Operationalizing Fairness for Responsible Machine Learning

A dissertation submitted towards the degree
Doctor of Engineering (Dr.-Ing.)
of the Faculty of Mathematics and Computer Science
of Saarland University

by
Preethi Lahoti

Saarbrücken
2021

Defense Colloquium

Date: 20 May 2022

Dean of the Faculty: Prof. Dr. Jürgen Steimle

Examination Committee

Chair: Prof. Dr. Kurt Mehlhorn

Reviewer, Advisor: Prof. Dr. Gerhard Weikum

Reviewer, Co-Advisor: Prof. Dr. Krishna P. Gummadi

Reviewer: Moritz Hardt, PhD

Reviewer: Dr. Solon Barocas

Academic Assistant: Dr. Paramita Mirza

The future is already here – it's just not very evenly distributed.

William Gibson

Dedicated to Mumma and Pappa, for prioritizing girl education and for raising their daughters to be strong independent women.

Acknowledgments

I am deeply indebted to my PhD advisors Gerhard and Krishna for their mentorship and their role in shaping my career and life going forward. I am grateful for all the opportunities and for allowing me the freedom to pursue my research interests. Thank you Gerhard for shaping me into a better researcher and a better academic. Thank you for always being available and sharing your wisdom and guidance at every step of the journey, and helping me reach the best of my ability. Thank you Krishna for teaching me to broaden my perspective and to target bold goals. I will always cherish the countless hours of interactions that helped me grow both professionally and personally. I am forever grateful to you both.

I would like to thank the remaining members of my doctoral committee Moritz Hardt and Solon Barocas for their valuable feedback, and for allowing me the opportunity to learn from them. Thank you to Kurt Mehlhorn and Paramita Mirza for serving on my examination committee, and offering their valuable time and support.

Thank you to all my friends, colleagues and administrative staff at MPI-INF, MPI-SWS, and Saarland University for all the support and for making this journey enjoyable. Graduate school would not have been the same without you. Thanks to the International Max Planck Research School and ERC Synergy grant for supporting my research.

Some of the most memorable times during my doctoral studies were my internships. Thank you to Alex Beutel, Jilin Chen, Ed Chi, Nithum Thain, Marie Pellat, Fernando Diaz, Lucas Dixon, and all my collaborators and colleagues at Google Brain and PAIR team for providing such a great environment, and for shaping my research.

A special thanks to Aris Gionis, my long time mentor and trusted advisor. He is the one who first kindled my passion for research, and introduced me to the topic of fairness in machine learning. Thank you to Alessandra Sala for being a great mentor during my internship at Bell Labs, and for playing a pivotal role in my decision to pursue a PhD. Additionally, I would like to thank my other fantastic mentors – Alex Beutel, Asia Biega, Jilin Chen, Fernando Diaz and Jilles Vreeken. From technical insights to career opportunities your guidance has been invaluable. I am forever indebted to all my teachers over the years at Ananda Jyothi High school, Vasavi College of Engineering, and Aalto University. Everything that I am today is the culmination of years of your guidance and teaching.

Thank you to my parents and sisters for being my pillar of support in spite of the distance. It is your unreasonable confidence in me that has brought me here today. Heartfelt thanks to my German family for always being there, and for making me feel at home in Europe. Finally, this thesis would not have been possible without the unconditional support of my loving husband Robert. Thank you for being a trusted advisor, supporting partner, copy editor, and even a generous officemate during the pandemic.

Preethi

Abstract

As machine learning (ML) is increasingly used for decision making in scenarios that impact humans, there is a growing awareness of its potential for unfairness. A large body of recent work has focused on proposing formal notions of fairness in ML, as well as approaches to mitigate unfairness. However, there is a growing disconnect between the ML fairness literature and the needs to operationalize fairness in practice. This thesis addresses the need for responsible ML by developing new models and methods to address challenges in operationalizing fairness in practice. Specifically, it makes the following contributions.

First, we tackle a key assumption in the group fairness literature that sensitive demographic attributes such as race and gender are known upfront, and can be readily used in model training to mitigate unfairness. In practice, factors like privacy and regulation often prohibit ML models from collecting or using protected attributes in decision making. To address this challenge we introduce the novel notion of *computationally-identifiable* errors and propose Adversarially Reweighted Learning (ARL), an optimization method that seeks to improve the worst-case performance over unobserved groups, without requiring access to the protected attributes in the dataset.

Second, we argue that while group fairness notions are a desirable fairness criterion, they are fundamentally limited as they reduce fairness to an average statistic over pre-identified protected groups. In practice, automated decisions are made at an individual level, and can adversely impact individual people irrespective of the group statistic. We advance the paradigm of individual fairness by proposing iFair (individually fair representations), an optimization approach for learning a low dimensional latent representation of the data with two goals: to encode the data as well as possible, while removing any information about protected attributes in the transformed representation.

Third, we advance the individual fairness paradigm, which requires that similar individuals receive similar outcomes. However, similarity metrics computed over observed feature space can be brittle, and inherently limited in their ability to accurately capture similarity between individuals. To address this, we introduce a novel notion of fairness graphs, wherein pairs of individuals can be identified as deemed similar with respect to the ML objective. We cast the problem of individual fairness into graph embedding, and propose PFR (pairwise fair representations), a method to learn a unified pairwise fair representation of the data.

Fourth, we tackle the challenge that production data after model deployment is constantly evolving. As a consequence, in spite of the best efforts in training a fair model, ML systems can be prone to failure risks due to a variety of unforeseen reasons. To ensure responsible model deployment, potential failure risks need to be predicted, and mitigation actions need to be devised, for example, deferring to a human expert when uncertain or collecting additional data to address model's blind-spots. We propose Risk Advisor, a model-agnostic meta-learner to predict potential failure risks and to give guidance on the sources of uncertainty inducing the risks, by leveraging information theoretic notions of aleatoric and epistemic uncertainty.

This dissertation brings ML fairness closer to real-world applications by developing methods that address key practical challenges. Extensive experiments on a variety of real-world and synthetic datasets show that our proposed methods are viable in practice.

Kurzfassung

Mit der zunehmenden Verwendung von Maschinellem Lernen (ML) in Situationen, die Auswirkungen auf Menschen haben, nimmt das Bewusstsein über das Potenzial für Unfairness zu. Ein großer Teil der jüngeren Forschung hat den Fokus auf das formale Verständnis von Fairness im Zusammenhang mit ML sowie auf Ansätze zur Überwindung von Unfairness gelegt. Jedoch driften die Literatur zu Fairness in ML und die Anforderungen zur Implementierung in der Praxis zunehmend auseinander. Diese Arbeit beschäftigt sich mit der Notwendigkeit für verantwortungsvolles ML, wofür neue Modelle und Methoden entwickelt werden, um die Herausforderungen im Fairness-Bereich in der Praxis zu bewältigen. Ihr wissenschaftlicher Beitrag ist im Folgenden dargestellt.

In Kapitel 3 behandeln wir die Schlüsselprämisse in der Gruppenfairnessliteratur, dass sensible demografische Merkmale wie etwa die ethnische Zugehörigkeit oder das Geschlecht im Vorhinein bekannt sind und während des Trainings eines Modells zur Reduzierung der Unfairness genutzt werden können. In der Praxis hindern häufig Einschränkungen zum Schutz der Privatsphäre oder gesetzliche Regelungen ML-Modelle daran, geschützte Merkmale für die Entscheidungsfindung zu sammeln oder zu verwenden. Um diese Herausforderung zu überwinden, führen wir das Konzept der *Komputational-identifizierbaren* Fehler ein und stellen Adversarially Reweighted Learning (ARL) vor, ein Optimierungsverfahren, das die Worst-Case-Performance bei unbekannter Gruppenzugehörigkeit ohne Wissen über die geschützten Merkmale verbessert.

In Kapitel 4 stellen wir dar, dass Konzepte für Gruppenfairness trotz ihrer Eignung als Fairnesskriterium grundsätzlich beschränkt sind, da Fairness auf eine gemittelte statistische Größe für zuvor identifizierte geschützte Gruppen reduziert wird. In der Praxis werden automatisierte Entscheidungen auf einer individuellen Ebene gefällt, und können unabhängig von der gruppenbezogenen Statistik Nachteile für Individuen haben. Wir erweitern das Konzept der individuellen Fairness um unsere Methode iFair (individually fair representations), ein Optimierungsverfahren zum Erlernen einer niedrigdimensionalen Darstellung der Daten mit zwei Zielen: die Daten so akkurat wie möglich zu enkodieren und gleichzeitig jegliche Information über die geschützten Merkmale in der transformierten Darstellung zu entfernen..

In Kapitel 5 entwickeln wir das Paradigma der individuellen Fairness weiter, das ein ähnliches Ergebnis für ähnliche Individuen erfordert. Ähnlichkeitsmetriken im beobachteten Featureraum können jedoch unzuverlässig und inhärent beschränkt darin sein, Ähnlichkeit zwischen Individuen korrekt abzubilden. Um diese Herausforderung anzugehen, führen wir den neue Konzept der Fairnessgraphen ein, in denen Paare (oder Sets) von Individuen als ähnlich im Bezug auf die ML-Aufgabe identifiziert werden. Wir übersetzen das Problem der individuellen Fairness in eine Grapheinbindung und stellen PFR (pairwise fair representations) vor, eine Methode zum Erlernen einer vereinheitlichten paarweisen fairen Abbildung der Daten.

In Kapitel 6 gehen wir die Herausforderung an, dass sich die Daten im Feld nach der Inbetriebnahme des Modells fortlaufend ändern. In der Konsequenz können ML-Systeme trotz größter Bemühungen, ein faires Modell zu trainieren, aufgrund einer Vielzahl an unvorhergese-

henen Gründen scheitern. Um eine verantwortungsvolle Implementierung sicherzustellen, gilt es, Risiken für ein potenzielles Versagen vorherzusehen und Gegenmaßnahmen zu entwickeln, z.B. die Übertragung der Entscheidung an einen menschlichen Experten bei Unsicherheit oder das Sammeln weiterer Daten, um die blinden Flecken des Modells abzudecken. Wir stellen mit Risk Advisor einen modell-agnostischen Meta-Learner vor, der Risiken für potenzielles Versagen vorhersagt und Anhaltspunkte für die Ursache der zugrundeliegenden Unsicherheit basierend auf informationstheoretischen Konzepten der aleatorischen und epistemischen Unsicherheit liefert.

Diese Dissertation bringt Fairness für verantwortungsvolles ML durch die Entwicklung von Ansätzen für die Lösung von praktischen Kernproblemen näher an die Anwendungen im Feld. Umfassende Experimente mit einer Vielzahl von synthetischen und realen Datensätzen zeigen, dass unsere Ansätze in der Praxis umsetzbar sind.

Contents

Chapter 1

Introduction

Contents

1.1 Motivation

Recent decades have seen tremendous improvements in machine learning (ML) models with ground-breaking model performance, even rivaling that of human experts. At the same time the use of ML models for decision making has moved beyond decision-making in industrial automation, and ML is increasingly used for predictive decision making in scenarios that impact human individuals such as credit lending, college admissions, hiring, criminal justice, healthcare, and beyond.

While focusing on improving model accuracy was an acceptable performance goal for industrial automation, deployment of ML in mission-critical real-world systems calls for complementary considerations including fairness, accountability, trust and privacy. This thesis focuses on the fairness considerations in building responsible ML systems.

A crucial component of fair ML systems is that the ML system's predictions do not adversely affect individuals based on user's demographic attributes (which include sensitive or protected attributes such as gender, race). That is, they can be safely applied to make predictions for individuals from all demographic groups. Over the past several years, we have witnessed growing concerns about ML systems being "unfair" by introducing or perpetuating discriminatory behavior in predictive decision making. We point the reader to [Barocas and Selbst 2016; Crawford and Calo 2016; Angwin et al. 2016; Barocas et al. 2018; Kearns and Roth 2019] for an introduction to the topic and popular examples of bias and discrimination in algorithmic decision making.

The problem has garnered significant attention in the scientific machine learning community, and has motivated a subfield of research called fairness in ML. A large body of recent work has focused on proposing formal notions of fairness in ML, as well as approaches to mitigate unfairness. Despite the volume and velocity of research publications, there is a growing disconnect between ML fairness methods and the needs of the ML practitioners to operationalize fairness in practice.

This dissertation contributes to research in fairness in ML by developing models and methods that address challenges in putting fairness to practice.

1.1.1 Sources of Unfairness in the ML Pipeline

A typical ML pipeline involves a series of choices from problem formulation, data collection, model learning to model deployment and monitoring. Unfairness can be introduced at any point in the pipeline. Developing fair and responsible ML systems requires careful fairness considerations at all steps of the ML pipeline. Figure 1.1 visualizes a typical ML pipeline from a responsible machine learning perspective, and presents various fairness related considerations that arise at each stage of the pipeline.

Next, we briefly describe a typical ML lifecycle to help frame sources of unfairness that can arise in each step. This will be useful background information as we later introduce approaches for fairness in ML, and highlight key challenges in operationalizing them.

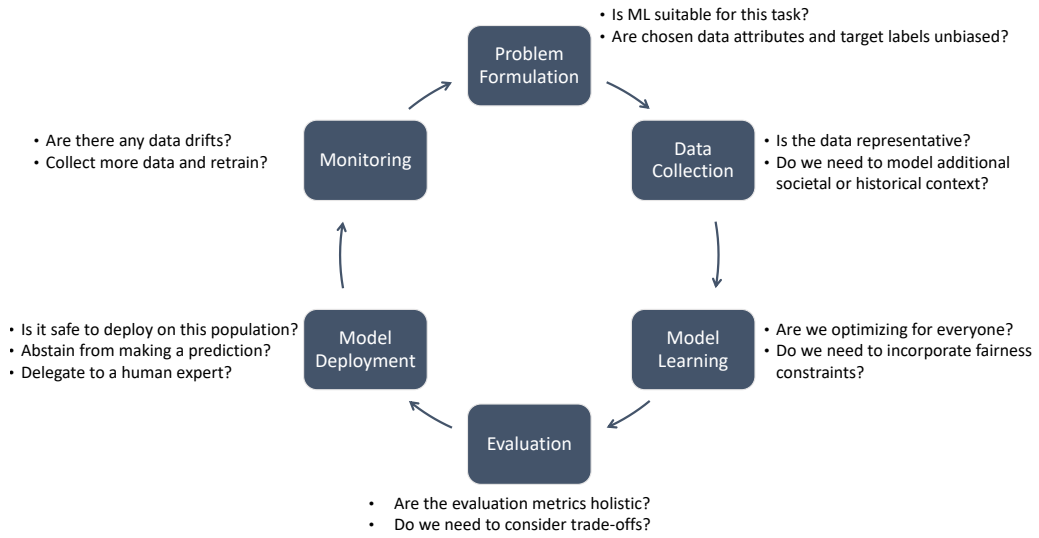


Figure 1.1: A schematic depiction of a typical machine learning pipeline and various fairness considerations that arise at each stage in the pipeline.

1.1.1.1 Problem Formulation and Data Collection

The first step of ML pipeline typically starts with translating the business problem into a predictive ML task, and collecting training data for learning. This involves selecting training population to collect data samples from, and making choices about which data attributes and target labels are to be collected. Each of these choices can potentially introduce bias into the training data. As ML algorithms see the world through the lenses of training data, any harmful correlations or biases in the data are mirrored, and potentially exacerbated.

Example: College Admission Consider the task of selecting students for Graduate School in the US. Performance in SAT (Scholastic Assessment Test) exam is often used as a proxy to gauge student's ability to successfully graduate after admission. It is well known that SAT exams can be taken multiple times, and only the best score is reported for admissions. Further, it is common to employ SAT tutors to receive additional coaching. However, professional tutoring, as well as each attempt to re-take the SAT exam come at a financial cost. Due to complex interplay of historical subordination and social circumstances, it is known that, on average, SAT scores for African-American students are lower than for white students [Brooks 1992]. Unless explicitly corrected, using such data to train predictive models could lead to unfair models predictions for African-American students.

1.1.1.2 Model Learning and Evaluation

ML models are trained and evaluated using aggregated measures of their predictive performance such as maximizing average accuracy (for classification tasks) or minimizing mean squared error (for regression tasks). However, optimizing for average loss is problematic if the training dataset is not a representative sample of all the groups in the population. Similarly, relying on aggregated metrics (e.g., average accuracy) for evaluation can be problematic as model performance may not be uniform across groups and aggregated metrics can hide disparities in model performance across groups.

Further, different population groups in a dataset can have different feature distribution and differ in their relationship to the prediction target variable. Optimizing for average accuracy could lead to a model that fits better to the majority population group (simply due to more data for this group), leading to more errors for individuals from under-represented groups in the training data.

Example: Face Recognition [Buolamwini and Gebru 2018] studied popular commercial face recognition systems and discovered that while all systems achieved ∼90% accuracy at a classification task, there was a significant difference in model performance across groups: female subjects were less accurate than male, dark-skinned less accurate than light-skinned, and intersectional groups like dark-skinned females the worst, differing in accuracy by as much as 34%.

1.1.1.3 Model Deployment and Monitoring

During model development, it is commonly assumed that training data is representative of the target population after model deployment. In practice, however, this is rarely the case. Models are often deployed in production on new target populations (e.g., geographic

locations, age groups, etc) far away from the training data. Further, models are often misused and deployed in ways that are not inline with the purpose for which they were trained. Such differences between training and deployment environments can lead to unforeseen model failures. Unfortunately, systems often fail silently without any warning, often while showing high confidence in their predictions.

Example: Pneumonia Screening [Zech et al. 2018] observed that ML models trained for automated pneumonia screening from chest X-ray images showed promising results on one hospital data. However, when the models were deployed on X-ray images from another hospital the model's ability to diagnose pneumonia significantly deteriorated.

1.2 Challenges

Next, we give a brief overview of popular fairness notions and techniques for incorporating fairness criterion into ML systems, and highlight key challenges in operationalizing them in practice.

1.2.1 Incorporating Fairness Criterion into ML Models

Training fair ML models requires going beyond optimizing for model's average performance, and considering alternate model objectives and fairness constraints which account for holistic performance for individuals from all protected groups. At a high level, algorithmic fairness literature has focused on two families of fairness notions: group notions of fairness and individual notions of fairness.

Group Notions of Fairness Group fairness notions are the most popular fairness definitions. At a high level, group fairness notions seek to achieve equality (of some metric of interest) across all protected groups in the population (e.g., based on demographic attributes like gender or race). For instance, *statistical parity* (also known as demographic parity) and its variants [Calders et al.; Kamiran et al. 2010; Feldman et al. 2015; Pedreschi et al. 2008; Barocas and Selbst 2016] seek to achieve parity in the proportion of positive outcomes across groups. Group fairness notions are popular as they are intuitive, and can be easily incorporated as statistical constraints in the ML objective, and solved as a constrained optimization problem.

A key assumption in group fairness notions is that protected attributes such as race and gender are specified upfront and that the model has access to protected attributes at training and inference time to mitigate unfairness [Hashimoto et al. 2018]. However, this is rarely the case. In practice, factors like privacy and regulation often prohibit ML models from collecting and/or using protected features for decision making. For instance, regulators like CFBP prohibit creditors from collecting or using information about an applicant's race, color, religion, national origin, or sex for decision-making.[1] Further, there can be many

[1] Creditors may not request or collect information about an applicant's race, color, religion, national origin, or sex. Exceptions to this rule generally involve situations in which the information is necessary to test for compliance with fair lending rules. [CFBP Consumer Law and Regulations, 12 CFR §1002.5]

"intersectional" subgroups, a term used to refer to groups formed at the intersection of several protected attributes (e.g., black female)[Kearns et al. 2017a]. As the number of features (and their arity) increases, the number of intersectional subgroups can exponentially increase. As a consequence, the requirement of achieving equality across all computable groups in the dataset can become too stringent making it computationally intractable.

Individual Notions of Fairness While effective at countering group-based unfairness in decision outcomes, *group fairness* notions do not address unfairness in outcomes at the level of individual users. For instance, it is natural for individuals to compare their outcomes with those of others with similar qualifications (independently of their group membership) and perceive any differences in outcomes amongst individuals with similar standing as unfair.

Dwork et al. [Dwork et al. 2012a] formalized this intuition, and proposed a notion of fairness called *individual fairness* (also known as metric fairness), which asks that "two individuals who are similar with respect to the predictive ML task should be classified similarly". The authors envisioned that a quantitative distance metric would be provided by fairness experts which captures the similarity between individuals with respect to the ML task at hand (e.g., suitability of candidate for college admission).

However, eliciting such a quantitative specification of a distance metric from human experts has been the most challenging aspect of the individual fairness framework. Consequently, despite its intuitive appeal and its potential to tackle unfairness beyond group-level, the alternative paradigm of *individual fairness* has received relatively little attention.

1.2.2 Learning Fair Data Representations

With a few exceptions, the vast majority of work on fairness in ML treats data as a given, and focus on incorporating fairness criterion (e.g., equal group error rates) into the ML learning objective. However, even so called "fair" models can lead to unfair outcomes for individuals by simply mirroring the biases present in the training data.

Obfuscating protected information Fair representation learning approaches [Zemel et al. 2013; Beutel et al. 2017; Feldman et al. 2015; Edwards and Storkey 2016] aims to "de-bias" the data by learning transformations of the original data that retain as much task-relevant information as possible while removing information about protected attributes. The high level idea being that such data representations can be freely used to train models for downstream ML tasks as group membership cannot be inferred from the transformed representations, i.e., subsequent ML models cannot differentiate between users based on group membership.

The key challenge in learning fair representations is its operationalization. Merely removing protected attributes is not enough as there are many data attributes that are not deemed protected, but have a relationship with one or more protected attributes. Not considering these attributes can result in scenarios where we erroneously believe that the models are fair, while they can be indirectly discriminating via correlated features (e.g., redlining in the US based on zip-code of residents).

Modeling and Correcting Data Bias While in certain cases it is useful to learn fair representations of the data which remove any information on group memberships. In other cases it is valuable to learn data representations that reconcile differences in feature distributions of individuals from different groups, for example by incorporating domain-specific societal and historical contexts into the learned representations.

As an example, recall the college admissions example introduced earlier (subsec. 1.1.1.1) wherein equally qualified candidates could potentially have different distributions for SAT scores due to differences in their financial standings. Unless such background fairness information is explicitly modeled and incorporated into model training, future ML models can mirror historical or societal biases present in the training data [Olteanu et al. 2019].

Overcoming biases embedded in the data would require knowledge of how the data generation/collection process is biased [Gebru et al. 2018], and actively seeking to counter this bias for example by collecting expert fairness judgments and enhancing the data with the desired fairness properties. Such nuanced fairness considerations require involvement of multiple stakeholders such as fairness experts and policy makers.

1.2.3 Safe Deployment, Risk Mitigation, and Monitoring

Often fair ML approaches assume that ML systems operate in a static setting, i.e., production data after deployment comes from the same distribution as the training data. They assume that once a ML model is trained by taking all fairness criteria into consideration, it can be safely used to make reliable predictions on newly seen production data. However, real life ML systems operate in dynamic environments where the data is constantly evolving. Further, models are frequently deployed on new target populations (e.g., geographic locations, demographic groups, etc) on which the model was not trained, and often in ways that are not inline with the purpose for which they were trained [Saria and Subbaswamy 2019].

Unfortunately, systems often fail silently without any warning, despite showing high confidence in predictions[Nguyen et al. 2015; Jiang et al. 2018; Goodfellow et al. 2015]. Thus, we need responsible model deployment and risk mitigation strategies that can anticipate failure risks and provide guidance on the appropriate risk mitigation actions. For instance, if an introspection component indicates non-negligible likelihood of being erroneous, the system could abstain or defer the decision to a human expert. When significant data drift is detected in the deployed environment, judiciously collecting additional training samples and updating the model would be a remedy.

1.3 Contributions

In the context of the described problems in developing fair and responsible machine learning models, this dissertation tackles the following specific research questions:

RQ:1 *Can we achieve group fairness without access to protected attributes at training or inference time? How can we go beyond assuming pre-specified (group of) protected attribute(s), and improve fairness for any combination of protected attribute values?*

In Chapter 3 we scrutinize one of the major assumption in operationalizing *group fairness*, namely that the protected attributes (e.g., race, gender) are specified upfront, and membership to the protected group is known. In practice, however, factors like privacy and regulation often prohibit ML models from collecting or using protected features for decision making. Therefore, in this chapter, we address the research question: "How can we train a ML model to improve group fairness when we do not have access to protected data attributes neither at training nor inference time?"

We propose Adversarially Reweighted Learning (ARL), a meta-learning approach that leverages the notion of computationally-identifiable errors. In particular, we hypothesize that non-protected features and task labels are valuable for computationally-identifying systematic errors due to unfairness, and can be used to co-train an adversarial reweighting approach for improving performance for worst-case unobserved groups. Our results show that ARL improves Rawlsian Max-Min fairness, with notable AUC improvements for worst-case protected groups in multiple datasets, outperforming state-of-the-art method [Hashimoto et al. 2018]. This work was published in the proceedings of NeurIPS 2020 [Lahoti et al. 2020].

