKFace dataset. Higher is better for AUC, BACC, WA and AF; lower is better for EO-Diff and AE-Diff.

interventions, specifically where in the network fairness constraints are applied, matter significantly for the attainable fairness–performance balance.

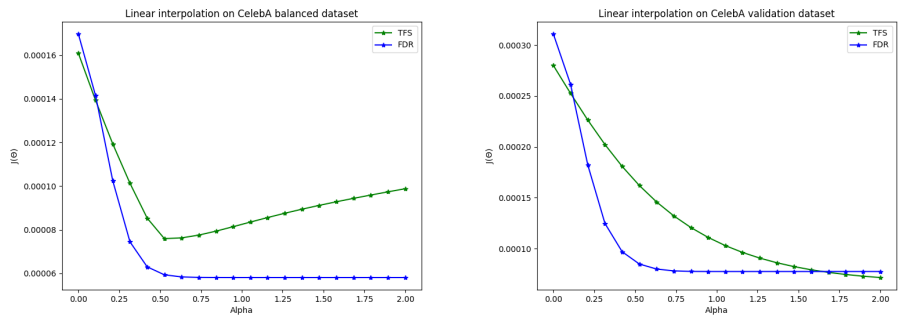
4.6.3. Loss Function Visualization

To further illuminate the behavioural difference between TFS and FDR, we visualise their respective objective functions using the 1-dimensional linear interpolation technique of Goodfellow et al. [GVS14]. The approach traces the loss $J(\theta) = \mathscr{L}(\theta(\alpha))$ along the line connecting a randomly initialised parameter vector θ_0 and the final trained parameter vector θ^*, parametrised as $\theta(\alpha) = (1-\alpha)\theta_0 + \alpha\theta^*$. Evaluating this trajectory on both the balanced training set and the validation set allows us to assess not only how well each method fits its training objective, but also how well that solution transfers to held-out data.

The results, shown in Figure 4, reveal a telling asymmetry. On the balanced dataset, FDR achieves a lower loss than TFS throughout the interpolation path, indicating that last-layer fine-tuning is highly effective at minimising the training objective under fairness constraints. However, on the validation dataset, this

advantage reverses: TFS reaches a lower final loss, while FDR's solution does not transfer as well to unseen data. This gap between training and validation behaviour is a classic signature of overfitting, in this case overfitting to the fairness objective on the balanced training set rather than learning a representation that generalises.

This observation is directly consistent with the theoretical concern raised in Section 4.3.1: if the last layer of a pretrained network retains minimal information about the input by the end of training, then forcing fairness constraints through that layer alone is akin to a superficial adjustment that fits the training distribution without addressing the deeper representational source of bias. TFS, by contrast, intervenes at an earlier point in the network where bias-relevant information is still present, producing a solution that is both fairer and more generalisable. The alignment between the loss visualisation in Figure 4 and the test-set results in Table 3 thus provides converging evidence that the architectural choice of where to apply fairness constraints, not merely whether to apply them, is an important determinant of the fairness–performance trade-off.



(a) Loss on CelebA Balanced Dataset (b) Loss on CelebA Validation Dataset

Figure 4: The linear interpolation curves for Resnet18 model using TFS and FDR frameworks on CelebA balanced and validation datasets.

4.7. Summary and impact

To our knowledge, at the time of publication of the article [SR24], this was the first work to challenge the conventional wisdom about the effectiveness of the last layer as an approach for achieving a good trade-off between fairness and performance. In particular, this contribution presented an innovative debiasing technique, 'The Fairness Stitch (TFS)', which combines model stitching and fairness constraints to address the trade-off between fairness and performance in machine learning. In addition, we evaluated the efficacy of our method on two popular open-source datasets, CelebA and UTKFace. Moreover, by comparing our results with the baseline method, our research found that TFS provides a better trade-off between fairness and accuracy. Besides, our work presents a practical debiasing method

with respect to both computational and sample complexity, especially for deep learning models. Therefore, our proposed method in this contribution challenges the conventional wisdom about the effectiveness of the last layer in mitigating bias. Furthermore, 'TFS' complements the surgical fine-tuning concept [Lee+22] but in the fairness context and provokes us to rethink the efficacy of the last layer for fairness in machine learning.

5. GLCS: ADVANCING EQUAL OPPORTUNITY FAIRNESS AND GROUP ROBUSTNESS THROUGH GROUP-LEVEL COST-SENSITIVE DEEP LEARNING (CONTRIBUTION II)

This chapter is based on the second contribution [SMR25] titled 'GLCS: Advancing Equal Opportunity Fairness and Group Robustness through **G**roup-**L**evel **C**ost-**S**ensitive Deep Learning'. While the previous chapter focused on the effect of the architecture of neural networks on the trade-off between fairness and performance for fair classification, this chapter consider the effects of group-level class-imbalanced datasets on the trade-off problem and, on that basis, proposes a new technique to improve it. More specifically, we proposed Group-Level Cost-Sensitive Deep Learning (GLCS) a pioneering cost-sensitive deep learning framework that addresses the problem of class imbalance at different group levels in the dataset, enabling more nuanced handling of demographic disparities in machine learning systems. Moreover, 'GLCS' is a novel cost-sensitive optimization technique that mitigates performance disparities by strategically balancing group-level representations, thereby improving equal opportunity fairness without compromising overall model accuracy. In addition, one of the main findings of this contribution is the laying of foundations at the intersection of cost-sensitive deep learning, group fairness, and group robustness in machine learning. Furthermore, this contribution provided extensive experimental results that validate the generalizability and effectiveness of our approach, showcasing consistent improvements in both group robustness and group fairness, with a particular focus on equal opportunity.

5.1. Introduction

In many real-world applications, the label distribution can differ substantially across sensitive groups. Such imbalances may lead a model to allocate capacity unevenly, which can in turn produce disparate error rates across groups even when the overall accuracy is high. This phenomenon is particularly evident under fairness notions that condition on the true label, such as *equal opportunity*, which aims to reduce group disparities in true-positive rates. At the same time, enforcing fairness constraints can incur a performance cost, and understanding this fairness–utility trade-off requires metrics that go beyond average accuracy.

A fundamental challenge that pervades the pursuit of fairness in machine learning is the phenomenon of spurious correlations. Rather than learning genuine causal relationships between features and outcomes, models trained on real-world datasets frequently capture misleading statistical associations that have no causal grounding. Such correlations can emerge through several mechanisms: sampling procedures that do not adequately represent the underlying population, historical

imbalances embedded in archival data, or coincidental co-occurrence patterns that happen to be present in a particular training set but do not generalise beyond it.

The consequences of spurious correlations are not confined to degraded predictive performance in a narrow technical sense. When a model's decisions are driven by features that merely correlate with, rather than cause, the target outcome, there is a substantial risk that pre-existing societal biases are reproduced, or even amplified, at inference time. This is particularly concerning for minority groups, who are liable to bear a disproportionate share of discriminatory outcomes produced by such models.

A natural response to these concerns is the pursuit of *group robustness*: the requirement that a model perform well not only on average across the full dataset, but specifically across every predefined subgroup within it. This property is typically operationalised through the *worst-group accuracy* metric, defined as the minimum accuracy attained over all groups. Even this more demanding objective is, however, not insulated from the difficulties introduced by spurious correlations. Models optimised via standard empirical risk minimisation (ERM) are well known to underperform on underrepresented groups, a failure that can be attributed to both the geometric and statistical skew present in the input training data.

The second contribution of this thesis [SMR25] examines the trade-off between equal opportunity and performance in machine learning by addressing class imbalance at different group levels in the dataset using a cost-sensitive deep learning framework. In general, the admissible trade-offs between equal opportunity and accuracy have been studied by Pinzón et al. [Pin+24; Pin+22]. In particular, these works characterize conditions on the underlying data source under which equal opportunity can be compatible with non-trivial predictive accuracy.

Beyond the fairness–performance trade-off, this chapter also establishes a connection between group fairness and group robustness that is rarely made explicit in the existing literature. It is shown that a framework designed to improve a group fairness metric simultaneously promotes group robustness, in the sense that it encourages more consistent predictive performance across all subgroups. This dual benefit is not incidental; it reflects the fact that both objectives are undermined by the same underlying pathology, namely the tendency of models optimised via standard empirical risk minimisation to exploit spurious statistical patterns in ways that disproportionately harm minority groups.

To address these interrelated challenges in a unified manner, this chapter proposes the *Group-Level Cost-Sensitive* (GLCS) framework. Building on the principles of cost-sensitive deep learning [Kha+17; ZL05], the GLCS framework incorporates misclassification costs that are defined and applied at the subgroup level, embedding fairness considerations directly into the learning objective.

The remainder of this chapter is structured as follows. Section 5.2 establishes the mathematical preliminaries that underpin the proposed framework. Section 5.3 sets out the formal problem statement and the corresponding learning objective. Section 5.4 describes the datasets and baseline methods used in the empirical

evaluation, while Section 5.5 defines the evaluation metrics. Section 5.6 details the experimental configuration. The subsequent sections present the results and discussion.

The work presented in this chapter draws on *Publication II* [SMR25], of which the author of this thesis is the primary author. Certain portions of the text, figures, and tables have been reproduced or adapted from that publication.

5.2. Mathematical Preliminaries

This section introduces the definitions of different concepts used in our second contribution [SMR25]. In particular, we define *group robustness*, which we use in our experiments on the CivilComments-WILDS dataset (Section 5.4) to assess the group robustness of our proposed method, GLCS. Moreover, we provide formal definitions of the augmented Lagrangian method and the different dataset partitioning schemes, which form the basis of the proposed approach (Section 5.3).

5.2.1. Problem Setup

Consider a binary classification problem with a deep neural network 'DNN' classifier $f_\theta(\cdot)$ parameterized by θ, and denote $F(\theta)$ as the associated loss function. Let M denote our dataset. More specifically, suppose we have the dataset

$$M = \{(x_i, s_i, y_i)\}_{i=1}^{m},$$

where $x_i \in \mathbb{R}^d$ is the feature vector, $y_i \in \{0,1\}$ is the label, and s_i is the protected attribute. For example, s_i can be race, gender, etc.

Definition 5.2.1 (Group Robustness)**.** Group robustness refers to maintaining consistent model performance across all subgroups [Liu+21; LMK24]. It focuses on identifying and mitigating spurious correlations, improving performance on the worst-performing groups, and maintaining high overall accuracy while improving minority-group performance.

Definition 5.2.2 (Augmented Lagrangian Method (ALM))**.** Augmented Lagrangian Method (ALM), also known as the method of multipliers, is a powerful optimization technique that bridges the gap between constrained and unconstrained optimization problems. Introduced by Bertsekas [Ber76].

Definition 5.2.3 (Class-based Partitioning)**.** The dataset M can be partitioned into positive (critical) and negative classes as follows:

$$\begin{aligned} P &= \{x_i^p\}_{i=1}^{|P|} \quad \text{(positive class samples)} \\ N &= \{x_i^n\}_{i=1}^{|N|} \quad \text{(negative class samples)} \end{aligned} \tag{5.1}$$

where $|P| < |N|$, indicating P represents the minority class [San+21].

Definition 5.2.4 (Dataset Partitioning by Protected Attribute and Label)**.** Let $s \in \{0,1\}$ denote the protected attribute (e.g., gender or race). The dataset M is first partitioned into two disjoint subsets according to this attribute:

$$\begin{aligned} Z_1 &= \{x_i^{s_1}\}_{i=1}^{|Z_1|} \quad \text{(group with } s=1\text{)} \\ Z_0 &= \{x_i^{s_0}\}_{i=1}^{|Z_0|} \quad \text{(group with } s=0\text{)} \end{aligned} \tag{5.2}$$

where $|Z_1| < |Z_0|$, so that Z_0 denotes the majority (non-protected) group and Z_1 the minority (protected) group. Each group is then further partitioned into intersectional subgroups according to the true label $y \in \{0,1\}$, yielding four disjoint subsets in total:

$$\begin{aligned} Z_{1,1} &= \{x_i^{s_1,y_1}\}_{i=1}^{|Z_{1,1}|} \quad \text{(protected group, positive class)} \\ Z_{1,0} &= \{x_i^{s_1,y_0}\}_{i=1}^{|Z_{1,0}|} \quad \text{(protected group, negative class)} \\ Z_{0,1} &= \{x_i^{s_0,y_1}\}_{i=1}^{|Z_{0,1}|} \quad \text{(non-protected group, positive class)} \\ Z_{0,0} &= \{x_i^{s_0,y_0}\}_{i=1}^{|Z_{0,0}|} \quad \text{(non-protected group, negative class)} \end{aligned}$$

where $|Z_{1,1}| < |Z_{1,0}|$ and $|Z_{0,1}| < |Z_{0,0}|$, indicating that the positive class constitutes the minority class within each protected attribute group.

5.3. GLCS Framework

This chapter proposes the Group-Level Cost-Sensitive (GLCS) framework, a method formulated as a constrained optimisation problem with the objective of achieving equal opportunity in classification. The framework builds on the constrained optimisation approach introduced by Sangalli et al. [San+21], which addressed imbalanced dataset classification through constraints (5.3a) and (5.3b). The present work extends that foundation by introducing two additional constraints, (5.3c) and (5.3d), which enforce equal opportunity across protected groups. The full problem formulation is expressed as:

$$\min_{\theta} F(\theta) \quad \text{subject to:} \tag{5.3a}$$

$$\sum_{k=1}^{|N|} \max\left(0, -(f_\theta(x_j^p) - f_\theta(x_k^n)) + \delta\right) = 0, \quad \forall j \in \{1,\ldots,|P|\} \tag{5.3b}$$

$$\sum_{k=1}^{|Z_{1,0}|} \max\left(0, -(f_\theta(x_l^{s_1,y_1}) - f_\theta(x_k^{s_1,y_0})) + \delta\right) = 0, \quad \forall l \in \{1,\ldots,|Z_{1,1}|\} \tag{5.3c}$$

$$\sum_{k=1}^{|Z_{0,0}|} \max\left(0, -(f_\theta(x_r^{s_0,y_1}) - f_\theta(x_k^{s_0,y_0})) + \delta\right) = 0, \quad \forall r \in \{1,\ldots,|Z_{0,1}|\} \tag{5.3d}$$

where: $f_\theta(\cdot) : \mathscr{X} \to [0,1]$ is the DNN's output probability function, θ represents the DNN parameters and $\delta > 0$ is the margin parameter. Subsequently, we derive an equivalent unconstrained form of the above constrained system defined in equations (5.3a)-(5.3d) using the augmented Lagrangian method (ALM):

$$\begin{aligned}\mathscr{L}_\mu(\theta,\lambda) = F(\theta) &+ \underbrace{\frac{\mu_1}{2|P||N|}\sum_{j=1}^{|P|} q_j^2 + \frac{1}{|P||N|}\sum_{j=1}^{|P|}\lambda_j q_j}_{\text{Global Class Separation}} \\ &+ \underbrace{\frac{\mu_2}{2|Z_{1,1}||Z_{1,0}|}\sum_{l=1}^{|Z_{1,1}|} q_l^2 + \frac{1}{|Z_{1,1}||Z_{1,0}|}\sum_{l=1}^{|Z_{1,1}|}\lambda_l q_l}_{\text{Class Separation Within Protected Group}} \\ &+ \underbrace{\frac{\mu_3}{2|Z_{0,1}||Z_{0,0}|}\sum_{r=1}^{|Z_{0,1}|} q_r^2 + \frac{1}{|Z_{0,1}||Z_{0,0}|}\sum_{r=1}^{|Z_{0,1}|}\lambda_r q_r}_{\text{Class Separation Within Non-Protected Group}}\end{aligned} \tag{5.4}$$

where the constraint violations (q_j, q_l and, q_r) are defined as:

$$q_j = \sum_{k=1}^{|N|} \max\left(0, -(f_\theta(x_j^p) - f_\theta(x_k^n)) + \delta\right) \qquad \text{(global)}$$

$$q_l = \sum_{k=1}^{|Z_{1,0}|} \max\left(0, -(f_\theta(x_l^{s_1,y_1}) - f_\theta(x_k^{s_1,y_0})) + \delta\right) \qquad \text{(protected)}$$

$$q_r = \sum_{k=1}^{|Z_{0,0}|} \max\left(0, -(f_\theta(x_r^{s_0,y_1}) - f_\theta(x_k^{s_0,y_0})) + \delta\right) \qquad \text{(non-protected)}$$

In this formulation, $\mu_1, \mu_2, \mu_3 > 0$ denote the penalty coefficients associated with the quadratic terms, and $\lambda_j, \lambda_l, \lambda_r$ are the Lagrange multipliers corresponding to positive samples within each respective group, updated iteratively as $\lambda_j^{(t+1)} = \lambda_j^{(t)} + \mu \cdot q_j^{(t)}$. The parameter $\delta > 0$ controls the margin.

This unconstrained reformulation confers two properties that are central to the design of the GLCS framework. First, it enables *asymmetric treatment* of positive and negative classes across the full dataset M and the subgroups Z_1 and Z_0, reflecting the differing relative importance of errors in each class. Second, it directs *performance focus* towards reducing the False Positive Rate (FPR) at high True Positive Rate (TPR) across all groups, thereby prioritising the part of the operating range most consequential for fairness-aware evaluation.

5.4. Datasets and Baselines

The empirical evaluation presented in this chapter draws on three datasets: CelebA [Liu+15] and UTKFace [ZSQ17] for facial attribute analysis and demographic

fairness assessment, and CivilComments-WILDS [Koh+21] for evaluating group robustness under distribution shifts.

5.4.1. Datasets

The CelebFaces Attributes (CelebA) dataset [Liu+15] comprises 202,599 celebrity images annotated with 40 binary attributes [Han+24b]. Following the preprocessing protocol of the Fair Fairness Benchmark (FFB) [Han+24b], the task considered here is binary classification of the "Wavy Hair" attribute (y), with "Gender" treated as the protected attribute (s). The dataset is partitioned into training (80%, 162,770 samples), validation (10%, 19,867 samples), and test sets (10%, 19,962 samples). The distribution of samples across gender groups ($s = 1$ for male, $s = 0$ for female) and target labels ($y = 1$ for wavy hair) is reported in Table 5.

The UTKFace dataset [ZSQ17] contains over 20,000 facial images annotated with age, gender, and ethnicity. The dataset exhibits broadly balanced distributions across major demographic factors, with 12,661 young and 11,044 older subjects, and near-equal gender representation (12,391 male, 11,314 female). As shown in Table 6, notable age–gender interactions are present. Following Han et al. [Han+24b], images are standardised to 48×48 pixels across three colour channels and partitioned into training (18,964 samples), validation (2,371 samples), and test (2,370 samples) sets.

The CivilComments-WILDS dataset [Koh+21], derived from [Bor+19a], comprises approximately 450,000 online comments annotated for toxicity and for mentions of eight demographic identities: gender (male, female), sexual orientation (LGBTQ), race (black, white), and religion (Christian, Muslim, or other). This dataset is particularly relevant to the study of group robustness, as it may contain spurious correlations between demographic mentions and toxicity labels. Following Koh et al. [Koh+21], 16 overlapping groups are defined as (a, toxic) and $(a, \text{non-toxic})$ for each demographic identity a. The distribution of comments across dataset splits is summarised in Table 7. An initial analysis using Empirical Risk Minimisation (ERM) identified comments mentioning Christian identity as the worst-performing group; this group is consequently designated as the sensitive attribute within the GLCS framework.

5.4.2. Baselines

For the group fairness evaluation on CelebA and UTKFace, the GLCS framework is compared against two baselines: Empirical Risk Minimisation (ERM) [Vap91] and DiffEopp [CM21; HPS16].

For the group robustness evaluation on CivilComments-WILDS, a broader set of state-of-the-art methods is considered as baselines: ERM [Vap91], Just Train Twice (JTT) [Liu+21], Deep Feature Reweighting (DFR) [KIW22], Automatic Feature Reweighting (AFR) [Qiu+23], and Group Distributionally Robust Optimisation (Group-DRO) [Sag+19].

Target Attribute Distribution (Wavy Hair)		
Positive Class ($y = 1$)		51,982
Negative Class ($y = 0$)		110,788
Protected Attribute Distribution (Gender)		
Male ($s = 1$)		68,261
Female ($s = 0$)		94,509
Intersectional Distribution		
Male with Wavy Hair	$P(s = 1\|y = 1)$	9,762
Male without Wavy Hair	$P(s = 1\|y = 0)$	58,499
Female with Wavy Hair	$P(s = 0\|y = 1)$	42,220
Female without Wavy Hair	$P(s = 0\|y = 0)$	52,289

Table 5: CelebA Dataset Statistics and Demographic Distribution

Demographic	Age Group		Total
Group	Old ($y = 1$)	Young ($y = 0$)	Sample
Male	6,854	5,537	12,391
Female	4,190	7,124	11,314
Total	11,044	12,661	23,705

Table 6: Demographic Distribution of Age Categories in UTKFace Dataset

5.5. Metrics

This section describes the evaluation metrics employed in the experiments reported in this chapter. The metrics are organised into five categories: (i) threshold-agnostic performance metrics, (ii) threshold-dependent performance metrics, (iii) group fairness metrics, (iv) nuanced metrics, and (v) a group robustness metric.

5.5.1. Threshold-Agnostic Performance Metrics:

The assessment of binary classification models relies on several threshold-agnostic performance metrics, each of which captures a distinct aspect of model behaviour across the full range of decision thresholds.

- **Precision-Recall Area Under the Curve** (PR-AUC): It is derived from the precision-recall curve and is particularly valuable in scenarios with class imbalance, as it focuses on the positive class and is less affected by a large number of true negatives. It effectively captures the trade-off between precision (positive predictive value) and recall (sensitivity) across various classification thresholds.
- **Receiver Operating Characteristic Area Under the Curve** (ROC-AUC): Conversely, the ROC-AUC, based on the ROC curve, illustrates the model's

Split	Number of Comments	Distribution (%)
Training	269,038	59.79
Validation	45,180	10.04
Test	133,782	29.73
Total	450,000	100.00

Table 7: Data Distribution in CivilComments-WILDS Dataset

ability to discriminate between classes by plotting the true positive rate against the false positive rate at various threshold settings. ROC-AUC is invariant to class distribution, making it a robust metric for comparing model performance across different datasets.

- **The Brier score**: The Brier score is another crucial metric that measures the mean squared difference between the predicted probability and the actual outcome. Ranging from 0 to 1, where lower scores indicate better calibration, the Brier score is particularly useful for assessing the accuracy of probabilistic predictions. It not only evaluates a model's discrimination power but also its calibration, providing a holistic view of predictive performance.
- **Average Precision (AP)**: AP summarizes the precision-recall curve by computing a weighted average of the precision values at all thresholds, where each weight corresponds to the increase in recall between consecutive thresholds.
- **AUC-PR Gain**: AUC-PR Gain gives an improvement over traditional PR analysis by introducing normalized gain metrics that enable more meaningful model comparisons [FK15].

Together, these metrics offer a multifaceted evaluation framework, enabling us to comprehensively assess and compare the performance of binary classification models across various operational contexts and decision thresholds.

5.5.2. Threshold-Dependent Performance Metrics

In binary classification, a crucial prerequisite for computing several threshold-dependent performance metrics is selecting an appropriate classification threshold. This threshold determines how probability scores are converted into discrete class predictions, significantly impacting the resulting metrics. Optimal threshold selection is paramount for fair model comparison and real-world applicability, as it balances the trade-off between different types of errors (e.g., false positives and false negatives) and accounts for class imbalance and misclassification costs. The following threshold-dependent performance metrics used in our experiments are particularly sensitive to threshold selection:

- **Balanced Accuracy**: This metric addresses the limitations of standard accuracy in imbalanced datasets by calculating the arithmetic mean of sensitivity (true positive rate) and specificity (true negative rate) [Bro+10]. Formally

defined as $\frac{1}{2}(\frac{TP}{TP+FN} + \frac{TN}{TN+FP})$, balanced accuracy provides an unbiased performance measure even when classes are of very different sizes.

- **F1 Score**: The harmonic mean of precision and recall, the F1 score is calculated as $2 \cdot \frac{precision \cdot recall}{precision + recall}$. It provides a single score that balances precision and recall, making it particularly useful when seeking an optimal balance between the two.
- **Matthews Correlation Coefficient (MCC)**: Regarded as a balanced measure that can be used even if the classes are of very different sizes, the MCC is essentially a correlation coefficient between the observed and predicted binary classifications.
  It is defined as $\frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$ and ranges from -1 to +1.
- **Precision**: Also known as positive predictive value, precision measures the proportion of true positive predictions among all positive predictions: $\frac{TP}{TP+FP}$. It is particularly important in applications where false positives are costly.
- **Recall**: Also termed sensitivity or true positive rate, recall quantifies the proportion of actual positive instances that were correctly identified: $\frac{TP}{TP+FN}$. It is crucial in scenarios where missing positive instances (false negatives) are particularly undesirable.

These metrics, while providing valuable insights into model performance, are inherently dependent on the chosen classification threshold. Therefore, when reporting these metrics, it is essential to specify the threshold used and the method used to determine it. Furthermore, in cost-sensitive learning scenarios or when dealing with imbalanced datasets, optimizing the threshold for each model separately ensures a fairer comparison and more accurately reflects each model's potential performance in real-world applications.

In Section 5.6.2, we provide a comprehensive analysis of the threshold optimization technique employed in our experiments for the evaluation of threshold-dependent performance metrics. This critical aspect of our methodology addresses the inherent sensitivity of various performance metrics to classification thresholds in binary classification tasks.

By providing this in-depth examination of threshold optimization, we aim to ensure transparency in our evaluation process and facilitate a more nuanced understanding of model performance in real-world applications. This detailed exposition not only underpins the reliability of our experimental results but also offers valuable insights into broader challenges and considerations in selecting and interpreting performance metrics within the domains of group fairness and group robustness in machine learning.

5.5.3. Group Fairness Metrics

We utilized the following metrics to evaluate the group fairness of our technique. We used the same fairness metrics as described by Han et al. [Han+24b]. In their

work, Han et al. [Han+24b] introduced the FAIR FAIRNESS BENCHMARK (FFB), a benchmarking framework specifically designed for in-processing group fairness methods. The authors considered a variety of fairness metrics, including demographic parity, p-rule, equality of opportunity, equalized odds, the area between CDF curves, and more. The implementations of these metrics in our experiments are based on the implementation in the Fairness Benchmark (FFB) [Han+24b].

- **Demographic Parity (dp)** [Zem+13; Dwo+12]: A classifier satisfies demographic parity if the predicted outcome $\hat{Y}$ is independent of the sensitive attribute S, i.e.,
$$P(\hat{Y}|S=0) = P(\hat{Y}|S=1).$$
- **p-Rule (prule)** [Zaf+17]: A classifier satisfies the p-rule if the ratio of the probability of individuals with a certain sensitive attribute value receiving a positive outcome to the probability of those without that value receiving a positive outcome is no less than $p/100$, i.e.,
$$\left|\frac{P(\hat{Y}=1|S=1)}{P(\hat{Y}=1|S=0)}\right| \leq \frac{p}{100}.$$
- **Equalized Odds (eodd)** [HPS16]: A classifier satisfies equalized odds if the predicted outcome $\hat{Y}$ is independent of the sensitive attribute S for both $Y=0$ and $Y=1$. In Section 2.3.3, we provide a formal mathematical definition of this metric.
- **Equality of Opportunity (eopp)** [HPS16]: A classifier satisfies equality of opportunity if individuals with positive true labels ($Y=1$) have equal probabilities of receiving a positive prediction ($\hat{Y}=1$) regardless of their sensitive attribute (S). In Section 2.3.3, we provide a formal mathematical definition of this metric.
- **ROC AUC Parity (AUCP)**: A classifier satisfies ROC AUC parity if the area under the receiver operating characteristic (ROC) curve is the same across different sensitive attribute groups.
- **Balance for Negative Class (BFN)** [KMR16]: A classifier satisfies balance for the negative class if the average predicted probability for the negative class is the same across different sensitive attribute groups, i.e.,
$$\mathbb{E}[f(X)|Y=0,S=0] = \mathbb{E}[f(X)|Y=0,S=1].$$
- **Balance for Positive Class (BFP)** [KMR16]: A classifier satisfies balance for the positive class if the average predicted probability for the positive class is the same across different sensitive attribute groups, i.e.,
$$\mathbb{E}[f(X)|Y=1,S=0] = \mathbb{E}[f(X)|Y=1,S=1].$$

Fairness Metrics Notation in Contribution II [SMR25]. For the group fairness metrics, we adopt a systematic notation that distinguishes between probability-based and threshold-based evaluations. When utilizing output probability estimates, we denote demographic parity, equal opportunity, equalized odds, and p-Rule as *dpe*, *eoppe*, *eodde*, and *prulee*, respectively. Conversely, when evaluating binary predictions derived from threshold-based classification, these metrics are denoted as *dp*, *eopp*, *eodd*, and *prule*. This notational convention aligns with the experimental framework and code implementation used in FFB [Han+24b], providing consistency in metric interpretation across probability and binary domains.

5.5.4. Nuanced Metrics:

The pinned AUC metric, introduced by Dixon et al. [Dix+18], is designed to measure the unintended bias in machine learning models. It provides a threshold-agnostic assessment of unintended bias in classification models, inspired by the ROC-AUC metric. However, Borkan et al. [Bor+19b] addresses the limitations of the pinned AUC. Furthermore, Borkan et al. [Bor+19a] proposed alternative metrics, including subgroup-AUC, Background Positive Subgroup Negative AUC (BPSN-AUC), and Background Negative Subgroup Positive AUC (BNSP-AUC), to identify various types of biases. For instance, in the CivilComments dataset, these metrics split the data into two subgroups: one representing identity groups, which includes both toxic and non-toxic comments, and another representing a background group, which also includes both toxic and non-toxic comments.

- **Subgroup-AUC:** This metric calculates the AUC using the identity subgroup and reflects the model's understanding of this subgroup. A lower AUC indicates that the model struggles to distinguish between toxic and non-toxic comments containing identity-related terms within the subgroup.
- **BPSN-AUC:** This metric calculates the AUC by comparing toxic comments from the background group with non-toxic comments from the identity subgroup. A low AUC suggests that the model confuses identity mentions in non-toxic comments with identity mentions in toxic comments.
- **BNSP-AUC:** This metric computes the AUC by comparing non-toxic comments from the background group with toxic comments from the identity subgroup. A lower AUC indicates that the model struggles to differentiate between toxic comments mentioning identity and non-toxic comments that do not mention identity.

5.5.5. Group Robustness Metric

Worst-Group Accuracy: Using the CivilComments-WILDS dataset, we evaluate the group robustness of a model based on its worst-group accuracy, which is defined as the lowest accuracy observed across 16 predefined groups. A higher worst-group accuracy, relative to average accuracy, suggests that the model is less likely to mistakenly associate demographic identities with toxicity. This approach

allows us to evaluate performance across multiple test distributions, corresponding to different subsets of the test set based on demographic identities and labels, and to report the worst performance observed [Koh+21].

5.6. Experimental Setting

5.6.1. Architecture and Training Configuration

The experiments conducted on the CelebA and UTKFace image datasets adopt the ResNet-18 architecture [He+16] as the backbone network, following the implementation detailed in the Fair Fairness Benchmark (FFB) [Han+24b]. For the CivilComments-WILDS dataset, the GLCS framework is built on the Bidirectional Encoder Representations from Transformers (BERT) model [Dev18]. To ensure comparability, the baseline methods on this dataset, ERM, DFR, Group-DRO, JTT, and AFR, follow the implementations specified in [Qiu+23].

Early Stopping Criterion. Determining an appropriate stopping criterion in fair machine learning is particularly challenging, given the inherent tension between competing objectives such as group robustness, model utility, and fairness. This aspect of the training procedure has received limited attention in the existing literature. Han et al. [Han+24b] proposed a deterministic stopping strategy within the FFB framework, based on learning rate decay. An alternative empirical approach was introduced in **Sulaiman** and Roy [SR24], whereby training is halted when a satisfactory trade-off among utility and fairness metrics is observed on the validation set, typically within the early epochs of training. The experiments in this chapter follow the latter approach [SR24].

Computational overhead and scalability. The proposed GLCS training procedure introduces additional computations compared to standard ERM due to the constraint terms and performing the augmented Lagrangian updates. In our implementation, all constraint terms are computed *within each mini-batch* using GPU operations, so the overhead mainly comes from the additional pairwise comparisons in the batch; the multiplier/penalty updates are lightweight and add only a negligible constant cost per iteration. In practice, the wall-clock overhead is modest and scales similarly to ERM. Across our experiments, GLCS typically increases training time by about 10%-20% relative to ERM on the same hardware. Overall, GLCS remains practically scalable, and runtime is still dominated by the backbone model and dataset size, with GLCS introducing only a small multiplicative factor.

5.6.2. Classification Thresholds

Standard binary classification assumes a decision threshold of 0.5, which implicitly presupposes balanced class representation. This assumption is violated in several of the datasets considered here. In CelebA, for instance, positive instances account for only 32% of the training data (51,982 out of 162,770 samples), making the

default threshold systematically disadvantageous for the minority class. In such settings, the threshold should be recalibrated to approximately 0.32 for CelebA.

Empirical Methodology. Threshold optimisation for the CelebA and UTKFace experiments is carried out using the `binclass-tools` package[1] for each model (ERM, DiffEopp, and GLCS).

Calibrating Neural Networks. Deep neural networks often produce poorly calibrated probability estimates, tending towards overconfidence or underconfidence even when discriminative performance is strong. A well-calibrated model should produce confidence scores that reflect true empirical frequencies: among all predictions assigned a confidence of 0.8, for instance, approximately 80% should be correct. Reliable calibration is particularly important in high-stakes settings where probability estimates directly inform decisions. To address this, Temperature Scaling [Guo+17] is applied as a post-processing step in our experiments.

5.6.3. GLCS Methodology with CivilComments-WILDS Dataset

The experimental procedure for the CivilComments-WILDS dataset proceeds in two stages. In the first stage, a baseline model is trained using Empirical Risk Minimisation (ERM), which serves both to establish a performance reference and to identify group-specific disparities in the data. As reported in Section 5.7.3, this initial evaluation reveals that the Christian demographic group suffers the most pronounced accuracy degradation, and it is therefore designated as the sensitive group for the subsequent intervention, where we use a fixed decision threshold of 0.5 for GLCS and for all baseline methods, matching the standard practice adopted by the compared approaches in our experimental setup. In the second stage, the GLCS framework is applied with a targeted intervention directed at improving performance for the Christian group.

5.6.4. Hyperparameters

This section details the hyperparameter configurations used for each dataset.

CelebA and UTKFace Datasets. For the GLCS framework proposed in Section 5.3, we selected hyperparameters based on the unique characteristics of each dataset. On the CelebA dataset, characterized by substantial intersectional distribution disparities (as evidenced in Table 5), we employed $\mu_1 = \mu_2 = \mu_3 = 0.5$ and $\delta = 0.2$. Conversely, the UTKFace dataset, exhibiting minimal group distribution variations (detailed in Table 6), warranted a more nuanced approach with $\mu_1 = \mu_2 = \mu_3 = \varepsilon$ and $\delta = 0.1$, where ε is a small number. For comparative methods, namely ERM and DiffEopp, we consistently utilized the hyperparameters established in the Fair Fairness Benchmark (FFB) [Han+24b] across both datasets.

CivilComments-WILDS Dataset. For the CivilComments-WILDS dataset, we used $\mu_1 = \mu_2 = \mu_3 = 1$ and $\delta = 0.2$. For comparative methods, we used the hyperparameters for Just Train Twice (JTT) as specified in Liu et al. [Liu+21], and

[1] `https://github.com/lucazav/binclass-tools`

those for ERM, AFR, DFR, and Group-DRO as in Qiu et al. [Qiu+23]. Consistent with prior work by Qiu et al. [Qiu+23] and Liu et al. [Liu+21], we applied the standard 0.5 threshold for metric evaluation across all methods.

5.7. Main findings

This section presents the results from our second contribution [SMR25]. In particular, we evaluated the proposed GLCS approach against standard baselines on three datasets: CelebA, UTKFace, and CivilComments-WILDS. We also considered a calibrated variant, *Calibrated GLCS* (GLCS followed by Temperature Scaling). Our analysis examined how well our proposed GLCS approach balances predictive performance with fairness objectives. In what follows, we discuss the findings for each dataset in turn.

5.7.1. Experimental Evaluation on CelebA Dataset

Analysis of Threshold-Agnostic Performance Metrics. Table 8 reports the threshold agnostic performance metrics for all methods on the CelebA dataset. ERM achieves the strongest discriminative performance, while GLCS and Calibrated GLCS occupy an intermediate position between ERM and DiffEopp across all metrics. The two GLCS variants share identical discriminative performance but differ notably in calibration: Calibrated GLCS attains a substantially lower Brier Score (0.2044) than standard GLCS (0.3349).

Metric	ERM	DiffEopp	GLCS	Calibrated GLCS
ROC AUC ↑	0.8576	0.7815	0.8443	0.8443
PR AUC ↑	0.7913	0.6011	0.7371	0.7371
Brier Score ↓	0.1475	0.1861	0.3349	0.2044
AUPR Gain ↑	0.7911	0.6008	0.7369	0.7369

Table 8: Invariant performance metrics on CelebA.

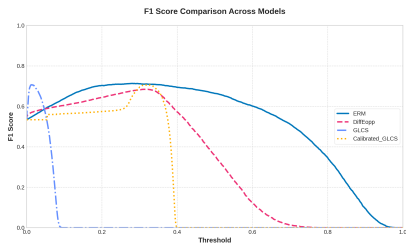
Analysis of Threshold-Dependent Performance Metrics. Table 9 reports the threshold-dependent performance metrics for all methods on the CelebA dataset. ERM achieves the highest balanced accuracy (BACC: 0.7737) and a strong F1 score (0.7132) at a threshold of 0.281. Calibrated GLCS attains comparable performance (BACC: 0.7715, F1: 0.7100) at a threshold of 0.315, with only a marginal reduction relative to ERM despite operating under the group-level fairness constraints. DiffEopp achieves the highest recall (0.8601) but the lowest precision (0.5680), suggesting a tendency towards positive predictions.

The base GLCS requires an unusually low threshold (0.012) to attain its best F1 score (0.7672), in stark contrast to the other methods, which operate in the range 0.281–0.315. This confirms the probability space compression effect induced by the group-level constraints, which temperature scaling effectively corrects: Calibrated GLCS restores the decision threshold to 0.315. The Matthews Correlation

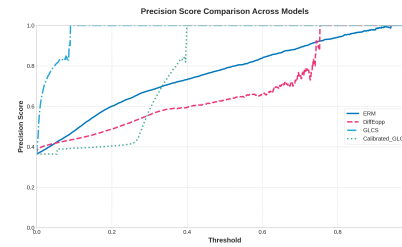
Coefficient, which is particularly informative under class imbalance, ranks ERM and Calibrated GLCS highest at 0.5341 and 0.5357, respectively, consistent with their F1 score rankings.

Method	Threshold	Bal. Acc. ↑	F1 ↑	MCC ↑	Precision ↑	Recall ↑
ERM	0.281	0.7737	0.7132	0.5341	0.6678	0.7652
DiffEopp	0.313	0.7428	0.6842	0.4698	0.5680	0.8601
GLCS	0.012	0.7672	0.7066	0.5161	0.6345	0.7972
Calibrated GLCS	0.315	0.7715	0.7100	0.5357	0.6864	0.7354

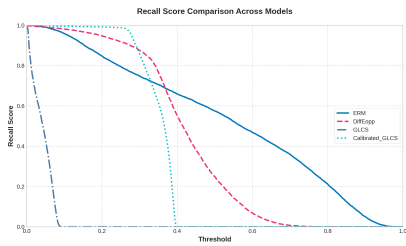
Table 9: Performance metrics on CelebA (threshold chosen for best F1).



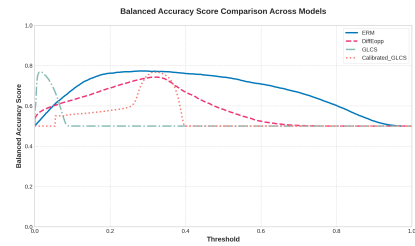
(a) F1 Score



(b) Precision Score



(c) Recall Score



(d) BACC Score

Figure 5: Performance Metrics Across Threshold Spectrum on CelebA

Performance Metrics Across Threshold Spectrum. Figures 5a and 5d illustrate the threshold sensitivity of each method. ERM maintains stable F1 and balanced accuracy across a broad threshold range (0.2–0.8), while GLCS exhibits a notably narrow operational range, attributable to probability space compression induced by the group-level fairness constraints. Temperature scaling in Calibrated GLCS restores a more standard threshold behaviour, broadening the robust performance region while preserving the fairness properties of the base GLCS. The recall–precision trade-offs across thresholds are shown in Figures 5b and 5c.

Subgroup Performance Analysis using Nuanced Metrics. Table 10 reports the nuanced subgroup metrics on the CelebA dataset. ERM achieves strong within-group discrimination but exhibits a marked asymmetry between BPSN-AUC (0.958) and BNSP-AUC (0.512), suggesting systematic cross-group bias. Dif-

fEopp reduces this asymmetry at the cost of within-group performance, particularly for females (Subgroup-AUC: 0.714). GLCS and Calibrated GLCS preserve competitive within-group performance (male: 0.795, female: 0.807) while exhibiting cross-group behaviour comparable to ERM.

Method	Subgroup	Subgroup Size	Subgroup AUC	BPSN AUC	BNSP AUC
ERM	Male	7,715	0.8050	0.9583	0.5118
	Female	12,247	0.8307	0.5118	0.9583
DiffEopp	Male	7,715	0.7976	0.8734	0.6150
	Female	12,247	0.7142	0.6150	0.8734
GLCS & Calibrated GLCS	Male	7,715	0.7954	0.9564	0.4866
	Female	12,247	0.8067	0.4867	0.9564

Table 10: Nuanced metrics on CelebA.

Empirical Analysis of Fairness Metrics: CelebA Dataset. Table 11 reports the fairness metric results for ERM, DiffEopp, GLCS, and Calibrated GLCS on the CelebA dataset. Across the majority of evaluation criteria, GLCS achieves the strongest fairness performance.

In terms of equal opportunity (eoppe), GLCS attains an error rate of 2.84%, substantially lower than DiffEopp (4.60%) and ERM (30.78%). The equalized odds metric (eodde) follows a similar pattern: GLCS records a minimal error of 4.20%, compared to Calibrated GLCS (12.80%), DiffEopp (18.07%), and ERM (47.25%). For demographic parity (dpe), GLCS achieves 2.51%, again outperforming DiffEopp (15.97%) and ERM (29.38%). The Calibrated GLCS variant attains a p-rule score (prulee) of 72.74%, compared to DiffEopp's 56.76% and ERM's 31.67%. The AUCP analysis shows that GLCS and Calibrated GLCS exhibit minimal disparity (1.13%), whereas ERM and DiffEopp record higher values of 2.57% and 8.34%, respectively.

With respect to class-level error balance, GLCS achieves the lowest disparities, with a balanced false positive rate (bfp) of 2.84% and a balanced false negative rate (bfn) of 1.36%. Calibrated GLCS follows with 5.24% bfp and 7.56% bfn. DiffEopp shows moderate imbalance (4.60% bfp, 13.47% bfn), while ERM exhibits the largest disparities (30.78% bfp, 16.47% bfn), reflecting the limitations of optimising for aggregate performance without fairness constraints.

Threshold Sensitivity Analysis of Equal Opportunity. Figure 6a illustrates the Equal Opportunity (eopp) score variations across different classification thresholds for ERM, DiffEopp, GLCS, and Calibrated GLCS models. The analysis reveals several notable patterns: ERM exhibits the highest sensitivity to threshold selection, with eopp scores peaking at approximately 0.5 around the 0.4 threshold and gradually declining toward both extremes. In contrast, both GLCS and Calibrated GLCS demonstrate remarkable stability across most threshold values, maintaining

Metric	ERM	DiffEopp	GLCS	Calibrated GLCS
p-Rule (prulee) ↑	31.67	56.76	22.68	72.74
Equal Opp.(eoppe) ↓	30.78	4.60	2.84	5.24
Equalized Odds (eodde) ↓	47.25	18.07	4.20	12.80
Demographic Parity (dpe) ↓	29.38	15.97	2.51	9.05
Balance for Positive Class (bfp) ↓	30.78	4.60	2.84	5.24
Balance for Negative Class (bfn) ↓	16.47	13.47	1.36	7.56
AUCP ↓	2.57	8.34	1.13	1.13

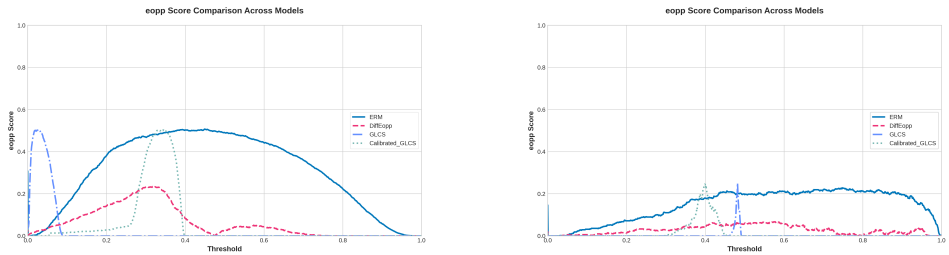
Table 11: Fairness metrics on CelebA.

consistently low eopp scores (< 0.1) except for a brief spike in GLCS at very low thresholds (< 0.1). DiffEopp shows intermediate performance with moderate threshold sensitivity, reaching a maximum eopp score of approximately 0.25 around 0.35 threshold. Notably, Calibrated GLCS exhibits a localized increase in eopp score around the 0.35 threshold region but quickly returns to stable performance. This comprehensive analysis suggests that GLCS and Calibrated GLCS provide more robust, threshold-invariant fairness guarantees than traditional ERM and DiffEopp approaches, making them more reliable choices for applications that require consistent fairness across different operating points.