RQ:2 *Can we go beyond statistical notions of group fairness, and mitigate unfairness at an individual level? Can we learn fair data representations that retain as much task-relevant information as possible, while removing information about protected attributes?*

Research on how to incorporate fairness into ML predictive tasks, including the previously introduced *ARL* approach, has largely focused on *group fairness*: giving "fair share of beneficial outcome" to protected groups. In Chapter 4, we take a critical look at the prevalently pursued paradigm of *group fairness*, highlight its limitations, and advance the alternate paradigm of *individual fairness*.

We propose *iFair* (individually fair representations), an optimization approach whose goal is to learn a generalized data representation that preserves fairness-aware similarity between individual records, while also aiming to minimize or bound the data loss. More formally, *iFair* assumes a restricted form of *fairness-aware* distance metric (weighted Minkowski p-metric) over the input feature space with learnable weights for data attributes. Once we have learned such a fair transformation $\phi : X \rightarrow \tilde{X}$, one can freely train any unconstrained predictor (e.g., classifier or ranker) without having to worry about individual fairness in prediction outcomes. Experimental studies with classification and regression tasks for downstream applications, empirically show that *iFair* can reconcile individual fairness with high utility, and outperforms state-of-the-art prior work [Zemel et al. 2013]. This work was published published in the proceedings of ICDE 2019 [Lahoti et al. 2019b].

RQ:3 *Can we achieve individual fairness without requiring an explicit specification of a quantitative distance metric d? How can we elicit and incorporate expert fairness judgements to counter data bias?*

In Chapter 5, we advance the original notion of individual fairness proposed by [Dwork et al. 2012a]. A key limitation of the prior work in operationalizing individual fairness, including our previous approach iFair is that they assume a distance metric over the input attribute

space. However, two individuals can be similar with respect to the ML task, while being far apart in attribute space (e.g., due to differences in feature distributions across groups). This simplification of the individual fairness notion largely limits the scope of the original idea of Dwork et al. : "... a (near ground-truth) approximation agreed upon by the society of the extent to which two individuals are deemed similar with respect to the task ... ".

In this work, we address this key limitation by proposing a practically viable operationalization of the individual fairness paradigm that does not rely on a human specification of a distance metric. Instead, we propose easier and more intuitive forms of eliciting expert fairness knowledge, and formalize them into a novel notion of pairwise *fairness graphs*. We propose *PFR* (pairwise fair representations), a representation learning approach, which aims to capture both data-driven similarity between individuals and pairwise fairness judgements in fairness graphs by casting it into a *graph embedding problem*. Comprehensive experiments with synthetic and real-life data demonstrate the practical viability of our model and its advantages over state-of-the-art prior works [Zemel et al. 2013; Lahoti et al. 2019b]. This work was published published in the proceedings of PVLDB 2019 [Lahoti et al. 2019a].

RQ:4 *Can we predict failure risks when deploying ML systems on production data? Can we identify the kind of uncertainty inducing the failure risk? How can we employ proactive monitoring and risk mitigation techniques?*

In Chapter 6, we address the challenge of predicting, analyzing and mitigating failure risks for classifier systems. The goal is to provide the system with *uncertainty scores* for its predictions, so as to (a) reliably predict test-time inputs for which the system is likely to fail, and (b) detect the kind of uncertainty that induces the risk, so that (c) appropriate mitigation actions can be pursued.

In this work, we propose the *Risk Advisor*, a post-hoc *meta-learning* approach to estimate uncertainties of a *fully trained* classifier, and to give guidance on the underlying sources of uncertainty. *Risk Advisor* is *model-agnostic*, and can be applied to any ML system, given only black-box access to the classifier, its training data, and its classification outputs. The *Risk Advisor* leverages the notions of *aleatoric* and *epistemic* uncertainties to distinguish between risks caused by distribution shifts between training data and deployment data, inherent data variability, and model limitations. Consequently, our approach for detecting failure risks is constructive by offering guidance on potential mitigation actions, like abstentions for criticial deployment-data points or requesting more training samples for re-building the production system. Comprehensive experiments with a variety of synthetic and real-world datasets show that *Risk Advisor* effectively predicts deployment-time failure risks and the source of uncertainty, outperforming state-of-the-art baselines [Jiang et al. 2018]. This work was published published in the proceedings of ICDM 2021 [Lahoti et al. 2021].

1.4 Publications

The results of this dissertation have been published in the following publications:

- **iFair: Learning Individually Fair Data Representations for Algorithmic Decision Making**.
  Preethi Lahoti, Krishna Gummadi, and Gerhard Weikum.
  IEEE International Conference on Data Engineering (ICDE), 2019

- **Operationalizing Individual Fairness with Pairwise Fair Representations**.
  Preethi Lahoti, Krishna Gummadi, and Gerhard Weikum.
  VLDB Endowment Vol. 13 No. 4 (PVLDB), 2019

- **Fairness without Demographics through Adversarially Reweighted Learning**.
  Preethi Lahoti[2], Alex Beutel, Jilin Chen, Kang Lee, Flavien Prost, Nithum Thain, Xuezhi Wang, Ed H. Chi
  Advances in Neural Information Processing Systems (NeurIPS), 2020

- **Detecting and Mitigating Test-time Failure Risks via Model-agnostic Uncertainty Learning**.
  Preethi Lahoti, Krishna Gummadi, and Gerhard Weikum.
  IEEE International Conference on Data Mining (ICDM), 2021

During her PhD, the author of this thesis has co-authored further papers related to fairness in ML, which are not included in this dissertation.

- **An Empirical Study on Learning Fairness Metrics for COMPAS Data with Human Supervision**.
  Hanchen Wang, Nina Grgic-Hlaca, Preethi Lahoti, Krishna Gummadi, Adrian Weller.
  In the NeurIPS Workshop on Human-centric Machine Learning, 2019

- **Accounting for Model Uncertainty in Algorithmic Discrimination**.
  Junaid Ali, Preethi Lahoti, and Krishna Gummadi
  AAAI Conference on Artifical Intelligence, Ethics and Society (AIES), 2021

- **A Practical Approach to Counterfactual Fairness using Generative Models**.
  Ayan Majumdar, Preethi Lahoti, Krishna Gummadi, and Isabel Valera.
  Under Submission

[2]Part of this research was performed while the author was an intern at Google Research. Preethi Lahoti co-authored the research proposal, designed the model, and was the main author and sole developer.

1.5 Organization

The remainder of the thesis is organized as follows. Chapter 2 gives an introduction to existing approaches for incorporating fairness criteria into machine learning systems. After an overview of popular fairness notions, and general paradigms in operationalizing fairness in machine learning systems, we highlight the assumptions made by the respective approaches, and the resulting challenges in practice. The remaining chapters each propose technical solutions to address some of these challenges. First, we address the challenge of achieving fairness at a group level without upfront knowledge of the protected attributes and group membership information (Chapter 3). Second, we cover the problems of mitigating fairness at an individual level by learning individually fair representations of the data (Chapter 4). Third, we present approaches to incorporate additional fairness context to counter data biases by eliciting and modeling nuanced expert input on pairwise similarity (Chapter 5). Fourth, in the context of responsible model deployment, we propose techniques for detecting and mitigating failure risks by modeling uncertainty in model predictions (Chapter 6). Chapter 7 concludes this dissertation and presents an outlook on future directions.

Chapter 2

Background

Contents

2.1 Preliminaries

This background chapter provides an overview of fairness in machine learning (ML). First, in Section 2.2 we provide a survey of algorithmic fairness definitions, and discuss the assumptions they make and their limitations. Next, in Section 2.3 we introduce techniques for incorporating fairness definitions into the ML pipeline.

In this background chapter we will limit our discussion to fairness in the classification setting. However, some of the approaches introduced in this thesis (in Chapters 4 and 5) are applicable beyond the classification setting, and can be freely applied to other ML task such as regression, and ranking. There is also extensive fairness literature in other ML domains such as recommender systems, computer vision, natural language processing, clustering, and many others which are beyond the scope of this thesis.

Notation

- X is an input dataset of n users in an m-dimensional feature space with binary or numerical values (i.e., after unfolding or encoding categorical attributes). We use X to denote both the dataset and the population of individuals $x_i \in X$.

$$X = [x_1, x_2, \cdots x_n] \in \mathbb{R}^{m \times n}$$

- Y denotes target class labels, and $y_i \in Y$ is the ground-truth class label corresponding to each individual $x_i \in X$.

- $\hat{Y}$ denotes predicted class labels, and $\hat{y}_i \in \hat{Y}$ is the predicted class label corresponding to each individual $x_i \in X$.

- S denotes the sensitive or protected attributes in the dataset, e.g., race, gender. We use S to denote both the protected attributes, as well as the attribute values it can take e.g., female, male, African-American.

Classification Task Given a training dataset $\mathcal{D} = \{(x_i, y_i) \cdots (x_n, y_n)\} \subset X \times Y$ drawn from an unknown data-generating distribution $\mathcal{P} \sim X \times Y$. Suppose H is a set of hypotheses (i.e., learned models) and $\ell(\cdot)$ is a loss function, the goal of the *classifier* is to learn a hypothesis that minimizes the expected empirical risk over observed training distribution $\mathcal{D}$.

$$h := \arg\min_{h \in H} \mathbb{E}_{(x_i, y_i) \in \mathcal{D}} \ell(h_\theta(x_i), y_i) \tag{2.1}$$

where $h_\theta(\cdot)$ is a classifier's predictor function with parameters θ, $\hat{y}_i = h_\theta(x_i)$ is the corresponding predicted class label, and $\ell(h_\theta(x_i), y_i)$ is standard classifier loss function (e.g., binary cross-entropy) between Y and $\hat{Y}$.

2.2 Algorithmic Fairness Notions

In this thesis, we focus on the algorithmic definitions of fairness introduced in the ML literature [Barocas et al. 2018; Friedler et al. 2019; Kearns and Roth 2019; Chouldechova and Roth 2020]. However, there is rich literature on fairness, discrimination and justice in other disciplines such as ethics, philosophy, social sciences, and law. A broader discussion on this topic from the viewpoint of other disciplines is beyond the scope of this thesis. We point the reader to [Green and Hu 2018; Binns 2018; Wachter et al. 2021] for further interdisciplinary reading on this topic. Next, we give a brief overview of algorithmic fairness notions, their assumptions and challenges in their operationalization.

2.2.1 Group Notions of Fairness

Group fairness notions ask that members of all protected groups in the population (e.g., based on demographic attributes like gender or race) receive their "fair share of beneficial outcomes" in a downstream task. To this end, one or more *protected attributes* and respective values are specified, and given special treatment in machine learning models. While the core idea of equality across groups is the same across all fairness definitions in the group fairness family, they differ in their choice of metric for "beneficial outcome". Following is a brief survey of popular group fairness definitions:

Statistical Parity and Disparate Impact The most widely used criterion is *statistical parity* (also known as demographic parity) and its variants [Calders et al.; Kamiran et al. 2010; Kamishima et al.; Pedreschi et al. 2008; Feldman et al. 2015; Fish et al. 2016]. Statistical parity seeks statistical independence between protected attribute S and the predicted outcomes $\hat{Y}$. In case of binary classification this translates to the probability of receiving a favourable outcome $\hat{Y} = 1$ for members of protected group to be the same as the probability of a favorable outcome for the members of non-protected group.

$$P(\hat{Y} = 1 | S = 1) = P(\hat{Y} = 1 | S = 0) \tag{2.2}$$

Inspired by the U.S. law of *disparate impact* or 80% rule [Feldman et al. 2015; Barocas and Selbst 2016], a number of variants of *statistical parity* have been proposed with the name *disparate impact* [Barocas and Selbst 2016; Feldman et al. 2015; Zafar et al. 2017b]. In contrast to parity, these notions consider the ratio of empirical probabilities of receiving positive outcome between the protected and non-protected groups, wherein a ratio of less than or equal to 0.8 is considered unfair.

$$\frac{P(\hat{Y}=1|S=1)}{P(\hat{Y}=1|S=0)} \leq \tau = 0.8 \tag{2.3}$$

The aforementioned fairness notions were defined in the context of classification. Similar variants of statistical parity or disparate impact have been proposed in other settings, including ranking [Asudeh et al. 2019; Elbassuoni et al. 2019; Zehlike et al. 2017], set selection [Stoyanovich et al. 2018], and clustering [Chierichetti et al. 2017], which seek approximately equal representation in the results.

A common criticism of statistical parity for operationalizing fairness in ML systems is that they do not consider data properties (e.g., ground-truth class label, data attributes). For many applications, such as risk assessment for healthcare systems, where risk scores are assigned to individuals, statistical parity may not be desirable as it would mandate that the fractions of positive outcomes across groups be the same irrespective of their data properties.

Equal Odds and Equal Opportunity The notion of *equal odds* [Hardt et al. 2016] addresses some of the drawbacks of statistical parity by considering *ground-truth* class label Y and the *predicted* outcome $\hat{Y}$ in its definition. Equal odds requires that the rates of false positives (FPR) and false negatives (FNR) be the same for protected and non-protected groups. Intuitively, this punishes classifiers which perform well only on specific groups. A similar notion of fairness was concurrently proposed by Zafar et al. [2017b], who refer to it as *disparate mistreatment*.

$$P(\hat{Y}=1|S=1,Y=y) = P(\hat{Y}=1|S=0,Y=y), \quad y \in \{0,1\} \tag{2.4}$$

Hardt et al. [2016] also proposed a relaxed version of equal odds called *equal opportunity* which demands only the equality of FPR (or FNR if the favorable outcome is encoded as $\hat{Y}=0$).

$$P(\hat{Y}=1|S=1,Y=0) = P(\hat{Y}=1|S=0,Y=0) \tag{2.5}$$

Predictive Parity and Calibration In contrast to the previous notions which were defined based on the predicted outcome $\hat{Y}$, calibration-based fairness notions [Kleinberg et al.; Chouldechova 2017] take the predicted probability or score (R) into account and ask that for a given predicted score ($R=r$) the predictions of the model $\hat{Y}$ should come true with the same rate across groups.

$$P(\hat{Y}=Y|S=1,R=r) = P(\hat{Y}=Y|S=0,R=r) \tag{2.6}$$

When R takes binary values, the aforementioned definition reduces to parity in positive predictive value (PPV) and negative predictive value (NPV) across groups

$$P(Y = 1|S = 1, \hat{Y} = 1) = P(Y = 1|S = 0, \hat{Y} = 1) \tag{2.7}$$

$$P(Y = 0|S = 1, \hat{Y} = 0) = P(Y = 0|S = 0, \hat{Y} = 0) \tag{2.8}$$

Fairness without Access to Protected Attributes A crucial limitation for all the aforementioned group fairness definitions is that they assume that protected attributes (e.g., race or gender) are specified upfront, and membership to the protected groups is known Holstein et al. [2019]. Recently techniques such as distributionally robust optimization Hashimoto et al. [2018], transfer learning Coston et al. [2019], secure multiparty computation [Kilbertus et al. 2018; Hu et al. 2019; Veale and Binns 2017] and differential privacy [Jagielski et al. 2019] have been applied to solve this key challenge. This thesis contributes to the group fairness line of research by developing methods for achieving group fairness without requiring that the protected attributes are specified upfront, and without knowledge of the protected group memberships of the individuals in the dataset. This contribution is presented in Chapter 3.

Subgroup Fairness Another key concern for group fairness notions is that a ML model can "seem" to be fair, while in fact introducing "fairness gerrymandering" [Kearns et al. 2018], a term used to refer to the situation wherein a ML model is fair in outcomes for each protected group (e.g., female vs male, African-American vs Caucasian), but is unfair towards individuals in intersectional groups such as Female African-american. To address *fairness gerrymandering*, a number of *subgroup fairness* notions have been proposed, which aims to achieve parity across many rich subgroups given by cross-product over all the protected attribute values [Kearns et al. 2018; Kim et al. 2018; Hébert-Johnson et al. 2017; Kim et al. 2019].

However, as the number of features (and their arity) increases, the number of intersectional subgroups can exponentially increase making it intractable to operationalize parity in practice. For instance, in the presence of just 6 protected attributes which take 4 values each, the number of subgroups in our collection is $\approx 10^7$ (given by $\Pi_{i \in m}(2^{|X^i|} - 1)$ where m is the number of protected-attributes).

This thesis contributes to the subgroup fairness line of research by developing methods for achieving subgroup fairness without requiring that the protected attributes are specified upfront, and without explicitly computing all intersectional groups in the dataset. Instead we propose the notion of "computationally-identifiable" errors to identify subgroups in the dataset that have *systematic errors*, and improve the model performance for such subgroups. This contribution is presented in Chapter 3.

Impossibility Results and Other Considerations Recently, several researchers have highlighted the inherent trade-offs between *fairness* and *utility* (e.g., accuracy) goals, the incompatibility between different notions of group fairness and the impossibility of achieving

them simultaneously [Kleinberg et al.; Chouldechova 2017; Friedler et al. 2016; Corbett-Davies et al. 2017].

It is also important to note that the group fairness definitions which seek "parity" are not always helpful in improving the performance for the underperforming group as the "parity" constraint equally allows decreasing the model's predictive performance (e.g., accuracy) for the better performing group. To avoid such scenarios, group fairness notions that explicitly seek to improve per-group performance have been proposed, such as Pareto-fairness [Balashankar et al. 2019] and Rawlsian Max-Min fairness [Rawls 2001; Hashimoto et al. 2018; Zhang and Shah 2014; Mohri et al. 2019; Diana et al. 2021]. In this thesis in Chapter 3, we follow the Rawlsian Max-Min fairness notion, and seek to improve the utility for the worst performing group in the dataset.

2.2.2 Individual Notions of Fairness.

Individual Fairness A key limitation of the group fairness notions is that they reduce fairness to an aggregate statistic over groups, without any fairness guarantees for individuals. Individual fairness aims to address this limitation by imposing fairness criteria at an individual level. In their seminal work, Dwork et al. [2012b] proposed the notion of *individual fairness*, which operates over individuals in the dataset and mandates that *similar individuals should be treated similarly*. Dwork et al. [2012b] assumed that the algorithm designer is provided with a *distance metric* on individuals $d : X \times X \to \mathbb{R}$, which measures similarity (distance) between individuals with respect to the predictive ML task. Given such a distance metric, they propose a theoretical framework for mapping individuals to a probability distribution over outcomes $M : X \to \Delta(Y)$, which satisfies the Lipschitz property (i.e., distance preservation) in the mapping as follows:

$$D(M(x_i), M(x_j)) \leq d(x_i, x_j) \quad \forall x_i, x_j \in X \tag{2.9}$$

where $D : \Delta(Y) \times \Delta(Y) \to \mathbb{R}$ and $d : X \times X \to \mathbb{R}$ denote distance metrics computed over probability distributions over outcomes and in the input space, respectively. In practice, however, such distance metrics are rarely available, and the lack of such distance metrics has been the central challenge in operationalizing individual fairness, and the focus of subsequent work. Zemel et al. [2013] proposed a method for learning fair data representations which satisfy individual fairness by learning a fair distance metric from the data. This thesis contributes to the individual fairness line of research by proposing approaches to overcome this challenges of specifying a distance metric, and by developing methods to learning individually fair representations of the data. This contribution is presented in Chapter 4 and Chapter 5.

Subsequent to our proposed methods for achieving individual fairness [Lahoti et al. 2019a,b], there has been some recent work on learning individually fair representations Yurochkin et al. [2020]; Ruoss et al. [2020]. A number of works propose learning a fair distance metric from the data via metric-learning[Ilvento 2020; Mukherjee et al. 2020; Bechavod et al. 2020; Gillen et al. 2018; Wang et al. 2019]. [Jung et al. 2021] made a similar argument as [Lahoti et al. 2019a] and proposed eliciting pairwise judgments on fairness, and an oracle efficient algorithm to enforce fairness constraints.

2.2.3 Other Notions of Fairness

Counterfactual Fairness There is an emerging line of works that consider fairness through the lens of causality [Kilbertus et al. 2017, 2018; Kusner et al. 2017; Bonchi et al. 2017; Coston et al. 2020; Zhang and Bareinboim 2018], and apply techniques and tool from causality to draw causal inferences with respect to the *counterfactual world* to define fairness. Kusner et al. [2017] propose the notion of *counterfactual fairness* which asks that an outcome received by an individual in the *real world* should be the same the as the outcome received in a *counterfactual world* in which the protected attributes of the individual are changed.

$$P(\hat{Y} = 1|X = x, S = s) = P(\hat{Y} = 1|X = x, S = s') \quad \forall x_i, x_j \in X, s, s' \in S \tag{2.10}$$

Note that this definition requires modeling the causal graph, and performing *interventions* on the causal graph to generate counterfactual datapoints [Pearl 2009].

Procedural Fairness In contrast to the majority of fairness notions which seek to achieve fairness over *outcomes* of ML task, Grgic-Hlaca et al. [2018b,a] propose *procedural fairness* which relates to the fairness in the *process* that leads up to the outcomes. Grgic-Hlaca et al. [2018b] argue that using certain features (e.g., protected attributes) in the predictive models can be deemed procedurally unfair irrespective to whether the eventual outcome is fair. Grgic-Hlaca et al. [2018a] investigate human perceptions of fairness by conducting extensive user studies, and analyze why people deem usage of certain features as unfair.

For the remainder of this thesis we will focus on group fairness and individual fairness notions.

2.3 Techniques for Operationalizing Fairness

A parallel line of research work in fairness in ML uses specific definitions of fairness and proposes approaches to incorporate fairness criteria into the ML pipeline. To this end, there are three general strategies: (i) pre-processing (ii) in-processing and (iii) post-processing as depicted in Figure 2.1. In what follows, we will introduce each of these strategies separately, and discuss selected existing ML fairness approaches that fall into these categories. We point the reader to [Friedler et al. 2019; Caton and Haas 2020; Islam et al. 2021] for a through survey of various approaches in each of these categories.

Pre-processing Methods The first strategy consists of data pre-processing methods that aim to incorporate fairness criteria into the training data before it is fed into the *model learning* stage of the ML pipeline. Early approaches in this category devised data pre-processing techniques that performed data perturbation such as modifying the values of the class labels for certain points in the training data to satisfy certain fairness conditions [Kamiran et al. 2010], "repairing" the training data by perturbing the values of data attributes [Feldman et al. 2015; Salimi et al. 2019] or distorting the training dataset until fairness constraints are satisfied in the outcomes [Pedreschi et al. 2008; Hajian and Domingo-Ferrer

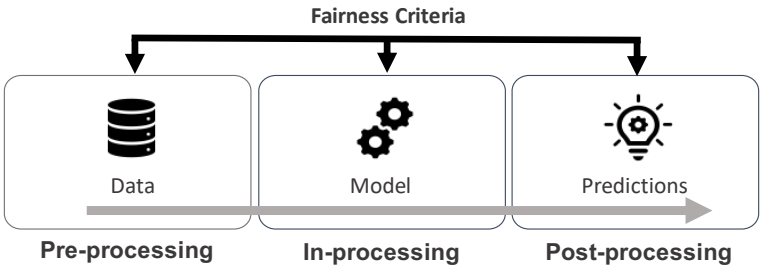


Figure 2.1: A schematic depiction of techniques for incorporating fairness criteria into the ML systems at various stages of the ML pipeline.

2013]. Most of these early approaches focused on the group fairness criterion of *statistical parity* or *disparate impact*.

Another approach for fair preprocessing aims to transform the original data into a low-rank latent representation such that downstream ML models trained on the learned representations satisfy fairness criteria. For instance, Zemel et al. [2013] propose a method to learn low-rank representations of the data that satisfy both *statistical parity* and *individual fairness*. Recently, approaches from adversarial learning have been applied to learn censored representations for fair classifiers that satisfy group fairness notions of *statistical parity* or *equal opportunity* [Louizos et al. 2016; Edwards and Storkey 2016; Beutel et al. 2017; Zhang et al. 2018; Madras et al. 2018]. Our proposed fair representation learning methods *iFair* (Chapter 4) and *PFR* (Chapter 5) fall into this category.

A key advantage of fair representation learning approaches is that once we have learned a fair transformation, one can freely train any unconstrained ML model without having to worry about individual fairness in prediction outcomes. However, these approaches suffer from a disadvantage that the learned representations are *latent*, and can affect other criteria such as interpretability or explainability of the downstream ML models.

In-processing Methods The second strategy consists of modifying the ML algorithm to incorporate fairness constraints into *model learning*. Typically, these approaches introduce fairness constraints as additional regularization terms in the original ML objective function, and fair variants of the model are learned via *constrained optimization*. For example, Kamishima et al. [2012] impose *statistical parity* constraints as fairness regularization terms into logistic regression model's objective function. Zafar et al. [2017b,a] develop approaches to incorporate *statistical parity* and *equal opportunity* as fairness constraints and propose convex proxies to solve the constrained optimization problem. Kearns et al. [2018] seek to prevent fairness gerrymandering by achieving *subgroup fairness*. Our proposed Adversarially Reweighted Learning approach *ARL* (Chapter 3) falls into this category, and aims to achieve *Rawlsian Max-Min* group fairness.