Performance and Equal Opportunity Trade-Off on CelebA Dataset. Our comprehensive experimental evaluation unveils intricate trade-offs between predictive performance and fairness metrics across heterogeneous threshold configurations on the CelebA Dataset. To ensure a rigorous and fair comparative analysis, we employ threshold-agnostic metrics: AUC-PR Gain and the threshold-agnostic Equal Opportunity Difference (eoppe) metric. The proposed GLCS and Calibrated GLCS approaches demonstrate remarkable fairness characteristics, consistently exhibiting substantially lower Equal Opportunity Difference scores. Specifically, the Calibrated GLCS achieved an eoppe of 5.24, while the GLCS method realized an eoppe of 2.84, in stark contrast to the Empirical Risk Minimization (ERM) baseline (eoppe = 30.78) and the DiffEopp approach (eoppe = 4.60). Notably, these improved fairness metrics are attained without compromising predictive performance. Both GLCS variants maintained competitive AUC-PR Gain scores (0.7369), comparable to DiffEopp (0.6008) and ERM (0.7911). This empirical evidence suggests that the proposed GLCS methodologies offer a principled approach to mitigating discriminatory outcomes while preserving high-fidelity predictive precision.

5.7.2. Experimental Evaluation on UTKFace Dataset

Analysis of Threshold-Agnostic Performance Metrics. Table 12 reports the threshold-agnostic performance metrics on the UTKFace dataset. ERM again



(a) eopp Score on CelebA

(b) eopp Score on UTKFace

Figure 6: eopp Metric Across Threshold Spectrum on CelebA and UTKFace

achieves the strongest overall performance, with GLCS and Calibrated GLCS positioned between ERM and DiffEopp across all discriminative metrics. As with CelebA, the two GLCS variants share identical discriminative performance but differ in calibration, with Calibrated GLCS attaining a lower Brier Score (0.2276) than standard GLCS (0.2410).

Metric	ERM	DiffEopp	GLCS	Calibrated GLCS
ROC AUC ↑	0.9015	0.8754	0.8866	0.8866
PR AUC ↑	0.8993	0.8609	0.8752	0.8752
Brier Score ↓	0.1266	0.1459	0.2410	0.2276
AUPR Gain ↑	0.8983	0.8599	0.8741	0.8741

Table 12: Invariant performance metrics on UTKFace.

Analysis of Threshold-Dependent Performance Metrics. Table 13 reports the threshold-dependent performance metrics for all methods on the UTKFace dataset. ERM achieves the strongest overall performance, with the highest BACC (0.8069), F1 score (0.8010), and precision (0.7606) at a threshold of 0.406. Calibrated GLCS attains closely comparable results (BACC: 0.8039, F1: 0.7997) at a threshold of 0.385, with only a marginal reduction relative to ERM despite operating under the group-level fairness constraints. DiffEopp achieves the highest recall (0.8614) but lower precision (0.7248) and F1 score (0.7873), indicating a tendency towards positive predictions. GLCS achieves intermediate performance (BACC: 0.7950, F1: 0.7925) at a notably higher threshold (0.479).

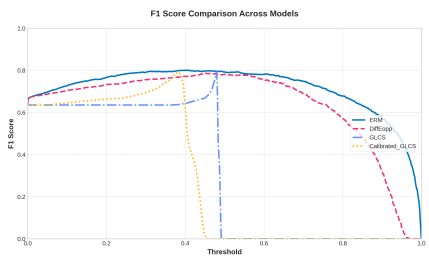
The Matthews Correlation Coefficient confirms the performance ordering, with ERM and Calibrated GLCS achieving the highest values (0.6128 and 0.6075, respectively), consistent with their F1 and BACC rankings. Taken together, these results indicate that Calibrated GLCS maintains competitive predictive performance relative to ERM while satisfying the proposed fairness constraints.

Performance Metrics Across Threshold Spectrum. Figure 7 illustrates threshold sensitivity across all methods on the UTKFace dataset. ERM and DiffEopp maintain stable F1 and balanced accuracy across a broad threshold range (0.2–0.6), while GLCS exhibits a sharp, localised performance spike near threshold 0.48,

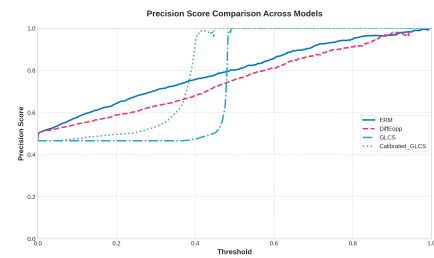
Metric	ERM	DiffEopp	GLCS	Calibrated GLCS
BACC ↑	0.8069	0.7881	0.7950	0.8039
F1 ↑	0.8010	0.7873	0.7925	0.7997
MCC ↑	0.6128	0.5782	0.5908	0.6075
Precision ↑	0.7606	0.7248	0.7364	0.7510
Recall ↑	0.8460	0.8614	0.8578	0.8551
Best Threshold	0.406	0.452	0.479	0.385

Table 13: Best-F1 thresholds for each method on UTKFace and their corresponding metrics

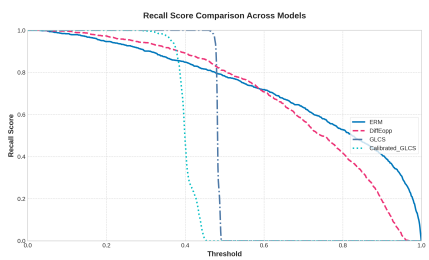
and Calibrated GLCS shows a more gradual improvement up to 0.4 followed by an abrupt drop. The recall–precision trade-offs (Figures 7b and 7c) show that both GLCS variants display step-like transitions around their respective optimal thresholds, in contrast to the more gradual transitions exhibited by ERM and DiffEopp.



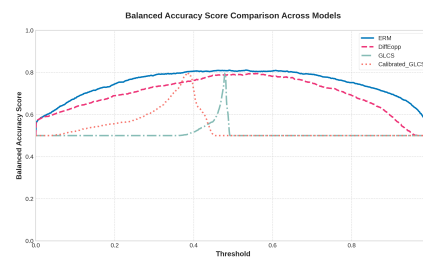
(a) F1 Score



(b) Precision Score



(c) Recall Score



(d) BACC Score

Figure 7: Performance Metrics Across Threshold Spectrum on UTKFace

Subgroup Performance Analysis using Nuanced Metrics. Table 14 reports the nuanced subgroup metrics on the UTKFace dataset. ERM achieves strong within-group discrimination (male Subgroup-AUC: 0.8927, female: 0.9015) with moderate cross-group asymmetry (BPSN-AUC: 0.9578, BNSP-AUC: 0.7946). DiffEopp reduces this asymmetry modestly at a small cost to within-group performance. GLCS and Calibrated GLCS maintain competitive Subgroup-AUC scores (male:

0.8874, female: 0.8760) with cross-group behaviour similar to ERM.

Method	Subgroup	Subgroup AUC	BPSN AUC	BNSP AUC	Size
ERM	Male	0.8927	0.9578	0.7946	1131
	Female	0.9015	0.7946	0.9578	1239
DiffEopp	Male	0.8723	0.8950	0.8446	1131
	Female	0.8717	0.8446	0.8950	1239
GLCS & Calibrated GLCS	Male	0.8874	0.9521	0.7654	1131
	Female	0.8760	0.7654	0.9521	1239

Table 14: Nuanced Metrics for each method on UTKFace Dataset

Empirical Analysis of Fairness Metrics: UTKFace Dataset. Table 15 reports the fairness metric results for all methods on the UTKFace dataset. GLCS achieves the strongest performance across most of the fairness criteria, attaining an eoppe of 0.23%, eodde of 0.65%, dpe of 0.62%, and a p-rule score of 98.69%. Calibrated GLCS follows closely, while DiffEopp, despite being specifically designed for equal opportunity, records a higher eoppe (2.16%) than GLCS. ERM exhibits the largest disparities across all metrics.

Metric	ERM	DiffEopp	GLCS	Calibrated GLCS
p-Rule (prulee) ↑	61.78	80.22	98.69	92.01
Equal Opportunity (eoppe) ↓	14.61	2.16	0.23	1.25
Equalized Odds (eodde) ↓	27.24	7.09	0.65	3.28
Demographic Parity (dpe) ↓	22.03	10.69	0.62	3.00
Balance for Positive (bfp) ↓	14.61	2.16	0.23	1.25
Balance for Negative (bfn) ↓	12.63	4.93	0.42	2.03
AUCP ↓	0.88	0.06	1.15	1.15

Table 15: Fairness metrics for each method on UTKFace.

Threshold Sensitivity Analysis of Equal Opportunity. Figure 6b demonstrates the superior fairness characteristics of GLCS-based approaches compared to alternative methods. While the ERM baseline exhibits persistent unfairness with eopp scores steadily increasing to approximately 0.2 across the 0.4-0.8 threshold range, and DiffEopp showing moderate improvement with scores around 0.05, both GLCS and Calibrated GLCS demonstrate remarkable fairness preservation across most threshold values, maintaining near-zero eopp scores throughout the majority of the threshold spectrum. The tiny elevation in eopp scores around the threshold 0.4 for these methods can be interpreted as a controlled trade-off point at which the models actively adjust their decision boundaries to maintain long-term fairness stability.

This localized behavior suggests a sophisticated fairness optimization strategy, where the models temporarily accept a minor fairness deviation to establish a robust equilibrium across the broader threshold range. Particularly noteworthy is how both GLCS variants achieve nearly perfect Equal Opportunity (eopp ≈ 0) across extensive threshold regions (0.0-0.35 and 0.45-1.0), demonstrating their ability to maintain consistent fairness guarantees without the continuous fairness drift observed in ERM and DiffEopp. This comprehensive analysis suggests that GLCS-based approaches offer superior fairness preservation by uniquely enabling stable equal opportunity metrics across diverse operating conditions.

Performance and Equal Opportunity Trade-Off on UTKFace Dataset.. Our experimental evaluation reveals nuanced trade-offs between predictive performance and fairness metrics across varying threshold configurations. To ensure a rigorous and fair comparative analysis, we use threshold-agnostic metrics: AUC-PR Gain and eoppe metric. The proposed GLCS and Calibrated GLCS demonstrate remarkable fairness characteristics, consistently exhibiting substantially lower equal opportunity difference scores (eoppe). Specifically, the Calibrated GLCS achieved an eoppe of 1.25, while the GLCS method realized an eoppe of 0.23, in stark contrast to ERM baseline (eoppe = 14.61) and the DiffEopp approach (eoppe = 2.16). Notably, these improved fairness metrics are attained without compromising predictive performance for our approach. Both GLCS variants maintained competitive AUC-PR Gain scores (0.8741), comparable to DiffEopp (0.8599) and ERM (0.8983). This empirical evidence suggests that the proposed GLCS methodologies offer a principled approach to mitigating discriminatory outcomes while preserving high-fidelity predictive precision. The results underscore the potential of GLCS methods in domains requiring stringent algorithmic fairness, particularly in high-stakes decision-making contexts where balancing performance and equitable outcomes is paramount.

5.7.3. Experimental Evaluation on CivilComments-WILDS Dataset

Analysis of Performance and Fairness Metrics. Table 16 reports the performance metrics for GLCS and ERM on the CivilComments-WILDS dataset. The two methods achieve comparable predictive performance: GLCS obtains an average precision (ap) of 74.62% and a ROC AUC of 94.53%, while ERM achieves 74.74% and 94.55% respectively, indicating a negligible performance difference.

Table 17 presents the corresponding fairness metrics. Across all criteria, GLCS attains lower disparity than ERM. In terms of equal opportunity (eoppe), GLCS records 4.91% compared to ERM's 8.14%. For equalized odds (eodde), GLCS achieves 5.83% against ERM's 8.51%. Demographic parity (dpe) is reduced from 2.43% under ERM to 0.86% under GLCS. The p-rule score improves from 79.56% for ERM to 95.85% for GLCS, indicating more balanced prediction rates across demographic groups. These results demonstrate that the fairness improvements are achieved with no meaningful sacrifice in predictive performance.

Metric	GLCS	ERM
AP	74.62	74.74
ROC AUC	94.53	94.55

Table 16: Performance Metrics Comparison on CivilComments-WILDS Dataset

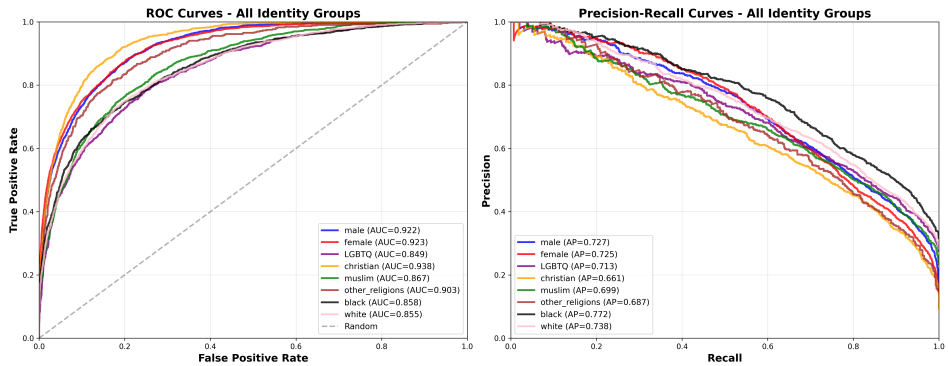


Figure 8: Per-group ROC (left) and Precision–Recall (right) curves on CivilComments-WILDS across identity groups.

Analysis of Per-group ROC and Precision–Recall curves. As shown in Fig. 8, we report per-identity-group ROC and Precision–Recall (PR) curves on the CivilComments WILDS dataset setting for our proposed method GLCS. The ROC curves for all groups lie well above the random diagonal, with AUCs ranging from 0.849 (LGBTQ) to 0.938 (Christian), indicating strong separability across groups. The PR curves (more informative under class imbalance) show Average Precision (AP) values in the range 0.661–0.772 (lowest for Christian at 0.661 and highest for black at 0.772), highlighting that some groups remain more challenging but still exhibit meaningful ranking quality.

Analysis of Group Robustness. Tables 18 and 19 present the group robustness results on the CivilComments-WILDS dataset. The ERM baseline achieves an average accuracy of 92.4%, but its worst-group accuracy is only 58.3%, corresponding to the Christian demographic group (toxic comments), indicating substantial performance disparities across groups. When the Christian group is designated as the sensitive group within the GLCS framework, a more balanced performance profile is obtained. GLCS achieves an average accuracy of 91.3% and a worst-group accuracy of 70.7%, representing an improvement of 12.4 percentage points in worst-group performance relative to ERM, at the cost of only 1.1 percentage points in average accuracy.

As shown in Table 19, GLCS also outperforms all other group robustness baselines on this dataset. DFR achieves a worst-group accuracy of 70.1%, GroupDRO 69.9%, JTT 69.3%, and AFR 68.7%, all of which fall below the 70.7% attained by GLCS, while each incurs a larger reduction in average accuracy relative to ERM.

Metric	GLCS	ERM
Demographic Parity Error (dpe) ↓	0.86	2.43
Equality of Opportunity Error (eoppe) ↓	4.91	8.14
Equalized Odds Error (eodde) ↓	5.83	8.51
p-Rule Error (prulee) ↑	95.85	79.56

Table 17: Fairness Metrics Comparison on CivilComments-WILDS Dataset

Group	# Samples		GLCS Accuracy		ERM Accuracy	
	NT	T	NT	T	NT	T
Male	12,092	2,203	0.898	0.737	0.937	0.647
Female	14,179	2,270	0.912	0.725	0.946	0.640
LGBTQ	3,210	1,216	0.784	0.745	0.880	0.620
Christian	12,101	1,260	0.935	0.707	0.962	0.583
Muslim	5,355	1,627	0.820	0.744	0.903	0.607
Other religions	2,980	520	0.882	0.746	0.935	0.623
Black	3,335	1,537	0.737	0.798	0.856	0.680
White	5,723	2,246	0.760	0.784	0.866	0.660

Table 18: Group accuracies and sample counts for GLCS and ERM methods. NT = Non-Toxic, T = Toxic.

5.8. Summary and impact

This chapter introduced the Group-Level Cost-Sensitive (GLCS) framework, a method that addresses the intersection of cost-sensitive learning, group fairness, and group robustness by incorporating group-level misclassification costs directly into the learning objective. Unlike conventional approaches, the GLCS framework modifies the loss function itself, providing a more principled mechanism for mitigating bias without sacrificing much predictive performance. Another result of this chapter is the empirical validation of a synergy between group robustness and group fairness. The result was demonstrated across multiple datasets, including facial attribute classification tasks (CelebA and UTKFace) and a large-scale text classification setting (CivilComments-WILDS), providing evidence of the possibility for framework's generalisability across different data modalities and fairness criteria.

The results indicate that fairness-aware learning need not come at a high cost of competitive predictive performance. This has practical relevance for high-stakes application domains such as healthcare, finance, and criminal justice, where both accuracy and equitable treatment across demographic groups are essential requirements.

Future work includes extending the GLCS framework to additional domains, investigating more sophisticated cost-sensitive optimisation strategies, and develop-

Method	Average Accuracy ↑	Worst-Group Accuracy ↑
ERM	0.924	0.583
DFR	0.872	0.701
Group-DRO	0.889	0.699
JTT	0.911	0.693
AFR	0.898	0.687
GLCS (Ours)	0.913	0.707

Table 19: Comparative performance of group robustness methods.

ing evaluation metrics that capture the interplay between group fairness and group robustness more comprehensively.

6. GFLC: GRAPH-BASED FAIRNESS-AWARE LABEL CORRECTION FOR FAIR CLASSIFICATION (CONTRIBUTION III)

In the previous chapters, our focus was on the first and second contributions related to the main challenge of balancing predictive utility with algorithmic fairness in classification. However, such approaches assume that the provided labels are completely accurate, which is rarely the case in practice. Wu et al. [Wu+22] noted that label noise in the training data can increase bias, and that fairness-aware learning algorithms that enforce fairness constraints might become even more biased than unconstrained ones. Therefore, in this chapter, we introduce the third contribution [SR25], titled 'GFLC: Graph-based Fairness-aware Label Correction for Fair Classification'.

6.1. Introduction

In many applications, labels are obtained through imperfect processes (e.g., crowdsourcing, heuristics, or weak supervision). When label noise exists, it can distort both model training and fairness evaluation. Specifically, fairness constraints may be optimized with respect to corrupted labels, leading to unexpected behaviors in debiasing procedures, reduced predictive performance, or even increased group disparities. This problem is especially severe when the noise is "structured," such as when label corruption rates vary across sensitive groups or classes, creating misleading patterns that a model can exploit.

In the literature on learning with noisy labels, it is conventional to distinguish between several taxonomic categories of label noise [AU21]. To make these distinctions precise, let x denote the instance features, s the sensitive group membership, and y the true class label. The simplest case is that of random noise, in which the corruption process is entirely stochastic and bears no systematic relationship to either the instance features or the true label y. A more structured form is Y-dependent noise, where the probability of label corruption is conditioned on the true class y but remains independent of the instance features x and s. The most general category, XY-dependent noise, arises when the noise mechanism depends jointly on both the instance features (x and s) and the true class label y.

A particularly consequential special case emerges when the noise process is jointly conditioned on a sensitive attribute s and its associated class label y [Sil+24]. This configuration is variously described in the fairness literature as *group-dependent* noise [WLL21] or *instance-dependent* noise [Wu+22], and it is precisely this setting that constitutes the central concern of the present chapter. Under such conditions, the probability that a given instance carries an incorrect label is modulated simultaneously by the class y to which it belongs and the sensitive group s with which it is associated, a structure that is inherently discriminatory in

character. As will be argued throughout this chapter, this form of label noise is of especial relevance when deploying bias mitigation techniques, since the noise itself is entangled with the very group-level disparities that such methods seek to redress.

Lamy et al. [Lam+19], Liu and Wang [LW21], Wang et al. [WLL21], and Fogliato et al. [FCG20] have shown that enforcing parity constraints on noisy labels can negatively affect classifier accuracy for groups that are not affected by label noise. This shows that label noise adversely affects the performance of machine learning models. Moreover, label noise significantly affects fairness metrics and can make fairness-aware algorithms more biased than those not designed with fairness in mind [WLL21].

Publication III addresses this problem by proposing GFLC (Graph-based Fairness-aware Label Correction), a method that aims to correct noisy labels while accounting for fairness considerations. The central idea is to build a k-Nearest Neighbors (kNN) graph over the feature space, use curvature-based geometric signals to identify unreliable local relations, and leverage these signals to guide label correction in a way that improves both utility and group fairness. By explicitly incorporating fairness and a margin term into the label-correction process, GFLC targets a failure mode that standard debiasing methods that assume clean labels do not address.

The remainder of the chapter is organized as follows. Section 6.2 introduces the preliminaries and notation needed for GFLC, including the kNN graph construction, Forman–Ricci curvature, and the discrete Ricci flow update. Section 6.3 presents GFLC method. Section 6.4 describes the experimental setup and datasets, and the subsequent sections report results and discussion.

The work presented in this chapter draws on *Publication III* [SR25], of which the author of this thesis is the primary author. Certain portions of the text, figures, and tables have been reproduced or adapted from that publication.

6.2. Preliminaries and Notation

This section collects the mathematical notions needed for our proposed method 'GFLC' and explains how they fit together in the label-correction pipeline. We start in Section 6.2.1 (*Noisy Labels*) by clarifying what 'noisy' means in our setting and how it manifests in practice: labels are deemed *suspicious* when they are inconsistent with the local structure of the data and/or induce systematic group-level disparities (tracked by the fairness metric *DP* in our controlled noise protocol). We then present kNN graph (Section 6.2.2) to capture neighborhood relations in the feature space, providing a concrete substrate for measuring such inconsistencies. Next, we introduce Forman–Ricci curvature (Section 6.2.3) as an edge-level notion that quantifies local cohesiveness and irregularity in this graph, and we describe the discrete Ricci flow update (Section 6.2.4) that uses curvature to iteratively reweight edges, strengthening coherent (positively curved) connections while attenuating

unstable (negatively curved) ones. Finally, in Section 6.2.5 we tie these notions back to the main goal of GFLC by explaining how curvature-driven smoothing improves the reliability of the graph signals used to rank candidates and perform label correction.

6.2.1. Noisy Labels

Label noise is a common phenomenon in both the learning-with-noisy-labels literature and real-world applications. In general, determining whether labels are noisy depends on (i) the type of noise one aims to detect (e.g., random vs. structured noise), and (ii) the availability of appropriate diagnostics or metrics that can reveal that type of corruption.

In the experimental setting used to develop and evaluate GFLC (Section 6.3), the label noise process is *controlled by design*: noise is injected in a group-dependent manner, i.e., the flip probability is specified as a function of sensitive group membership and class. Consequently, within our current evaluation protocol, the question of whether the noise is group-dependent, class-dependent, or random is largely answered *a priori*, since the corruption mechanism is known and intentionally biased to emulate fairness-related label corruption. This is the setting that GFLC (and the baseline Fair-OBNC method, Section 6.4.3) aims to mitigate.

In our experiments, the disparity induced by this injection is tracked and diagnosed using the demographic-parity metric *DP* (Eq. 6.12), which we treat as the main criterion for whether the targeted notion of fairness is achieved (consistent with the baseline Fair-OBNC).

6.2.2. k-NN Graph Construction

To capture the underlying geometric structure of the data, a k-Nearest Neighbour (kNN) graph is constructed from the feature representations. Formally, let $G = (V, E, W)$ denote an undirected graph with vertex set V, edge set E, and a positive weight $w_{uv} \in W$ associated with each edge $e_{uv} = \{u, v\} \in E$. The notation $x \sim u$ is used throughout to indicate that vertex x is a neighbour of u, that is, $\{u, x\} \in E$. In a kNN graph, each node corresponds to a data point and is connected to its k nearest neighbours according to a chosen distance metric.

Given the dataset $\mathscr{D} = \{\mathbf{x}_i, \tilde{y}_i, s_i\}_{i=1}^{N}$, an undirected k-nearest-neighbour graph $G = (V, E)$ is constructed over the feature space with the aim of approximating the data manifold. Edge weights are assigned according to an inverse distance weighting scheme, so that pairs of nodes lying close together in feature space receive larger weights, reflecting their greater similarity, while pairs that are farther apart receive correspondingly smaller weights. Concretely, the weight associated with the edge between nodes i and j is defined as:

$$w_{ij} = \frac{1}{\max(d(\mathbf{x}_i, \mathbf{x}_j), \varepsilon)} \tag{6.1}$$

where $d(\mathbf{x}_i, \mathbf{x}_j)$ denotes the Euclidean distance (L_2 norm) between the feature vectors of nodes i and j, and $\varepsilon = 10^{-8}$ is a small stabilising constant introduced to prevent division by zero.

6.2.3. Forman–Ricci Curvature and Its Simplified Form

A key geometric quantity employed in this chapter is the Forman–Ricci curvature, originally introduced by Sreejith et al. [Sre+16] as a discrete analogue of Ricci curvature tailored to undirected networks. Within this combinatorial framework, a scalar curvature value $F(e_{uv})$ is assigned to each edge e_{uv}, reflecting the local connectivity structure in the neighbourhood of the endpoints u and v. Intuitively, the curvature of an edge measures its strength relative to the dispersion of the surrounding neighbourhood. A notable feature of this formulation is that it incorporates both node and edge weights naturally, making it applicable to the analysis of both unweighted and weighted complex networks alike.

The general Forman–Ricci curvature of an edge e_{uv} is given by [Sre+16]:

$$F(e_{uv}) = w_{uv}\left(\frac{w_u}{w_{uv}} + \frac{w_v}{w_{uv}} - \sum_{x\sim u}\frac{w_u}{\sqrt{w_{uv}\,w_{ux}}} - \sum_{x\sim v}\frac{w_v}{\sqrt{w_{uv}\,w_{vx}}}\right) \tag{6.2}$$

where the sums range over all neighbours x of u and v respectively, excluding v from the sum over u and vice versa. This expression is derived from Forman's general discretization of Ricci curvature on CW-complexes [For03], specialised to the graph setting [Sre+16; WSJ17].

For the purposes of this chapter, however, a simplified variant of the above formulation is adopted. The derivation of this simplified form from the general formula in Equation 6.2 is provided in Appendix A. More precisely, for an edge $e_{uv} = \{u, v\} \in E$, the discrete Forman–Ricci curvature $F(e_{uv})^{(t)}$ at iteration t is defined as:

$$F(e_{uv})^{(t)} = w_{uv}^{(t)}\left(1 - \frac{1}{2}\left(\sum_{x\in\mathcal{N}(u)\setminus\{v\}}\sqrt{\frac{w_{uv}^{(t)}}{w_{ux}^{(t)}}} + \sum_{x\in\mathcal{N}(v)\setminus\{u\}}\sqrt{\frac{w_{uv}^{(t)}}{w_{vx}^{(t)}}}\right)\right) \tag{6.3}$$

where $w_{uv}^{(t)}$ denotes the weight of edge (u, v) at iteration t, the ratio $w_{uv}^{(t)}/w_{ux}^{(t)}$ compares the current edge weight to that of a neighbouring edge incident to u, and $w_{uv}^{(t)}/w_{vx}^{(t)}$ performs the analogous comparison from the perspective of v. To make the structure of this expression more transparent, one may define the auxiliary quantities:

$$\mathrm{sum}_u = \underbrace{\sum_{x\sim u}\sqrt{\frac{w_{uv}^{(t)}}{w_{ux}^{(t)}}}}_{\text{neighbours of } u} \qquad \text{and} \qquad \mathrm{sum}_v = \underbrace{\sum_{x\sim v}\sqrt{\frac{w_{uv}^{(t)}}{w_{vx}^{(t)}}}}_{\text{neighbours of } v} \tag{6.4}$$

so that the curvature may be written compactly as:

$$F(e_{uv})^{(t)} = w_{uv}^{(t)} \cdot \left(1 - \frac{\text{sum}_u + \text{sum}_v}{2}\right) = w_{uv}^{(t)} \cdot (1 - A)$$

The sign of the curvature admits a natural structural interpretation [Ni+19; Top+21]. When $A < 1$, the curvature is positive ($F(e_{uv})^{(t)} > 0$), indicating that the edge (u, v) is relatively strong in comparison to its local neighbourhood and that the surrounding connectivity is healthy. Conversely, when $A > 1$, the curvature is negative ($F(e_{uv})^{(t)} < 0$), signalling that the edge is comparatively weak and may represent a structural bottleneck or a critical bridge in a sparsely connected region of the graph. The boundary case $A = 1$ yields zero curvature ($F(e_{uv})^{(t)} = 0$), which corresponds to a state of balanced local connectivity.

6.2.4. Ricci Flow Update Rule

The Ricci flow, in Hamilton's original continuous formulation [Ham82], evolves the metric tensor g_{ij} of a manifold according to its Ricci curvature:

$$\frac{\partial g_{ij}}{\partial t} = -2\,\text{Ric}_{ij} \tag{6.5}$$

The negative sign encodes the smoothing behaviour of the flow: regions of positive curvature undergo metric contraction, while regions of negative curvature expand, driving the geometry toward uniformity. Translating this idea to the discrete graph setting, Ni et al. [Ni+19] introduced a discrete Ricci flow in which all edge weights are updated simultaneously at each iteration. In this chapter, the following update rule is adopted:

$$w_{ij}^{(t+1)} = \max\left(w_{ij}^{(t)} + \eta \cdot F(e_{uv})^{(t)},\ \varepsilon\right) \tag{6.6}$$

where $\eta = 0.1$ is the flow step size, and $\varepsilon = 10^{-8}$ is a small stabilising constant that ensures edge weights remain strictly positive throughout the iteration.

The geometric interpretation of this update rule follows directly from the sign of the curvature. Edges with positive curvature ($F(e_{uv})^{(t)} > 0$) receive increased weights, strengthening intra-community connections and reinforcing well-connected regions of the graph. Conversely, edges with negative curvature ($F(e_{uv})^{(t)} < 0$) have their weights reduced, which widens the separation between distinct network modules and prunes weak connections in sparse, tree-like regions. Together, these two effects amplify the modularity structure of the graph in a manner that mirrors the curvature-driven smoothing of the continuous flow. This iterative weight refinement plays a central role in the proposed GFLC method, as it enhances the graph Laplacian (see Section 6.3.1) used to score and rank data points that may carry noisy labels.

6.2.5. How Curvature Supports Label-Noise Correction

In GFLC, the Forman–Ricci curvature is computed on the constructed tabular kNN graph and is used to promote a smoother (more regular) graph geometry. This has a denoising effect: it reduces the influence of local irregularities in the connectivity/edge weights, which can arise from noisy samples or mislabeled/outlier points. Furthermore, by incorporating curvature into our method, potential label noise can be detected through inconsistency signals on the kNN graph (e.g., local neighborhood disagreement and weak support), which encourages the use of a graph-based geometric regularizer.

More generally, applying a discrete Ricci flow updates edge weights to improve the graph's geometric properties and connectivity patterns (see Section 6.2.4, *Ricci Flow Update Rule*). The update rule in Eq. (6.6) follows the intuition that *positively curved* edges contract (their effective weight increases), whereas *negatively curved* edges expand (their effective weight decreases). As a result, the flow strengthens coherent, well-supported relations and attenuates unstable connections that are more likely to be induced by noise. Thus, by considering our main intuition for the role of Forman–Ricci curvature, the curvature is computed on the constructed tabular kNN graph to quantify local geometric (cohesiveness vs. irregularity) of edges and neighborhoods. By applying a discrete Ricci flow update, the method strengthens well-supported (positively curved) connections and attenuates unstable (negatively curved) ones, thereby denoising the graph structure and improving the reliability of the subsequent label-correction step.

6.3. GFLC

Having established the foundations in the preceding sections, this section introduces *Graph-based Fairness-aware Label Correction* GFLC, the method proposed in *Publication III* [SR25], which present a unified label correction framework designed to operate under instance-dependent, group-conditioned label noise. In particular, GFLC integrates prediction confidence, geometric structure via graph Laplacian and Ricci flow, and fairness via demographic parity for fairness-aware label noise correction. GFLC takes as input a labeled dataset with features $X = \{x_i\}_{i=1}^{n}$, (possibly noisy) labels $y = \{y_i\} \in \{0,1\}$ and a sensitive attribute $s = \{s_i\} \in \{0,1\}$ indicating group membership. It outputs corrected labels $\tilde{y}$ that aim to improve fairness across groups while respecting the geometric structure of the data and the classifier's confidence.

GFLC proceeds as follows. First, train a classifier on (X, y) to obtain class probabilities $p_i = \Pr(y_i = 1 \mid x_i)$, which are used to compute the margin term. *Second,* build a kNN graph (Section 6.2.2). *Third,* compute discrete Forman–Ricci curvature on the edges and apply a Ricci-flow update to adjust the graph structure. *Fourth,* compute the fairness term. *Finally,* compute a *Combined Correction Score* (score_i) that balances the margin term, the graph Laplacian regularization term, and

a fairness incentive to determine which labels should be corrected. The combined correction score for each data point x_i is computed as:

$$\text{score}_i = \underbrace{\alpha(1-M_i)}_{\text{Margin term}} + \underbrace{\beta L_i}_{\text{Graph Laplacian}} + \underbrace{\gamma \Delta\text{DP}_i}_{\text{Fairness incentive}} \tag{6.7}$$

The resulting multi-component scoring function ensures high-quality corrections for each data point by scoring the instances using $score_i$ in descending order, reflecting the most uncertain instances via the margin term, the weight via the Graph Laplacian for instance x_i, and the highest fairness gain. After that, we begin by categorizing the scores (score_i) into two classes: S^+ for positive instances and S^- for negative instances. Following this, we select the top K^+ positive candidates and the top K^- negative candidates, as described in Section 6.3.4. Finally, we implement the flips, updating the labels to $y_i' = 1 - y_i$ for all i in the combined set of K^+ and K^-. In the upcoming sections, we will provide a detailed explanation of each term in Equation (6.7).

6.3.1. Graph Laplacian

Having defined the graph structure and refined its geometry through the Ricci flow update rule, it is now necessary to establish a mechanism for quantifying how consistent each node's label is with respect to its local neighbourhood. The graph Laplacian serves precisely this purpose: it provides a node-level disagreement score that, as will be seen, becomes a core component of the combined correction score used in GFLC.

The graph Laplacian is a fundamental tool for enforcing smoothness constraints in semi-supervised learning and is widely used in manifold regularization frameworks. Its conceptual foundation is elegant: we construct a graph in which nodes correspond to data points and edges reflect pairwise similarities. Once established, the Laplacian can be used as a regularizer that discourages sharp label variations between neighboring nodes. However, in our GFLC approach, we consider the Laplacian term (L_i) [SG23] over the k-NN graph structure, which is defined as:

$$L_i = \sum_{j \in \mathcal{N}(i)} w_{ij} \cdot (y_i - y_j)^2 \tag{6.8}$$

where $\mathcal{N}(i)$ represents the set of k-nearest neighbors of instance x_i, while w_{ij} denotes the edge weight between nodes i and j. The term $(y_i - y_j)^2$ penalizes label disagreement between connected nodes based on their labels. The quantity L_i calculates the weighted sum of squared differences between the label of node i and those of its neighbors. This serves as a measure of the disagreement between the label of node i and its neighbors, with the weights reflecting the strength of the connections (edge weights). Additionally, the weights w_{ij} in Equation (6.8) are updated according to the update rule defined in Equation (6.6).

Laplacian Term Analysis. Higher L_i indicates higher disagreement with neighbors and higher correction priority. In particular, $L_i = 0$ when $y_i = y_j$ for all neighbors j, which means perfect local consistency, and L_i is maximized when instance x_i has different labels from all its neighbors. Moreover, under the graph smoothness assumption, nearby instances should have similar labels; a high L_i suggests that y_i violates local structure and may be incorrect, and correcting such instances improves overall graph consistency.

Majority and Minority Group Nodes. In graph-based learning, *majority group nodes* refer to densely connected clusters where nodes share similar characteristics (e.g., a predominant class in classification tasks). Conversely, *minority group nodes* represent sparsely connected regions with distinctive properties (e.g., under-represented classes). The majority group nodes typically have higher degree connections within their group (intra-group edges). Conversely, minority nodes often exhibit more connections *across* groups (inter-group edges) (i.e. more edges to neighbors whose values y_j differ). Hence, L_i will typically be larger for those minority nodes than for the majority nodes. In other words, we are considering the following situation: (i) Majority nodes usually sit in a tight cluster where most neighbors j also belong to the same class (or have very similar y_j). That means $(y_i - y_j)^2$ is small on almost every intra-group edge, so their total $\sum w_{ij}(y_i - y_j)^2$ stays relatively low. (ii) Minority nodes, by contrast, often have fewer same-group neighbors and relatively more edges "across" to the majority. Along each of those inter-group edges, y_i and y_j are more likely to differ, so $(y_i - y_j)^2$ is larger. Even if the weights w_{ij} are the same size, having more of those "large-difference" terms in the sum drives L_i up. Therefore, nodes with higher L_i get penalized more heavily. A minority node's tendency to connect across groups thus "costs" it more in the Laplacian penalty than a majority node that mostly connects inside its own (homogeneous) cluster.

Next, we explain the following cases for the inter-group edges and their connection to our GFLC approach:

Inter-group edges. The weight reduction after updating on inter-group edges ($y_i \neq y_j$) when Forman curvature $F(e_{ij}) < 0$, acts as an *adaptive smoothing filter*. Thus, this helps for noise robustness by making spurious connections between groups (e.g., mislabeled nodes) experience weight decay.

Strong inter-group edges. Suppose there exists at least one neighbor j for which $y_i \neq y_j$ but $w_{ij}$is very large. Since $(y_i - y_j)^2 = 1$ whenever y_i and y_j differ, the term $w_{ij}(y_i - y_j)^2$ becomes large due to the high weight w_{ij}. Consequently, that single edge contributes a large value to the sum, driving L_i to be high. In other words, a large-weight edge prefers $y_i = y_j$, so observing $y_i \neq y_j$ despite w_{ij} being large is a strong indicator that y_i itself is likely erroneous. Therefore, in our GFLC approach, high L_i is a signal that node i is a candidate for having a mislabeled y_i.

6.3.2. Margin Term

While the graph Laplacian quantifies label inconsistency from a geometric perspective, it does not directly account for the classifier's own confidence in a given prediction. A label that is geometrically inconsistent with its neighbourhood is a strong candidate for correction, but so is one for which the model itself is highly uncertain. The margin term introduced here captures precisely this complementary signal, and together with the Laplacian term it forms the basis of the combined correction score in GFLC.

The margin term $(1 - M_i)$ converts prediction confidence into a correction priority score, ensuring that uncertain predictions, where label noise is most likely to occur, are prioritised for correction. Concretely, an ensemble model (e.g., Random Forest) is trained to obtain initial predicted probabilities, and the class threshold is subsequently optimised via ROC analysis. For each node i, the quantity M_i is defined in terms of its predicted probability $p_i \in [0,1]$ and a threshold interval $[\tau^-, \tau^+]$ that demarcates the ambiguous region. Specifically, M_i measures how far p_i lies outside this central ambiguous region, with a large positive value indicating a confident prediction that may override a potentially noisy label.

$$\tau^+ = \underset{\tau}{\operatorname{argmax}} \left(\mathrm{TPR}(\tau) - 5 \cdot \mathrm{FPR}(\tau)\right) \qquad \textit{and} \qquad \tau^- = 1 - \tau^+ \tag{6.9}$$

$$M_i = \begin{cases} p_i - \tau^+ & p_i \geq \tau^+ \\ \tau^- - p_i & p_i \leq \tau^- \\ 0 & \text{otherwise} \end{cases} \tag{6.10}$$

Higher margin term $(1 - M_i)$ values are better for correction purposes because: larger $(\text{margin term})_i$ indicates the model is less confident about sample i (high uncertainty), and less confident samples are more likely to have noisy/incorrect labels. Therefore, they should be prioritized for potential label correction.

6.3.3. Fairness Term

The graph Laplacian and margin terms established in the preceding subsections capture, respectively, geometric inconsistency and predictive uncertainty, two signals that together indicate which labels are likely to be noisy. However, neither term encodes any notion of group-level equity: a correction that reduces label noise may still leave, or even widen, disparities across sensitive groups. To address this, a fairness term is introduced that explicitly quantifies the impact of each candidate label flip on demographic parity, ensuring that the correction process is guided not only by what is geometrically or statistically suspect, but also by what is fair.

Fairness Term (DP_i*).* The fairness term measures the impact of flipping label y_i on demographic parity [Dwo+12] across sensitive groups and is defined as follows:

$$\Delta \mathrm{DP}_i \;=\; \mathrm{DP}_{\mathrm{new}}^{(i)} - \mathrm{DP}_{\mathrm{original}} \tag{6.11}$$

Both $\mathrm{DP}_{\mathrm{new}}^{(i)}$ and $\mathrm{DP}_{\mathrm{original}}$ measure demographic parity (DP) and are computed as the ratio between the minimum and maximum positive rates based on the sensitive group [Sil+24]. $\mathrm{DP}_{\mathrm{original}}$ is the current demographic parity ratio and is independent of i. Moreover, $\mathrm{DP}_{\mathrm{new}}^{(i)}$ is the hypothetical demographic parity ratio after flipping y_i. Both ratios range from 0 to 1, where 1 indicates perfect demographic parity. Mathematically, we can define Demographic Parity (DP) as follows:

$$\mathrm{DP} = \frac{\min\limits_{s}\big(\Pr(\hat{y}=1 \mid S=s)\big)}{\max\limits_{s}\big(\Pr(\hat{y}=1 \mid S=s)\big)} \tag{6.12}$$

Why We Use a Signed Fairness Change for Label Correction. Using the signed change in Equation (6.11) is advantageous for fair label-correction selection because it preserves *direction*: (i) $\Delta DP_i > 0$ indicates that flipping y_i improves demographic parity, (ii) $\Delta DP_i < 0$ indicates that it degrades it, and (iii) $\Delta \mathrm{DP}_i = 0$ means that flipping y_i has no impact on demographic parity. In contrast, using $|\Delta DP_i|$ would discard the sign and could assign the same score to flips that help and flips that harm fairness, which is undesirable when selecting corrections in our case.

6.3.4. Determining Number of Instances to Flip

The previous scoring function ensures that among all possible flips, we select the *most appropriate* candidates. This addresses the critical question: *'How many instances should be flipped?'*. In order to determine the total number of positive instances to flip to negative (top K^-) and the total number of negative instances to flip to positive (top K^+), we consider using some parts of Fair-OBNC [Sil+24] to achieve this as follows:

First, we define the overall positive rate as $P(y) = \frac{1}{n}\sum_{i=1}^{n} y_i$. Moreover, we define the minimum and maximum prevalence values as $P_{\mathrm{lower}} = P(y)\cdot(1-D)$ and $P_{\mathrm{upper}} = P(y)\cdot(1+D)$, where $D \in [0,1]$ is the disparity tolerance parameter.

Second, for each sensitive group $s \in \mathscr{S}$ we define $n_s = |\{i : s_i = s\}|$ as group size and $P(y|s) = \frac{1}{n_s}\sum_{i:s_i=s} y_i$ as group positive rate.

Third, we use the following flip determination logic [Sil+24]:

$$\text{For each group } (s \in \{0,1\}): \begin{cases} \text{If } P(y|s) < P_{\mathrm{lower}}: & \text{Flip negatives to positives} \\ \text{If } P(y|s) > P_{\mathrm{upper}}: & \text{Flip positives to negatives} \\ \text{Otherwise}: & \text{No flips needed} \end{cases} \tag{6.13}$$

Finally, we are computing the required flips K^+ and K^- as follows:

$$K^{+} = \sum_{s:P(y|s)<P_{\text{lower}}} \left\lceil n_s \cdot (P_{\text{lower}} - P(y|s)) \right\rceil \tag{6.14}$$

$$K^{-} = \sum_{s:P(y|s)>P_{\text{upper}}} \left\lceil n_s \cdot (P(y|s) - P_{\text{upper}}) \right\rceil \tag{6.15}$$

6.4. Experimental setup

This section describes the experimental configuration used to evaluate whether GFLC can learn a robust and fair classifier under conditions of label noise.