In-processing has been the most popular fairness strategy as it allows model designers to implement desired fairness constraints directly into the learning objective, and thus ensure better fairness guarantees. However, this property is also its key limitation as it requires the objective function to be modifiable, which may not be that easy for certain model classes.

Similarly, it may be challenging (if not infeasible) to modify complex ML systems deployed in real-world settings.

Post-processing Methods Post-processing approaches operate post-hoc on the model's predictions (e.g., predicted outcomes $\hat{Y}$ or probability scores) to ensure fairness in outcomes. For instance, Fish et al. [2016] propose approaches to shift the decision boundaries of classic ML algorithms such as adaptive boosting, SVM, and logistic regression to achieve *statistical parity* in the outcomes. Hardt et al. [2016] modify the decision score thresholds of a trained model to *equal odds* or *equal opportunity* across groups. Kamiran et al. [2010] propose a method for re-labeling the nodes of decision tree classifiers to ensure *statistical parity*. Pleiss et al. [2017] modify predicted class labels $\hat{Y}$ for random tuples to ensure that a calibrated classifier satisfies *equal opportunity*.

A key advantage of post-processing methods is their ease of implementation as they do not need any changes to data processing or model learning, and can be applied post-hoc to the eventual outcomes. However, this flexibility in their mechanism limits their ability to balance the accuracy-fairness trade-off [Islam et al. 2021].

2.4 Further Considerations in Fair Machine Learning

Changes in Deployment Environment Irrespective of the choice of the fairness notions (i.e., group or individual fairness) or the technique for operationalizing fairness (i.e., pre-processing, in-processing, or post-processing), the main idea in fair ML is that once a model is trained taking fairness criteria into consideration, it can be applied freely to production data on deployment. However, it is important to note that existing fairness techniques derive a fair model under the assumption that training and test data are identically and independently drawn (iid) from the same distribution.

A recent line of work investigates fairness under data shifts and proposes fairness approaches robust to changes between training and test distributions [Mohri et al. 2019; Rezaei et al. 2021, 2020; Biswas and Mukherjee 2021; Singh et al. 2021; Jiang et al. 2018]. In Chapter 3 we will draw connections between our proposed Adversarially Reweighted Learning (ARL) approach and this line of research, and show that ARL is robust to ensuring fairness even under worst-case distribution changes between training and test environments. In Chapter 5 we further contribute to this line of research by developing methods for predicting changes in deployment environment, and proposing techniques to counter failure risks due to distribution shifts by collecting more training samples in a judicious way.

Long Term Impact of Fairness As depicted in Figure 1.1, a typical ML pipeline in the real-world does not end after *model learning* or *model deployment*. Instead, the ML life-cycle continues with post-deployment tasks such as further data collection to perform model updates, i.e., retrain future versions of the ML system. The current version of the model determines which data/labels are collected for future training, thus inadvertently introducing feedback loops into the system. For example, in a credit lending system possible loan defaults (target class label) can only be observed if the current lending systems grants

a loan in the first place [Liu et al. 2019]. Similar feedback loops have been observed in predictive policing, recommender systems, pretrial detention and employment [Lum and Isaac 2016; Ferraro et al. 2021; Ensign et al. 2018].

Similarly, [Liu et al. 2019; Zhang et al. 2020] investigate long term impact of fairness interventions in ML systems and show that short term fairness goals can introduce unexpected and counter-intuitive harms to the protected group. For instance, enforcing fairness criteria such as *statistical parity* in the predictive outcomes may have undesirable effects such as increased loan default rate amongst the members of the protected group, which can in turn affect future loan opportunities for the protected group members. D'Amour et al. [2020] built on these results and developed a simulation framework to study long term dynamics in fair ML systems.

Privacy-Fairness Trade-off Typically sensitive data such as protected data attributes (e.g., race, gender) are required to train fair machine models. However, usage of such data attributes may compromise the user's privacy. An interesting line of work tackles this problem by employing cryptographic tools to train fair models in an encrypted form via secure multi-party computation [Veale and Binns 2017; Kilbertus et al. 2018; Hu et al. 2019], or in a privacy preserving form by employing differentially private learning [Veale and Binns 2017; Jagielski et al. 2019].

This thesis contributes to this line of work by developing fair machine learning models that can be trained without knowledge of the protected group memberships. In other words, without collecting or storing protected attributes in the dataset. This contribution is presented in Chapter 3.

Chapter 3

Fairness without Demographics

Contents

In this chapter we go beyond the assumption that protected attributes – i.e., sensitive demographic data such as race or gender – are specified upfront and are readily available to use in model training to mitigate unfairness. Our goal is to design fair classifiers that can achieve group fairness without requiring knowledge of protected group memberships. To address this challenge, we introduce the novel notion of *computationally-identifiable* errors and propose Adversarially Reweighted Learning (ARL), a method that seeks to improve Rawlsian Max-Min fairness without requiring access to protected attributes in the dataset. Specifically, *ARL* proposes a two-player game between a *learner* and an *adversary* wherein the *learner* aims to minimize the classification loss, while the *adversary* acts as a meta-learner whose goal is to identify regions of input space with systematic errors, and to guide the *learner* to improve its predictive performance in such regions by re-weighting the training examples. Extensive experiments on publicly available real-world and synthetic datasets shows that *ARL* improves Rawlsian Max-Min fairness, with notable AUC improvements for the worst-case protected groups in multiple datasets, outperforming state-of-the-art methods.

3.1 Introduction

We start from the observation that much of the previous group fairness research (e.g., [Zafar et al. 2017b; Hardt et al. 2016]) assumes that protected attributes such as race and sex are accessible in the dataset, and relies upon them to mitigate unfairness. However, in practice factors like privacy and regulation often preclude the collection of protected features, or their use for model training or inference, severely limiting the applicability of traditional fairness research in practice [Veale and Binns 2017; Holstein et al. 2019]. For instance, regulators like CFBP require that creditors comply by fairness, yet prohibit them from using demographic information for decision-making.[1] Similarly, GDPR imposes heightened prerequisites to collect and use protected features. Yet, in spite of these restrictions on access to protected features and their usage in ML models, it is imperative that our systems ensure fairness. Recent surveys of ML practitioners from both public sector [Veale and Binns 2017] and industry [Holstein et al. 2019] highlight this conundrum, and identify "addressing fairness without demographics" as a crucial open problem with high practical significance.Therefore, in this chapter, we ask the research question:

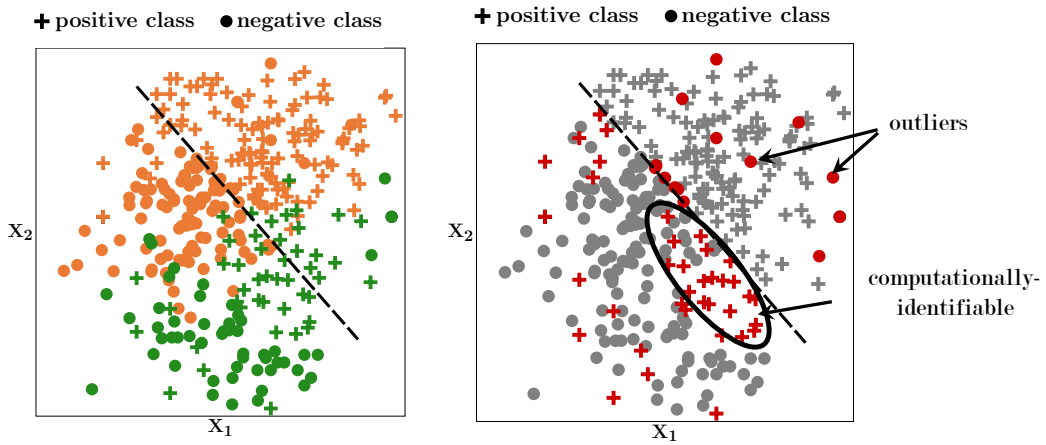
How can we train a ML model to improve group fairness when we do not have knowledge of protected group memberships?

Fairness Goal In this chapter, we follow the Rawlsian *Max-Min* fairness for distributive justice [Rawls 2001]. In Section 3.3.1, we formalize our *Max-Min* fairness goal: to train a model that maximizes the minimum expected utility across protected groups with the additional challenge that *we do not know protected group memberships*. It is worth noting that, unlike parity based notions of group fairness [Hardt et al. 2016; Zafar et al. 2017c], which aim to minimize gap across groups, *Max-Min* fairness notion permits inequalities. For many high-stakes ML applications, such as *healthcare*, improving the utility of worst-off groups is an important goal, and in some cases, parity notions that equally accept decreasing the accuracy of better performing groups are often not reasonable.

Key Idea: Computationally-Identifiable Errors While the system does not have direct access to protected groups, we hypothesize that unobserved protected features S are correlated with the observed features X (e.g., race is correlated with zip-code) and class labels Y (e.g., due to imbalanced class labels). As we will see in Table 3.8, this is frequently true. While correlates of protected features are a common cause for concern in the fairness literature, we show that this property can be valuable for *improving* fairness metrics. Next, we illustrate how this correlated information can be valuable with a toy example.

Illustrative Example Consider a binary classification task with data points from positive class ("+") and negative class ("o") as shown in Figure 3.1. Our dataset consists of individuals with membership to one of the two protected groups: "orange" data points and "green" data points. The dashed black line denotes the decision boundary learned by an empirical risk minimizing (ERM) classifier. The ERM classifier does not have access to the protected attribute (color), i.e., it only observes a point's position on the X_1 and X_2 axes.

[1] "Creditors may not request or collect information about an applicant's race, color, religion, national origin, or sex. Exceptions to this rule generally involve situations in which the information is necessary to test for compliance with fair lending rules." [CFBP Consumer Law and Regulations, 12 CFR §1002.5]



(a) Training Data **(b)** Classification Errors

Figure 3.1: An illustrative example of computationally-identifiable errors: (a) binary classification data consisting of datapoints from two groups: "orange" and "green" (b) classifications errors made by an ERM classifier on the training data are highlighted in "red".

Although the two classes ("+" and "o") are well-separable within each group alone (see Figure 3.1a), we see in Figure 3.1b that the ERM classifer over the full data makes more errors for the green group. Although the model does not have access to the protected attribute (i.e., color), observe that X_2 is correlated with the group membership. Even without access to group memberships (i.e., color), we can quickly identify a region of the input space with systematic errors (marked in ellipse) with low X_2 value and a positive label (+). In Section 3.3.2, we will define the notion of *computationally-identifiable* errors that correspond to such region. These errors are in contrast to errors due to noisy class labels (marked "outliers") which are randomly distributed across the X_1-X_2 input space.

The closest prior work to ours is *DRO* [Hashimoto et al. 2018]. Similar to us, *DRO* has the goal of fairness without demographics, and aims to achieve Rawlsian Max-Min Fairness for unknown protected groups. However, to achieve this, *DRO* uses distributionally robust optimization to optimize for any worst-case groups exceeding a size α. But as the authors point out, this runs the risk of focusing the optimization on noisy outliers in the training data. In contrast, we hypothesize that focusing on addressing computationally-identifiable errors will better improve fairness for the unobserved groups.

Adversarially Reweighted Learning With this hypothesis, we propose Adversarially Reweighted Learning (ARL), a method that leverages the notion of computationally-identifiable errors through an adversary $f_\phi(X, Y)$ to improve worst-case utility over outcomes for unobserved protected groups S. Our experimental results show that ARL achieves high AUC for worst-case protected groups, high overall AUC, and robustness against training data biases. Taken together, we make the following contributions:

- **Fairness without Demographics:** In Section 3.3, we propose Adversarially Reweighted Learning (*ARL*), a modeling approach that aims to improve the utility for worst-off

protected groups, without access to protected features at training or inference time. Our key insight is that when improving model performance for worst-case groups, it is valuable to focus the objective on *computationally-identifiable* regions of errors.

- **Empirical Benefits:** In Section 3.4, we evaluate *ARL* on three real-world datasets. Our results show that *ARL* yields significant AUC improvements for worst-case protected groups, outperforming state-of-the-art alternatives on all the datasets, and even improves the overall AUC on two of three datasets.

- **Understanding ARL:** In Section 3.5 we do a thorough experimental analysis and present insights into the inner-workings of ARL by analyzing the learned example weights. In addition, we perform a synthetic study to investigate robustness of *ARL* to worst-case training distributions. We observe that *ARL* is quite robust to representation bias, and differences in group base-rate. However, similar to prior approaches, *ARL* degrades with noisy class labels.

3.2 Related Work

We now discuss work most closely related to *ARL*.

Fairness without demographics: An interesting line of work tackles this problem by relying on trusted third parties that collect and store protected demographic data necessary for incorporating fairness. They generally assume that the ML model has access to the protected features, albeit in encrypted form via secure multi-party computation [Veale and Binns 2017; Kilbertus et al. 2018; Hu et al. 2019], or in a privacy preserving form by employing differentially private learning [Veale and Binns 2017; Jagielski et al. 2019].

Some works address this problem *approximately* by using proxy features [Wang et al. 2020] or assuming that the protected attribute is slightly perturbed [Awasthi et al. 2020]. However, using proxies can in itself be prone to estimation bias [Kallus et al. 2019; Chen et al. 2019]. Multiple works have explored addressing access to limited amount of demographic data via transfer learning and domain adaptation [Coston et al. 2019; Madras et al. 2018; Creager et al. 2021]. For example, Coston et al. [2019] focus on domain adaptation of fairness in settings where the protected group memberships are known for either source or target dataset. Mohri et al. [2019] consider a federated learning setting, wherein given training data from K *known* domains (equivalent to groups) with *unknown* sampling distributions, the model optimizes for a worst-case target distribution.

The closest prior work to ours is *DRO* [Hashimoto et al. 2018], which uses techniques from distributionally robust optimization to achieve Rawlsian Max-Min fairness without having access to protected attributes. A key difference between *DRO* and *ARL* is the type of worst-case groups identified by them in the dataset: DRO considers any worst-case distribution exceeding a given size α as a potential protected group. Concretely, given a lower bound on size of the smallest protected group, say α, *DRO* optimizes for improving the worst-case loss of any set of examples exceeding size α. In contrast, *ARL* relies on the notion of computational-identifiability to identify training samples with systematic errors.

Computational-Identifiability: Related to our algorithm, a number of works [Kearns et al. 2018; Kim et al. 2018; Hébert-Johnson et al. 2017; Kim et al. 2019] address intersectional fairness by optimizing for group fairness between all computationally identifiable groups in the input space. While the perspective of learning over computationally identifiable groups is similar, they differ from us in that they assume the protected group features are available in their input space, and that they aim to minimize the *gap* in utility across groups via regularization.

Modeling Technique Inspirations: In terms of technical machinery, our proposed ARL approach draws inspiration from a wide variety of prior modeling techniques. Re-weighting [Kahn and Marshall 1953; Höfler et al. 2005; Little and Rubin 1986] is a popular paradigm typically used to address problems such as class imbalance by upweighting examples from minority class. Adversarial learning [Goodfellow et al. 2014; Asif et al. 2015] is typically used to train a model to be robust with respect to adversarial examples. Focal loss [Lin et al. 2017] encourages the learning algorithm to focus on more *difficult* examples by up-weighting examples proportionate to their losses. Domain adaptation work requires a model to be robust and generalizable across different domains, under either covariate shift [Zadrozny 2004; Shimodaira 2000] or label shift Lipton et al. [2018].

3.3 Model

We now dive into the precise problem formulation and our proposed modeling approach.

3.3.1 Problem Formulation

In this chapter we consider a binary classification setup (though the approach can be generalized to other settings). We are given a training dataset consisting of n individuals $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$ where $x_i \sim X$ is an m-dimensional input vector of non-protected features, and $y_i \sim Y$ represents a binary class label. We use S to denote both the protected attributes (e.g., race or gender), as well as the protected groups (given by a cross-product over the values that the protected attributes take), e.g., Male African-American. Concretely, S is a random variable over $\{k\}_{k=0}^K$ where K is the number of subgroups in the dataset. For each $x_i \in X$ there exists an unobserved $s_i \sim S$ denoting its group membership. The set of examples (i.e., data points) with membership in group s is given by $\mathcal{D}_s := \{(x_i, y_i) : s_i = s\}_{i=1}^n$. Again, we do not observe a distinct set $\mathcal{D}_s$ but include the notation for formulation of the problem. To be more precise, we assume that protected attributes S are unobserved and not available at training or inference times. However, we will frame our definition and evaluation of fairness in terms of groups S.

Problem Definition Given dataset $\mathcal{D} \in X \times Y$, but *no observed protected group memberships* S, learn a model $h_\theta : X \to Y$ that is fair to all the groups in $s \in S$.

A natural next question is: what is a "fair" model? As in DRO [Hashimoto et al. 2018], we follow the Rawlsian *Max-Min* fairness principle of distributive justice [Rawls 2001]: we aim to maximize the minimum utility U a model has across all groups $s \in S$ as given by Definition 1. Here, we assume that when a model predicts an example correctly, it increases

utility for that example. As such U can be considered any one of standard accuracy metrics in machine learning that models are designed to optimize for.

Definition 1 (Rawlsian Max-Min Fairness)**.** *Suppose H is a set of hypotheses, and $U_{\mathcal{D}_s}(h)$ is the expected utility of the hypothesis h for the individuals in group s, then a hypothesis h^* is said to satisfy Rawlsian* Max-Min *fairness principle [Rawls 2001] if it maximizes the utility of the worst-off group, i.e., the group with the lowest utility.*

$$h^* = \arg\max_{h \in H} \min_{s \in S} U_{\mathcal{D}_s}(h) \tag{3.1}$$

In our evaluation in Section 3.4, we use AUC as a utility metric, and report the minimum utility over protected groups S as AUC(min).

3.3.2 Adversarial Reweighted Learning

Given this fairness definition and goal, how do we achieve it? As with traditional machine learning, most utility/accuracy metrics are not differentiable, and convex loss functions are used instead. The traditional ML task is to learn a model h that minimizes the loss over the training data $\mathcal{D}$:

$$h^*_{\text{avg}} = \arg\min_{h \in H} L_{\mathcal{D}}(h), \tag{3.2}$$

where $L_{\mathcal{D}}(h) = \mathbb{E}_{(x_i,y_i)\sim\mathcal{D}}[\ell(h(x_i), y_i]$ for some loss function $\ell(\cdot)$ (e.g., binary cross entropy).

Therefore, we take the same perspective in turning Rawlsian Max-Min Fairness as given in Eq. (3.1) into a learning objective. Replacing the expected utility with an appropriate loss function $L_{\mathcal{D}_s}(h)$ over the set of individuals in group s, we can formulate our fairness objective as:

$$h^*_{\max} = \arg\min_{h \in H} \max_{s \in S} L_{\mathcal{D}_s}(h) \tag{3.3}$$

where $L_{\mathcal{D}_s}(h) = \mathbb{E}_{(x_i,y_i)\sim\mathcal{D}_s}[\ell(h(x_i), y_i]$ is the expected loss for the individuals in group s.

Minimax Problem: Similar to Agnostic Federal Learning (AFL) [Mohri et al. 2019], we can formulate the Rawlsian *Max-Min Fairness* objective function in Eq. (3.3) as a zero-sum game between two players θ and λ. The optimization comprises of T game rounds. In round t, player θ learns the best parameters θ that minimize the expected loss. In round $t+1$, player λ learns an assignment of weights λ that maximize the weighted loss.

$$\begin{aligned} J(\theta, \lambda) &:= \min_{\theta} \max_{\lambda} L(\theta, \lambda) = \min_{\theta} \max_{\lambda} \sum_{s \in S} \lambda_s L_{\mathcal{D}_s}(h) \\ &= \min_{\theta} \max_{\lambda} \sum_{i=0}^{n} \lambda_{s_i} \ell(h(x_i), y_i) \end{aligned} \tag{3.4}$$

To derive a concrete algorithm we need to specify how the players pick θ and λ. For the player θ, one can use any iterative learning algorithm for classification tasks. For player λ, if the group memberships were known, the optimization problem in Eq. 3.4 could be solved by projecting θ on a probability simplex over S groups given by $\lambda = \{[0,1]^S : \|\lambda\| = 1\}$ as in AFL [Mohri et al. 2019]. Unfortunately, for us, as we do not observe S, we cannot directly optimize this objective as in AFL [Mohri et al. 2019]. DRO [Hashimoto et al. 2018] deals with this by effectively setting weights λ_i based on $\ell(h(x_i), y_i)$ to focus on the

largest errors. Instead, we will leverage the concept of *computationally-identifiable* subgroups [Hébert-Johnson et al. 2017].

Computational-Identifiability: Given a family of binary functions $\mathcal{F}$, we say that a subgroup S is computationally-identifiable if there is a function $f : X \times Y \to \{0, 1\}$ in $\mathcal{F}$ such that $f(x, y) = 1$ if and only if $(x, y) \in S$. Building on this definition, we define $f_\phi : X \times Y \to [0, 1]$ to be an adversarial neural network parameterized by ϕ whose task, implicitly, is to identify regions where the learner makes significant errors, e.g. regions $Z := \{(x, y) : \ell(h(x), y) \geq \varepsilon\}$. Since our adversarial network f_ϕ is not a binary classifier, we do not explicitly specify ε but rather train f_ϕ such that it returns a higher value in higher loss regions. The adversarial example weights $\lambda_\phi : f_\phi \to \mathbb{R}$ can then be defined by appropriately rescaling f_ϕ to put a high weight on regions with a high likelihood of errors, encouraging h_θ to improve in these regions. Rather than explicitly enforcing a binary set of weights, as would be implied by the original definition of computational identifiability, our adversary uses a sigmoid activation to map $f_\phi(x, y)$ to [0,1].

ARL Objective: We formalize this intuition, and propose an Adversarially Reweighted Learning approach, called *ARL*, which considers a minimax game between a *learner* and *adversary*: Both *learner* and *adversary* are learnt models, trained alternatively. The *learner* optimizes for the main classification task, and aims to learn the best parameters θ that minimizes expected loss. The *adversary* learns a function mapping $f_\phi : X \times Y \to [0, 1]$ to *computationally-identify* regions with high loss, and makes an adversarial assignment of weight vector $\lambda_\phi : f_\phi \to \mathbb{R}$ so as to maximize the expected loss. The *learner* then adjusts itself to minimize the adversarial loss.

$$J(\theta, \phi) = \min_\theta \max_\phi \sum_{i=1}^{n} \lambda_\phi(x_i, y_i) \cdot \ell(h_\theta(x_i), y_i) \tag{3.5}$$

If the adversary was perfect, it would *adversarially* assign all the weight to training examples in computationally-identifiable regions where the *learner* makes significant errors, and thus improve *learner's* performance in such regions. It is worth highlighting that the design and complexity of the adversary model f_ϕ plays an important role in controlling the granularity of *computationally-identifiable* regions of error. More expressive f_ϕ leads to finer-grained computationally-identifiable regions (and hence finer grained re-weighting) but runs the risk of overfitting to errors due to outliers.

Observe that without any constraints on λ the objective in Eq. 3.5 is ill-defined. There is no finite λ that maximizes the loss, as an even higher loss could be achieved by scaling up λ. Thus, it is crucial that we constrain the values λ. In addition, it is necessary that $\lambda_i \geq 0$ for all i, since minimizing the negative loss can result in unstable behaviour. Further, we do not want λ_i to fall to 0 for any examples, so that all examples can contribute to the training loss. Finally, to prevent exploding gradients, it is important that the weights are normalized across the dataset (or current batch). In principle, our optimization problem is general enough to accommodate a wide variety of constraints. In this work we perform a normalization step that rescales the adversary $f_\phi(x, y)$ to produce the weights λ_ϕ. We center

the output of f_ϕ and add 1 to ensure that all training examples contribute to the loss.

$$\lambda_\phi(x_i, y_i) = 1 + n \cdot \frac{f_\phi(x_i, y_i)}{\sum_{i=1}^{n} f_\phi(x_i, y_i)}$$

3.4 Experiments

We now evaluate the effectiveness of our proposed *ARL* approach, and investigate the inner-workings of *ARL* through experiments on real-world and synthetic datasets.

Datasets and Pre-processing We perform our experiments on three real-world, publicly available datasets, previously used in the literature on algorithmic fairness:

- Census Income: The UCI Adult dataset [R. Kohavi 1996] contains US census income survey records. We use the binarized "income" feature as the target variable for our classification task to predict if an individual's income is above $50k$.
- Law School: The Law School dataset [Wightman 1998] from the law school admissions council's national longitudinal bar passage study to predict whether a candidate would pass the bar exam. It consists of law school admission records. We use the binary feature "isPassBar" as the target variable for classification.
- COMPAS: The COMPAS dataset [Angwin et al. 2016] for recidivism prediction consists of criminal records comprising offender's criminal history and demographic features (sex, race). We use the ground truth on whether the offender was re-arrested (binary) as the target variable for classification.

We transform all categorical attributes using one-hot encoding, and standardize all features vectors to have zero mean and unit variance. Python scripts for preprocessing the datasets are accessible along with the rest of the code of this chapter.