6.4.1. Dataset

The experiments are conducted on the bank account fraud dataset [Jes+22], specifically Variant II of the suite, which is designed for binary classification with the objective of detecting fraud in bank account opening applications. The applicant's age is adopted as the sensitive attribute, encoded as a binary variable: applicants aged 50 and older constitute the first sensitive group ($s = A$), while those under 50 form the second group ($s = B$). The dataset configuration follows that of [Sil+24]. In total, the dataset comprises one million application instances, each described by 30 features [Jes+22]. The class distribution is heavily imbalanced: the positive class (fraud) accounts for approximately 1.15% of instances, with the remaining 98.85% belonging to the negative class (legitimate applications).

Label noise injection. To evaluate the effectiveness of our GFLC method compared to the baseline across various biased data scenarios, we inject label noise into unbiased versions of the dataset as done in the works Silva et al. [Sil+23] and Silvaa et al. [Sil+24]. The process outlined in the work Silvaa et al. [Sil+24] begins by generating an independent and identically distributed (IID) dataset on the sensitive attribute and the data splits. First, the sensitive attribute column is shuffled to eliminate any existing relationships between this column, the labels, and the features. Next, the instances are randomly shuffled into training, validation, and test sets to prevent potential data drift in the original splits. The subsequent injection of label noise is based on both the label and the sensitive group to which each instance belongs.

Label Noise. In our experiments, we focus exclusively on the scenario of applying label noise equally to both labels. This differs from the approach taken in Silvaa et al. [Sil+24], which considered three scenarios: applying label noise equally to both labels, applying noise to samples with positive labels, and applying noise to samples with negative labels. Nevertheless, the study by Silvaa et al. [Sil+24] found that when only adding noise to the positive label instances, the performance of the models showed little variation across different methods and noise rates, and this is likely due to the small number of positive cases (and, therefore, mislabeled instances) in the dataset. Additionally, the results for the

other two scenarios—injecting noise into both labels equally and applying it only to samples with negative labels—were very similar. As a result, similar conclusions can be drawn when analyzing the effects of injecting noise solely into the negative label and across both labels [Sil+24]. Furthermore, as noted in Silvaa et al. [Sil+24], we examine three different noise rates: 5%, 10%, and 20%.

Sensitive group. In the context of sensitive groups, as discussed in Silvaa et al. [Sil+24], we apply noise uniformly to instances in one specific group (Group A only), simulating a scenario where the sensitive attribute influences the likelihood of receiving an incorrect label.

IID Test Set. In the domain of learning with noisy labels, it is essential to evaluate the trained models on a clean, IID test set after obtaining corrected labels for the noisy training dataset and training on this corrected dataset. This evaluation helps us understand how the models perform when the biases present in the training data are mitigated in the resulting training dataset [Sil+24].

6.4.2. Experimental Settings in GFLC

In our GFLC method, we are using the following parameters in all experimental settings. For the K-NN graph (Section 6.2.2), we are using $k = 10$. Moreover, for Equation (6.7) we are using $\alpha = 0.2$, $\beta = 0.6$ and $\gamma = 0.2$. In Equation (6.6), the weights are updated for 2 iterations ($ricci\_iter = 2$). Furthermore, we select $D = 0.05$ (Section 6.3.4) as the target of disparity. Finally, after obtaining the modified/corrected training data set using our proposed GFLC method (Section 6.3), we train a LightGBM model configured with 100 estimators and a learning rate of 0.1 using the corrected training data set. The final trained model will be used to make predictions for the IID test set (Section 6.4.1).

6.4.3. Baseline: Fair-OBNC

We use Fair Ordering-Based Noise Correction (Fair-OBNC) [Sil+24] as a baseline. It is a fairness-aware label noise correction method, extending the Ordering-Based Label Noise Correction (OBNC) algorithm [FB15] with fairness considerations. This method enhances demographic parity in training datasets, making it a good baseline for the intersection of fairness and noise correction. For Fair-OBNC, we follow the same experimental settings outlined in Silvaa et al. [Sil+24], where 50 hyperparameter configurations are randomly sampled and applied to the training set. However, unlike Silvaa et al. [Sil+24], which used a fixed decision threshold, we use the same procedure across decision thresholds ranging from 0 to 0.6 in our experiments. Furthermore, after obtaining the modified training sets, LightGBM models are trained on them [Sil+24]. These models then make predictions on the IID clean test set, and we finally average the performance and fairness metrics across the 50 runs. Additionally, the implementation code for Fair-OBNC is available in the Aequitas package[1] under the name LabelFlipping.

[1] `https://github.com/dssg/aequitas`

6.4.4. Performance & Fairness Metrics

The metrics reported in this chapter are those not already defined in earlier chapters, selected to reflect the specific objectives of the bank account fraud detection task, where the primary goal is to identify as many fraudulent applications as possible [Sil+24]. Since both GFLC and the Fair-OBNC baseline [Sil+24] (Sections 6.3.3 and 6.4.3) target *Demographic Parity* as the fairness criterion during label correction, this metric serves as the primary measure of group fairness in the evaluation. *Equal Opportunity* and *Equalized Odds* are additionally reported to provide a broader assessment of fairness across groups, even though they were not the central objective of the correction process.

Definition 6.4.1. True Positive Rate (TPR), which is also known as sensitivity or recall, is a key performance metric in binary classification. It measures the proportion of actual positive cases that the model correctly identifies. Mathematically, TPR is defined as TPR = TP/(TP + FN). In this formula, TP represents true positives, and FN represents false negatives. Essentially, TPR answers the question: out of all the actual positive cases in the dataset, how many did the model successfully identify?

Definition 6.4.2. True Negative Rate (TNR), also known as specificity, measures the proportion of actual negative cases that are correctly identified as negative by the model. Mathematically expressed as TNR = TN / (TN + FP), where TN represents true negatives, and FP represents false positives. Therefore, the TNR addresses the question: Of all the actual negative cases in the dataset, how many did the model correctly classify as negative?

Definition 6.4.3. False Positive Rate (FPR) measures the proportion of actual negative cases incorrectly identified as positive by a model. FPR can be calculated as the ratio of false positives (FP) to the sum of false positives and true negatives (TN), expressed as FPR=FP/(FP+TN).

Definition 6.4.4. False Negative Rate (FNR) is the proportion of actual positives erroneously classified as negatives. It is computed as the ratio of false negatives (FN) to the sum of false negatives and true positives (TP), given by FNR=FN/(FN+TP).

Definition 6.4.5. To compute the Demographic Parity DP metric, we calculate the ratio between the lowest and highest predicted prevalence (Equation 6.12). In this metric, values close to 1 indicate an equilibrium in predicted prevalence across all groups in the dataset, while values close to 0 indicate greater disparities in the fairness metric. In our experiments, this metric is called *pprev_ratio* (predicted prevalence), same as it is named in the Aequitas toolkit [Sal+18; Jes+24; Sil+24].

6.5. Results

The following subsections present the results of GFLC against the Fair-OBNC baseline. Each approach yields a trained model fitted on the training set with its

corresponding corrected labels, and all models are subsequently evaluated on the same IID test set.

6.5.1. Evaluation of performance

Table 20 reports the AUC scores of GFLC and Fair-OBNC under varying noise levels. At 5% noise, GFLC achieves an AUC of 0.874 versus 0.813 for Fair-OBNC. When the noise rate rises to 10%, GFLC maintains its lead with an AUC of 0.843 compared to Fair-OBNC's 0.783. Even at 20% noise, GFLC outperforms Fair-OBNC, recording 0.799 against 0.752.

Table 20: Performance comparison across noise rates

Metric	Noise Rate 5%		Noise Rate 10%		Noise Rate 20%	
	GFLC	Fair-OBNC	GFLC	Fair-OBNC	GFLC	Fair-OBNC
AUC ↑	0.874	0.813	0.843	0.783	0.799	0.752

6.5.2. Results at Noise Rate 5%

The experimental results presented in Figures (9, 10) demonstrate a comprehensive comparison between GFLC (red, dotted) and Fair-OBNC (blue, dashed) methods across various fairness and performance metrics at a 5% label-noise rate. The analysis reveals distinct behavioral patterns and trade-offs between the two approaches across different threshold values in terms of fairness and performance.

True-Positive Rate (TPR). As the threshold increases (Figure 9a), Fair-OBNC's TPR initially stays slightly higher, down to ≈ 0.50 at threshold 0.1 versus GFLC's ≈ 0.45. However, after that, it drops off faster. Beyond ≈ 0.12 threshold, GFLC retains a higher TPR, for instance ≈ 0.3 TPR at threshold 0.2 compared to Fair-OBNC's ≈ 0.21.

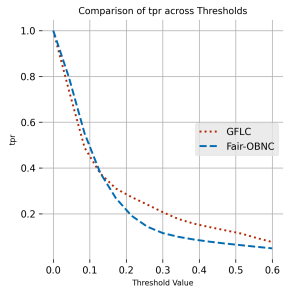
False-Positive Rate (FPR). At very low thresholds (near 0), Figure 9e, both methods produce nearly 100% false-positives (i.e. everyone is predicted positive). As the threshold increases, GFLC's FPR drops more sharply: by a threshold of ≈ 0.1, it's already down near 5%, while Fair-OBNC is still around 10%. Beyond ≈ 0.2, both methods drive FPR very close to zero. This indicates that GFLC results in fewer false alarms for every threshold choice while maintaining a comparable True Positive Rate (TPR) when compared to Fair-OBNC.

Precision. Low thresholds (< 0.1), Figure 9c, lead to extremely low precision for both models (because nearly everyone is called positive). In the mid-range (0.1–0.3), GFLC and Fair-OBNC track closely until about 0.2, where they both reach $\approx$0.20 precision. At higher thresholds (> 0.2), GFLC pulls ahead—reaching up to $\approx$0.36 precision at threshold 0.55 versus around 0.24 for Fair-OBNC.

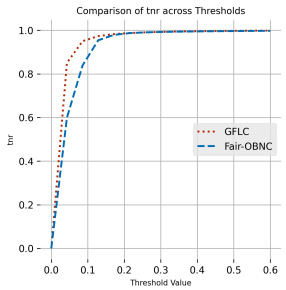
False Negative Rate (FNR). Figure 9d presents the false negative rate (FNR) across thresholds, where both methods show similar trends but with notable differences in magnitude. Both GFLC and Fair-OBNC demonstrate increasing FNR

with higher thresholds. The convergence occurs around threshold 0.15, after which Fair-OBNC maintains higher FNR values.

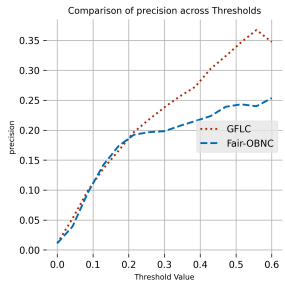
True Negative Rate (TNR). Figure 9b demonstrates true negative rate (TNR) performance, where both methods achieve high performance levels above 0.95 for thresholds greater than 0.2. At lower thresholds (< 0.2), GFLC shows slightly faster convergence to optimal TNR values.



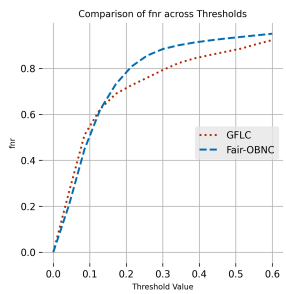
(a) TPR



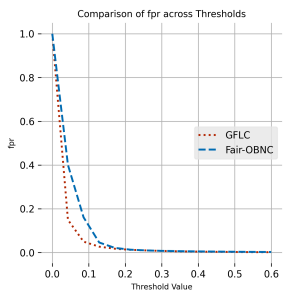
(b) TNR



(c) Precision



(d) FNR

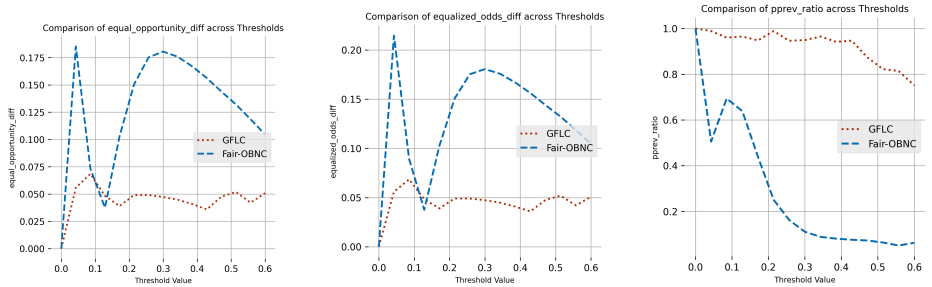


(e) FPR

Figure 9: Performance Metrics Comparison Across Different Thresholds at 5% Noise Rate

Demographic Parity. The demographic parity ratio results in Figure 10c reveal the most dramatic performance differences between the methods, where higher values indicate better demographic parity. GFLC maintains exceptional demographic parity with ratios consistently near 1.0 across all threshold values, indicating optimal demographic balance across groups. Conversely, Fair-OBNC shows substantial degradation in demographic parity with increasing thresholds, dropping from approximately 1.0 at threshold 0.0 to roughly 0.05 at threshold 0.6. This represents a severe failure in maintaining demographic parity, suggesting that Fair-OBNC's focus on other fairness metrics comes at an unacceptable cost to demographic balance.

Equal Opportunity. Figure 10a illustrates the equal opportunity difference across varying threshold values, where lower values indicate better fairness. GFLC demonstrates superior and more consistent fairness with equal opportunity dif-



(a) Equal Opportunity ↓ (b) Equalized odds ↓ (c) Demographic Parity ↑

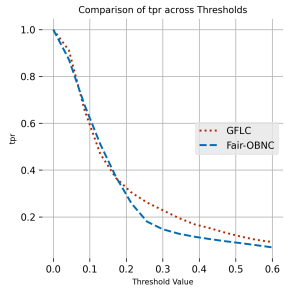
Figure 10: Fairness Metrics Comparison Across Different Thresholds at 5% Noise Rate

ferences maintained around 0.05 across all thresholds, indicating stable fairness performance. In contrast, Fair-OBNC exhibits significantly worse fairness performance, with peaks reaching approximately 0.18 at threshold values around 0.05 and 0.3. Moreover, Fair-OBNC shows good Equal Opportunity at approximately 0.04 at threshold 0.11. Furthermore, Fair-OBNC shows a problematic bimodal distribution with substantial fairness degradation at low thresholds (0.05) and moderate thresholds (0.3), suggesting threshold-dependent instability that compromises equal opportunity fairness.

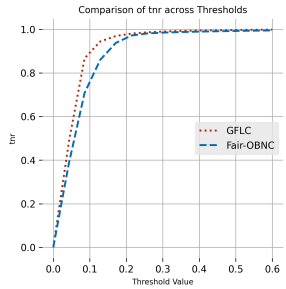
Equalized Odds. The equalized odds difference, shown in Figure 10b, follows a similar pattern to the Equal Opportunity metric. GFLC again demonstrates superior fairness performance with equalized odds differences remaining consistently low around 0.05, showing minimal variation and maintaining stable fairness across all threshold values. Fair-OBNC exhibits poor fairness performance with peaks reaching 0.21 at very low thresholds and 0.175 at moderate thresholds around 0.3. The method shows dramatic fairness deterioration from threshold 0.0 to 0.05, followed by some recovery (approximately at threshold 0.11) but continued instability, indicating unreliable equalized odds performance.

6.5.3. Results at Noise Rate 10%

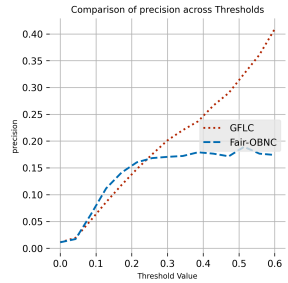
At a 10% label noise rate, all plots in Figures (11 and 12) compare the behavior of GFLC (in red, dotted line) and Fair-OBNC (in blue, dashed line) as we vary the decision threshold from 0 to 0.6. Generally, these figures show results similar to those observed at a 5% noise rate. In this context, we will skip detailing the results for the 10% label noise rate and instead focus on those for the 20% label noise rate, as they exhibit more interesting behavior and highlight the advantages of our GFLC method at higher noise rates.



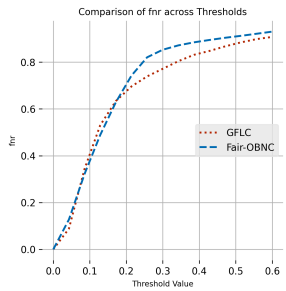
(a) TPR



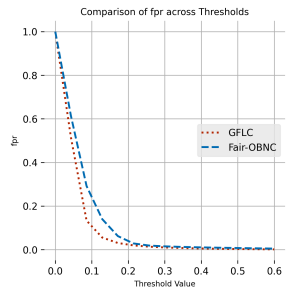
(b) TNR



(c) Precision

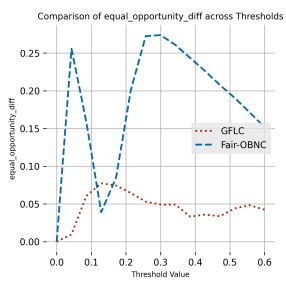


(d) FNR

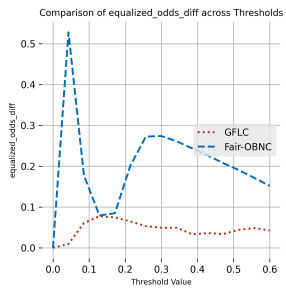


(e) FPR

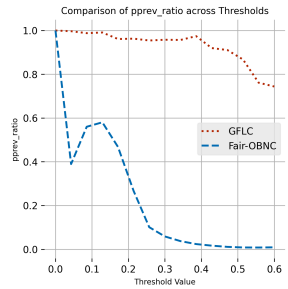
Figure 11: Performance Metrics Comparison Across Different Thresholds at 10% Noise Rate



(a) Equal Opportunity ↓



(b) Equalized odds ↓



(c) Demographic Parity ↑

Figure 12: Fairness Metrics Comparison Across Different Thresholds at 10% Noise Rate

6.5.4. Results at Noise Rate 20%

The plots in (Figure 13 and Figure 14) compare how GFLC (red, dotted) and Fair-OBNC (blue, dashed) behave as we change the decision threshold from 0 up to 0.6, at a 20% label noise rate. The following analysis at a 20% label noise rate reveals the strength of our GFLC approach compared to the baseline method.

True-Positive Rate (TPR). GFLC's TPR, shown in Figure 13a, shows slightly higher values than Fair-OBNC's TPR as the threshold increases from 0 to 0.15. After the threshold of 0.15, Fair-OBNC's TPR remains slightly higher. This is the opposite of the results observed at 5% or 10% label noise, where GFLC's True Positive Rate (TPR) was very similar to Fair-OBNC's TPR as the threshold increased from 0 to 0.15. Furthermore, after the threshold exceeded 0.15, GFLC's TPR at a noise rate of 20% exhibited behavior that was contrary to its TPR at the earlier noise rates of 5% and 10%, and was the same as the Fair-OBNC.

False-Positive Rate (FPR). As the threshold increases, shown in Figure 13e, from 0 to 0.08, GFLC's FPR is higher than Fair-OBNC's. Between 0.08 and 0.11, both methods exhibit comparable behavior. However, above the 0.11 threshold, Fair-OBNC's FPR is higher than GFLC's.

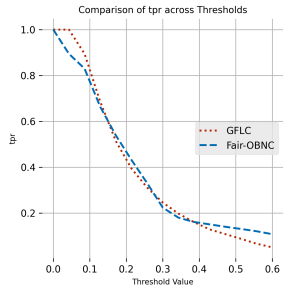
Precision. For low thresholds (less than 0.1) the Figure 13c shows very low precision for both models again. In the mid-range (0.1–0.3), Fair-OBNC shows better precision. However, beyond thresholds of 0.3, GFLC demonstrates superior precision compared to Fair-OBNC.

False Negative Rate (FNR). Figure 13d presents the false negative rate (FNR) across thresholds, where both methods show similar trends but with notable differences in magnitude. Both GFLC and Fair-OBNC demonstrate increasing FNR with higher thresholds. The convergence occurs around threshold 0.15, after which Fair-OBNC maintains higher FNR values.

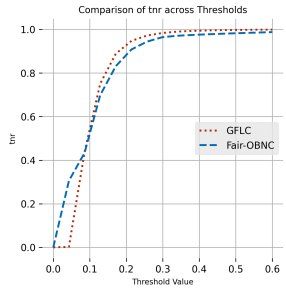
True Negative Rate (TNR). Figure 13b demonstrates true negative rate (TNR) performance, where both methods achieve high performance levels above 0.95 for thresholds greater than 0.2. At lower thresholds (< 0.2), GFLC shows slightly faster convergence to optimal TNR values.

Demographic Parity. The demographic parity ratio shown in Figure 14c indicates that the GFLC method maintains exceptional demographic parity, with ratios consistently close to 1.0 across all threshold values. However, the Fair-OBNC method shows significant degradation in the demographic parity metric as the threshold increases, decreasing from approximately 1.0 at a threshold of 0.0 to around 0 at a threshold of 0.6. This illustrates the advantages of the GFLC method, particularly at the high noise rates of 20 % compared to Fair-OBNC.

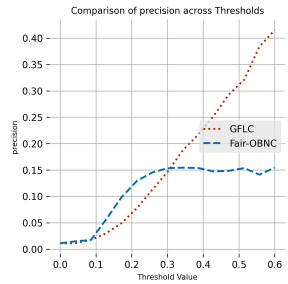
Equal Opportunity and Equalized Odds. Figures 14a and 14b show that our GFLC method outperforms the baseline method Fair-OBNC across all threshold values at this high noise rate of 20% for equal opportunity and equalized odds. This highlights the advantages of our GFLC method for different fairness definitions in high-noise scenarios.



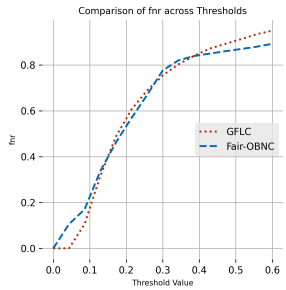
(a) TPR



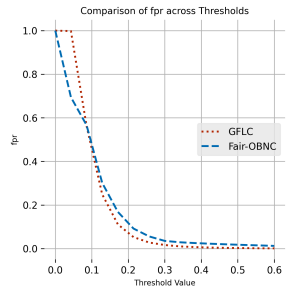
(b) TNR



(c) Precision



(d) FNR

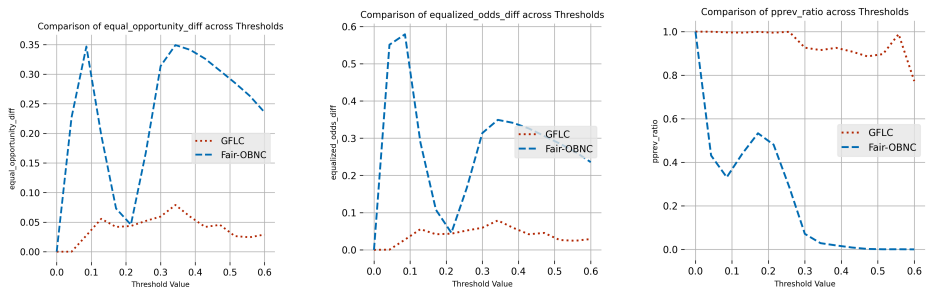


(e) FPR

Figure 13: Performance Metrics Comparison Across Different Thresholds at 20% Noise Rate

6.6. Conclusion

In this chapter, we introduced GFLC, a novel approach for learning with instance-dependent noisy labels. Our method is designed to correct labels while ensuring fairness across demographic groups. GFLC combines graph-based techniques, fairness-aware learning, and a prediction confidence measure. It effectively leverages the structural insights from k-nearest neighbor (k-NN) graphs, discrete Forman–Ricci curvature, and discrete Ricci flow. Moreover, we demonstrated the effectiveness of our approach using a real dataset, showing that it improves the trade-off between discriminative performance and model fairness. This work builds on key aspects of the literature on instance-dependent noisy labels and highlights important fairness issues. Our work contributes to the development of safer machine learning models for use in high-impact societal contexts.



(a) Equal Opportunity ↓ (b) Equalized odds ↓ (c) Demographic Parity ↑

Figure 14: Fairness Metrics Comparison Across Different Thresholds at 20% Noise Rate

7. CONCLUSION

This thesis examined the fairness-performance trade-off in supervised learning through three ways: model architecture, data composition, and label quality. Across these approaches, the guiding objective was consistent: to reduce group disparities without sacrificing too much of the core objective (performance) of machine learning models. To that end, we introduced three contributions.

Contribution I: TFS

First, we asked which architectural modifications can improve the balance between fairness and performance. We answered by proposing *The Fairness Stitch* 'TFS' [SR24]. TFS is an approach that can be considered as an in-processing debiasing architectural framework. TFS demonstrated that effective debiasing often requires interventions beyond the last-layer fine-tuning. We demonstrate in *The Fairness Stitch* framework that stitching intermediate representations within a single pre-trained neural network model can reduce group disparities while maintaining task performance. Crucially, it shows the limitations of treating the classifier head as the sole locus of debiasing and solving the problem of overfitting for fairness. When sensitive correlations percolate through earlier layers, fine-tuning the last layer alone is often insufficient.

Risks and assumptions. Our evaluation of TFS follows the same balanced-data protocol used for the last-layer fairness fine-tuning baseline [Mao+23], i.e., constructing a sub-dataset that is balanced with respect to both the class label and the sensitive attribute. This design choice helps isolate the fairness–performance behaviour of architectural interventions under a controlled comparison setting; however, it also introduces an assumption that such balancing is feasible and representative of the deployment distribution. Consequently, a practical limitation is that the observed trade-offs may shift under the original (unbalanced) data distribution or under alternative balancing strategies.

A second assumption is the current placement choice of the fairness stitch, which was inserted directly before the final classification layer. While this placement is well-motivated by the goal of challenging the 'last-layer-only' claim and by the natural block structure of the backbone, the current study does not yet provide a systematic analysis of placement sensitivity. We explicitly acknowledge as a limitation that stitch placement should be more accurately guided. As a follow-up, we plan a micro-ablation that evaluates multiple alternative stitch positions on the same datasets (and extends to additional modalities such as text, images, and audio), and we will complement performance/fairness reporting with information-theoretic analyses that quantify how intermediate representations retain information about inputs, labels, and sensitive attributes.

Finally, because some fairness deltas can be modest but consistent, we treat reliability as a first-class evaluation dimension in planned extensions: future work

will include multi-seed experiments and uncertainty reporting (e.g., confidence intervals) to ensure that improvements are not run-specific.

Contribution II: GLCS

Second, we asked how to mitigate group-level class imbalance to enhance fairness without undue accuracy loss. GLCS [SMR25] answers this by translating imbalance into principled, group-aware costs within a constrained optimization that targets equal opportunity for the primary goal 'fairness-performance trade-off' and for the problem of group robustness at the same time.

Risks and assumptions. The GLCS framework introduces a small set of margin and penalty hyperparameters, namely δ and μ_1, μ_2, μ_3, which control the strength and behavior of the constrained optimization toward Equal Opportunity. A limitation of the current evaluation is that, although the reported configuration performs consistently in our experiments, we do not yet provide a systematic multi-dataset sensitivity slice over these parameters. We explicitly treat *hyperparameter stability across multiple datasets* as a prioritized future-work item: we plan to run a small grid sweep (e.g., 3-5 representative configurations per dataset) to verify that Equal Opportunity and utility remain stable over a practical range and that gains are not tied to a single configuration–dataset pairing.

In addition, GLCS outcomes (as with most classification approaches) can be influenced by the choice of decision threshold at deployment time. To avoid overstating threshold-specific behavior, we emphasize that GLCS should be evaluated and selected using threshold-robust criteria (e.g., threshold-agnostic Equal Opportunity differences and ranking-based utility measures), and that threshold choice should be aligned with application requirements. In future extensions, we will further expand sensitivity reporting to include threshold-related stability alongside hyperparameter stability to strengthen practitioner guidance. For the CivilComments-WILDS group-robustness experiment, we used a fixed decision threshold of 0.5 for GLCS and all baselines and reported per-group ROC/PR curves to ensure that comparisons are not driven by threshold re-selection but reflect ranking quality.

Contribution III: GFLC

Third, we asked how machine learning models can learn under biased or noisy supervision. For that, we proposed GFLC [SR25], a graph-based, fairness-aware label correction method to mitigate instance-dependent label noise under the demographic parity objective. GFLC approaches label noise as a structured phenomenon through a graph to propose fairness-aware corrections. The method recognizes a simple but consequential reality: fairness constraints can backfire when labels are systematically corrupted. Correcting supervision with explicit

sensitivity to demographic structure can provide trade-offs under demographic parity.

Risks and assumptions. GFLC relies on three practical assumptions that bound the scope of its conclusions. First, GFLC uses the *initial classifier's predicted probabilities* inside both the margin component and the ΔDP fairness incentive; therefore, if the initial model is overconfident or miscalibrated (potentially in a group-dependent manner), the probability magnitudes can influence which instances are selected for flipping and may risk amplifying teacher/model bias. We explicitly note the absence of a calibration analysis as a current limitation, and we plan a prioritized follow-up work that reports ECE/reliability diagrams and per-group Brier score and NLL (negative log-likelihood), and evaluates a temperature-scaling ablation comparing flip sets and downstream fairness/utility *with vs. without* calibration.

Second, the experimental setting assumes *group-dependent noise* by construction, since the noise is injected following the same protocol as the Fair-OBNC baseline. This assumption is aligned with the bias mechanism GFLC is designed to mitigate and explains why the ΔDP term is necessary to achieve the fairness objective in our development experiments; removing ΔDP (Graph-only) improved performance-related metrics but did not achieve the targeted fairness improvement. However, this also limits external generality to settings where group dependence is present. For real-world datasets where the noise mechanism is unknown, we plan a weak-gold diagnostic to quantify per-group error asymmetries, uncertainty patterns, and graph neighbor disagreement, and to gate the use of ΔDP accordingly, including an explicit Graph-only vs. full GFLC ablation.

Third, GFLC assumes that a tabular kNN graph provides a meaningful neighborhood structure and that applying curvature-based smoothing improves robustness by mitigating irregularities/noise in the graph. This choice is methodologically motivated, but it introduces sensitivity to the graph construction (e.g., feature representation and k), and it may be less effective when nearest-neighbor relations are weak or noisy. As future work, we will strengthen external validity with broader baselines and a small natural-noise check beyond the controlled injection setting.

Overall implications, limitations, and future work.

Taken together, the three contributions support a unified implication: improving fairness in supervised learning is most effective when interventions are applied at the stage where the bias is introduced or reinforced in the *representation/architecture* (TFS), the *data and optimization objective* (GLCS), or the *labels/supervision* (GFLC). Across these settings, a consistent lesson emerges that fairness is not a purely post hoc adjustment; rather, it is a pipeline-level design choice that benefits from targeted mechanisms, careful evaluation, and explicit assumptions. At the same time, the thesis has three corresponding limitations that motivate clear future work: (i) TFS requires more systematic guidance for stitch placement and broader

reliability reporting beyond the current controlled setting; (ii) GLCS would benefit from multi-dataset hyperparameter and threshold sensitivity slices to strengthen practitioner guidance and confirm robustness beyond a single configuration; and (iii) GFLC's external validity should be strengthened through calibration/reliability checks for the initial classifier, a weak-gold noise diagnostic when the noise mechanism is unknown, and broader baselines including a small natural-noise check. Addressing these items will improve portability across modalities and architectures, reduce run and threshold-specific variability, and provide stronger evidence that fairness gains remain stable under realistic deployment conditions.

To conclude, fairness in machine learning is often presented as a tension: intervene aggressively and risk degrading performance, or protect performance and risk entrenching disparities. This thesis proposes three techniques to deal with this problem. There is no universal recipe, as different applications will prioritize different risks, metrics, and constraints. Yet a consistent pattern emerges: interventions that respect the structure of the learning pipeline, features, data, and labels, tend to deliver improvements that are both measurable and durable. In this sense, the thesis contributes not only methods but also a stance. Fairness is not a single constraint, a post hoc adjustment, or a line in a compliance checklist. It is a design property that requires engineering, validation, maintenance, and possibly more. The societal stakes are real. Decisions informed by algorithms have a significant impact on access to credit, healthcare, education, employment, and safety. Responsible systems must not only optimize predictive loss but also account for who incurs that loss, under what conditions, and with what consequences. If the techniques developed here help practitioners make those trade-offs with greater clarity and control, then they have served their purpose.

BIBLIOGRAPHY

[Mao+23] Yuzhen Mao, Zhun Deng, Huaxiu Yao, Ting Ye, Kenji Kawaguchi, and James Zou. "Last-Layer Fairness Fine-tuning is Simple and Effective for Neural Networks". In: *arXiv preprint arXiv:2304.03935* (2023).

[SR24] **Modar Sulaiman** and Kallol Roy. "The Fairness Stitch: A Novel Approach for Neural Network Debiasing". In: *Acta Informatica Pragensia* 13.3 (Aug. 2024), pp. 359–373. ISSN: 1805-4951. DOI: 10.18267/j.aip.241. URL: http://dx.doi.org/10.18267/j.aip.241.

[SMR25] **Modar Sulaiman**, Nesma Talaat Abbas Mahmoud, and Kallol Roy. "Advancing Equal Opportunity Fairness and Group Robustness through Group-Level Cost-Sensitive Deep Learning". In: *Baltic Journal of Modern Computing* 13.1 (2025). ISSN: 2255-8950. DOI: 10.22364/bjmc.2025.13.1.06. URL: https://doi.org/10.22364/bjmc.2025.13.1.06.

[SR25] **Modar Sulaiman** and Kallol Roy. "GFLC: Graph-based Fairness-aware Label Correction for Fair Classification". In: *Baltic Journal of Modern Computing* 13.3 (2025). ISSN: 2255-8950. DOI: 10.22364/bjmc.2025.13.3.02. URL: https://doi.org/10.22364/bjmc.2025.13.3.02.

[Ali+22] Abdulalem Ali, Shukor Abd Razak, Siti Hajar Othman, Taiseer Abdalla Elfadil Eisa, Arafat Al-Dhaqm, Maged Nasser, Tusneem Elhassan, Hashim Elshafie, and Abdu Saif. "Financial fraud detection based on machine learning: a systematic literature review". In: *Applied Sciences* 12.19 (2022), p. 9637.

[Ste+21] Harald Steck, Linas Baltrunas, Ehtsham Elahi, Dawen Liang, Yves Raimond, and Justin Basilico. "Deep learning for recommender systems: A Netflix case study". In: *AI magazine* 42.3 (2021), pp. 7–18.

[Han+24a] Ryan Han, Julián N Acosta, Zahra Shakeri, John PA Ioannidis, Eric J Topol, and Pranav Rajpurkar. "Randomised controlled trials evaluating artificial intelligence in clinical practice: a scoping review". In: *The lancet digital health* 6.5 (2024), e367–e373.

[BS16] Solon Barocas and Andrew D Selbst. "Big data's disparate impact". In: *Calif. L. Rev.* 104 (2016), p. 671.

[Kus+17] Matt J Kusner, Joshua Loftus, Chris Russell, and Ricardo Silva. "Counterfactual fairness". In: *Advances in neural information processing systems* 30 (2017).

[Sey+21] Laleh Seyyed-Kalantari, Haoran Zhang, Matthew BA McDermott, Irene Y Chen, and Marzyeh Ghassemi. "Underdiagnosis bias of

artificial intelligence algorithms applied to chest radiographs in under-served patient populations". In: *Nature medicine* 27.12 (2021), pp. 2176–2182.

[Bol+16] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. "Man is to computer programmer as woman is to homemaker? debiasing word embeddings". In: *Advances in neural information processing systems* 29 (2016).

[Jen+19] Sophie Jentzsch, Patrick Schramowski, Constantin Rothkopf, and Kristian Kersting. "Semantics derived automatically from language corpora contain human-like moral choices". In: *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 2019, pp. 37–44.

[WM02] David H Wolpert and William G Macready. "No free lunch theorems for optimization". In: *IEEE transactions on evolutionary computation* 1.1 (2002), pp. 67–82.

[Mit80] Tom M Mitchell. "The need for biases in learning generalizations". In: (1980).

[Olt+19] Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kıcıman. "Social data: Biases, methodological pitfalls, and ethical boundaries". In: *Frontiers in big data* 2 (2019), p. 13.

[Fer24] Emilio Ferrara. "Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies". In: *Sci* 6.1 (2024), p. 3.

[Žli17] Indrė Žliobaitė. "Measuring discrimination in algorithmic decision making". In: *Data Mining and Knowledge Discovery* 31.4 (2017), pp. 1060–1089.

[Fje+20] Jessica Fjeld, Nele Achten, Hannah Hilligoss, Adam Nagy, and Madhulika Srikumar. "Principled artificial intelligence: Mapping consensus in ethical and rights-based approaches to principles for AI". In: *Berkman Klein Center Research Publication* 2020-1 (2020).

[ZG22] Han Zhao and Geoffrey J Gordon. "Inherent tradeoffs in learning fair representations". In: *Journal of Machine Learning Research* 23.57 (2022), pp. 1–26.

[XYZ23] Ruicheng Xian, Lang Yin, and Han Zhao. "Fair and optimal classification via post-processing". In: *International conference on machine learning*. PMLR. 2023, pp. 37977–38012.

[SG21] Harini Suresh and John Guttag. "A Framework for Understanding Sources of Harm throughout the Machine Learning Life Cycle". In: *Equity and Access in Algorithms, Mechanisms, and Optimization*. EAAMO '21. ACM, Oct. 2021, pp. 1–9. DOI: 10.1145/3465416.3483305. URL: http://dx.doi.org/10.1145/3465416.3483305.

[Dri21] Salim Dridi. “Supervised learning-a systematic literature review”. In: *preprint, Dec* (2021).

[JGR20] Tammy Jiang, Jaimie L Gradus, and Anthony J Rosellini. “Supervised machine learning: a brief primer”. In: *Behavior therapy* 51.5 (2020), pp. 675–687.

[SHG19] Pratap Chandra Sen, Mahimarnab Hajra, and Mitadru Ghosh. “Supervised classification algorithms in machine learning: A survey and review”. In: *Emerging technology in modelling and graphics: Proceedings of IEM graph 2018*. Springer, 2019, pp. 99–111.

[Kad19] Ammar Ismael Kadhim. “Survey on supervised machine learning techniques for automatic text classification”. In: *Artificial intelligence review* 52.1 (2019), pp. 273–292.

[MY15] Iqbal Muhammad and Zhu Yan. “SUPERVISED MACHINE LEARNING APPROACHES: a SURVEY”. In: *ICTACT Journal on Soft Computing* 05.03 (Apr. 2015), pp. 946–952. DOI: 10.21917/ijsc.2015.0133. URL: https://doi.org/10.21917/ijsc.2015.0133.

[Sal94] Steven L Salzberg. “Book Review: C4. 5: Programs for machine learning”. In: *Machine Learning* 16.3 (1994), p. 235.

[KZP+07] Sotiris B Kotsiantis, Ioannis Zaharakis, P Pintelas, et al. “Supervised machine learning: A review of classification techniques”. In: *Emerging artificial intelligence applications in computer engineering* 160.1 (2007), pp. 3–24.

[MRT18] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

[BBL03] Olivier Bousquet, Stéphane Boucheron, and Gábor Lugosi. “Introduction to statistical learning theory”. In: *Summer school on machine learning*. Springer, 2003, pp. 169–207.

[MCM86] Tom M Mitchell, Jaime G Carbonell, and Ryszard S Michalski. *Machine learning: a guide to current research*. Vol. 12. Springer Science & Business Media, 1986.

[Sar21] Iqbal H Sarker. “Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions”. In: *SN computer science* 2.6 (2021), pp. 1–20.

[Cal+17] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. “Optimized preprocessing for discrimination prevention”. In: *Advances in neural information processing systems* 30 (2017).

[ZLM18] 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*. 2018, pp. 335–340.

[La 23] William G La Cava. “Optimizing fairness tradeoffs in machine learning with multiobjective meta-models”. In: *Proceedings of the Genetic and Evolutionary Computation Conference*. 2023, pp. 511–519.

[Deb+00] Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and Tanaka Meyarivan. “A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II”. In: *International conference on parallel problem solving from nature*. Springer. 2000, pp. 849–858.

[Kam+18] Faisal Kamiran, Sameen Mansha, Asim Karim, and Xiangliang Zhang. “Exploiting reject option in classification for social discrimination control”. In: *Information Sciences* 425 (2018), pp. 18–33.

[She+24] Yi Sheng, Junhuan Yang, Jinyang Li, James Alaina, Xiaowei Xu, Yiyu Shi, Jingtong Hu, Weiwen Jiang, and Lei Yang. “Data-Algorithm-Architecture Co-Optimization for Fair Neural Networks on Skin Lesion Dataset”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2024, pp. 153–163.

[ST17] Ravid Shwartz-Ziv and Naftali Tishby. *Opening the Black Box of Deep Neural Networks via Information*. 2017. DOI: 10.48550/ARXIV.1703.00810. URL: https://arxiv.org/abs/1703.00810.

[TPB99] Naftali Tishby, Fernando C. Pereira, and William Bialek. “The Information Bottleneck Method”. In: *Proceedings of the 37th Annual Allerton Conference on Communication, Control and Computing*. 1999, pp. 368–377. URL: https://arxiv.org/abs/physics/0004057.

[BNB21] Yamini Bansal, Preetum Nakkiran, and Boaz Barak. “Revisiting model stitching to compare neural representations”. In: *Advances in neural information processing systems* 34 (2021), pp. 225–236.

[LV15] Karel Lenc and Andrea Vedaldi. “Understanding image representations by measuring their equivariance and equivalence”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 991–999.

[KIW22] Polina Kirichenko, Pavel Izmailov, and Andrew Gordon Wilson. “Last layer re-training is sufficient for robustness to spurious correlations”. In: *arXiv preprint arXiv:2204.02937* (2022).

[Lee+22] Yoonho Lee, Annie S Chen, Fahim Tajwar, Ananya Kumar, Huaxiu Yao, Percy Liang, and Chelsea Finn. “Surgical fine-tuning improves adaptation to distribution shifts”. In: *arXiv preprint arXiv:2210.11466* (2022).

[Kum+22] Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. “Fine-tuning can distort pretrained features and under-

perform out-of-distribution". In: *arXiv preprint arXiv:2202.10054* (2022).

[Che+21] Valeriia Cherepanova, Vedant Nanda, Micah Goldblum, John P Dickerson, and Tom Goldstein. "Technical challenges for training fair neural networks". In: *arXiv preprint arXiv:2102.06764* (2021).

[Wan+23] Mingyang Wan, Daochen Zha, Ninghao Liu, and Na Zou. "In-processing modeling techniques for machine learning fairness: A survey". In: *ACM Transactions on Knowledge Discovery from Data* 17.3 (2023), pp. 1–27.

[KAS11] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. "Fairness-aware learning through regularization approach". In: *2011 IEEE 11th International Conference on Data Mining Workshops*. IEEE. 2011, pp. 643–650.

[Jia+20] Ray Jiang, Aldo Pacchiano, Tom Stepleton, Heinrich Jiang, and Silvia Chiappa. "Wasserstein fair classification". In: *Uncertainty in artificial intelligence*. PMLR. 2020, pp. 862–872.

[Beu+19a] 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*. 2019, pp. 453–459.

[Beu+19b] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H Chi, et al. "Fairness in recommendation ranking through pairwise comparisons". In: *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2019, pp. 2212–2220.

[Zaf+19] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P Gummadi. "Fairness constraints: A flexible approach for fair classification". In: *The Journal of Machine Learning Research* 20.1 (2019), pp. 2737–2778.

[Zaf+17] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. "Fairness constraints: Mechanisms for fair classification". In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 962–970.

[Dwo+12] 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*. 2012, pp. 214–226.

[Has+18] Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. "Fairness without demographics in repeated loss minimization". In: *International Conference on Machine Learning*. PMLR. 2018, pp. 1929–1938.

[Kea+18] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. "Preventing fairness gerrymandering: Auditing and learning for subgroup fairness". In: *International conference on machine learning*. PMLR. 2018, pp. 2564–2572.

[KC12] Faisal Kamiran and Toon Calders. "Data preprocessing techniques for classification without discrimination". In: *Knowledge and information systems* 33.1 (2012), pp. 1–33.

[Ple+17] Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. "On fairness and calibration". In: *Advances in neural information processing systems* 30 (2017).

[CKP09] Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. "Building classifiers with independency constraints". In: *2009 IEEE international conference on data mining workshops*. IEEE. 2009, pp. 13–18.

[Zem+13] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. "Learning fair representations". In: *International conference on machine learning*. PMLR. 2013, pp. 325–333.

[HPS16] Moritz Hardt, Eric Price, and Nati Srebro. "Equality of opportunity in supervised learning". In: *Advances in neural information processing systems* 29 (2016).

[Liu+15] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Deep learning face attributes in the wild". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 3730–3738.

[Faw04] Tom Fawcett. "ROC graphs: Notes and practical considerations for researchers". In: *Machine learning* 31.1 (2004), pp. 1–38.

[PG20] 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.

[GBB19] Josh Gardner, Christopher Brooks, and Ryan Baker. "Evaluating the fairness of predictive student models through slicing analysis". In: *Proceedings of the 9th international conference on learning analytics & knowledge*. 2019, pp. 225–234.

[Bro+10] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M Buhmann. "The balanced accuracy and its posterior distribution". In: *2010 20th international conference on pattern recognition*. IEEE. 2010, pp. 3121–3124.