Dataset	Size	No. of features	Protected attributes	Protected groups	Prediction task
Census Income	40701	15	Race, Sex	{White, Black} × {Male, Female}	income ≥ 50k?
Law School	27479	12	Race, Sex	{White, Black} × {Male, Female}	Pass bar exam?
COMPAS	7215	11	Race, Sex	{White, Black} × {Male, Female}	recidivate in 2 years?

Table 3.1: Description of datasets

Baselines and Implementation In the experiments presented in Section 3.4 and 3.5, we use a standard feed-forward network to implement both *learner* and *adversary*. Our model for the learner is a fully connected two layer feed-forward network with 64 and 32 hidden units in the hidden layers, with $ReLU$ activation function. While our adversary is general enough to be a deep network, we observed that for the small academic datasets

used in our experiments, a *linear* adversary performed the best. Fig. 3.2 summarizes the computational graph of our proposed ARL approach. [2]

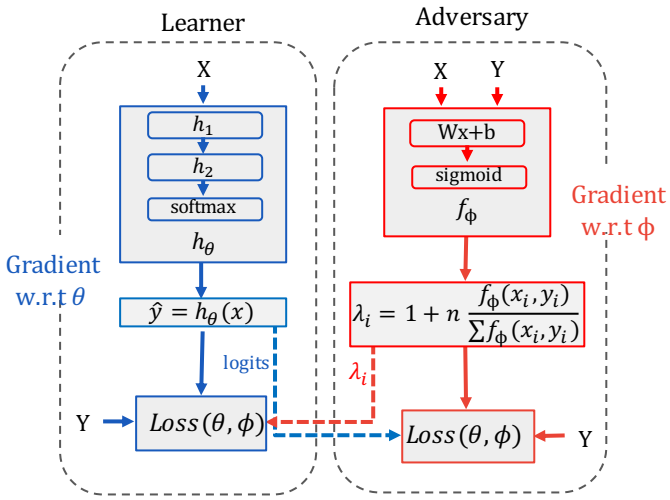


Figure 3.2: ARL's Computational Graph

We compare our proposed method *ARL* with the two group-agnostic (i.e., agnostic to protected group memberships) and two group-aware baselines. All approaches have the same DNN architecture, optimizer and activation functions. Following are the implementation details:

- **Baseline**: This is a vanilla *group-agnostic* baseline, which performs standard empirical risk minimization (ERM) with uniform example weights.
- **DRO**: Our main comparison is with the state-of-the-art method *DRO* [Hashimoto et al. 2018]. Similar to *ARL*, *DRO* is *group-agnostic*, and optimizes for Rawlsian Max-Min Fairness. We use the code shared by [Hashimoto et al. 2018].
- **IPW**: Inverse Probability Weighting (IPW) [Höfler et al. 2005] is a standard *group-aware* reweighting approach. IPW performs a weighted ERM with inverse probability weights $1/p(s)$, where $p(s)$ is the probability of observing an individual from group s. Additionally, we perform experiments on a variant of IPW called IPW (S+Y) with weights $1/p(s, y)$, where $p(s, y)$ is the joint probability of observing a data-point having membership to group s and class label y over empirical training distributions.
- **Min-Diff:** A group fairness approach with access to the protected attributes, which aims for *Equal Odds*, i.e., to minimize the difference (Min-diff) in group error rates. We use the code shared by the authors. As we are interested in improving performance for multiple subgroups at a time, we add one *Min-Diff* loss terms for each protected attribute (sex and race).

[2] The Python and Tensorflow implementation of the proposed method *ARL*, as well as all the baselines is available open-source at `https://github.com/google-research/google-research/tree/master/group_agnostic_fairness`

Experimental Setup and Parameter Tuning We use the same experimental setup and hyper-parameter tuning for all the methods. Each dataset is randomly split into 70% training and 30% test sets. Best hyper-parameter values for all approaches are chosen via grid-search by performing 5-fold cross validation on the training set optimizing for best *overall* AUC. We do not use protected group information for training or tuning.

For each approach, we choose the best learning-rate, and batch size by performing a grid search over an exhaustive hyper parameter space given by batch size (32, 64, 128, 256, 512) and learning rate (0.001, 0.01, 0.1, 1, 2, 5). We tune the hyper-parameters for *DRO* by performing grid search over the parameter space as reported in their paper. In addition to batch size, and learning rate, DRO [Hashimoto et al. 2018] has an additional fairness hyper-parameter η, which controls the performance for the worst-case group. In their paper, the authors present a specific hyperparameter tuning approach to choose the best value for η. Hence for the sake of fair comparison, we report results for two variants of DRO: (i) DRO, original approach with η tuned as detailed in their paper and (ii) DRO(auc) with η tuned to achieve best overall AUC performance. All results reported are averages across 10 independent runs (with different model parameter initialization) on an independent test set.

Evaluation Metrics We choose AUC (area under the ROC curve) as our utility metric as it is robust to class imbalance, i.e., unlike *Accuracy* it is not easy to receive high performance for trivial predictions. Further, it encompasses both *FPR* and *FNR*, and is threshold agnostic.

To evaluate fairness we stratify the test data by groups to compute AUC per protected group $s \in S$, and report the following aggregate metrics.

- AUC(avg): mean AUC across all the data points.
- AUC(min): minimum AUC over all protected groups $s \in S$.
- AUC(macro-avg): macro-average over all protected group AUCs.
- AUC(minority): AUC reported for the smallest protected group in the dataset.

For all metrics higher values are better. Values reported are averages over 10 runs. Note that the protected features are removed from the dataset, and are not used for training, validation or testing. The protected features are only used to compute subgroup *AUC* in order to evaluate fairness.

3.4.1 Main Results: Fairness without Demographics

Our main comparison is with *DRO* [Hashimoto et al. 2018], a *group-agnostic* distributionally robust optimization approach that optimizes for the worst-case subgroup. Additionally, we report results for the vanilla *group-agnostic Baseline*, which performs standard ERM with uniform weights. Tbl. 3.2 reports results based on average performance across runs, with the best average performance highlighted in bold. Detailed results for all protected groups in the dataset are reported in the Tbl. 3.3, 3.4, and 3.5. We make the following key observations:

ARL improves worst-case AUC: ARL outperforms DRO, and achieves best results for AUC (minority) for all datasets. We observe a 6.5 percentage point (pp) improvement over the baseline for Census Income, 0.8 pp for Law School, and 1.1 pp for COMPAS. Similarly,

ARL shows 2 pp and 1 pp improvement in AUC (min) over baseline for Census Income and Law School datasets respectively. For COMPAS dataset there is no notable difference in performance over baseline.It is worth noting that while ARL shows no notable utility gain or loss on COMPAS dataset, DRO shows a substantial drop in AUC for all groups. We believe this is due to DRO picking up on noisy outliers in the dataset as high loss examples.

These results are inline with our observations on computational-identifiability of protected groups (Tbl. 3.8) and robustness to label bias (Fig 3.3a) in Section 3.5. As we will later see, unlike Census Income and Law School datasets, protected-groups in COMPAS dataset are not computationally-identifiable. Further, ground-truth recidivism class labels in COMPAS dataset are known to be noisy [Eckhouse 2017]. We suspect that noisy data and biased ground-truth training labels play a role in the subpar performance of *ARL* and *DRO* for COMPAS dataset, as both these approaches are susceptible to performance degradation in the presence of noisy class labels (as they cannot differentiate between mistakes on correct vs noisy class labels) as we will later see in Section 3.5.2.

ARL improves overall AUC: Further, in contrast to the general expectation in fairness approaches, wherein utility-fairness trade-off is implicitly assumed, we observe that for Census Income and Law School datasets *ARL* in fact shows ~ 1 pp improvement in AUC (avg) and AUC (macro-avg). This is because *ARL*'s optimization objective of minimizing maximal loss is better aligned with improving overall *AUC*.

Dataset	Method	AUC avg	AUC macro-avg	AUC min	AUC minority
Census Income	Baseline	0.898	0.891	0.867	0.875
Census Income	DRO	0.874	0.882	0.843	0.891
Census Income	DRO (auc)	0.899	0.908	0.869	0.933
Census Income	ARL	**0.907**	**0.915**	**0.881**	**0.942**
Law School	Baseline	0.813	0.813	0.790	0.824
Law School	DRO	0.662	0.656	0.638	0.677
Law School	DRO (auc)	0.709	0.710	0.683	0.729
Law School	ARL	**0.823**	**0.820**	**0.798**	**0.832**
COMPAS	Baseline	**0.748**	**0.730**	**0.674**	0.774
COMPAS	DRO	0.619	0.601	0.572	0.593
COMPAS	DRO (auc)	0.699	0.678	0.616	0.704
COMPAS	ARL	0.743	0.727	0.658	**0.785**

Table 3.2: Main results: ARL vs DRO. Best results are in bold.

3.4.2 ARL vs Inverse Probability Weighting

Next, to better understand and illustrate the advantages of ARL over standard re-weighting approaches, we compare ARL with inverse probability weighting (*IPW*)[Höfler et al. 2005], which is the most common re-weighting choice used to address representational disparity

Method	AUC White	AUC Black	AUC Male	AUC Female	AUC White Male	AUC White Female	AUC Black Male	AUC Black Female
Baseline	0.894	0.919	0.882	0.882	0.867	0.881	0.914	0.875
DRO	0.869	0.908	0.848	0.902	0.843	0.901	0.897	0.891
DRO (auc)	0.894	0.931	0.873	0.928	0.869	0.925	0.909	0.933
ARL	**0.903**	**0.932**	**0.885**	**0.930**	**0.881**	**0.927**	**0.917**	**0.942**

Table 3.3: Census Income: values in the table are AUC (mean).

Method	AUC White	AUC Black	AUC Male	AUC Female	AUC White Male	AUC White Female	AUC Black Male	AUC Black Female
Baseline	0.799	0.828	0.816	0.808	0.805	**0.790**	0.824	**0.830**
DRO	0.639	0.668	0.658	0.668	0.638	0.638	0.677	0.662
DRO (auc)	0.687	0.733	0.709	0.710	0.691	0.683	0.729	0.737
ARL	**0.811**	**0.829**	**0.829**	**0.815**	**0.819**	**0.798**	**0.832**	0.825

Table 3.4: Law School: values in the table are AUC (mean).

Method	AUC White	AUC Black	AUC Male	AUC Female	AUC White Male	AUC White Female	AUC Black Male	AUC Black Female
Baseline	**0.713**	**0.753**	**0.749**	**0.717**	0.724	**0.674**	**0.738**	0.774
DRO	0.601	0.614	0.624	0.583	0.609	0.572	0.613	0.593
DRO (auc)	0.680	0.690	0.704	0.655	0.696	0.616	0.681	0.704
ARL	0.712	0.747	0.745	0.714	**0.725**	0.658	0.733	**0.785**

Table 3.5: COMPAS: values in the table are AUC (mean).

problems. Specifically, IPW performs a weighted ERM with example weights set as $1/p(s)$, where $p(s)$ is the probability of observing an individual from group s in the empirical training distribution. In addition to vanilla IPW, we also report results for a IPW variant with inverse probabilities computed jointly over protected-features S and class-label Y reported as IPW(S+Y). Tbl. 3.6 summarizes the results. We make following observations and key takeaways:

Firstly, observe that in spite of not having access to demographic features, *ARL* has comparable if not better results than both variants of the *IPW* on all datasets. This results shows that even in the absence of group labels, *ARL* is able to appropriately assign adversarial weights to improve AUC for protected groups.

Further, not only does ARL improve subgroup fairness, in most settings it even outperforms IPW, which has perfect knowledge of group membership. This result further highlights the strength of *ARL*. We observed that this is because unlike IPW, ARL does not equally upweight all examples from protected groups, but does so only if the model needs much more capacity to be classified correctly. We present evidence of this observation in Section 3.5.

dataset	method	AUC avg	AUC macro-avg	AUC min	AUC minority
Census Income	IPW(S)	0.897	0.892	0.876	0.883
Census Income	IPW(S+Y)	0.897	0.909	0.877	0.932
Census Income	ARL	**0.907**	**0.915**	**0.881**	**0.942**
Law School	IPW(S)	0.794	0.789	0.772	0.775
Law School	IPW(S+Y)	0.799	0.798	0.784	0.785
Law School	ARL	**0.823**	**0.820**	**0.798**	**0.832**
COMPAS	IPW(S)	**0.744**	**0.727**	**0.679**	0.759
COMPAS	IPW(S+Y)	0.727	0.724	0.678	0.764
COMPAS	ARL	0.743	**0.727**	0.658	**0.785**

Table 3.6: ARL vs Inverse Probability Weight

3.4.3 ARL vs Group-Fairness Approaches

Although our problem formulation is not the same as traditional group-fairness approaches - i.e., *ARL* is *group-agnostic* and seeks to improve Rawlsian Max-Min fairness - for the sake of completion, we compare *ARL* with a group fairness approach that aims for equal opportunity (EqOpp) [Hardt et al. 2016]. Amongst the many fairness methods that can achieve EqOpp [Hardt et al. 2016; Zafar et al. 2017c; Beutel et al. 2019], we choose *Min-Diff* [Beutel et al. 2019] as a comparison as it is the closest to *ARL* in terms of implementation and optimization. To ensure fair comparison we instantiate *Min-Diff* with similar neural architecture and model capacity as *ARL*. Further, as we are interested in performance for multiple protected groups, we add one *Min-Diff* loss term for each protected feature (sex and race). Tbl. 3.7 summarizes these results. We make the following observations:

dataset	method	AUC avg	AUC macro-avg	AUC min	AUC minority
Census Income	Baseline	0.898	0.891	0.867	0.875
Census Income	MinDiff	0.847	0.856	0.835	0.863
Census Income	ARL	**0.907**	**0.915**	**0.881**	**0.942**
Law School	Baseline	0.813	0.813	0.790	0.824
Law School	MinDiff	**0.826**	**0.825**	**0.805**	**0.840**
Law School	ARL	0.823	0.820	0.798	0.832
COMPAS	Baseline	**0.748**	**0.730**	**0.674**	0.774
COMPAS	MinDiff	0.730	0.712	0.645	0.748
COMPAS	ARL	0.743	0.727	0.658	**0.785**

Table 3.7: ARL vs Group-Fairness

Min-Diff improves gap but not worst-off group: True to its goal, Min-Diff decreases

the FPR gap between groups: FPR gap on sex is between 0.02 and 0.05, and FPR gap on race is between 0.01 and 0.19 for all datasets. However, this does not always lead to improved AUC for worst-off groups (observe AUC min and AUC minority). ARL substantially outperforms Min-Diff (by about 5 pp) for Census Income, and achieves comparable performance on Law School and COMPAS datasets.

This result highlights the intrinsic mismatch between fairness goals of group-fairness approaches vs the desire to improve performance for protected groups. We believe making models more inclusive by improving the performance for groups, not just decreasing the gap, is an important complimentary direction for fairness research.

Utility-Fairness Trade-off: Further, observe that Min-Diff incurs a 5 pp drop over baseline in overall AUC for Census Income dataset, and about 2 pp drop for COMPAS dataset. In contrast, as noted earlier *ARL* in fact shows an improvement in overall AUC for Census Income and Law School datasets. This result shows that unlike Min-Diff (or group fairness approaches in general) where there is an explicit utility-fairness trade-off, *ARL* achieves a better pareto allocation of overall and subgroup AUC performance. This is because the goal of *ARL*, which explicitly strives to improve the performance for protected groups is aligned better with achieving overall utility.

3.5 Analysis

In this section, we conduct analysis to gain insights into *ARL*. First, we verify our hypothesis that groups are *computationally-identifiable*. Next, we perform a synthetic study to investigate the robustness of *ARL* to distribution shifts between training and test datasets. Later, we investigate if the example weights learned by *ARL* are meaningful. Finally, we present extensions and variants of *ARL*, and provide further insights into ARL by varying the input to ARL's *adversary*.

3.5.1 Are Groups Computationally-identifiable?

First, we test our hypothesis that unobserved protected groups S are correlated with observed features X and class label Y. Thus, even when they are unobserved, they can be computationally-identifiable. We test this hypothesis by training a predictive model to infer S given X and Y. Tbl. 3.8 reports the predictive accuracy of a linear model.

We observe that Census Income and Law School datasets have significant correlations with unobserved protected groups, which can be adversarially exploited to computationally-identify protected-groups. In contrast, for the COMPAS dataset the protected groups are not as computationally-identifiable. As we saw earlier in Tbl. 3.2 and 3.5 these results align with ARL showing no gain or loss for COMPAS dataset, but improvements for Census Income and Law School.

	Census Income	Law School	COMPAS
Race	0.90	0.94	0.61
Sex	0.84	0.58	0.78

Table 3.8: Accuracy in predicting protected group memberships.

3.5.2 Robustness to Distribution Shifts

In this subsection, we perform additional experiments on a number of semi-synthetic datasets[3] to investigate robustness of ARL to distribution shifts. Specifically, we investigate robustness of the ARL and DRO methods to training data biases [Blum and Stangl 2020], such as bias in group sizes, i.e., the fraction of training examples from each group (*representation bias*) and bias due to noisy or incorrect ground-truth labels (*label bias*). We use the Census Income dataset and generate several semi-synthetic training sets with worst-case distributions (e.g., few training examples of "female" group) by re-sampling points from original training set. We then train our approaches on these worst-case training sets, and evaluate their performance on a fixed untainted original test set.

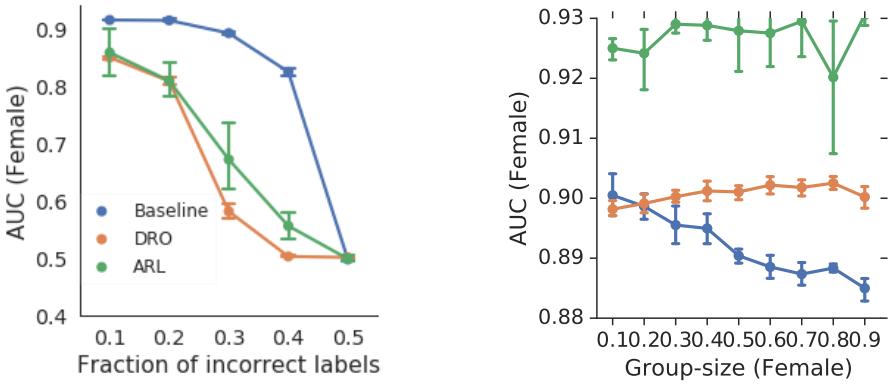
Concretely, to replicate *representation bias*, we vary the fraction of female examples in training set by under/over-sampling female examples from training set. Similarly, to replicate *label bias*, we vary the fraction of incorrect training labels by flipping the ground-truth class labels for a fraction of training data points uniformly at random. In all experiments, the size of the training set remains fixed. To mitigate the randomness in data sampling and optimization processes, we repeat the process 10 times and report results on a fixed untainted original test set (e.g., without adding label noise). Fig. 3.3 reports the results. For this experiment, we limit ourselves to the protected group "Female". For each training setting shown on the X-axis, we report the corresponding AUC for the "Female" subgroup on the Y-axis. The vertical bars in the plot are confidence intervals over 10 runs. We make the following observations:

Representation Bias: Both *DRO* and *ARL* are robust to the representation bias. *ARL* clearly outperforms *DRO* and *Baseline* at all points. Surprisingly, we see a drop in AUC for *Baseline* as the group-size increases. This is an artifact of having fixed training data size. As the fraction of female examples increases, we are forced to oversample female examples and downsample male examples; this leads to a decrease in the information present in training data and in turn leads to a worse performing model. In contrast, *ARL* and *DRO* cope better with this loss of information.

Label Bias: This experiment sheds interesting insights on the benefits of *ARL* over *DRO*. Recall that both approaches aim to focus on worst-case groups, however they differ in how these "groups" are formed. *DRO* is guaranteed to focus on worst-case risk for *any* group in the data exceeding size α. In contrast, ARL would only improve the performance for groups that are computationally-identifiable over (x, y).

We performed this experiment by setting the *DRO* hyperparameter α to 0.2. We observed

[3]The code to generate synthetic datasets is shared along with the rest of the code of this chapter.



(a) Label bias

(b) Representation bias

Figure 3.3: Robustness to distribution shifts between training and test distributions.

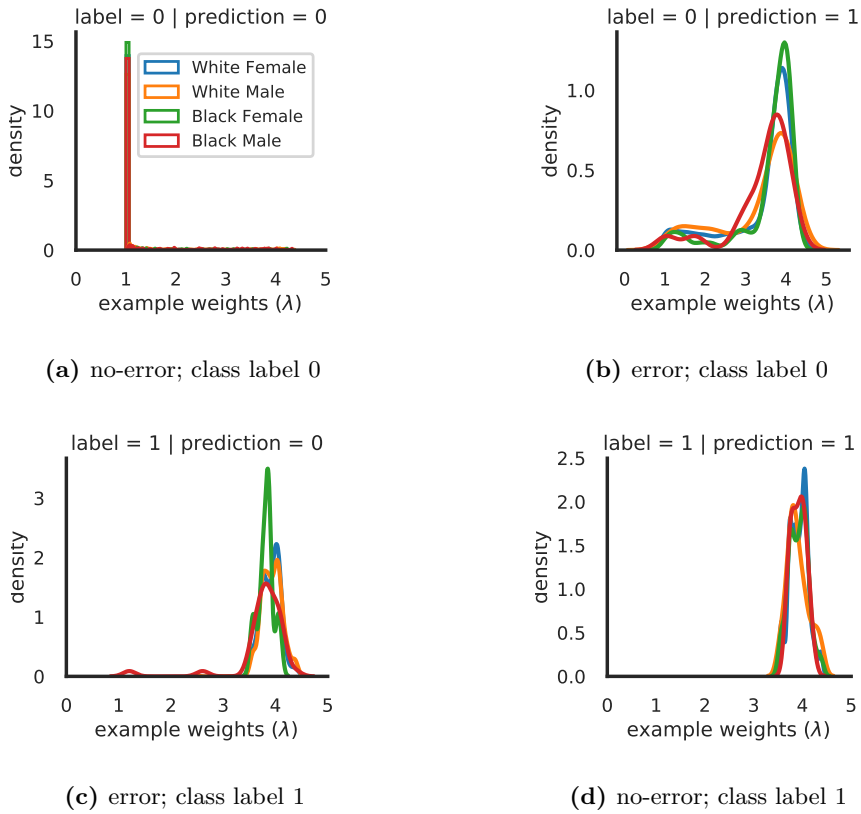
that while the fraction of incorrect ground truth class labels (i.e., outliers) is less than 0.2, the performance of both the methods is nearly the same. As the fraction of outliers in the training set exceeds 0.2 we observe that *DRO's* performance drops substantially. These results highlight that, as expected both the methods are sensitive to label noise (as they aim to up-weight examples with prediction error but cannot distinguish between true and noisy labels). However, as noted by Hashimoto et al. [2018], *DRO* becomes more sensitive to label noise once the noisy data points form a sizable fraction of training examples.

3.5.3 Are the learned Example Weights Meaningful?

Next, we investigate if the example weights learned by *ARL* are meaningful through the lense of training examples in the Census Income dataset. Fig. 3.4 visualizes the example weights assigned by *ARL* stratified into four quadrants of a confusion matrix. Each subplot visualizes the learned weights λ on x-axis and their corresponding density on y-axis. We make the following observations:

Misclassified examples are upweighted: As expected, misclassified examples are up-weighted (see Fig. 3.4b and 3.4c), whereas correctly classified examples are not upweighted (see Fig. 3.4a). Further, we observe that even though this was not our original goal, as an interesting side-effect *ARL* has also learned to address the class imbalance problem in the dataset. Recall that our Census Income dataset has class imbalance, and only 23% of examples belong to class 1. Observe that, in spite of making no errors *ARL* assigns high weights to all class 1 examples as shown in Fig. 3.4d (unlike in Fig. 3.4a where all class 0 example have weight 1).

ARL adjusts weights to base-rate: We smoothly vary the base-rate of female group in training data (i.e., we synthetically control fraction of female examples with class label 1 in training data). Fig. 3.5a visualizes training data base-rate on the x-axis and mean example weight learned for the subgroup on the y-axis. Observe that at female base-rate 0.1, i.e.,



(a) no-error; class label 0

(b) error; class label 0

(c) error; class label 1

(d) no-error; class label 1

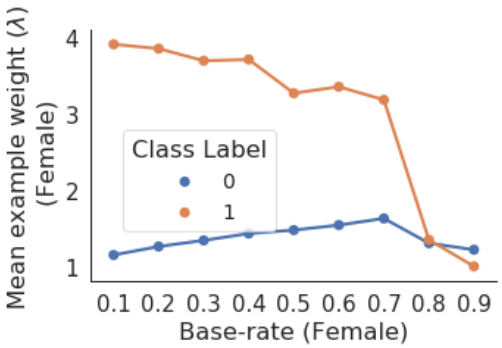
Figure 3.4: Example weights learned by ARL for four quadrants of a confusion matrix.

when only 10% of female training examples belong to class 1, the mean weight assigned for examples in class 1 is significantly higher than class 0. As the base-rate increases, i.e., as the number of class 1 examples increases, ARL correctly learns to decrease the weights for class 1 examples, and increases the weights for class 0 examples. These insights further explain the reason why *ARL* manages to improve overall AUC.