[Lah+20] Preethi Lahoti, Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, and Ed Chi. "Fairness without demographics through adversarially reweighted learning". In: *Advances in neural information processing systems* 33 (2020), pp. 728–740.

[Raw01] John Rawls. *Justice as fairness: A restatement*. Harvard University Press, 2001.

[ZSQ17] Zhifei Zhang, Yang Song, and Hairong Qi. "Age progression/regression by conditional adversarial autoencoder". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 5810–5818.

[Par+20] Sungho Park, Dohyung Kim, Sunhee Hwang, and Hyeran Byun. "Readme: Representation learning by fairness-aware disentangling method". In: *arXiv preprint arXiv:2007.03775* (2020).

[GVS14] Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. "Qualitatively characterizing neural network optimization problems". In: *arXiv preprint arXiv:1412.6544* (2014).

[Par+25] Otavio Parraga, Martin D More, Christian M Oliveira, Nathan S Gavenski, Lucas S Kupssinskü, Adilson Medronha, Luis V Moura, Gabriel S Simões, and Rodrigo C Barros. "Fairness in Deep Learning: A survey on vision and language research". In: *ACM Computing Surveys* 57.6 (2025), pp. 1–40.

[Meh+21] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. "A survey on bias and fairness in machine learning". In: *ACM computing surveys (CSUR)* 54.6 (2021), pp. 1–35.

[Du+20] Mengnan Du, Fan Yang, Na Zou, and Xia Hu. "Fairness in deep learning: A computational perspective". In: *IEEE Intelligent Systems* 36.4 (2020), pp. 25–34.

[CH24] Simon Caton and Christian Haas. "Fairness in machine learning: A survey". In: *ACM Computing Surveys* 56.7 (2024), pp. 1–38.

[KMR16] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. "Inherent trade-offs in the fair determination of risk scores". In: *arXiv preprint arXiv:1609.05807* (2016).

[Sel+19] Andrew D Selbst, Danah Boyd, Sorelle A Friedler, Suresh Venkatasubramanian, and Janet Vertesi. "Fairness and abstraction in sociotechnical systems". In: *Proceedings of the conference on fairness, accountability, and transparency*. 2019, pp. 59–68.

[Kha+17] Salman H Khan, Munawar Hayat, Mohammed Bennamoun, Ferdous A Sohel, and Roberto Togneri. "Cost-sensitive learning of deep feature representations from imbalanced data". In: *IEEE transactions on neural networks and learning systems* 29.8 (2017), pp. 3573–3587.

[ZL05] Zhi-Hua Zhou and Xu-Ying Liu. "Training cost-sensitive neural networks with methods addressing the class imbalance problem". In: *IEEE Transactions on knowledge and data engineering* 18.1 (2005), pp. 63–77.

[ZZ16] Siyuan Zhou and Ya Zhang. “Active learning for cost-sensitive classification using logistic regression model”. In: *2016 IEEE international conference on big data analysis (ICBDA)*. IEEE. 2016, pp. 1–4.

[Cui+19] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. “Class-balanced loss based on effective number of samples”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2019, pp. 9268–9277.

[Cao+19] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. “Learning imbalanced datasets with label-distribution-aware margin loss”. In: *Advances in neural information processing systems* 32 (2019).

[Qiu+23] Shikai Qiu, Andres Potapczynski, Pavel Izmailov, and Andrew Gordon Wilson. “Simple and fast group robustness by automatic feature reweighting”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 28448–28467.

[Han+24b] Xiaotian Han, Jianfeng Chi, Yu Chen, Qifan Wang, Han Zhao, Na Zou, and Xia Hu. “FFB: A Fair Fairness Benchmark for In-Processing Group Fairness Methods”. In: *Proceedings of the International Conference on Learning Representations*. 2024. DOI: 10.5555/TzAJbTClAz. URL: https://openreview.net/forum?id=TzAJbTClAz.

[San+21] Sara Sangalli, Ertunc Erdil, Andeas Hötker, Olivio Donati, and Ender Konukoglu. “Constrained optimization to train neural networks on critical and under-represented classes”. In: *Advances in neural information processing systems* 34 (2021), pp. 25400–25411.

[Ber76] Dimitri P Bertsekas. “Multiplier methods: A survey”. In: *Automatica* 12.2 (1976), pp. 133–145.

[HR] Martin Hirzel and Parikshit Ram. “Oversampling to Repair Bias and Imbalance Simultaneously”. In: *AutoML Conference 2023*.

[DKC22] Damien Dablain, Bartosz Krawczyk, and Nitesh Chawla. “Towards a holistic view of bias in machine learning: Bridging algorithmic fairness and imbalanced learning”. In: *arXiv preprint arXiv:2207.06084* (2022).

[Shu+22] Changjian Shui, Gezheng Xu, Qi Chen, Jiaqi Li, Charles X Ling, Tal Arbel, Boyu Wang, and Christian Gagné. “On learning fairness and accuracy on multiple subgroups”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 34121–34135.

[Sub+21] Shivashankar Subramanian, Afshin Rahimi, Timothy Baldwin, Trevor Cohn, and Lea Frermann. “Fairness-aware class imbalanced learning”. In: *arXiv preprint arXiv:2109.10444* (2021).

[YKF20] 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. 2020, pp. 1715–1724.

[Tar+23] Davoud Ataee Tarzanagh, Bojian Hou, Boning Tong, Qi Long, and Li Shen. "Fairness-aware class imbalanced learning on multiple subgroups". In: *Uncertainty in Artificial Intelligence*. PMLR. 2023, pp. 2123–2133.

[Bor+19a] Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. "Nuanced metrics for measuring unintended bias with real data for text classification". In: *Companion proceedings of the 2019 world wide web conference*. 2019, pp. 491–500.

[Bor+19b] Daniel Borkan, Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. "Limitations of pinned auc for measuring unintended bias". In: *arXiv preprint arXiv:1903.02088* (2019).

[Sag+19] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. "Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization". In: *arXiv preprint arXiv:1911.08731* (2019).

[Liu+21] Evan Z Liu, Behzad Haghgoo, Annie S Chen, Aditi Raghunathan, Pang Wei Koh, Shiori Sagawa, Percy Liang, and Chelsea Finn. "Just train twice: Improving group robustness without training group information". In: *International Conference on Machine Learning*. PMLR. 2021, pp. 6781–6792.

[LMK24] Tyler LaBonte, Vidya Muthukumar, and Abhishek Kumar. "Towards last-layer retraining for group robustness with fewer annotations". In: *Advances in Neural Information Processing Systems* 36 (2024).

[Koh+21] Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. "Wilds: A benchmark of in-the-wild distribution shifts". In: *International conference on machine learning*. PMLR. 2021, pp. 5637–5664.

[CM21] Ching-Yao Chuang and Youssef Mroueh. "Fair mixup: Fairness via interpolation". In: *arXiv preprint arXiv:2103.06503* (2021).

[LEN14] Zachary C Lipton, Charles Elkan, and Balakrishnan Naryanaswamy. "Optimal thresholding of classifiers to maximize F1 measure". In: *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2014, Nancy, France, September 15-19, 2014. Proceedings, Part II 14*. Springer. 2014, pp. 225–239.

[Koy+14] Oluwasanmi O Koyejo, Nagarajan Natarajan, Pradeep K Ravikumar, and Inderjit S Dhillon. "Consistent binary classification with generalized performance metrics". In: *Advances in neural information processing systems* 27 (2014).

[San16] Ignacio Enrique Sanchez. “Optimal threshold estimation for binary classifiers using game theory”. In: *F1000Research* 5 (2016).

[FM08] Elizabeth A Freeman and Gretchen G Moisen. “A comparison of the performance of threshold criteria for binary classification in terms of predicted prevalence and kappa”. In: *Ecological modelling* 217.1-2 (2008), pp. 48–58.

[Rob+20] Edgar Robles, Fatima Zaidouni, Aliki Mavromoustaki, and Payam Refael. “Threshold Optimization in Multiple Binary Classifiers for Extreme Rare Events using Predicted Positive Data.” In: *AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering (1)*. 2020.

[HFF12] José Hernández-Orallo, Peter Flach, and César Ferri Ramírez. “A unified view of performance metrics: Translating threshold choice into expected classification loss”. In: *Journal of Machine Learning Research* 13 (2012), pp. 2813–2869.

[KKS23] Hamid Reza Kazemi, Kaveh Khalili-Damghani, and Soheil Sadi-Nezhad. “Estimation of optimum thresholds for binary classification using genetic algorithm: An application to solve a credit scoring problem”. In: *Expert Systems* 40.3 (2023), e13203.

[Guo+17] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. “On calibration of modern neural networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 1321–1330.

[FK15] Peter Flach and Meelis Kull. “Precision-recall-gain curves: PR analysis done right”. In: *Advances in neural information processing systems* 28 (2015).

[He+16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[Dev18] Jacob Devlin. “Bert: Pre-training of deep bidirectional transformers for language understanding”. In: *arXiv preprint arXiv:1810.04805* (2018).

[Vap91] Vladimir Vapnik. “Principles of risk minimization for learning theory”. In: *Advances in neural information processing systems* 4 (1991).

[FCG20] Riccardo Fogliato, Alexandra Chouldechova, and Max G’sell. “Fairness evaluation in presence of biased noisy labels”. In: *International conference on artificial intelligence and statistics*. PMLR. 2020, pp. 2325–2336.

[Ngu+23] Khang Nguyen, Nong Minh Hieu, Vinh Duc Nguyen, Nhat Ho, Stanley Osher, and Tan Minh Nguyen. “Revisiting over-smoothing and over-squashing using ollivier-ricci curvature”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 25956–25979.

[Ham82] Richard S Hamilton. “Three-manifolds with positive Ricci curvature”. In: *Journal of Differential geometry* 17.2 (1982), pp. 255–306.

[Top+21] Jake Topping, Francesco Di Giovanni, Benjamin Paul Chamberlain, Xiaowen Dong, and Michael M Bronstein. “Understanding over-squashing and bottlenecks on graphs via curvature”. In: *arXiv preprint arXiv:2111.14522* (2021).

[Ni+19] Chien-Chun Ni, Yu-Yao Lin, Feng Luo, and Jie Gao. “Community detection on networks with Ricci flow”. In: *Scientific reports* 9.1 (2019), p. 9984.

[Sre+16] RP Sreejith, Karthikeyan Mohanraj, Jürgen Jost, Emil Saucan, and Areejit Samal. “Forman curvature for complex networks”. In: *Journal of Statistical Mechanics: Theory and Experiment* 2016.6 (2016), p. 063206.

[For03] Forman. “Bochner’s method for cell complexes and combinatorial Ricci curvature”. In: *Discrete & Computational Geometry* 29 (2003), pp. 323–374.

[Sam+18] Areejit Samal, RP Sreejith, Jiao Gu, Shiping Liu, Emil Saucan, and Jürgen Jost. “Comparative analysis of two discretizations of Ricci curvature for complex networks”. In: *Scientific reports* 8.1 (2018), p. 8650.

[WSJ17] Melanie Weber, Emil Saucan, and Jürgen Jost. “Characterizing complex networks with Forman-Ricci curvature and associated geometric flows”. In: *Journal of Complex Networks* 5.4 (2017), pp. 527–550.

[AZ06] Rie Ando and Tong Zhang. “Learning on graph with Laplacian regularization”. In: *Advances in neural information processing systems* 19 (2006).

[SG23] Or Streicher and Guy Gilboa. “Graph laplacian for semi-supervised learning”. In: *International Conference on Scale Space and Variational Methods in Computer Vision*. Springer. 2023, pp. 250–262.

[WJS16] Melanie Weber, Jürgen Jost, and Emil Saucan. “Forman-Ricci flow for change detection in large dynamic data sets”. In: *Axioms* 5.4 (2016), p. 26.

[FB15] Wei Feng and Samia Boukir. “Class noise removal and correction for image classification using ensemble margin”. In: *2015 IEEE international conference on image processing (ICIP)*. IEEE. 2015, pp. 4698–4702.

[Che+25] Mingcai Chen, Yuntao Du, Wei Tang, Baoming Zhang, and Chongjun Wang. “LaplaceConfidence: A graph-based approach for learning with noisy labels”. In: *Intelligent Data Analysis* 29.1 (2025), pp. 45–58.

[AB20] Bruno Klaus de Aquino Afonso and Lilian Berton. “Analysis of label noise in graph-based semi-supervised learning”. In: *Proceedings*

of the 35th Annual ACM Symposium on Applied Computing. 2020, pp. 1127–1134.

[Lam+19] Alex Lamy, Ziyuan Zhong, Aditya K Menon, and Nakul Verma. "Noise-tolerant fair classification". In: *Advances in neural information processing systems* 32 (2019).

[LW21] Yang Liu and Jialu Wang. "Can less be more? when increasing-to-balancing label noise rates considered beneficial". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 17467–17479.

[WLL21] 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*. 2021, pp. 526–536.

[Sil+24] Inês Oliveira e Silvaa, Sérgio Jesusa, Hugo Ferreiraa, Pedro Saleiroa, Inês Sousab, Pedro Bizarroa, and Carlos Soaresc. "Fair-OBNC: Correcting Label Noise for Fairer Datasets". In: (2024).

[Jes+22] Sérgio Jesus, José Pombal, Duarte Alves, André Cruz, Pedro Saleiro, Rita Ribeiro, João Gama, and Pedro Bizarro. "Turning the tables: Biased, imbalanced, dynamic tabular datasets for ml evaluation". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 33563–33575.

[Sil+23] Inês Oliveira e Silva, Carlos Soares, Inês Sousa, and Rayid Ghani. "Systematic analysis of the impact of label noise correction on ML Fairness". In: *Australasian Joint Conference on Artificial Intelligence*. Springer. 2023, pp. 173–184.

[Jes+24] Sérgio Jesus, Pedro Saleiro, Inês Oliveira e Silva, Beatriz M Jorge, Rita P Ribeiro, João Gama, Pedro Bizarro, and Rayid Ghani. "Aequitas flow: Streamlining fair ml experimentation". In: *Journal of Machine Learning Research* 25.354 (2024), pp. 1–7.

[Sal+18] Pedro Saleiro, Benedict Kuester, Loren Hinkson, Jesse London, Abby Stevens, Ari Anisfeld, Kit T Rodolfa, and Rayid Ghani. "Aequitas: A bias and fairness audit toolkit". In: *arXiv preprint arXiv:1811.05577* (2018).

[LT16] Tongliang Liu and Dacheng Tao. "Classification with Noisy Labels by Importance Reweighting". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38.3 (Mar. 2016), pp. 447–461. ISSN: 1939-3539. DOI: 10.1109/tpami.2015.2456899. URL: http://dx.doi.org/10.1109/TPAMI.2015.2456899.

[AU21] Görkem Algan and Ilkay Ulusoy. "Image classification with deep learning in the presence of noisy labels: A survey". In: *Knowledge-Based Systems* 215 (2021), p. 106771.

[Wu+22] Songhua Wu, Mingming Gong, Bo Han, Yang Liu, and Tongliang Liu. "Fair classification with instance-dependent label noise". In: *Con-*

ference on Causal Learning and Reasoning. PMLR. 2022, pp. 927–943.

[TW23] Donna Tjandra and Jenna Wiens. "Leveraging an alignment set in tackling instance-dependent label noise". In: *Conference on Health, Inference, and Learning*. PMLR. 2023, pp. 477–497.

[Can+24] Ygor Canalli, Filipe Braida, Leandro Alvim, and Geraldo Zimbrão. "Fair Transition Loss: From label noise robustness to bias mitigation". In: *Knowledge-Based Systems* 294 (2024), p. 111711.

[Pin+24] Carlos Pinzón, Catuscia Palamidessi, Pablo Piantanida, and Frank Valencia. "On the incompatibility of accuracy and equal opportunity". In: *Machine Learning* 113.5 (2024), pp. 2405–2434.

[Pin+22] Carlos Pinzón, Catuscia Palamidessi, Pablo Piantanida, and Frank Valencia. "On the impossibility of non-trivial accuracy in presence of fairness constraints". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 36. 7. 2022, pp. 7993–8000.

[Han+23] Xiaotian Han, Zhimeng Jiang, Hongye Jin, Zirui Liu, Na Zou, Qifan Wang, and Xia Hu. "Retiring Delta DP: New Distribution-Level Metrics for Demographic Parity". In: *arXiv preprint arXiv:2301.13443* (2023).

[Dix+18] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. "Measuring and mitigating unintended bias in text classification". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 2018, pp. 67–73.

APPENDIX A

Derivation of Simplified Formula for Forman Curvature

Here, we present the derivation of the simplified Forman curvature formula described in Section 6.2.3. This formula emphasizes edge-centric properties, indicating whether an edge serves as a bottleneck (negative curvature) or promotes expansion (positive curvature). We begin with the notion of discrete Ricci curvature for graphs or networks, namely, the Forman-Ricci curvature [For03; Sam+18], and derive the simplified version through a series of justified steps. For an edge e_{uv} connecting vertices u and v, the original Forman curvature formula is as follows:

$$F(e_{uv}) = w_{uv}\left(\frac{w_u}{w_{uv}} + \frac{w_v}{w_{uv}} - \sum_{x\sim u}\frac{w_u}{\sqrt{w_{uv}w_{ux}}} - \sum_{x\sim v}\frac{w_v}{\sqrt{w_{uv}w_{vx}}}\right) \tag{7.1}$$

where w_{uv} is the edge weight between u and v. Additionally, let w_u and w_v represent the weights, and let x be the neighbors of vertex u. Furthermore, we assume unit vertex weights ($w_u = w_v = 1$), which is common in applications within network science. Thus, we obtain the following.

$$F(e_{uv}) = w_{uv}\left(\frac{2}{w_{uv}} - \sum_{x\sim u}\frac{1}{\sqrt{w_{uv}w_{ux}}} - \sum_{x\sim v}\frac{1}{\sqrt{w_{uv}w_{vx}}}\right) \tag{7.2}$$

$$= w_{uv}\left(\frac{2}{w_{uv}} - \frac{1}{w_{uv}^{1/2}}\left[\sum_{x\sim u}\frac{1}{\sqrt{w_{ux}}} + \sum_{x\sim v}\frac{1}{\sqrt{w_{vx}}}\right]\right) \tag{7.3}$$

Next, we will divide the entire equation by 2.

$$\frac{F(e_{uv})}{2} = \frac{w_{uv}}{2}\left(\frac{2}{w_{uv}} - \frac{1}{w_{uv}^{1/2}}\left[\sum_{x\sim u}\frac{1}{\sqrt{w_{ux}}} + \sum_{x\sim v}\frac{1}{\sqrt{w_{vx}}}\right]\right) \tag{7.4}$$

$$= 1 - \frac{w_{uv}^{1/2}}{2}\left[\sum_{x\sim u}\frac{1}{\sqrt{w_{ux}}} + \sum_{x\sim v}\frac{1}{\sqrt{w_{vx}}}\right] \tag{7.5}$$

After that, we define the normalized curvature $F_{\text{norm}}(e_{uv})$.

$$F_{\text{norm}}(e_{uv}) := \frac{F(e_{uv})}{2} = 1 - \frac{1}{2}\sqrt{w_{uv}}\left[\sum_{x\sim u}\frac{1}{\sqrt{w_{ux}}} + \sum_{x\sim v}\frac{1}{\sqrt{w_{vx}}}\right] \tag{7.6}$$

Final Adjustment for GFLC: We apply the w_{uv} multiplication to the normalized curvature $F_{\text{norm}}(e_{uv})$ to ensure that strong edges dominate curvature calculations while weak edges in sparse areas do not appear artificially significant.

$$F_{\text{final}}(e_{uv}) = w_{uv} \cdot F_{\text{norm}}(e_{uv}) \tag{7.7}$$

$$F_{\text{final}}(e_{uv}) = w_{uv} \cdot F_{\text{norm}}(e_{uv}) \tag{7.8}$$

$$= w_{uv} \left(1 - \frac{1}{2} \sqrt{w_{uv}} \left[\sum_{x \sim u} \frac{1}{\sqrt{w_{ux}}} + \sum_{x \sim v} \frac{1}{\sqrt{w_{vx}}} \right] \right) \tag{7.9}$$

SISUKOKKUVÕTE

Andmetest õiglaste otsusteni: õigluse tagamine masinõppemudelites

Masinõppesüsteemide kasutamisest on saanud finants-, tervishoiu- ja haridussektori ning avaliku halduse otsustusprotsesside lahutamatu osa. Nende abil tehtud prognoosid mõjutavad üha enam seda, kes saab laenu, milliseid meditsiinilisi sekkumisi rakendatakse, milliseid sotsiaaltoetusi jagatakse või kui suure veebinähtavuse keegi saavutab. Kuigi sellised süsteemid võivad efektiivsust ja ennustuste täpsust märkimisväärselt suurendada, kätkevad need ka olulist riski: ajalooliselt kallutatud või struktuurselt tasakaalustamata andmestikel treenitud mudelid võivad süstemaatiliselt seada teatud demograafilised rühmad ebasoodsamasse olukorda. Siinses väitekirjas käsitletakse seda probleemi ning näidatakse, kuidas saavutada masinõppes parem tasakaal ennustustäpsuse ja õigluse vahel.

Väitekiri keskendub täpsuse ja õigluse tasakaalu leidmisele juhendatud masinõppes. Uurimistöös käsitletakse kolme peamist tegurit, mis üheskoos seda tasakaalu kujundavad: närvivõrkude arhitektuur, treeningandmete jaotus demograafiliste rühmade vahel ja siltide kvaliteet, millele juhendatud masinõpe tugineb. Iga nimetatud mõõtme puhul uuritakse väitekirjas, kuidas see mõjutab õigluse ja mudeli tulemuslikkuse vahelist tasakaalu.

Esimene panus keskendub mudeli arhitektuurile ning tutvustab treeningu käigus rakendatavat eelarvamuste vähendamise meetodit TFS (*The Fairness Stitch* – õigluse kiht). Õiglase klassifitseerimise tavapärane praktika koondab sekkumised enamasti mudeli viimasesse kihti, kus väljundkihti peenhäälestatakse õiglust tagavate piirangute all või kohandatakse otsustuskünniseid pärast treeningut. Doktoritöös väidetakse, et sellised ainult viimasele kihile keskenduvad strateegiad on struktuurselt piiratud, sest jätavad tähelepanuta koha, kus kallutatud esitusviisid tegelikult kujunevad – sügaval võrgustiku sees. TFS lahendab selle probleemi, lisades olemasolevate kihtide vahele väikese treenitava „õmbluskihi“ ja peenhäälestades just seda kihti õigluse piirangute all, hoides samal ajal ülejäänud põhimudeli parameetrid fikseerituna.