3.5.4 Further Extensions and Variants of ARL

Our proposed *ARL* approach is flexible and generalizes to many related works by varying the inputs to the adversary. In this subsection, our goal is to draw these connections, and gain further insights into ARL by performing experiments by varying the input to the *adversary*.

So far in this chapter, we operated under the assumption that protected features are not available in the dataset. However, in practice there are scenarios where protected features S are available but they are unknown to us. More concretely, we do not know a priori which subset of features amongst all features $X + S$ might be candidates for protected groups S. Examples of this setting include scenarios wherein a number of demographics features (e.g., age, race, sex) are present in the dataset. However, we do not know which group(s) amongst all the possible groups might need potential fairness treatment. Our proposed ARL approach naturally generalizes to this setting as well. The results for this setting are reported as the



(a) Base-rate vs example weights (λ).

ARL variant *ARL(adv: X+Y+S)*.

Similarly, there are many scenarios when protected features are known, and also specified upfront. However, there can be exponentially many intersectional groups (given by the cross-product over protected features). As a consequence, naively computing an exhaustive list of subgroups and explicitly enforcing subgroup fairness across all computable groups can be prohibitively expensive. Our proposed method *ARL* can be easily modified for this setting by varying the input variable to the adversary such that it takes only the protected features S. If the domain of our adversary $f_\phi(.)$ was S, i.e., it took only protected features as input, the regions that can be computationally-identified by the adversary boil down to all the possible intersectional subgroups, i.e., $Z \subseteq 2^S$. The resulting *ARL* objective in Eq. 3.5 would then reduce to minimizing the loss for the worst-off group amongst all *known* intersectional subgroups (similar to the subgroup fairness objective in Mohri et al. [2019]). The results for this setting are reported as the variants ARL(adv: S) and ARL(adv: S + Y).

Next, we perform experiments comparing the following variants of ARL:

- ARL (adv: X+Y) : vanilla *ARL* where the adversary takes non-protected features X and class label Y as input.
- ARL(adv: X+Y+S): a variant of *ARL* where the adversary takes all features $X + S$ and class label Y as input.
- ARL (adv: S): a variant of *ARL* where the adversary takes only protected features S as input.
- ARL (adv: S+Y): a variant of *ARL* with access to protected features S and class label Y as input.

A summary of results is reported in Tbl. 3.9. We make the following observations:

- Firstly, we observe that vanilla ARL without access to the protected features, i.e., *ARL (adv: X+Y)* is competitive, and sometimes even better than ARL variants with access to the protected features, i.e., *ARL (adv: S)*, *ARL (adv: S+Y)* and *ARL(adv: X+Y+S)* (except in the case of COMPAS dataset as observed earlier). These results

Dataset	Method	AUC	AUC macro-avg	AUC min	AUC minority
Census Income	Baseline	0.898	0.891	0.867	0.875
Census Income	ARL (adv: S)	0.900	0.894	0.875	0.879
Census Income	ARL (adv: S+Y)	**0.907**	0.907	**0.882**	0.907
Census Income	ARL (adv: X+Y+S)	**0.907**	0.911	0.881	**0.932**
Census Income	ARL (adv: X+Y)	**0.907**	**0.915**	0.881	0.942
Law School	Baseline	0.813	0.813	0.790	0.824
Law School	ARL (adv: S)	0.820	0.823	0.799	**0.846**
Law School	ARL (adv: S+Y)	0.824	**0.826**	0.801	0.845
Law School	ARL (adv: X+Y+S)	**0.826**	0.825	**0.808**	0.838
Law School	ARL (adv: X+Y)	0.823	0.820	0.798	0.832
COMPAS	Baseline	0.748	0.730	0.674	0.774
COMPAS	ARL (adv: S)	0.747	0.729	0.675	0.768
COMPAS	ARL (adv: S+Y)	0.747	**0.731**	**0.681**	0.771
COMPAS	ARL (adv: X+Y+S)	**0.748**	**0.731**	0.673	**0.778**
COMPAS	ARL (adv: X+Y)	0.743	0.727	0.658	0.785

Table 3.9: A comparison of variants of ARL

highlight the strength of ARL as an approach to achieve group fairness without access to protected features.

- Further, we observe that the performance of the ARL variants where protected features are explicitly specified upfront, i.e., *ARL(adv: Y+S)* and *ARL(adv: S)* is comparable to the variant where all the features and class labels are given as input to the adversary, i.e., *ARL(adv: X+Y+S)*. In certain cases (e.g., Census Income dataset), access to remaining features X even improves fairness. We believe this is because access to X helps the adversary to make fine-grained distinctions amongst a subset of disadvantaged candidates in a given group $s \in S$ that need fairness treatment.

- Finally, we observe that variants with class label (Y) generally outperform variants without class label. For instance, *ARL(S+Y)* has higher AUC than *ARL(S)* for all groups across all datasets, and the improvement is especially high for Census Income and Law School datasets, which have a class imbalance problem (observe base-rate in Tbl. 3.1). This is expected and can be explained as follows: variants without access to class label Y such as *ARL(S)* are forced to give the same weight to both positive and negative examples of a group. As a consequence, they do not cope well with differences in base-rates, particularly differences in base-rates across groups, as they cannot treat the positive and negative classes differently.

3.6 Conclusion

Improving group fairness in ML systems without directly observing protected features is a difficult and under-studied challenge for putting machine learning fairness goals into practice. The limited prior work has focused on improving model performance for any worst-case distribution, but as we show this is particularly vulnerable to noisy outliers. Our key insight is that when improving model performance for worst-case groups, it is valuable to focus the objective on *computationally-identifiable* regions of errors i.e., regions of the input and label space with significant errors. In practice, we find *ARL* is better at improving AUC for worst-case protected groups across multiple dataset and over multiple types of training data biases. As a result, we believe this insight and the *ARL* method provides a foundation for how to pursue fairness without access to protected demographic data.

Chapter 4

Learning Individually Fair Representations

Contents

People are rated, ranked and selected or not selected in an increasing number of applications based on machine learning (ML) models. Examples are approval or denial of loans or visas, ranking in job portals, or candidate selection for interviews. Research on how to incorporate fairness into ML tasks has prevalently pursued the paradigm of *group fairness*: balancing share of beneficial outcomes across protected groups. In contrast, the alternative paradigm of *individual fairness* has received relatively little attention. In this chapter we take a critical look at the group fairness paradigm, highlight its limitations, and advance the alternate paradigm of individual fairness. Our notion of individual fairness requires that users who are similar in all task-relevant attributes such as job qualification, and disregarding all potentially discriminating attributes such as gender should have similar outcomes. We cast this problem as a fair representation learning problem, and propose *iFair* (individually fair representations), an optimization approach for learning a low-rank latent representation of the data with two goals: to encode the data as well as possible, while removing any information about protected attributes in the transformed representation. Once we have learned a fair transformation: $\phi : X \rightarrow \tilde{X}$, one can freely train any unconstrained predictor (e.g., classifier or ranker) without having to worry about individual fairness in prediction outcomes. We demonstrate the versatility of our method by applying it to classification and learning-to-rank tasks on a variety of real-world and synthetic datasets. Our experiments show substantial improvements over the best prior work for this setting.

4.1 Introduction

We start from the observation that barring a few, a vast majority of the work in developing fair ML methods focuses on the *group fairness* notions, and narrowly considers a setting with a single binary protected attribute (e.g., gender: male vs female; race: African-american vs White). Typically, ML predictors are extended to incorporate protected demographic groups in their loss functions, e.g., as a regularization constraint to reflect legal boundary conditions and regulatory policies [Calders et al.; Kamiran et al. 2010; Kamishima et al.; Pedreschi et al. 2008; Feldman et al. 2015; Fish et al. 2016]. Most notably, the *statistical parity* notion of group fairness asks that the fraction of individuals from a protected group in the accepted class should be proportionate to their size in the population e.g., computing a shortlist of people invited for job interviews should have a gender mix that is proportional to the base population of job applicants. Other definitions of group fairness have been proposed [Hardt et al. 2016; Zafar et al. 2017b; Pleiss et al. 2017], and variants of group fairness have been applied to learning-to-rank tasks [Zehlike et al. 2017; Yang and Stoyanovich 2017; Singh and Joachims 2018]. In all these cases, fair classifiers or regression models need an explicit specification of a protected attribute such as race, and often the identification of a specific *protected (attribute-value) group* such as race equals African-American.

The Case for Individual Fairness Dwork et al. [2012b] argued that group fairness, while appropriate for policies regarding demographic groups, does not capture the goal of treating individual people in a fair manner. This led to the definition of *individual fairness*: similar individuals should be treated similarly. For binary classifiers, this means that individuals who are similar on the task-relevant attributes (e.g., job qualifications) should have nearly the same probability of being accepted by the classifier. This kind of fairness is intuitive and captures aspects that group fairness does not handle. Most importantly, it addresses potential discrimination of people by disparate treatment despite the same or similar qualifications (e.g., for loan requests, visa applications or job offers), and it can mitigate such risks.

The following example in Table 4.1 illustrates the points that a) individual fairness addresses situations that group fairness does not properly handle, and b) individual fairness must be carefully traded off against the utility of classifiers and rankings.

Example: Table 4.1 shows a real-world example for the issue of unfairness to individual people. Consider the ranked results for an employer's query "Brand Strategist" on the German job portal *Xing*; that data was originally used in Zehlike et al. [2017]. The top-10 results satisfy group fairness with regard to gender, as defined by Zehlike et al. [2017] where a top-k ranking τ is fair if for every prefix $\tau|i =< \tau(1), \tau(2), \cdots \tau(i) > (1 \leq i \leq k)$ the set $\tau|i$ satisfies *statistical parity* with statistical significance. However the outcomes in Table 4.1 are far from being fair for the individual users: people with very similar qualifications, such as Work Experience and Education Score ended up on ranks that are far apart (e.g., ranks 5 and 30). By the *position bias* [Joachims and Radlinski 2007] when searchers browse result lists, this treats the low-ranked people quite unfairly. This demonstrates that applications can satisfy group-fairness policies, while still being unfair to individuals.

Search Query	Work Experience	Education Experience	Candidate	Xing Ranking
Brand Strategist	146	57	male	1
Brand Strategist	327	0	female	2
Brand Strategist	502	74	male	3
Brand Strategist	444	56	female	4
Brand Strategist	139	25	male	5
Brand Strategist	110	65	female	6
Brand Strategist	12	73	male	7
Brand Strategist	99	41	male	8
Brand Strategist	42	51	female	9
Brand Strategist	220	102	female	10
		...		
Brand Strategist	3	107	female	20
Brand Strategist	123	56	female	30
Brand Strategist	3	3	male	40

Table 4.1: Top k results on `www.xing.com` (Jan 2017) for an employer's job search query "Brand Strategist".

Problem Statement: Unfortunately, the rationale for capturing individual fairness has not received much follow-up work – the most notable exception being Zemel et al. [2013] as discussed below. The current chapter advances the approach of individual fairness in its practical viability, and specifically addresses the key problem of coping with the critical trade-off between fairness and utility: *How can a data-driven system provide a high degree of individual fairness while also keeping the utility of classifiers and rankings high?* Is this possible in an application-agnostic manner, so that arbitrary downstream applications are supported? Can the system handle situations where protected attributes are not explicitly specified at all or become known only at decision-making time (i.e., after the system was trained and deployed)?

State of the Art and its Limitations: Prior work on fairness for ranking tasks has exclusively focused on group fairness disregarding the dimension of individual fairness. Biega et al. [2018] address individual fairness in rankings by giving fair exposure to individuals over a series of rankings. However, they explicitly assume to have access to scores that are already individually fair. As such, their work is complementary to ours as they do not address how such an individually fair score can be computed. For the restricted setting of binary classifiers, the most notable work on individual fairness is Zemel et al. [2013]. That work addresses the fundamental trade-off between utility and fairness by defining a combined loss function to learn a low-rank data representation. The loss function reflects a weighed sum of classifier accuracy, statistical parity for a single pre-specified protected group, and individual fairness in terms of reconstruction loss of data. This model, called *LFR*, is powerful and elegant, but has major limitations:

- It is geared for binary classifiers and does not generalize to a wider class of machine-learning tasks, dismissing regression models, i.e., learning-to-rank tasks.
- Its data representation is tied to a specific use case with a single protected group that

needs to be specified upfront. Once learned, the representation cannot be dynamically adjusted to different settings later.

- Its objective function strives for a compromise over three components: application utility (i.e., classifier accuracy), group fairness and individual fairness. This tends to burden the learning with too many aspects that cannot be reconciled.

Our approach overcomes these limitations by developing a model for representation learning that focuses on individual fairness and offers greater flexibility and versatility. Simple approaches like removing all protected attributes from the data and then performing a standard clustering technique do not reconcile these two conflicting goals, as standard clustering may lose too much utility and individual fairness needs to consider attribute correlations beyond merely masking the explicitly protected ones.

Approach and Contribution: The approach that we put forward in this chapter, called *iFair*, is to learn a generalized data representation that preserves the fairness-aware similarity between individual records while also aiming to minimize or bound the data loss. This way, we aim to reconcile individual fairness and application utility, and we intentionally disregard group fairness as an explicit criterion.

iFair resembles the model of Zemel et al. [2013] in that we also learn a representation via probabilistic clustering, using a form of gradient descent for optimization. However, our approach differs from Zemel et al. [2013] on a number of major aspects:

- iFair learns flexible and versatile representations, instead of committing to a specific downstream application like binary classifiers. This way, we open up applicability to arbitrary classifiers and support regression tasks (e.g., rating and ranking) as well.

- iFair does not depend on a pre-specified binary protected attribute. Instead, it supports multiple protected attributes where the "protected values" are known only at run-time after the application is deployed. For example, we can easily handle situations where the critical value for gender is female for some ranking queries and male for others.

- iFair does not consider any notion of group fairness in its objective function. This design choice relaxes the optimization problem, and we achieve much better utility with very good fairness in both classification and ranking tasks. Hard group fairness constraints, based on the downstream application requirements, can be enforced post-hoc by adjusting the outputs of iFair-based classifiers or rankings as demonstrated in subsection 4.4.4.

The novel contributions of iFair are: 1) the first method, to the best of our knowledge, that provides individual fairness for learning-to-rank tasks; 2) an application-agnostic framework for learning low-rank data representations that reconcile individual fairness and utility such that application-specific choices on sensitive attributes and values do not require learning another representation; 3) experimental studies with classification and regression tasks for downstream applications, empirically showing that iFair can indeed reconcile strong individual fairness with high utility.

4.2 Related Work

We now discuss work most closely related to *iFair*.

Fair Data Preprocessing: Among the three high level approaches for incorporating fairness in ML systems – (i) pre-processing, (ii) in-processing, and (iii) post-processing – iFair follows the pre-processing approach. This line of work uses a specific definition of fairness and proposes fair data pre-processing methods to transform the input data records into their fair representations. To this end, there are two general strategies: (i) Data Perturbations, and (ii) Learning Fair Representations

The first strategy consists of *de-biasing* the input data by appropriate preprocessing [Kamiran et al. 2010; Pedreschi et al. 2008; Feldman et al. 2015; Salimi et al. 2019]. This typically involves data perturbation such as modifying the value of protected attributes or class labels in the training data to satisfy certain fairness conditions, e.g., equal proportion of positive (negative) class labels in both protected and non-protected groups.

The second strategy consists of methods for learning fair data representations. This line of work is the most related to *iFair* in terms of technical mechanisms for operationalizing fairness (details follow).

Learning Fair Representations: Dwork et al. [2012b] gave the first definition of *individual fairness* and argued that *similar individuals should receive similar outcomes*. They further developed a theoretical framework for mapping individuals to a probability distribution over outcomes, which satisfies the Lipschitz property (i.e., distance preservation) in the mapping. In this chapter, we follow up on this definition of individual fairness and present a generalized framework for learning individually fair representations of the data.

The work of Zemel et al. [2013] is the closest to ours in that it is also learns low-rank representations by probabilistic mapping of data records. However, the methods deviates from our in important ways. First, its fair representations are tied to a particular classifier by assuming a binary classification problem with pre-specified labeling target attribute and a single protected group. In contrast, the representations learned by *iFair* are agnostic to the downstream learning tasks and thus easily deployable for new applications. Second, the optimization in Zemel et al. [2013] aims to combine three competing objectives: classifier accuracy, statistical parity, and data loss (as a proxy for individual fairness). The *iFair* approach, on the other hand, addresses a more streamlined objective function by focusing on classifier accuracy and individual fairness.

Approaches similar to Zemel et al. [2013] have been applied to learn censored representations for fair classifiers via adversarial learning [Louizos et al. 2016; Edwards and Storkey 2016; Madras et al. 2018]. These approaches, however, focus on group fairness and do not consider individual fairness at all.

Fairness in Ranking: Prior work on fairness in learning-to-rank tasks has primarily focused on group fairness. Yang and Stoyanovich [2017] introduced statistical parity in rankings. Zehlike et al. [2017] built on Yang and Stoyanovich [2017] and proposed to ensure statistical parity at all top-k prefixes of the ranked results. Singh and Joachims [2018] proposed a generalized fairness framework for a larger class of group fairness definitions (e.g., disparate treatment and disparate impact). However, all this prior work has focused on group fairness

alone. It implicitly assumes that individual fairness is taken care of by the ranking quality, disregarding situations where trade-offs arise between these two dimensions. The work of Biega et al. [2018] addresses individual fairness in rankings from the perspective of giving fair exposure to items over a series of rankings, thus mitigating the position bias in click probabilities. In their approach they explicitly assume to have access to scores that are already individually fair. As such, their work is complementary to ours as they do not address how such a score, which is individually fair can be computed.

Individual Fairness: Subsequent to our proposed method *iFair*, there has been some recent work on learning individually fair representations. Yurochkin et al. [2020] rely on Wasserstein distance as the fair distance metric and employ distributionally robust optimization to enforce individual fairness. Ruoss et al. [2020] propose learning certifiable individually fair representations by defining logical constraints. Some recent works [Ilvento 2020; Mukherjee et al. 2020; Gillen et al. 2018] including our own subsequent work [Wang et al. 2019] propose learning a distance metric from the data via metric-learning. An interesting line of work [Jung et al. 2019; Bechavod et al. 2020], including our own subsequent work [Lahoti et al. 2019a] operationalize individual fairness by elicting and modeling expert knowledge.

4.3 Model

We consider user records that are fed into a learning algorithm towards algorithm decision making. A fair algorithm should make its decisions solely based on non-protected attributes (e.g., technical qualification or education) and should disregard protected attributes that bear the risk of discriminating users (e.g., ethnicity/race). This dichotomy of attributes is specified upfront by domain experts, and follows legal regulations and policies. Ideally, one should consider also strong correlations (e.g., geo-area correlated with ethnicity/race), but this is usually beyond the scope of the specification. We start with introducing preliminary notations and definitions.

Input Data: The input data for n users with m attributes is an $n \times m$ matrix X with binary or numerical values (i.e., after unfolding or encoding categorical attributes). Without loss of generality, we assume that the attributes $1 \,..\, l$ are non-protected and the attributes $l+1 \,..\, m$ are protected. We denote the i-th user record consisting of all attributes as x_i and only non-protected attributes as x_i^*. Note that, unlike in prior works, the set of protected attributes is allowed to be empty (i.e., $l = m$). Also, we do not assume any upfront specification of which attribute values form a protected group. So a downstream application can flexibly decide on the critical values (e.g., male vs. female or certain choices of citizenships) on a case-by-case basis.

Output Data: The goal is to transform the input records x_i into representations $\tilde{x_i}$ that are directly usable by downstream applications and have better properties regarding fairness. Analogously to the input data, we can write the entire output of $\tilde{x_i}$ records as an $n \times m$ matrix $\tilde{X}$.

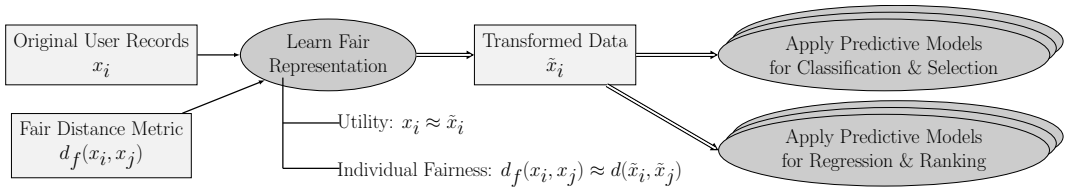


Figure 4.1: Overview of decision-making pipeline.

4.3.1 Problem Formulation

Individually Fair Representation: Inspired by the Dwork et al. [2012b] notion of individual fairness, *"individuals who are similar should be treated similarly"*, we propose *similar individuals* should be *indistinguishable* in their learned representation. We call such representations as *individually fair representations*. The overall decision-making pipeline is illustrated in Figure 4.1.

Definition 1. (Individually Fair Representation) *Given a* fairness-aware *distance metric* d_f*, a mapping* ϕ *of input records* x_i *into output records* $\tilde{x_i}$ *is individually fair if for every pair* x_i, x_j *we have*

$$|d_f(x_i, x_j) - d(\phi(x_i), \phi(x_j))| \leq \varepsilon \tag{4.1}$$

where d is a standard distance metric (e.g., Euclidean distance) in the transformed data space, and d_f is a *fairness-aware* distance metric in the original input space. The definition requires that individuals who are deemed similar according to some *fairness-aware* distance metric d_f should be mapped close-by in their transformed fair representations $\tilde{X}$. In more technical terms, a distance measure between user records should be preserved in the transformed space.

So far we have left the choice of the distance metric d_f open. Our methodology is general and can incorporate a wide suite of distance measures. However, for the actual optimization, we need to make a specific choice for d_f. In this chapter, we focus on the family of *Minkowski p-metrics*, which is indeed a metric for $p \geq 1$. A common choice is $p = 2$, which corresponds to a Gaussian kernel.

Definition 2. (Fairness-aware Distance Metric) *The distance between two data records* x_i, x_j *is*

$$d_f(x_i, x_j) = \big[\sum_{t=1}^{m} \alpha_t (x_{i,t} - x_{j,t})^p\big]^{1/p} \tag{4.2}$$

where α is an m-dimensional vector of tunable or learnable weights for the different data attributes.

The $m-$dimensional weight vector α controls the influence of each attribute. Our goal is to learn *fair* attribute weight vector α. A natural setting for fairness is to assign no weight to the protected attributes as these should not play any role in the similarity of (qualifications of) users, and near zero weights to attributes correlated to the protected attributes (e.g., zip-code vs race).

Utility Objective: Without making any assumptions on the downstream application, the best way of ensuring high utility is to minimize the data loss induced by ϕ.

Definition 3. (Data Loss) *The reconstruction loss between X and $\tilde{X}$ is the sum of squared errors*

$$L_{util}(X, \tilde{X}) = \sum_{i=1}^{n} ||x_i - \tilde{x}_i||_2 = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_{ij} - \tilde{x}_{ij})^2 \tag{4.3}$$

Individual Fairness Objective: Following the rationale in Definition 1, the desired transformation ϕ should preserve pair-wise distances between data records on non-protected attributes.

Definition 4. (Fairness Loss) *For input data X, with row-wise data records x_i and its transformed representation $\tilde{X}$ with row-wise $\tilde{x_i}$, the fairness loss L_{fair} is*

$$L_{fair}(X, \tilde{X}) = \sum_{i,j=1..m} \left(d_f(x_i, x_j) - d(\tilde{x_i}, \tilde{x_j})\right)^2 \tag{4.4}$$

4.3.2 Learning Fair Representations

As individual fairness needs to preserve similarities between records x_i, x_j, we cast the goal of computing good representations $\tilde{x_i}, \tilde{x_j}$ into a formal problem of *probabilistic clustering*. We aim for K clusters, each given in the form of a *prototype vector* v_k ($k = 1..K$), such that records x_i are assigned to clusters by a record-specific probability distribution that reflects the distances of records from prototypes. This can be viewed as a low-rank representation of the input matrix X with $k < m$, so that we reduce attribute values into a more compact form. As always with soft clustering, K is a hyper-parameter.

Definition 5. (Transformed Representation) *The fair representation $\tilde{X}$, an $n \times m$ matrix of row-vise output vectors $\tilde{x_i}$, consists of*

(i) $k < m$ prototype vectors v_k, each of dimensionality m,

(ii) a probability distribution u_i, of dimensionality k, for each input record x_i where u_{ik} is the probability of x_i belonging to the cluster of prototype v_k.

The transformed representation $\phi : x_i \rightarrow \tilde{x_i}$ is given by

$$\tilde{x_i} := \phi(x_i) = \sum_{k=1..K} u_{ik} \cdot v_k \tag{4.5}$$

or equivalently in matrix form: $\tilde{X} = U \times V^T$ where the rows of U are the per-record probability distributions and the columns of V^T are the prototype vectors.

Definition 6. (Probability Vector) *The probability vector u_i for record x_i is*

$$u_{i,k} = \frac{exp(-d_f(x_i, v_k))}{\sum_{j=1}^{K} exp(-d_f(x_i, v_j))} \tag{4.6}$$

The transformed representation $\tilde{x_i}$ given by the mapping $\phi : x_i \rightarrow \tilde{x_i}$ can be written as

$$\tilde{x_i} := \sum_{k=1..K} u_{ik} \cdot v_k = \sum_{k=1}^{K} \frac{exp(-d_f(x_i, v_k))}{\sum_{j=1}^{K} exp(-d_f(x_i, v_j))} \cdot v_k \tag{4.7}$$

The distance function d_f is applicable to original data records x_i, and prototype vectors v_k alike. In our model, we avoid the quadratic number of comparisons for all pairs $d_f(x_i, x_j)$ by instead computing distances only between records and prototype vectors $d_f(x_i, v_k)$ in Eq. 4.7 (cf. also Zemel et al. [2013]).

4.3.3 Optimization Problem

With these definitions in place, the task of learning fair representations $\tilde{X}$ now amounts to computing K prototype vectors v_k and the m-dimensional weight vector α in d_f such that the overall loss function L is minimized.

Definition 7. (Optimization Objective) *The optimization objective is to compute prototype vectors v_k $(k = 1..K)$ and fair attribute weights α_j $(j = 1..m)$ as argmin for the loss function*

$$\begin{aligned} L &= \lambda \cdot L_{util}(X, \tilde{X}) \quad + \quad \mu \cdot L_{fair}(X, \tilde{X}) \\ &= \lambda \cdot \sum_{i=1}^{M} \sum_{j=1}^{N} (x_{ij} - \tilde{x}_{ij})^2 + \mu \cdot \sum_{i,j=1..M} \left(d(\tilde{x_i}, \tilde{x_j}) - d_f(x_i, x_j)\right)^2 \end{aligned}$$

where L_{util} is the data loss, L_{fair} is the loss in individual fairness, and λ and μ are the corresponding hyper-parameters controlling the trade-off. Values for $\tilde{x}_{ij}$ and d_f are substituted using Equations 4.7 and 4.2.

Gradient Descent Optimization: Given this setup, the learning system minimizes the combined objective function given by

$$L = \lambda \cdot L_{util}(X, \tilde{X}) + \mu \cdot L_{fair}(X, \tilde{X}) \tag{4.8}$$

where L_{util} is the data loss, L_{fair} is the loss in individual fairness, and λ and μ are the hyper-parameters controlling the trade-off. We have two sets of *model parameters* to learn

(i) v_k $(k = 1..K)$, the $m-$dimensional prototype vectors,

(ii) α, the $m-$dimensional weight vector of the distance metric d_f in Equation 4.2.

We apply the *L-BFGS* algorithm [Liu and Nocedal 1989], a quasi-Newton method, to minimize Equation 4.8 and learn the model parameters.

We initialize the weight vector α with (near-)zero values to the protected attributes to reflect the intuition that protected attributes should be discounted in the distance-preservation of individual fairness (and avoiding zero values to allow slack for the numerical

computations in learning the model). The information bottleneck caused by the low-rank clustering encourages the model to map correlated attributes to the same prototypes v_k in order to minimize the data-loss, while the fairness objective encourages the model to assign low-weights to features correlated to the protected attributes in order to preserve the pairwise distances. In our experiments, we observe that indeed initializing the weight vector α with (near-)zero weights to the protected attributes increases the fairness of the learned data representations as opposed to randomly initialization the weight vector with values in $[0, 1]$ (see Section 4.4).

4.4 Experiments

The key hypothesis that we test in the experimental evaluation is whether *iFair* can indeed reconcile the two goals of *individual fairness* and *utility* reasonably well. As *iFair* is designed as an application-agnostic representation, we test its versatility by studying both classifier and learning-to-rank use cases, in subsections 4.4.1 and 4.4.2, respectively. We compare *iFair* to a variety of baselines including LFR [Zemel et al. 2013] for classification and FA*IR [Zehlike et al. 2017] for ranking. Although group fairness is not among the design goals of our approach, we include group fairness measures in reporting on our experiments – shedding light into this aspect from an empirical perspective.

Datasets and Pre-processing We apply the *iFair* framework to five publicly available real-world datasets, previously used in the literature on algorithmic fairness.

- **ProPublica's COMPAS** recidivism dataset [Angwin et al. 2016], a widely used test case for fairness in machine learning and algorithmic decision making. We set *race* as a protected attribute, and use the binary indicator of recidivism as the outcome variable Y.
- **Census Income** dataset consists of survey results of income of 48,842 adults in the US [Dheeru and Karra Taniskidou 2017]. We use gender as the protected attribute and the binary indicator variable of $income > 50K$ as the outcome variable Y.
- **German Credit** data has 1000 instances of credit risk assessment records [Dheeru and Karra Taniskidou 2017]. Following the literature, we set *age* as the sensitive attribute, and *credit worthiness* as the outcome variable.
- **Airbnb** data consists of house listings from five major cities in the US, collected from `http://insideairbnb.com/get-the-data.html` (June 2018). After appropriate data cleaning, there are $27,597$ records. For experiments, we choose a subset of 22 informative attributes (categorical and numerical) and infer host gender from *host name*, using lists of common first names. We use *gender* of the host as the protected attribute and *rating/price* as the ranking variable.
- **Xing** is a popular job search portal in Germany (similar to LinkedIn). We use the anonymized data given by Zehlike et al. [2017], consisting of top 40 profiles returned

for 57 job queries. For each candidate we collect information about job category, work experience, education experience, number of views of the person's profile, and gender. We set *gender* as the protected attribute. We use a weighted sum of work experience, education experience and number of profile views as a score that serves as the ranking variable.

The Compas, Census and Credit datasets are used for experiments on classification, and the Xing and Airbnb datasets are used for experiments on learning-to-rank regression. Table 4.2 gives details of experimental settings and statistics for each dataset, including base-rate (fraction of samples belonging to the positive class, for both the protected group and its complement), and dimensionality m (after unfolding categorical attributes). We choose the protected attributes and outcome variables to be in line with the literature. In practice, however, such decisions would be made by domain experts and according to official policies and regulations. The flexibility of our framework allows for multiple protected attributes, multivariate outcome variable, as well as inputs of all data types.

Table 4.2: Experimental settings and statistics of the datasets.

Dataset	Base-rate protected	Base-rate unprotected	n	m	Outcome	Protected
Compas	0.52	0.40	6901	431	recidivism	race
Census	0.12	0.31	48842	101	income	gender
Credit	0.67	0.72	1000	67	loan default	age
Airbnb	-	-	27597	33	rating/price	gender
Xing	-	-	2240	59	work + education	gender

Baselines and Implementation In each dataset, categorical attributes are transformed using one-hot encoding, and all features vectors are normalized to have unit variance. We randomly split the datasets into three parts. We use one part to train the model to learn model parameters, the second part as a validation set to choose hyper-parameters by performing a grid search (details follow), and the third part as a test set. We use the same data split to compare all methods.

We evaluate all data representations – *iFair* against various baselines – by comparing the results of a standard classifier (*logistic regression*) and a learning-to-rank regression model (*linear regression*) applied to

- **Full Data**: the original dataset.
- **Masked Data**: the original dataset without protected attributes.
- **SVD**: transformed data by performing dimensionality reduction via singular value decomposition (SVD) [Halko et al. 2011], with two variants of data: (a) full data and (b) masked data. We name these variants **SVD** and **SVD-masked**, respectively.

- **LFR**: the learned representation by the method of Zemel et al. [2013].
- **FA*IR**: this baseline does not produce any data representation. FA*IR [Zehlike et al. 2017] is a ranking method which expects as input a set of candidates ranked by their *deserved scores* and returns a ranked permutation which satisfies group fairness at every prefix of the ranking. We extended the code shared by Zehlike et al. [2017] to make it suitable for comparison (see Section 4.4.2).
- **iFair**[1]: the representation learned by our model. We perform experiments with two kinds of initializations for the model parameter α (attribute weight vector): (a) random initialization in $(0, 1)$ and (b) initializing protected attributes to (near-)zero values, to reflect the intuition that protected attributes should be discounted in the distance-preservation of individual fairness (and avoiding zero values to allow slack for the numerical computations in learning the model). We call these two methods **iFair-a** and **iFair-b**, respectively.

Experimental Setup and Parameter Tuning We use the same experimental setup and hyper-parameter tuning for all the methods. We initialize model parameters (v_k vectors and the α vector) to random values from uniform distribution in $(0, 1)$ (unless specified otherwise, for the *iFair-b* method). To compensate for variations caused due to initialization of model parameters, for each method and at each setting, we report the results from the best of 3 runs.

As for hyper-parameters (e.g., λ and μ in Equation 4.8 of *iFair*), including the dimensionality K of the low-rank representations, we perform a grid search over the set $\{0, 0.05, 0.1, 1, 10, 100\}$ for mixture coefficients and the set $\{10, 20, 30\}$ for the dimensionality K. Recall that the input data is pre-processed with categorical attributes unfolded into binary attributes; hence the choices for K. The mixture coefficients $(\lambda, \mu, \dots)$ control the trade-off between different objectives: utility, individual fairness, group fairness (when applicable). Since it is all but straightforward to decide which of the multiple objectives is more important, we choose these hyper-parameters based on different choices for the optimization goal (e.g., maximize utility alone or maximize a combination of utility and individual fairness). Thus, our evaluation results report multiple observations for each model, depending on the goal for tuning the hyper-parameters. When possible, we identify Pareto-optimal choices with respect to multiple objectives; that is, choices that are not consistently outperformed by other choices for all objectives.

Evaluation Metrics

- **Utility:** measured as accuracy (Acc) and the area under the ROC curve (AUC) for the classification task, and as Kendall's Tau (KT) and mean average precision at 10 (MAP) for the learning-to-rank task.
- **Individual Fairness:** measured as the *consistency* of the outcome $\hat{y}_i$ of an individual with the outcomes of his/her k=10 nearest neighbors. This metric has been introduced

[1]The Python implementation of our proposed method *iFair* is available open source at https://github.com/plahoti-lgtm/iFair

by Zemel et al. [2013] and captures the intuition that similar individuals should be treated similarly. Note that nearest neighbors of an individual, $kNN(x_i)$, are computed on the original attribute values x_i excluding protected attributes, whereas the predicted response variable $\hat{y}_i$ is computed on the output of the learned representations $\tilde{x}_i$.

$$\text{yNN} = 1 - \frac{1}{M} \cdot \frac{1}{k} \cdot \sum_{i=1}^{M} \sum_{j \in kNN(x_i^*)} |\hat{y}_i - \hat{y}_j|$$

- **Group Fairness:** measured as
  - *Equality of Opportunity* (*EqOpp*): One minus the difference in the *True Positives* rates between the the protected group X^+ and the non-protected group X^-
  - *Statistical Parity* defined as:

$$Parity = 1 - |\frac{1}{|X^+|} \sum_{i \in X^+} \hat{y}_i - \frac{1}{|X^-|} \sum_{j \in X^-} \hat{y}_j|$$

We use the notion of *EqOpp* as our primary metric of group fairness, but report the measure of *Parity* as well.

4.4.1 Evaluation on Classification Task

This section evaluates the effectiveness of *iFair* and its competitors on a classification task. We focus on the utility-(individual)fairness tradeoff that learned representations alleviate when used to train classifiers. For all methods, wherever applicable, hyper-parameters were tuned via grid search. Specifically, we chose the models that were Pareto-optimal with regard to AUC and yNN.

Results: Figure 4.2 shows the result for all methods and datasets, plotting utility (AUC) against individual fairness (yNN). The dotted lines show models that are Pareto-optimal with regard to AUC and yNN. We observe that there is a considerable amount of unfairness in the original dataset, which is reflected in the results of *Full Data* in Figure 4.2. *Masked Data* and the two SVD variants show an improvement in fairness; however, there is still substantial unfairness hidden in the data in the form of correlated attributes. For the Compas dataset, which is the most difficult of the three datasets due to its dimensionality, SVD completely fails. The representations learned by *LFR* and *iFair* dominate all other methods in coping with the trade-off. *iFair-b* is the overall winner: it is consistently Pareto-optimal for all three datasets and all but the degenerate extreme points. For the extreme points in the trade-off spectrums, no method can achieve near-perfect utility without substantially losing fairness and no method can be near-perfectly fair without substantially losing utility.

Table 4.3 shows detailed results for three choices of tuning hyper-parameters (via grid search): (a) considering utility (AUC) only, (b) considering individual fairness (yNN) only, (e) using the harmonic mean of utility and individual fairness as tuning target. Here we focus on the *LFR* and *iFair* methods, as the other baselines do not have hyper-parameters to control trade-offs and are good only at extreme points of the objective space anyway. The results confirm and further illustrate the findings of Figure 4.2. The two *iFair* methods,

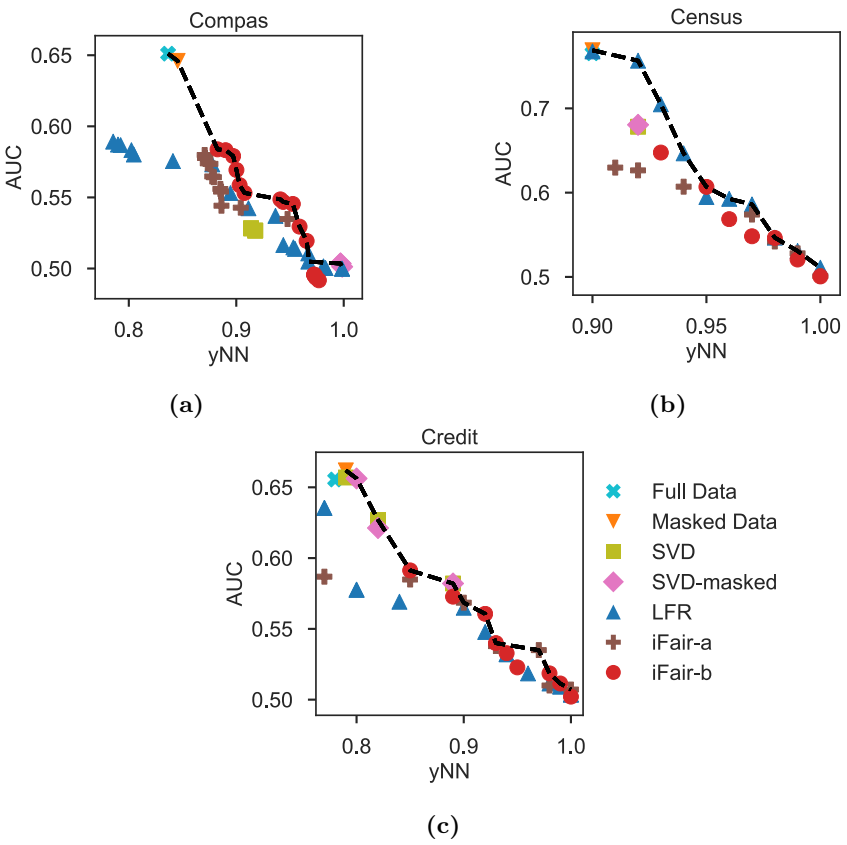


Figure 4.2: Utility vs. individual fairness trade-off for classification task. Dashed lines represent Pareto-optimal points.

Tuning	Method	Compas					Census					Credit				
		Acc	AUC	EqOpp	Parity	yNN	Acc	AUC	EqOpp	Parity	yNN	Acc	AUC	EqOpp	Parity	yNN
Baseline	Full Data	0.66	0.65	0.70	0.72	0.84	0.84	0.77	0.90	0.81	0.90	0.74	0.66	0.82	0.81	0.78
Max	LFR	**0.60**	**0.59**	0.60	0.62	0.79	**0.81**	**0.77**	0.81	0.75	0.90	0.71	**0.64**	0.78	0.77	0.77
Utility	iFair-a	**0.60**	0.58	**0.91**	**0.91**	0.87	0.78	0.63	**0.96**	**0.91**	0.91	0.69	0.61	0.84	0.86	0.74
(a)	iFair-b	0.59	0.58	0.84	0.84	**0.88**	0.78	0.65	0.78	0.85	**0.93**	**0.73**	0.59	**0.97**	**0.98**	**0.85**
Max	LFR	0.54	0.51	**0.99**	0.99	**0.97**	**0.76**	0.51	**1.00**	0.99	**1.00**	0.72	0.51	**0.99**	0.98	0.98
Fairness	iFair-a	**0.56**	**0.53**	0.97	0.99	0.95	**0.76**	0.51	0.95	**1.00**	0.99	**0.73**	**0.53**	**0.99**	0.98	0.97
(b)	iFair-b	0.55	0.52	0.98	**1.00**	**0.97**	**0.76**	**0.52**	0.98	0.99	0.99	0.72	0.51	**0.99**	**1.00**	**0.99**
	LFR	0.59	0.57	0.72	0.77	0.88	**0.78**	**0.76**	**0.94**	0.74	0.92	0.71	**0.64**	0.78	0.77	0.77
Optimal	iFair-a	**0.60**	**0.58**	**0.91**	**0.91**	0.87	0.77	0.63	0.93	**0.90**	0.92	**0.73**	0.57	0.94	0.94	**0.90**
(c)	iFair-b	0.59	**0.58**	0.83	0.84	**0.89**	**0.78**	0.65	0.78	0.85	**0.93**	**0.73**	0.59	**0.97**	**0.98**	0.85

Table 4.3: A comparison of IFR vs iFair for Classification task, with hyper parameter tuning for criterion (a) max utility: best AUC (b) best Individual Fairness: best consistency, and (c) "Optimal": best harmonic mean of AUC and consistency.

tuned for the combination of utility and individual fairness (case (c)), achieve the best overall results: iFair-b shows an improvement of 6 percent in consistency, for a drop of 10 percent in Accuracy for Compas dataset. (+3.3% and -7% for Census, and +9% and

-1.3% for Credit). Both variants of *iFair* outperform *LFR* by achieving significantly better individual fairness, with on-par or better values for utility.

4.4.2 Evaluation on Learning-to-Rank Task

This section evaluates the effectiveness of *iFair* on a regression task for ranking people on Xing and Airbnb dataset. We report ranking utility in terms of Kendall's Tau (KT), average precision (AP), individual fairness in terms of consistency (yNN) and group fairness in terms of fraction of protected candidates in top-10 ranks (statistical parity equivalent for ranking task). To evaluate models in a real world setting, for each dataset we constructed multiple queries and corresponding ground truth rankings. In case of Xing dataset we follow Zehlike et al. [2017] and use the 57 job search queries. For Airbnb dataset, we generated a set of queries based on attributes values for *city*, *neighborhood* and *home type*. After filtering for queries which had at least 10 listings we were left with 43 queries.

As stated in the beginning of Section 4.4, for the Xing dataset, the deserved score is a weighted sum of the true qualifications of an individual, i.e., work experience, education experience and the number of profile views. To test the sensitivity of our results for different choices of weights, we varied the weights over a grid of values in $[0.0, 0.25, 0.5, 0.75, 1.0]$. We observe that the choice of weights has no significant effect on the measurs of interest. Table 4.4 shows details. For the remainder of this section, the reported results correspond to uniform weights.

Weights			Base-rate	MAP	KT	yNN	% Protected
w_{work}	w_{edu}	w_{views}	Protected				in output
0.00	0.50	1.00	33.57	0.76	0.58	1.00	31.07
0.25	0.75	0.00	33.57	0.83	0.69	0.95	35.54
0.50	1.00	0.25	32.68	0.74	0.56	1.00	31.07
0.75	0.00	0.50	32.68	0.75	0.55	1.00	31.07
0.75	0.25	0.00	31.25	0.84	0.74	0.96	33.57
1.00	0.25	0.75	32.86	0.75	0.56	1.00	31.07
1.00	1.00	1.00	32.68	0.76	0.57	1.00	31.07

Table 4.4: Sensitivity of iFair to weights in ranking scores for Xing dataset.

Note that the baseline *LFR* used for the classification experiment, is not geared for regression tasks and thus omitted here. Instead, we compare *iFair* against the *FA*IR* method of Zehlike et al. [2017], which is specifically designed to incorporate group fairness into rankings.

Baseline FA*IR: This ranking method takes as input a set of candidates ranked according to a precomputed score, and returns a ranked permutation which satisfies group fairness without making any changes to the scores of the candidates. Here is a summary of the algorithm: First, we feed masked data to a linear regression model and compute a score for each candidate. We then give this candidate set as input to FA*IR algorithm. FA*IR operates on two priority queues (sorted by previously computed scores): P_0 for non-protected candidates and P_1 for protected candidates. For each rank k, it computes the minimum

number of protected candidates required to satisfy statistical parity (via significance tests) at position k. If the parity constraint is satisfied, it chooses the best candidate and its score from $P_0 \cup P_1$. If the constraint is not satisfied, it chooses the best candidate from P1 for the next rank and leaves a placeholder for the score. Since one cannot measure consistency directly on rankings, we make a minor modification to FA*IR such that it also returns fair scores along with a fair ranking. Our extension linearly interpolates the scores to fill the placeholders, and thus returns a ranked list along with "fair scores".

Results: Table 4.5 shows a comparison of experimental results for the ranking task for all methods across all datasets. We report mean values of average precision (MAP), Kendall's Tau (KT) and consistency (yNN) over all 57 job search queries for Xing and 43 house listing queries for Airbnb. Similar to the classification task, *Full Data* and *Masked Data* have the best utility (MAP and KT), whereas iFair has the best individual fairness (yNN). iFair clearly outperforms both variants of SVD by achieving significantly better individual fairness (yNN) for comparable values of utility. As expected, *FA*IR*, which optimizes to satisfy statistical parity across groups, has the highest fraction of protected candidates in the top 10 ranks, but does not achieve any gains on individual fairness. This is not surprising, though, given its design goals. It also underlines our strategic point that individual fairness needs to be explicitly taken care of as a first-order objective. Between *FA*IR* and *iFair*, there is no clear winner, given their different objectives. We note, though, that the good utility that *FA*IR* achieves in some configurations critically hinges on the choice of the value for its parameter p.

Dataset	Method	MAP (AP@10)	KT (mean)	yNN (mean)	% Protected in top 10
Xing (57 queries)	Full Data	**1.00**	**1.00**	0.93	32.50
	Masked Data	**1.00**	**1.00**	0.93	32.68
	SVD	0.74	0.59	0.81	31.79
	SVD-masked	0.67	0.50	0.78	32.86
	FA*IR (p = 0.5)	0.93	0.94	0.92	38.21
	FA*IR (p = 0.9)	0.78	0.78	0.85	**48.57**
	iFair-b	0.76	0.57	**1.00**	31.07
Airbnb (43 queries)	Full Data	**0.68**	**0.53**	0.72	47.44
	Masked Data	0.67	0.53	0.72	47.44
	SVD	0.66	0.49	0.73	48.37
	SVD-masked	0.66	0.49	0.73	48.37
	FA*IR (p = 0.5)	0.67	0.52	0.72	48.60
	FA*IR (p = 0.6)	0.65	0.51	0.73	**51.16**
	iFair-b	0.60	0.45	**0.80**	49.07

Table 4.5: Experimental results for ranking task. Reported values are means over multiple query rankings for the criterion "Optimal": best harmonic mean of MAP and yNN.

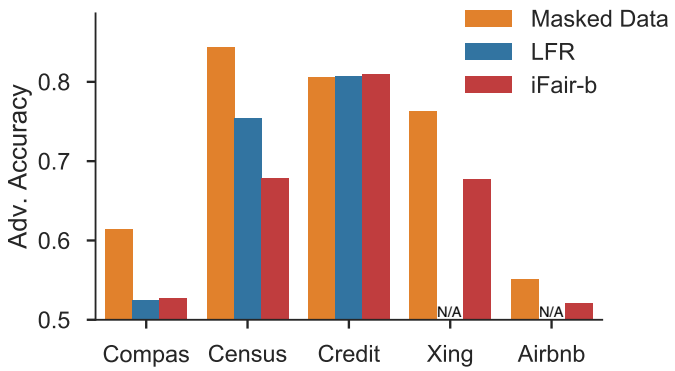


Figure 4.3: Adversarial accuracy of predicting protected group membership. Lower values are better.

4.4.3 Obfuscating Protected Information

We also investigate the ability of our model to obfuscate information about protected attributes. A reasonable proxy to measure the extent to which protected information is still retained in the *iFair* representations is to predict the value of the protected attribute from the learned representations. We trained a logistic-regression classifier to predict the protected group membership from: (i) Masked Data (ii) learned representations via *LFR*, and (iii) learned representations via *iFair-b*.

Results: Figure 4.3 shows the adversarial accuracy of predicting the protected group membership for all 5 datasets (with LFR not applicable to Xing and Airbnb). For all datasets, *iFair* manages to substantially reduce the adversarial accuracy. This signifies that its learned representations contain little information on protected attributes, despite the presence of correlated attributes. In contrast, *Masked Data* still reveals enough implicit information on protected groups and cannot prevent the adversarial classifier from achieving fairly good accuracy.