Teine panus keskendub andmestiku koostisele ning tutvustab raamistikku GLCS (*Group-Level Cost-Sensitive Deep Learning* – rühmatasandi kulutundlik süvaõpe), mis on loodud rühmapõhise klassi tasakaalustamatuse leevendamiseks ja nii rühmatasandi õigluse kui ka rühmatasandi stabiilsuse suurendamiseks. Paljudes realistlikes rakendustes on positiivsed ja negatiivsed klassid mitte üksnes globaalselt tasakaalust väljas, vaid ka demograafiliste rühmade vahel ebaühtlaselt jaotunud. Mõned alarühmad on ühtaegu alaesindatud ja seotud suurema valeklassifitseerimisriskiga, mis toob nende rühmade jaoks kaasa oluliselt kehvemad tulemused isegi siis, kui mudeli üldine täpsus paistab suur. GLCS vormistab selle probleemi otseselt õpieesmärgi osana, määrates valeklassifitseerimise kulu erinevatel rühmatasemetel ja optimeerides piiratud riski eesmärgiga tagada võrdsed võimalused.

Kolmas panus tegeleb siltide kvaliteedi küsimusega ja tutvustab meetodit GFLC (*Graph-based Fairness-aware Label Correction* – graafipõhine õiglusest lähtuv siltide parandamine), mis on loodud näitepõhise sildimüra käsitlemiseks viisil, mis sõnaselgelt arvestab õigluse nõudeid. Suurem osa õiglust käsitlevast teaduskirjandusest eeldab vaikimisi, et treeningsildid on korrektsed. Praktikas võivad sildid aga olla mürased märgendusvigade, piiripealsete või mitmeti tõlgendatavate juhtumite ja neid silte loonud inimeste tehtud otsuste kallutatuse tõttu. Oluline on, et selline müra ei jaotu enamasti ühtlaselt: teatud rühmad saavad süstemaatiliselt mürasemad sildid kui teised, mis võib süvendada just neid erinevusi, mida õiglussekkumised püüavad vähendada. GFLC lähtub sellest tähelepanekust ning kavandab sildiparanduse mehhanismi, mis kasutab korraga ära andmete geomeetrilist struktuuri, mudeli ebakindlust ja õiglusega seotud kaalutlusi.

Need kolm panust üheskoos pakuvad sidusat lahendust õigluse ja tulemuslikkuse dilemmale. *The Fairness Stitch* näitab, et õiglust saab suurendada, muutes sihipäraselt närvivõrkude sisemist arhitektuuri, mitte toetudes üksnes väljundkihi järel tehtavatele kohandustele. *Group-Level Cost-Sensitive Deep Learning* näitab, et rühmapõhised kulutundlikud eesmärgid võivad muuta mudeli treenimise õiglasemaks, säilitades samal ajal prognooside kvaliteedi ja suurendades mudeli stabiilsust jaotuse nihke suhtes. *Graph-based Fairness-aware Label Correction* laiendab õiglase teabe arvestamise ulatust olukordadele, kus sildid on mürased, ning illustreerib, kuidas graafistruktuuri ja õigluse piiranguid saab kasutada teoreetiliselt põhjendatud sildiparanduste suunamiseks. Kõigi kolme uurimissuuna raames tuuakse väitekirjas esile meetodid, mis on modulaarse ülesehitusega, sobituvad olemasolevate süvaõppe töövoogudega ja on rakendatavad realistlikel andmestikel.

Väitekirja peamine sõnum on, et on võimalik luua klassifitseerijaid, mis on mitte ainult täpsed, vaid ka õiglased ja võrdseid võimalusi tagavad. Selle saavutamiseks tuleb keskenduda mudeli arhitektuurile, andmestiku jaotusega seotud küsimustele ja andmesiltidele.

PUBLICATIONS

CURRICULUM VITAE

Personal data

Full Name:	Modar Sulaiman
Date of birth:	31.01.1992
Citizenship:	Syria
Contact:	modar.m.sulaiman@gmail.com

Education

2020–2026	University of Tartu, Estonia, Doctoral degree in Computer Science
2015–2017	University of L'Aquila, Italy, Master's degree in Mathematical Engineering (Scientific Computing); University of Silesia in Katowice, Poland, Master's degree in Mathematics (Mathematical Modelling)
2009–2013	Tishreen (Latakia) University, Syria, Bachelor's degree in Mathematics

Employment

2020–2026	Junior Research Fellow of Artificial Intelligence, University of Tartu, Estonia
2018–2019	Research Intern, SAP
2014–2015	Teaching Assistant, Tishreen (Latakia) University, Syria

Scientific work

Main fields of interest:

- Natural Language Processing
- Fairness in Machine Learning
- Deep Learning

ELULOOKIRJELDUS

Isikuandmed

Täisnimi:	Modar Sulaiman
Sünniaeg:	31.01.1992
Kodakondsus:	Süüria
Kontakt:	modar.m.sulaiman@gmail.com

Haridus

2020–2026	Tartu Ülikool, Eesti, doktorikraad arvutiteaduses
2015–2017	L'Aquila Ülikool, Itaalia, magistrikraad matemaatilises inseneriteaduses (teaduslik arvutus); Sileesia Ülikool Katowices, Poola, magistrikraad matemaatikas (matemaatiline modelleerimine)
2009–2013	Tishreeni (Latakia) Ülikool, Süüria, bakalaureusekraad matemaatikas

Teenistuskäik

2020–2026	Tehisintellekti nooremteadur, Tartu Ülikool, Eesti
2018–2019	Uurimispraktikant, SAP
2014–2015	Õppeassistent, Tishreeni (Latakia) Ülikool, Süüria

Teadustegevus

Peamised uurimisvaldkonnad:

- Loomuliku keele töötlus
- Õiglus masinõppes
- Sügavõpe

DISSERTATIONES INFORMATICAE PREVIOUSLY PUBLISHED IN DISSERTATIONES MATHEMATICAE UNIVERSITATIS TARTUENSIS

19. **Helger Lipmaa.** Secure and efficient time-stamping systems. Tartu, 1999, 56 p.
22. **Kaili Müürisep.** Eesti keele arvutigrammatika: süntaks. Tartu, 2000, 107 lk.
23. **Varmo Vene.** Categorical programming with inductive and coinductive types. Tartu, 2000, 116 p.
24. **Olga Sokratova.** Ω-rings, their flat and projective acts with some applications. Tartu, 2000, 120 p.
27. **Tiina Puolakainen.** Eesti keele arvutigrammatika: morfoloogiline ühestamine. Tartu, 2001, 138 lk.
29. **Jan Villemson.** Size-efficient interval time stamps. Tartu, 2002, 82 p.
45. **Kristo Heero.** Path planning and learning strategies for mobile robots in dynamic partially unknown environments. Tartu 2006, 123 p.
49. **Härmel Nestra.** Iteratively defined transfinite trace semantics and program slicing with respect to them. Tartu 2006, 116 p.
53. **Marina Issakova.** Solving of linear equations, linear inequalities and systems of linear equations in interactive learning environment. Tartu 2007, 170 p.
55. **Kaarel Kaljurand.** Attempto controlled English as a Semantic Web language. Tartu 2007, 162 p.
56. **Mart Anton.** Mechanical modeling of IPMC actuators at large deformations. Tartu 2008, 123 p.
59. **Reimo Palm.** Numerical Comparison of Regularization Algorithms for Solving Ill-Posed Problems. Tartu 2010, 105 p.
61. **Jüri Reimand.** Functional analysis of gene lists, networks and regulatory systems. Tartu 2010, 153 p.
62. **Ahti Peder.** Superpositional Graphs and Finding the Description of Structure by Counting Method. Tartu 2010, 87 p.
64. **Vesal Vojdani.** Static Data Race Analysis of Heap-Manipulating C Programs. Tartu 2010, 137 p.
66. **Mark Fišel.** Optimizing Statistical Machine Translation via Input Modification. Tartu 2011, 104 p.
67. **Margus Niitsoo**. Black-box Oracle Separation Techniques with Applications in Time-stamping. Tartu 2011, 174 p.
71. **Siim Karus.** Maintainability of XML Transformations. Tartu 2011, 142 p.
72. **Margus Treumuth.** A Framework for Asynchronous Dialogue Systems: Concepts, Issues and Design Aspects. Tartu 2011, 95 p.
73. **Dmitri Lepp.** Solving simplification problems in the domain of exponents, monomials and polynomials in interactive learning environment T-algebra. Tartu 2011, 202 p.

74. **Meelis Kull.** Statistical enrichment analysis in algorithms for studying gene regulation. Tartu 2011, 151 p.
77. **Bingsheng Zhang.** Efficient cryptographic protocols for secure and private remote databases. Tartu 2011, 206 p.
78. **Reina Uba.** Merging business process models. Tartu 2011, 166 p.
79. **Uuno Puus.** Structural performance as a success factor in software development projects – Estonian experience. Tartu 2012, 106 p.
81. **Georg Singer.** Web search engines and complex information needs. Tartu 2012, 218 p.
83. **Dan Bogdanov.** Sharemind: programmable secure computations with practical applications. Tartu 2013, 191 p.
84. **Jevgeni Kabanov.** Towards a more productive Java EE ecosystem. Tartu 2013, 151 p.
87. **Margus Freudenthal.** Simpl: A toolkit for Domain-Specific Language development in enterprise information systems. Tartu, 2013, 151 p.
90. **Raivo Kolde.** Methods for re-using public gene expression data. Tartu, 2014, 121 p.
91. **Vladimir Šor.** Statistical Approach for Memory Leak Detection in Java Applications. Tartu, 2014, 155 p.
92. **Naved Ahmed.** Deriving Security Requirements from Business Process Models. Tartu, 2014, 171 p.
94. **Liina Kamm.** Privacy-preserving statistical analysis using secure multi-party computation. Tartu, 2015, 201 p.
100. **Abel Armas Cervantes.** Diagnosing Behavioral Differences between Business Process Models. Tartu, 2015, 193 p.
101. **Fredrik Milani.** On Sub-Processes, Process Variation and their Interplay: An Integrated Divide-and-Conquer Method for Modeling Business Processes with Variation. Tartu, 2015, 164 p.
102. **Huber Raul Flores Macario.** Service-Oriented and Evidence-aware Mobile Cloud Computing. Tartu, 2015, 163 p.
103. **Tauno Metsalu.** Statistical analysis of multivariate data in bioinformatics. Tartu, 2016, 197 p.
104. **Riivo Talviste.** Applying Secure Multi-party Computation in Practice. Tartu, 2016, 144 p.
108. **Siim Orasmaa.** Explorations of the Problem of Broad-coverage and General Domain Event Analysis: The Estonian Experience. Tartu, 2016, 186 p.
109. **Prastudy Mungkas Fauzi.** Efficient Non-interactive Zero-knowledge Protocols in the CRS Model. Tartu, 2017, 193 p.
110. **Pelle Jakovits.** Adapting Scientific Computing Algorithms to Distributed Computing Frameworks. Tartu, 2017, 168 p.
111. **Anna Leontjeva.** Using Generative Models to Combine Static and Sequential Features for Classification. Tartu, 2017, 167 p.
112. **Mozhgan Pourmoradnasseri.** Some Problems Related to Extensions of Polytopes. Tartu, 2017, 168 p.

113. **Jaak Randmets.** Programming Languages for Secure Multi-party Computation Application Development. Tartu, 2017, 172 p.
114. **Alisa Pankova.** Efficient Multiparty Computation Secure against Covert and Active Adversaries. Tartu, 2017, 316 p.
116. **Toomas Saarsen.** On the Structure and Use of Process Models and Their Interplay. Tartu, 2017, 123 p.
121. **Kristjan Korjus.** Analyzing EEG Data and Improving Data Partitioning for Machine Learning Algorithms. Tartu, 2017, 106 p.
122. **Eno Tõnisson.** Differences between Expected Answers and the Answers Offered by Computer Algebra Systems to School Mathematics Equations. Tartu, 2017, 195 p.

DISSERTATIONES INFORMATICAE UNIVERSITATIS TARTUENSIS

1. **Abdullah Makkeh**. Applications of Optimization in Some Complex Systems. Tartu 2018, 179 p.
2. **Riivo Kikas**. Analysis of Issue and Dependency Management in Open-Source Software Projects. Tartu 2018, 115 p.
3. **Ehsan Ebrahimi**. Post-Quantum Security in the Presence of Superposition Queries. Tartu 2018, 200 p.
4. **Ilya Verenich**. Explainable Predictive Monitoring of Temporal Measures of Business Processes. Tartu 2019, 151 p.
5. **Yauhen Yakimenka**. Failure Structures of Message-Passing Algorithms in Erasure Decoding and Compressed Sensing. Tartu 2019, 134 p.
6. **Irene Teinemaa**. Predictive and Prescriptive Monitoring of Business Process Outcomes. Tartu 2019, 196 p.
7. **Mohan Liyanage.** A Framework for Mobile Web of Things. Tartu 2019, 131 p.
8. **Toomas Krips.** Improving performance of secure real-number operations. Tartu 2019, 146 p.
9. **Vijayachitra Modhukur.** Profiling of DNA methylation patterns as biomarkers of human disease. Tartu 2019, 134 p.
10. **Elena Sügis.** Integration Methods for Heterogeneous Biological Data. Tartu 2019, 250 p.
11. **Tõnis Tasa.** Bioinformatics Approaches in Personalised Pharmacotherapy. Tartu 2019, 150 p.
12. **Sulev Reisberg.** Developing Computational Solutions for Personalized Medicine. Tartu 2019, 126 p.
13. **Huishi Yin.** Using a Kano-like Model to Facilitate Open Innovation in Requirements Engineering. Tartu 2019, 129 p.
14. **Faiz Ali Shah.** Extracting Information from App Reviews to Facilitate Software Development Activities. Tartu 2020, 149 p.
15. **Adriano Augusto**. Accurate and Efficient Discovery of Process Models from Event Logs. Tartu 2020, 194 p.
16. **Karim Baghery.** Reducing Trust and Improving Security in zk-SNARKs and Commitments. Tartu 2020, 245 p.
17. **Behzad Abdolmaleki.** On Succinct Non-Interactive Zero-Knowledge Protocols Under Weaker Trust Assumptions. Tartu 2020, 209 p.
18. **Janno Siim.** Non-Interactive Shuffle Arguments. Tartu 2020, 154 p.
19. **Ilya Kuzovkin.** Understanding Information Processing in Human Brain by Interpreting Machine Learning Models. Tartu 2020, 149 p.
20. **Orlenys López Pintado.** Collaborative Business Process Execution on the Blockchain: The Caterpillar System. Tartu 2020, 170 p.
21. **Ardi Tampuu.** Neural Networks for Analyzing Biological Data. Tartu 2020, 152 p.

22. **Madis Vasser.** Testing a Computational Theory of Brain Functioning with Virtual Reality. Tartu 2020, 106 p.
23. **Ljubov Jaanuska.** Haar Wavelet Method for Vibration Analysis of Beams and Parameter Quantification. Tartu 2021, 192 p.
24. **Arnis Parsovs.** Estonian Electronic Identity Card and its Security Challenges. Tartu 2021, 214 p.
25. **Kaido Lepik.** Inferring causality between transcriptome and complex traits. Tartu 2021, 224 p.
26. **Tauno Palts.** A Model for Assessing Computational Thinking Skills. Tartu 2021, 134 p.
27. **Liis Kolberg.** Developing and applying bioinformatics tools for gene expression data interpretation. Tartu 2021, 195 p.
28. **Dmytro Fishman.** Developing a data analysis pipeline for automated protein profiling in immunology. Tartu 2021, 155 p.
29. **Ivo Kubjas.** Algebraic Approaches to Problems Arising in Decentralized Systems. Tartu 2021, 120 p.
30. **Hina Anwar.** Towards Greener Software Engineering Using Software Analytics. Tartu 2021, 186 p.
31. **Veronika Plotnikova.** FIN-DM: A Data Mining Process for the Financial Services. Tartu 2021, 197 p.
32. **Manuel Camargo.** Automated Discovery of Business Process Simulation Models From Event Logs: A Hybrid Process Mining and Deep Learning Approach. Tartu 2021, 130 p.
33. **Volodymyr Leno.** Robotic Process Mining: Accelerating the Adoption of Robotic Process Automation. Tartu 2021, 119 p.
34. **Kristjan Krips.** Privacy and Coercion-Resistance in Voting. Tartu 2022, 173 p.
35. **Elizaveta Yankovskaya.** Quality Estimation through Attention. Tartu 2022, 115 p.
36. **Mubashar Iqbal.** Reference Framework for Managing Security Risks Using Blockchain. Tartu 2022, 203 p.
37. **Jakob Mass.** Process Management for Internet of Mobile Things. Tartu 2022, 151 p.
38. **Gamal Elkoumy.** Privacy-Enhancing Technologies for Business Process Mining. Tartu 2022, 135 p.
39. **Lidia Feklistova.** Learners of an Introductory Programming MOOC: Background Variables, Engagement Patterns and Performance. Tartu 2022, 151 p.
40. **Mohamed Ragab.** Bench-Ranking: A Prescriptive Analysis Approach for Large Knowledge Graphs Query Workloads. Tartu 2022, 158 p.
41. **Mohammad Anagreh.** Privacy-Preserving Parallel Computations for Graph Problems. Tartu 2023, 181 p.
42. **Rahul Goel.** Mining Social Well-being Using Mobile Data. Tartu 2023, 104 p.

43. **Anti Ingel.** Algorithms using information theory: classification in brain-computer interfaces and characterising reinforcement-learning agents. Tartu 2023, 142 p.
44. **Shakshi Sharma.** Fighting Misinformation in the Digital Age: A Comprehensive Strategy for Characterizing, Identifying, and Mitigating Misinformation on Online Social Media Platforms. Tartu 2023, 158 p.
45. **Kristiina Rahkema.** Quality Analysis of iOS Applications with Focus on Maintainability and Security Aspects. Tartu 2023, 182 p.
46. **Ivan Slobozhan.** Studying Online Social Media Engagement in CIS Countries during Protests, Mass Demonstrations and War. Tartu 2023, 81 p.
47. **Nurlan Kerimov.** Building a catalogue of molecular quantitative trait loci to interpret complex trait associations. Tartu 2023, 248 p.
48. **Pavlo Tertychnyi.** Machine Learning Methods for Anti-Money Laundering Monitoring. Tartu 2023, 117 p.
49. **Abasi-amefon Obot Affia.** A Framework and Teaching Approach for IoT Security Risk Management. Tartu 2023, 180 p.
50. **Raimond-Hendrik Tunnel.** Video Game Design and Development Bachelor's Curriculum for Estonia. Tartu 2024, 137 p.
51. **Ahto Salumets.** Bioinformatics analysis of various aspects in immunology. Tartu 2024, 198 p.
52. **Mohammed Abdulhameed Shaif Ali.** Deep Learning Methods for Cell Microscopy Image Analysis. Tartu 2024, 143 p.
53. **Pille Pullonen-Raudvere**. Foundations of Efficient and Secure Algorithm Development for Secure Multiparty Computation. Tartu 2024, 265 p.
54. **Marili Rõõm.** Multiple approaches to learners' success and factors affecting it in computer programming MOOCs. Tartu 2024, 170 p.
55. **Shivananda Rangappa Poojara.** Design and Orchestration of Scalable, Event-Driven Serverless Data Pipelines for Internet of Things (IoT) Applications. Tartu 2024, 172 p.
56. **Hassan Abdulgaleel Hassan Salim Eldeeb.** Empowering Machine Learning Pipelines with Automated Feature Engineering. Tartu 2024, 121 p.
57. **Muhammad Uzair.** Soft decision making for agri-food 4.0. Tartu 2024, 158 p.
58. **Kirill Milintsevich.** Estimation of Depression Level from Text: Symptom-Based Approach, External Knowledge, Dataset Validity. Tartu 2024, 130 p.
59. **Maksym Del.** Multilingual and Multi-Domain Representational Patterns Across Trpansformer-Based Models. Tartu 2024, 131 p.
60. **Kristo Raun.** Adaptive Out-of-order Handling in Streaming Conformance Checking. Tartu 2024, 118 p.
61. **Toivo Vajakas.** Towards integration of mobile network data into analyzing human mobility. Tartu 2024, 103 p.
62. **Katsiaryna Lashkevich.** Data-Driven Analysis and Optimization of Waiting Times in Business Processes. Tartu 2024, 169 p.
63. **Alejandra Duque-Torres.** Classifying, Constraining and Ranking Metamorphic Relations. Tartu 2025, 159 p.

64. **Mariia Bakhtina.** A Method for Information Security and Privacy Management in Smart Solutions. Tartu 2025, 199 p.
65. **Andre Tättar.** Multilingual Machine Translation for Under-Resourced Languages. Tartu 2025, 170 p.
66. **Mahmoud Shoush.** Prescriptive Process Monitoring Under Uncertainty and Resource Constraints. Tartu 2025, 178 p.
67. **Alireza Akhavi Zadegan.** A Multimodal approach for refining Mapping and Localization by Integrating Generative AI and Pedestrian-Centric Data. Tartu 2025, 147 p.
68. **Eerik Muuli.** Automating the assessment and feedback processes in IT teaching – improving creation and maintenance from the teaching staff perspective. Tartu 2025, 196 p.
69. **Kateryna Kubrak.** Towards User-Centered Prescriptive Process Monitoring Systems. Tartu 2025, 151 p.
70. **Zhigang Yin.** Computing and Sensing in a Smart Ring. Tartu 2025, 251 p.
71. **Abdul-Rasheed Olatunji Ottun.** Practical Trustworthy Artificial Intelligence with Human Oversight. Tartu 2025, 239 p.
72. **Sander Mikelsaar.** Analysis and Optimization of Iteratively Decodable Codes. Tartu 2025, 146 p.
73. **Marharyta Domnich.** Advancing Human-Centric Counterfactual Explanations in Explainable AI. Tartu 2025, 210 p.
74. **Viacheslav Komisarenko.** Aligning Training Loss to Evaluation Metrics in Deep Learning. Tartu 2026, 165 p.
75. **Heidi Taveter.** Using Programming-Process Data of Introductory Programming Courses: Finding Solver Types, Giving Feedback, and Detecting Plagiarism. Tartu 2026, 184 p.
76. **Daniel Majoral Lopez.** Deep neural networks for microscopy images. Tartu 2026, 81 p.
77. **Mahir Gulzar.** Addressing Real-world Scenarios via Motion Prediction in Autonomous Driving. Tartu 2026, 141 p.
78. **Hele-Andra Kuulmets.** Cross-Lingual Transfer Learning and Evaluation in Low-Resource Settings. Tartu 2026, 180 p.
79. **Simmo Saan.** Correctness Witnesses for Thread-Modular Program Analysis. Tartu 2026, 251 p.
80. **Tarun Khajuria.** Scene understanding in human and computer vision. Tartu 2026, 151 p.