Relation to Group Fairness: Consider the notions of group fairness defined in Section 4.4. Statistical parity requires the probability of predicting positive outcome to be independent of the protected attribute: $P(\hat{Y} = 1|S = 1) = P(\hat{Y} = 1|S = 0)$. Equality of opportunity requires this probability to be independent of the protected attribute conditioned on the true outcome Y: $P(\hat{Y} = 1|S = 1, Y = 1) = P(\hat{Y} = 1|S = 0, Y = 1)$. Thus, forgetting information about the protected attribute indirectly helps improving group fairness; as algorithms trained on the individually fair representations carry largely reduced information on protected attributes. Subsequent work by Binns [2020] supports this argument and carefully draws a connection between group fairness and individual fairness.

This observation is supported by our empirical results on group fairness for all datasets. In Table 4.3, although group fairness is not an explicit goal, we observe substantial improvements by more than 10 percentage points; the performance for other datasets is similar.

However, the extent to which *iFair* also benefits group fairness criteria depends on the base rates $P(Y = 1|S = 1)$ and $P(Y = 1|S = 0)$ of the underlying data. Therefore, in applications where statistical parity is a legal requirement, additional steps are needed, as discussed next.

4.4.4 Enforcing Group Fairness in Downstream Task

By its application-agnostic design, it is fairly straightforward to enhance *iFair* by post-processing steps to enforce statistical parity, if needed. Obviously, this requires access to the values of protected attributes, but this is the case for most group fairness methods.

We demonstrate the extensibility of our framework by applying the *FA*IR* [Zehlike et al. 2017] technique as a post-processing step to the *iFair* representations of the Xing and Airbnb data. For each dataset, we generate top-k rankings by varying the target minimum fraction of protected candidates (parameter p of the FA*IR algorithm). Figure 4.4 reports ranking utility (MAP), percentage of protected candidates in top 10 positions, and individual fairness (yNN) for increasing values of the FA*IR parameter p.

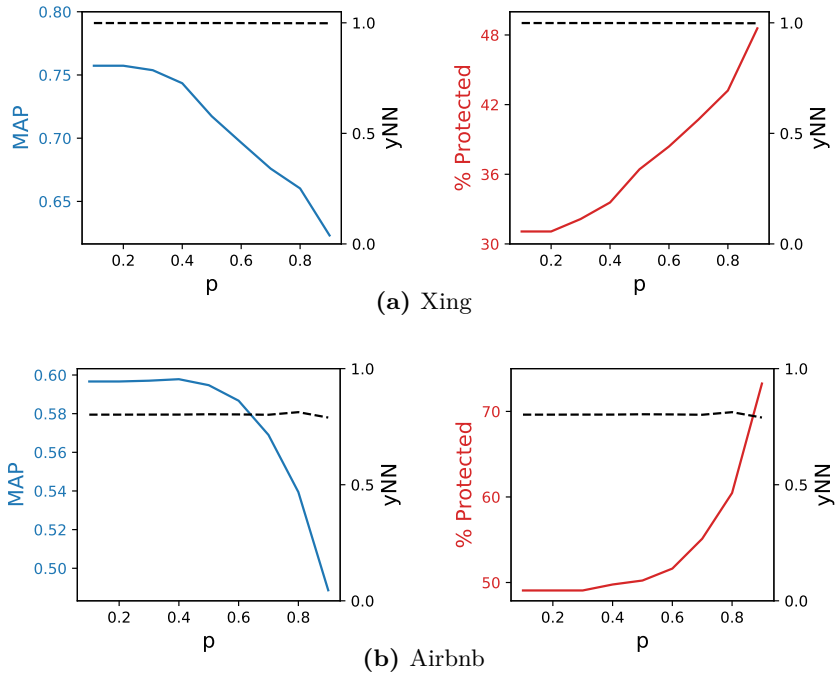


Figure 4.4: Applying FA*IR algorithm to *iFair* representations.

The key observation is that the combined model *iFair + FA*IR* can indeed achieve whatever the required share of protected group members is, in addition to the individual fairness property of the learned representation.

4.5 Analysis

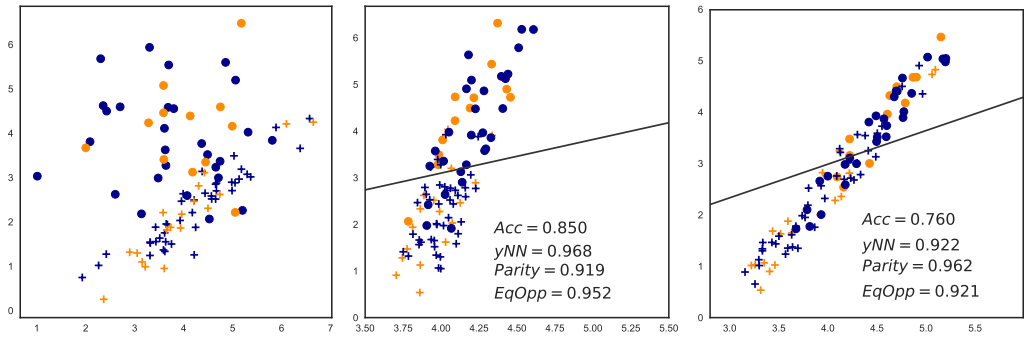
In this section, we investigate the learned representations of *iFair* and empirically compare *iFair* to the *LFR* model by performing additional experiments on synthetic datasets. We are interested in gaining further insights into the general behavior of methods for learned representations, to what extent they can reconcile utility and individual fairness at all, and how they relate to group fairness criteria (although *iFair* does not consider these in its optimization). To this end, we generate *synthetic data* with systematic parameter variation as follows. We restrict ourselves to the case of a binary classifier.

Synthetic Dataset: We generate 100 data points with 3 attributes: 2 real-valued and non-sensitive attributes $X1$ and $X2$ and 1 binary attribute A which serves as the protected attribute. We first draw two-dimensional datapoints from a mixture of Gaussians with two components: (i) isotropic Gaussian with unit variance and (ii) correlated Gaussian with covariance 0.95 between the two attributes and variance 1 for each attribute. To study the influence of membership to the protected group (i.e., A set to 1), we generate three variants of this data:

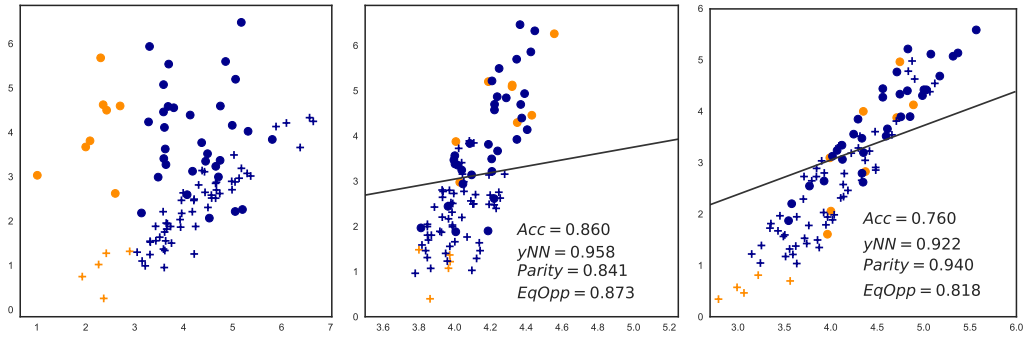
- Random: A is set to 1 with probability 0.3 at random.
- Correlation with $X1$: A is set to 1 if $X1 \leq 3$.
- Correlation with $X2$: A is set to 1 if $X2 \leq 3$.

So the three synthetic datasets have the same values for the non-sensitive attributes $X1$ and $X2$ as well for the outcome variable Y. The datapoints differ only on membership to the protected group and its distribution across output classes Y.

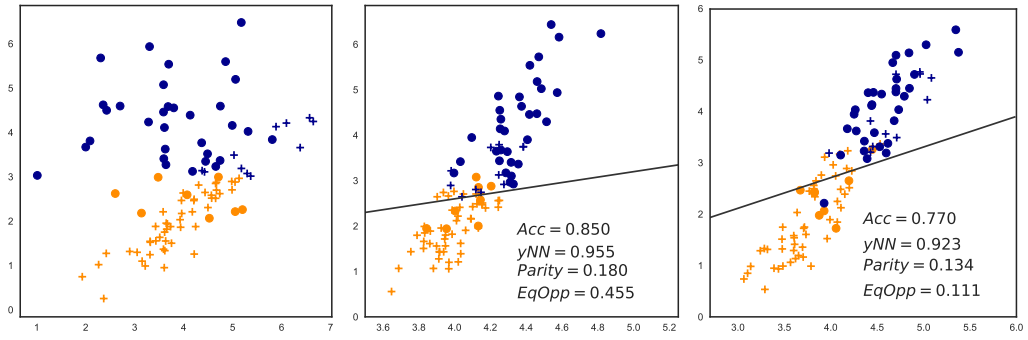
Figure 4.5 shows these three cases row-wise: subfigures a-c, d-f, g-i, respectively. The left column of the figure displays the original data, with the two class labels for output Y depicted by marker: "o" for $Y = 0$ and "+" for $Y = 1$ and the membership to the protected group by color: orange for $A = 1$ and blue for $A = 0$. The middle column of Figure 4.5 shows the learned *iFair* representations, and the right column shows the representations based on *LFR*. Note that the values of importance in Figure 4.5 (middle and right column) are the positions of the data points in the two-dimensional latent space and the classifier decision boundary (solid line). The color of the datapoints and the markers (o and +) depict the true class and true group membership, and not the learned values. They are visualized to aid the reader in relating original data with transformed representations. Furthermore, small differences in the learned representation are expected due to random initializations of model parameters. The solid line in the charts denotes the predicted classifiers' decision boundary applied on the learned representations. Hyper-parameters for both *iFair* as well as *LFR* are chosen by performing a grid search on the set $\{0, 0.05, 0.1, 1, 10, 100\}$ for optimal individual fairness of the classifier. For each of the nine cases, we indicate the resulting classifier accuracy *Acc*, individual fairness in terms of consistency yNN with regard to the $k = 10$ nearest neighbors, the statistical parity *Parity* with regard to the protected group $A = 1$, and equality-of-opportunity *EqOpp* notion of group fairness.



(a) original data (random) **(b)** *iFair* **(c)** *LFR*



(d) original data ($X1 \leq 3$) **(e)** *iFair* **(f)** *LFR*



(g) original data ($X2 \leq 3$) **(h)** *iFair* **(i)** *LFR*

Figure 4.5: Illustration of properties of data representations on synthetic data. (left: original data, center: representation learned via *iFair* model. right: representation learned via *LFR* model). Output class labels: o for Y=0 and + for Y=1. Membership in protected group: blue for A=0 and orange for A=1. Solid line depicts the decision boundary of a logistic regression model trained on the respective learned representations. *iFair* outperforms *LFR* on all metrics except for statistical parity.

Main findings: Two major insights from this study are: (i) representations learned via *iFair* remain nearly the same irrespective of changes in group membership, and (ii) *iFair* significantly outperforms *LFR* on accuracy, consistency and equality of opportunity, whereas *LFR* wins on statistical parity. In the following we further discuss these findings and their implications.

- **Influence of Protected Group:** The middle column in Figure 4.5 shows that the *iFair* representation remains largely unaffected by the changes in the group memberships of the datapoints. In other words, changing the value of the protected attribute of a datapoint, while all other attribute values remain the same, has hardly any influence on its learned representation; consequently it has nearly no influence on the outcome made by the decision-making algorithms trained on these representations. This is an important and interesting characteristic to have in a fair representation, as it directly relates to the definition of *individual fairness*. In contrast, the membership to the protected group has a pronounced influence on the learned representation of the *LFR* model (refer to Figure 4.5 right column). Recall that the color of the datapoints as well as the markers (o and +) are taken from the original data. They depict the true class and membership to group of the datapoints, and are visualized to aid the reader.
- **Tension in Objective Function:** The optimization via *LFR* has three components: classifier accuracy as utility metric, individual fairness in terms of data loss, and group fairness in terms of statistical parity. We observe that by pursuing group fairness and individual fairness together, the tension with utility is very pronounced. The learned representations are stretched on the compromise over all three goals, ultimately leading to sacrificing utility. In contrast, *iFair* pursues only utility and individual fairness, and disregards group fairness. This helps to make the multi-objective optimization more tractable. *iFair* clearly outperforms *LFR* not only on accuracy, with better decision boundaries, but also wins in terms of individual fairness. This shows that the tension between utility and individual fairness is lower than between utility and group fairness.
- **Trade-off between Utility and Individual Fairness:** The improvement that *iFair* achieves in individual fairness comes at the expense of a small drop in utility. The trade-off is caused by the loss of information in learning representative prototypes. The choice of the mapping function in Equation 4.7 and the fairness distance function $d_f(.)$ in Definition 4.2 affects the ability to learn prototypes. Our framework is flexible and easily supports other kernels and distance functions. Exploring these influence factors is a direction for future work.

4.6 Conclusions

We propose *iFair*, a generic and versatile, unsupervised framework to perform a probabilistic transformation of data into individually fair representations. Our approach accommodates two important criteria. First, we view fairness from an application-agnostic view, which allows us to incorporate it in a wide variety of tasks, including general classifiers and regression for learning-to-rank. Second, we treat individual fairness as a property of the

dataset (in some sense, like privacy), which can be achieved by pre-processing the data into a transformed representation. This stage does not need access to protected attributes. If desired, we can also post-process the learned representations and enforce group fairness criteria such as statistical parity.

We applied our model to five real-world datasets, empirically demonstrating that utility and individual fairness can be reconciled to a large degree. Applying classifiers and regression models to *iFair* representations leads to algorithmic decisions that are substantially more consistent than the decisions made on the original data. Our approach is the first method to compute individually fair results in learning-to-rank tasks. For classification tasks, it outperforms the state-of-the-art prior work.

Chapter 5

Individual Fairness via Pairwise Fairness Graphs

Contents

In this Chapter, we revisit the notion of individual fairness. A central challenge in operationalizing individual fairness is the difficulty in eliciting a human specification of a similarity metric. In this chapter, we propose an operationalization of individual fairness that does not rely on a human specification of a distance metric. Instead, we propose novel approaches to elicit and leverage side-information on equally deserving individuals to counter subordination between social groups. We model this knowledge as a fairness graph, and learn a unified Pairwise Fair Representation (PFR) of the data that captures both data-driven similarity between individuals and the pairwise side-information in fairness graph. We elicit fairness judgments from a variety of sources, including human judgments for two real-world datasets on recidivism prediction (COMPAS) and violent neighborhood prediction (Crime & Communities). Our experiments show that the PFR model for operationalizing individual fairness is practically viable.

5.1 Introduction

5.1.1 Motivation

The Case for Individual Fairness: The fairness notions explored by the bulk of the works can be broadly categorized as targeting either *group fairness* [Pedreschi et al. 2008; Feldman et al. 2015] or *individual fairness* [Dwork et al. 2012b]. Group fairness notions attempt to ensure that members of all protected groups in the population (e.g., based on demographic attributes like gender or race) receive their "fair share of beneficial outcomes" in a downstream task. To this end, one or more *protected attributes* and respective values are specified, and given special treatment in machine learning models. Numerous operationalizations of group fairness have been proposed and evaluated including demographic parity [Feldman et al. 2015], equality of opportunity [Hardt et al. 2016], equalized odds [Hardt et al. 2016], and envy-free group fairness [Zafar et al. 2017c]. These operationalizations differ in the measures used to quantify a group's "fair share of beneficial outcomes" as well as the mechanisms used to optimize for the fairness measures.

While effective at countering group-based discrimination in decision outcomes, group fairness notions do not address unfairness in outcomes at the level of individual users. For instance, it is natural for individuals to compare their outcomes with those of others with similar qualifications (independently of their group membership) and perceive any differences in outcomes amongst individuals with similar standing as unfair.

Challenges in Operationizaling Individual Fairness: In their seminal work [Dwork et al. 2012b], Dwork et al. introduced a powerful notion of fairness called individual fairness, which states that "similar individuals should be treated similarly". In the original form of individual fairness introduced in [Dwork et al. 2012b], the authors envisioned that a task-specific similarity metric would be provided by human experts which captures the similarity between individuals (e.g., "a student who studies at University W and has a GPA X is similar to another student who studies at University Y and has GPA Z"). The individual fairness notion stipulates that individuals who are deemed similar according to this *task-specific similarity metric* should receive similar outcomes. Operationalizing this strong notion of fairness can help in avoiding unfairness at an individual level.

However, eliciting such a quantitative measure of similarity from humans has been the most challenging aspect of the individual fairness framework, and little progress has been made on this open problem. Two noteworthy subsequent works on individual fairness are [Zemel et al. 2013] and [Lahoti et al. 2019b], wherein the authors operationalize a simplified notion of similarity metric. Concretely, they assume a distance metric (similarity metric) such as a *weighted* Euclidean distance over a feature space of data atttributes, and aim to learn *fair feature weights* for this distance metric. This simplification of the individual fairness notion largely limits the scope of the original idea of [Dwork et al. 2012b]: "... a (near ground-truth) approximation agreed upon by the society of the extent to which two individuals are deemed similar with respect to the task ...".

In this work we revisit the original notion of individual fairness. There are two main challenges in its operationalization: First, it is very difficult, if not impossible for humans to come up with a precise quantitative similarity metric that can be used to measure "who is

similar to whom". Second, even if we assume that humans are capable of giving a precise similarity metric, it is still challenging for experts to model subjective side-information such as "who should be treated similar to whom" as a quantitative similarity metric.

Examples: The challenge is illustrated by two scenarios:

- Consider the task of selecting researchers for academic jobs. Due to the difference in publication culture of various communities, the citation counts of *successful* researchers in programming language are known to be typically lower than that of *successful* machine learning researchers. An expert recruiter might have the background information for fair selection that "an ML researcher with high citations is similarly strong and thus equally deserving as a PL researcher with relatively lower citations". It is all but easy to specify this background knowledge as a similarity metric.

- Consider the task of selecting students for Graduate School in the US. It is well known that SAT tests can be taken multiple times, and only the best score is reported for admissions. Further, each attempt to re-take the SAT test comes at a financial cost. Due to complex interplay of historical subordination and social circumstances, it is known that, on average, SAT scores for African-American students are lower than for white students [Brooks 1992]. Keeping historical subordination in mind, a fairness expert might deem an African-American student with a relatively lower SAT score to be similar to and equally deserving as a white student with a slightly higher score. Once again, it is not easy to model this information as a similarity metric.

Research Questions: We address the following research questions in this chapter.

- [RQ1] How to elicit and model various kinds of expert knowledge on individual fairness?
- [RQ2] How to encode this background information, such that downstream tasks can make use of it for data-driven predictions and decision making?

5.1.2 Proposed Approach

[RQ1] From Distance Metric to Fairness Graph.

Key Idea: It is difficult, if not impossible, for human experts to judge "the extent to which two individuals are similar", much less formulate a precise *similarity metric*. In this chapter, we posit that it is much easier for experts to make pairwise judgments about who is equally deserving and should be treated similar to whom.

We propose to capture these pairwise judgments as a *fairness graph*, G, with edges between pairs of individuals deemed similar with respect to the given task. We view this as valuable side information, but we consider it to be subjective and noisy. Aggregation over many users can mitigate this, but we cannot expect G to be perfectly fair. Further, for generality, we do not assume that these are always complete. In many applications, only partial and sometimes sparse fairness judgments would be available. In our experiments, we study the sensitivity to the amount of data in G in Subsection 5.5.1. In Subsection 5.3.2 we address some of the practical challenges that arise in eliciting pairwise judgments such as

comparing individuals from diverse groups, and we present various methods to construct fairness graphs.

It is worth highlighting that we only need pairwise judgments for a small sample of individuals in the training data for the application task. Naturally, no human judgments are elicited for test data (unseen data). So once the prediction model for the application at hand has been learned, only the regular data attributes of individuals are needed.

[RQ2] Learning Pairwise Fair Representations.

Given a fairness graph G, the goal of an individually fair algorithm is to minimize the inconsistency (differences) in outcomes for pairs of individuals connected in graph G. Thus, every edge in graph G represents a fairness constraint that the algorithm needs to satisfy. In Section 5.3, we propose a model called *PFR* (for Pairwise Fair Representations), which learns a new data representation with the aim of preserving the utility of the input feature space (i.e., retaining as much information of the input as possible), while incorporating the fairness constraints captured in the fairness graph.

Specifically, *PFR* aims to learn a latent data representation that preserves the local neighborhoods in the input data space, while ensuring that individuals connected in the fairness graph are mapped to nearby points in the learned representation. Since local neighborhoods in the learned representation capture individual fairness, once a fair representation is learned, any out-of-the-box downstream predictor can be directly applied. *PFR* takes as input (i) data records for individuals in the form of a feature matrix X for training a predictor, and (ii) a (sparse) fairness graph G that captures pairwise similarity for a subsample of individuals in the training data. The output of *PFR* is a mapping from the input feature space to the new representation space that can be applied to data records of novel unseen individuals.

5.1.3 Contribution

The key contributions of this chapter are:

- A practically viable operationalization of the individual fairness paradigm that overcomes the challenge of human specification of a distance metric, by eliciting easier and more intuitive forms of human judgments.
- Novel methods for transforming such human judgments into pairwise constraints in a fairness graph G.
- A mathematical optimization model and representation learning method, called *PFR*, that combines the input data X and the fairness graph G into a unified representation by learning a latent model with graph embedding.
- Demonstrating the effectiveness of our approach at achieving both individual and group fairness using comprehensive experiments with synthetic as well as real-life data on recidivism prediction (Compas) and violent neighborhoods prediction (Crime and Communities).

5.2 Related Work

We now discuss work most closely related to *PFR*.

Individual Fairness: The closest prior work to *PFR* is *LFR* by Zemel et al. [2013] and our own prior work *iFair* [Lahoti et al. 2019b]. Similar to *PFR*, both *LFR* and *iFair* aim to operationalize individual fairness by learning a low rank fair representations of the data. Like *LFR* and *iFair* our method can also be used to find representations for new individuals not seen in the training data. However, unlike us, *LFR* and *iFair* by learning a restricted form of distance metric from the data alone. Specifically, they aim to learn fair representations for individuals that obfuscate protected information in the learnt representations. In contrast, *PFR* aims to learn fair representations that reconcile differences between feature distributions of similarly qualified individuals, by elicting and incorporating background information on pairwise similarity between individuals.

Some works use the objective of the learning algorithm itself to implicitly define the similarity metric [Speicher et al. 2018; Biega et al. 2018; Kearns et al. 2017b]. For instance, when learning a classifier, these works would use the class labels in the training data or predicted class labels to measure similarity. However, fairness notions are meant to address societal inequities that are not captured in the training data (with potentially biased labels and missing features). In such scenarios, the fairness objectives are in conflict with the learning objectives.

Work building on PFR: Subsequent to our proposed method *PFR*, there has been growing work on operationalizing individual fairness. Ruoss et al. [2020] build on our work and propose learning certifiable individually fair representations by defining logical constraints. Jung et al. [2021] make a similar argument as *PFR*, and propose operationalizing fairness in an online setting by eliciting on expert side-information. An interesting line of work[Bechavod et al. 2020; Ilvento 2020; Mukherjee et al. 2020; Gillen et al. 2018; Yurochkin et al. 2020], including our own subsequent work [Wang et al. 2019] proposes operationalizing individual fairness by learning a distance metric from the data via metric-learning.

Graph Embedding: Finally, the core optimization problem we formulate relates to graph embedding and representation learning [Hamilton et al. 2017]. The aim of graph embedding approaches is to a learn a representation for the nodes in the graph encoding the edges between nodes as well as the attributes of the nodes [Lin et al. 2005; Amid and Ukkonen 2015]. Similarly, we wish to learn a representation encoding both the features of individuals as well as their interconnecting edges in the fairness graph.

5.3 Model

5.3.1 Notation

- X is an input data matrix of N data records and M numerical or categorical attributes. We use X to denote both the matrix and the population of individuals x_i:

$$X = [x_1, x_2, x_3, \cdots x_N] \in R^{M \times N}$$

- Z is a low-rank representation of X in a D-dimensional space where $D \ll M$.

$$Z = [z_1, z_2, z_3, \cdots z_N] \in R^{D \times N}$$

- S is a random variable representing the values that the protected-group attribute can take. We assume a single attribute in this role; if there are multiple attributes which require fair-share protection, we simply combine them into one. We allow more than two values for this attribute, going beyond the usual binary model (e.g., gender = male or female, race = white or others). $X_s \subset X$ denotes the subset of individuals in X who are members of group $s \in S$.
- W^X is the adjacency matrix of a k-nearest-neighbor graph over the input space X:

$$W_{ij}^X = \begin{cases} \exp\left(\frac{-\|\mathbf{x_i}-\mathbf{x_j}\|^2}{t}\right), \text{if } \mathbf{x_i} \in N_p(x_j) \text{ or } \mathbf{x_j} \in N_p(x_i) \\ 0 \qquad\qquad\qquad\qquad , \text{otherwise} \end{cases}$$

  where $N_p(x_i)$ denotes the set of p nearest neighbors of x_i in Euclidean space (excluding the protected attributes), and t is a scalar hyper-parameter.
- W^F is the adjacency matrix of the fairness graph G whose nodes are individuals and whose edges are connections between individuals that are equally deserving and must be treated similarly.

5.3.2 Eliciting and Modeling Expert Knowledge on Fairness

In this section we address the question of how to elicit side-information on individual fairness and model it as a fairness graph G and its corresponding adjacency matrix as W^F. The key idea of our approach is rooted in the following observations:

- Humans have a strong intuition about whether two individuals are similar or not. However, it is difficult for humans to specify a quantitative *similarity metric*.
- In contrast, it is more natural to make other forms of judgments such as (i)"Is A similar to B with respect to the given task?", or (ii)"How suitable is A for the given task (e.g., on a Likert scale)".
- However, these kinds of judgments are difficult to elicit when the pairs of individuals belong to diverse, incomparable groups. In such cases, it is easier for humans to compare individuals within the same group, as opposed to comparing individuals between groups. Pairwise judgements can be beneficial even if they are available only sparsely, that is, for samples of pairs.

Next, we present two models for constructing fairness graphs, which overcome the outlined difficulties via

(i) eliciting (binary) pairwise judgments of individuals who should be treated similarly, or grouping individuals into equivalence classes (see Subsection 5.3.2.1) and

(ii) eliciting within-group rankings of individuals and connecting individuals across groups who fall within the same quantiles of the per-group distributions (see Subsection 5.3.2.2).

5.3.2.1 Fairness Graph for Comparable Individuals

The most direct way to create a fairness graph is to elicit (binary) pairwise similarity judgments about a small sample of individuals in the input data, and to create a graph W^F such that there is an edge between two individuals if they are deemed similarly qualified for a certain task (e.g., being invited for job interviews).

Another alternative is to elicit judgments that map individuals into discrete equivalence classes. Given a number of such judgments for a sample of individuals in the input dataset, we can construct a fairness graph W_F by creating an edge between two individuals if they belong to the same equivalence class irrespective of their group membership.

Definition 8. (Equivalence Class Graph) *Let $[x_i]$ denote the equivalence class of an element $x_i \in X$. We construct an undirected graph W^F associated to X, where the nodes of the graph are the elements of X, and two nodes x_i and x_j are connected if and only if $[x_i] = [x_j]$.*

The fairness graph built from such equivalence classes identifies equally deserving individuals – a valuable asset for learning a fair data representation. Note that the graph may be sparse, if information on equivalence can be obtained merely for sampled representatives.

5.3.2.2 Fairness Graph for Incomparable Individuals

However, at times, our individuals are from diverse and incomparable groups. In such cases, it is difficult if not infeasible to ask humans for pairwise judgments about individuals *across groups*. Even with the best intentions of being fair, human evaluators may be misguided by wide-spread bias. If we can elicit a ranked ordering of individuals per-group, and pool them into quantiles (e.g., the top-10-percent), then one could assume that individuals from different groups who belong to the same quantile in their respective rankings, are similar to each other. Arguments along these lines have been made also by Kearns et al. [2017b] in their notion of meritocratic fairness.

Specifically, our idea is to first obtain within-group rankings of individuals (e.g., rank men and women separately) based on their suitability for the decision task at hand, and then construct a between-group fairness graph by linking all individuals ranked in the same k^{th} quantile across the different groups (e.g., link programming language researcher and machine learning researcher who are similarly ranked in their own groups). The relative rankings of individuals within a group, whether they are obtained from human judgments or from secondary data sources, are less prone to be influenced by discriminatory (group-based) biases.

Formally, given (X_s, Y_s) for all $s \in S$, where Y_s is a random variable depicting the ranked position of individuals in X_s. We construct a *between-group quantile graph* using Definitions 9 and 10 as follows.

Definition 9. (k-th quantile) *Given a random variable Y, the k-th quantile Q_k is that value of y in the range of Y, denoted y_k, for which the probability of having a value less than or equal to y is k.*

$$Q(k) = \{y : Pr(Y \leq y) = k\} \quad \text{where} \quad 0 < k < 1 \tag{5.1}$$

For the non-continuous behavior of discrete variables, we would add appropriate ceil functions to the definition, but we skip this technicality.

Definition 10. (Between-group quantile graph) *Let $X_s^k \subset X$ denote the subset of individuals who belong to group $s \in S$ and whose scores lie in the k-th quantile. We can construct a multipartite graph W^F whose edges are given by:*

$$W_{ij}^F = \begin{cases} 1 & , \text{ if } x_i \in X_s^k \quad \text{and} \quad x_j \in X_{s'}^k \quad , \quad s \neq s' \\ 0 & , \text{ otherwise} \end{cases} \tag{5.2}$$

That is, there exists an edge between a pair of individuals $\{x_i, x_j\} \in X$ if x_i and x_j have different group memberships and their scores $\{y_i, y_j\}$ lie in the same quantile. For the case of two groups (e.g., gender is male or female), the graph is a bipartite graph.

This model of creating between-group quantile graphs is general enough to consider any kind of per-group ranked judgment. Therefore, this model is not necessarily limited to legally protected groups (e.g., gender, race), it can be used for any socially salient groups that are incomparable for the given task (e.g., machine learning vs. programming language researchers). Note again that the pairwise judgements may be sparse, if such information is obtained only for sampled representatives.

5.3.3 Learning Pairwise Fair Representations

In this section we address the question: How to encode the background information such that downstream tasks can make use of it for the decision making?

5.3.3.1 Objective Function

In fair machine learning, such as fair classification models, the objective usually is to maximize the classifier accuracy (or some other quality metric) while satisfying constraints on group fairness statistics such as parity. For learning fair data representations that can be used in any downstream application – classifiers or regression models with varying target variables unknown at learning time – the objective needs to be generalized accordingly. To this end, the *PFR* model aims to combine the utility of the learned representation and, at the same time, preserve the information from the pairwise fairness graph. Starting with matrix X of N data records $x_1 \dots x_N$ and M numeric or categorial attributes, *PFR* computes a lower-dimensional latent matrix Z of N records each with $D < M$ values.

We model utility into the notion of preserving local neighborhoods of user records in the attribute space X in the latent representation Z

Reflecting the fairness graph in the learner's optimization for Z is a demanding and a priori open problem. Our solution *PFR* casts this issue into a graph embedding that is incorporated into the overall objective function. The following discusses the technical details of *PFR* 's optimization.

Preserving the input data: For each data record x_i in the input space, we consider the set $N_p(x_i)$ of its p nearest neighbors with regard to the distance defined by the kernel function given by W_{ij}^X. For all points x_j within $N_p(x_i)$, we want the corresponding latent

representations z_j to be close to the representation z_i, in terms of their L2-norm distance. This is formalized by the *Loss in* W^X, denoted by $Loss_X$.

$$Loss_X = \sum_{i,j=1}^{N} \|z_i - z_j\|^2 W_{ij}^X \tag{5.3}$$

Note that this objective requires only local neighborhoods in X to be preserved in the transformed space. We disregard data points outside of p-neighborhoods. This relaxation increases the feasible solution space for the dimensionality reduction.

Learning a fair graph embedding: Given a fairness graph W^F, the goal for Z is to preserve neighborhood properties in W^F. In contrast to $Loss_X$, however, we do not need any distance metric here, but can directly leverage the fairness graph. If two data points x_i, x_j are connected in W^F, we aim to map them to representations z_i and z_j close to each other. This is formalized by the *Loss in* W^F, denoted by $Loss_F$.

$$Loss_F = \sum_{i,j=1}^{N} \|z_i - z_j\|^2 W_{ij}^F \tag{5.4}$$

Intuitively, for data points connected in W^F, we add a penalty when their representations are far apart in Z.

Combined objective: Based on the above considerations, a fair representation Z is computed by minimizing the combined objectives of Equations 5.3 and 5.4. The parameter γ weighs the importance tradeoff between W^X and W^F. As γ increases influence of the fairness graph W^F increases. An additional orthonormality constraint on Z is imposed to avoid trivial results. The trivial result being that all the datapoints are mapped to same point.

$$\begin{aligned} &\text{Minimize } (1-\gamma) \sum_{i,j=1}^{N} \|z_i - z_j\|^2 W_{ij}^X + \gamma \sum_{i,j=1}^{N} \|z_i - z_j\|^2 W_{ij}^F \\ &\text{subject to} \quad Z^T Z = I \end{aligned} \tag{5.5}$$

5.3.3.2 Equivalence to Trace Optimization Problem

Next, we show that the optimization problem in Equation 5.5 can be transformed and solved as an equivalent eigenvector problem. To do so, we assume that the learnt representation Z is a linear transformation of X given by $Z = V^T X$.

We start by showing that minimizing $\|z_i - z_j\|^2 W_{ij}$ is equivalent to minimizing the trace $Tr(V^T X L X^T V)$. Here we use W to denote W^X or W^F, as the following mathematical

derivation holds for both of them analogously:

$$\begin{aligned} Loss &= \sum_{i,j=1}^{N} \|z_i - z_j\|^2 W_{ij} \\ &= \sum_{i,j=1}^{N} Tr((z_i - z_j)^T (z_i - z_j)) W_{ij} \\ &= 2 \cdot Tr(\sum_{i,j=1}^{N} z_i^T z_i D_{ii} - \sum_{i,j=1}^{N} z_i^T z_j W_{ij}) \\ &= 2 \cdot Tr(V^T X L X^T V) \end{aligned}$$

where $Tr(.)$ denotes the trace of a matrix, D is a diagonal matrix whose entries are column sums of W, and $L = D - W$ is the graph Laplacian constructed from matrix W. Analogous to L, we use L^X to denote graph laplacian of W^X, and L^F to denote graph laplacian of W^F.

5.3.3.3 Optimization Problem

Considering the results of Subsection 5.3.3.2, we can transform the above combined objective in Equation 5.5 into a trace optimization problem as follows:

$$\begin{aligned} &\text{Minimize } J(V) = \quad Tr\{V^T X((1-\gamma)L^X + \gamma L^F)X^T V\} \\ &\text{subject to} \quad V^T V = I \end{aligned} \tag{5.6}$$

We aim to learn an $M \times D$ matrix V such that for each input vector $x_i \in X$, we have the low-dimensional representation $z_i = V^T x_i$, where $z_i \in Z$ is the mapping of the data point x_i on to the learned basis V. The objective function is subjected to the constraint $V^T V = I$ to eliminate trivial solutions.

Applying Lagrangian multipliers, we can formulate the trace optimization problem in Equation 5.6 as an eigenvector problem

$$X((1-\gamma)L^X + \gamma L^F)X^T \mathbf{v_i} = \lambda \mathbf{v_i} \tag{5.7}$$

It follows that the columns of optimal V are the eigenvectors corresponding to D smallest eigenvalues denoted by $V = [\mathbf{v_1 v_2 v_3} \cdots \mathbf{v_D}]$, and γ is a regularization hyper-parameter. Finally, the d-dimensional representation of input X is given by $Z = V^T X$.

Implementation: The above standard eigenvalue problem for symmetric matrices can be solved in $O(N^3)$ using iterative algorithms. In our implementation we use the standard eigenvalue solver in scipy.linalg.lapack python library [Anderson et al. 1990].

5.3.3.4 Inference

Given an input vector x_i for a previously unseen individual, the *PFR* method computes its fair representation as $z_i = V^T x_i$ where z_i is the projection of the datapoint x_i on the learned basis V. It is important to note that the fairness graph W^F is only required during the training phase to learn the basis V. Once the $M \times D$ matrix V is learned, we do not need any fairness labels for newly seen data.

5.4 Experiments

This section reports on experiments with synthetic and real-life datasets. Subsection 5.4.1 introduces the experimental setup and baselines. In Subsection 5.4.2 and 5.4.3, we report our main results answering the following key questions.

- [Q1] What do the learned representations look like?
- [Q2] What is the trade-off between individual fairness and utility?
- [Q3] What is the influence on group fairness?

5.4.1 Setup and Baselines

Datasets and Downstream Task: We evaluate the performance of a variety of fairness methods on a downstream classification task using three datasets: (i) a synthetic dataset for US university admission with 203 numerical features, and two real-world datasets: (ii) crime and communities dataset for violent neighbourhood predictions with 96 numerical features and 46 one-hot encoded features (for categorical attributes), and (iii) compas dataset for recidivism prediction with 9 numerical and 420 one-hot encoded features. In order to check the *"true"* dimensionality of the datasets we computed the smallest rank k for SVD that achieves a relative error of at most 0.01 for the Frobenius norm difference between the SVD reconstruction and the original data. For the three datasets, these dimensionalities are 156, 69, and 117 respectively. Table 5.1 summarizes the statistics for each dataset, including base-rate (fraction of samples belonging to the positive class, for both the protected group and its complement). In all experiments, the representation learning based fair methods are followed by an out-of-the-box logistic regression classifier trained on the corresponding representations.

Table 5.1: Experimental settings and dataset statistics

Dataset	No of. records	No. of features	True Rank	Base-rate (s = 0)	Base-rate (s = 1)	Protected attribute
Synthetic	1000	203	156	0.51	0.48	Race
Crime	1993	142	69	0.35	0.86	Race
Compas	8803	429	117	0.41	0.55	Race

Baselines: We compare the performance of the following methods

- *Original representation:* a naive representation of the input dataset wherein the protected attributes are masked.
- *iFair* [Lahoti et al. 2019b]: a representation learning method, which optimizes for two objectives: (i) individual fairness in W^X, and (ii) obfuscating protected attributes.
- *LFR*[Zemel et al. 2013]: a representation learning method, which optimizes for three objectives: (i) accuracy (ii) individual fairness in W^X and (iii) statistical parity.

- *Hardt* [Hardt et al. 2016]: a post-processing method that aims to minimize the difference in error rates between groups by optimizing for the group-fairness measure *EqOdd* (Equality of Odds).

- *PFR*: Our unsupervised representation learning method that optimizes for two objectives (i) individual fairness as per W^F and (ii) individual fairness as per W^X.

Augmenting Baselines: For fair comparison we compare *PFR* with augmented versions of all methods (named with *suffix* +). In the augmented version, we give each method an advantage by enhancing it with the information in the fairness graph W^F. Since none of the methods can be naturally extended to incorporate the fairness graph as it is, we make our best attempt at modeling the fairness labels that are used to construct W^F as additional numerical features in the training data. Since we only have judgments for a sample of training data, we treat the rest as missing values and set them to -1. Note that this enhancement is only for training data as fairness labels are not available for unseen test data. This is in line with how PFR uses the pairwise comparisons: its representation is learned from the training data, but at test time, only data attributes X are available. Concrete details for each of the datasets follow in their respective subsections.

Hyper-parameter Tuning: We use the same experimental setup and hyper-parameter tuning techniques for all methods. Each dataset is split into separate training and test sets. On the training set, we perform 5-fold cross-validation (i.e., splitting into 4 folds for training and 1 for validation) to find the best hyper-parameters for each model via *grid search*. Once hyper-parameters are tuned, we use a independent test set to measure performance. All reported results are averages over 10 runs on independent test sets.

Evaluation Measures:

- **Utility** is measured as AUC (area under the ROC curve).
- **Individual Fairness** is measured as the *consistency* of outcomes between individuals who are similar to each other. We report consistency values as per both the similarity graphs, W^X and W^F.

$$Consistency = 1 - \frac{\sum_i \sum_j |\hat{y}_i - \hat{y}_j| \cdot W_{ij}}{\sum_i \sum_j W_{ij}} \quad \forall \quad i \neq j$$

- **Group Fairness**
  - *Equal Odds:* A binary classifier satisfies equal odds if the group-wise error rates are the same across all groups. In our experiments, we report group-wise false positive rate (FPR) and false negative rate (FNR).
  - *Statistical Parity:* A binary classifier satisfies statistical parity if the rate of positive predictions is the same across all groups $s \in S$. In our experiments, we report group-wise fraction of positive predictions.

$$P(\hat{Y} = 1|s = 0) = P(\hat{Y} = 1|s = 1) \tag{5.8}$$

5.4.2 Synthetic-Data-Experiments

5.4.2.1 Dataset and Setup

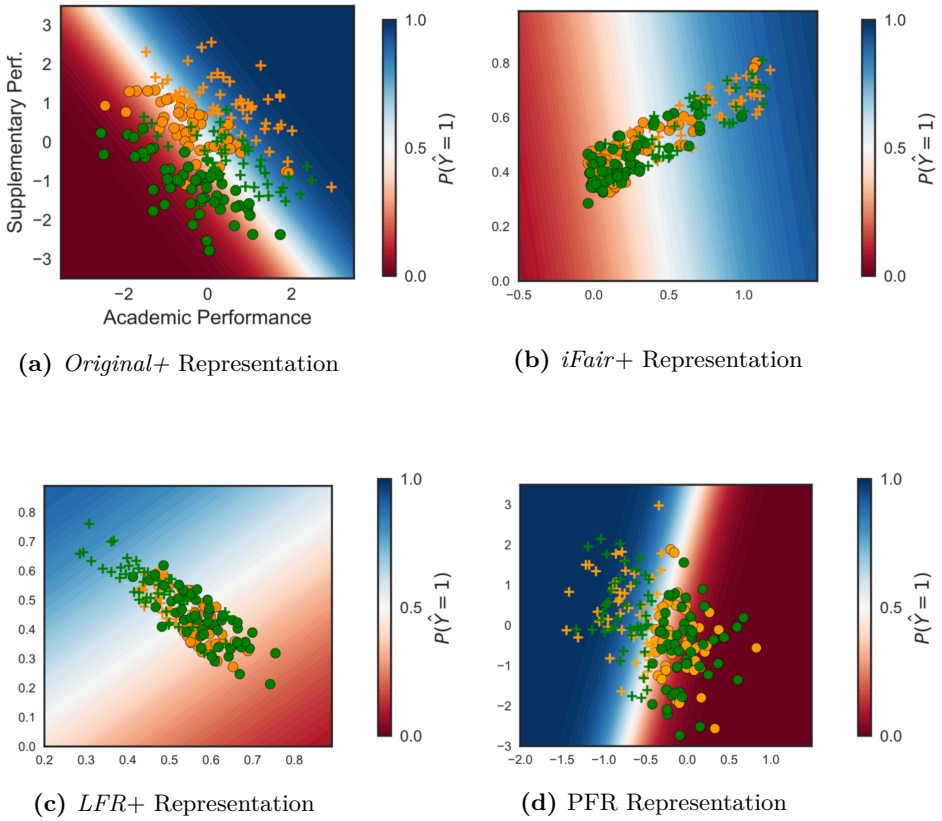
First, we evaluate the PFR's performance on a synthetic dataset by simulating the US graduate admissions scenario discussed in Section 5.1.1. Our task is it to predict the ability of a candidate to complete graduate school (binary classification). To this end, we imagine that the features in a college admission task can be grouped into two categories. First set of features which are related to their *academic performance* such as overall GPA, grades in each of the high schools subjects like Mathematics, Science, Languages, etc. Second set of features are related to their *supplementary performance* which constitute their overall application package such as SAT scores, admission essay, extracurricular activities, etc.

We assume that the scores for the second set of features can be inflated for individuals who have higher privilege in the society, for instance by re-taking SAT exam, and receiving professional coaching. Suppose we live in a society where our population consists of two groups $s = 0$ or 1, and the group membership has a high correlation with individual's privilege. This would result in a scenario where the two groups have different feature distributions. Further, if we assume that the inflation in the scores does not increase the ability of the candidate to complete college, the relevance functions for the two groups would also be different.

Creating Synthetic Datasets: We simulate this scenario by generating synthetic data for two population groups X_0 and X_1. Our dataset consists of three main features: group, *academic performance*, and *supplementary performance*. The correlation between *academic performance* and *supplementary performance* is set to 0.3. We have additional 100 numerical features with high correlation to *academic performance*, and 100 numerical features with high correlation to *supplementary performance*. We set the value of correlation between related features by drawing uniformly from [0.75, 1.0]. We use the correlation between features to construct the covariance matrix for a multivariate Gaussian distribution of dimensionality 203. To reflect the point that one groups has inflated scores for the features related to *supplementary performance*, we set the mean for these features for the non-protected group one standard deviation higher than the mean for the protected group.

In total we generate 600 samples for training, and 400 samples as a withheld test set. We run our experiments on two versions of the synthetic dataset: (i) a *low-dimensional* dataset, which is a subset of the high-dimensional data consisting of only three features: Group, Academic Performance and Supplementary performance, and (ii) a *high-dimensional* dataset with all 203 features. Experiments on the low-dimensional dataset are performed in order to be able to visually compare the original and learned representations. Dataset statistics are shown in Table 5.1.

Ground Truth Labels: Despite average score on *supplementary performance* features for group $X_{s=0}$ being higher than for the protected group $X_{s=1}$, we assume that the ability to complete graduate school is the same for both groups; that is, members of $X_{s=0}$ and $X_{s=1}$ are equally deserving if we adjust their *supplementary performance* scores. To implement this scenario, we set the *true* class label for group $X_{s=1}$ to positive (1) if *academic* + *supplementary* score ≥ 0 and for group $X_{s=0}$ as positive (1) if *academic* + *supplementary*



(a) *Original+* Representation

(b) *iFair+* Representation

(c) *LFR+* Representation

(d) PFR Representation

Figure 5.1: Comparison of (a)*Original+* (b)*iFair+* (c)*LFR+* and (d)*PFR* representations on a synthetic dataset. Colors depict membership to protected group (S): orange (non-protected) and green (protected). Markers denote *true* class labels: $Y = 1$ (marker +) and $Y = 0$ (marker o). Contour plots visualize decision boundary of a classifier trained on the representations. Blue color corresponds to prediction $\hat{Y} = 1$ and red corresponds to prediction $\hat{Y} = 0$. The more intensive the color, the higher or lower the score of the classifier.

score ≥ 1. Figure 5.1a visualizes the generated dataset. The colors depict the membership to groups (S): S = 0 (orange) and S = 1 (green). The markers denote *true* class labels Y = 1 (marker +) and Y = 0 (marker o).

Fairness Graph W^F: In this experiment we simulate the scenario for eliciting human input on fairness, wherein we have access to a fairness oracle who can make the judgments of the form "Is A similar to B?" described in Subsection 5.3.2.1. To this end, we randomly sample $N \log_2 N := 5538$ pairs (out of the possible $N^2 := 600 \times 600$). We then constructed our fairness graphs W^F by querying a fairness oracle for Yes/No answers to similarity pairs. If the two points are similar, we add an edge between the two nodes.

Fairness oracle for this task is a machine learning model consisting of two separate logistic regression models, one for each group, $X_{S=0}$ and $X_{S=1}$ respectively. Given a pair of points, if their prediction probabilities fall in the same quantile, they are deemed similar by the fairness oracle.

Augmenting Baselines: We cast each row of the matrix W^F (of the fairness graph) into n additional binary features for the respective individual. That is, for every user record, n additional 0/1 features indicate pairwise equivalence. All baselines have access to this information via the augmented input matrix X.

5.4.2.2 Results on Synthetic Low Dimension Dataset

What do the learned representations look like? In this subsection we inspect the original representations and contrast them with learned representations via *iFair+* [Lahoti et al. 2019b], *LFR+* [Zemel et al. 2013], and our proposed model *PFR*. Figure 5.1 visualizes the original dataset and the learned representations for each of the models with the number of latent dimensions set to $d = 2$ during the learning. The contour plots in (b), (c) and (d) denote the decision boundaries of logistic regression classifiers trained on the respective learned representations. Blue color corresponds to positive classification, red to negative; the more intensive the color, the higher or lower the score of the classifier. We observe several interesting points:

- First, in the original data, the two groups are separated from each other: *green* and *orange* datapoints are relatively far apart. Further, the deserving candidates of one group are relatively far away from the deserving candidates of the other group. That is, "green plus" are far from "orange plus", illustrating the inherent unfairness in the original data.
- In contrast, for all three representation learning techniques – *iFair+*, *LFR+* and *PFR* – the *green* and *orange* data points are well-mixed. This shows that these representations are able to make protected and non-protected group members indistinguishable from each other – a key property towards fairness.
- The major difference between the learned representations is that *PFR* succeeds in mapping the deserving candidates of one group close to the deserving candidates of the other group (i.e., "green plus" are close to "orange plus"). Neither *iFair+* nor *LFR+* can achieve this desired effect to the same extent.

Utility vs Individual Fairness: Figure 5.2 shows the best achievable trade-off between utility and the two notions of individual fairness. We make the following observations:

Individual fairness: PFR significantly outperforms all competitors in terms of *consistency* (W^F). This follows from the observation that, unlike *Original+*, *iFair+* and *LFR+* representations, *PFR* maps similarly deserving individuals close to each other in its latent space and similar performance as other approaches for *consistency* (W^X), but for a significantly better performance on AUC as shown in Figure 5.2.

Utility (AUC): PFR achieves by far the best *AUC*, even outperforming the original representation. While this may surprise on first glance, it is indeed an expected outcome. The fairness edges in W_F help *PFR* overcome the challenge of different groups having different feature distributions (observe Figure 5.1a). In contrast, *PFR* is able to learn a unified representation that maps deserving candidates of one group close to deserving candidates of the other group (observe Figure 5.1d), which helps in improving *AUC*.

Group Fairness: In addition to *Original+*, *iFair+*, *LFR+* and PFR, we include the *Hardt*

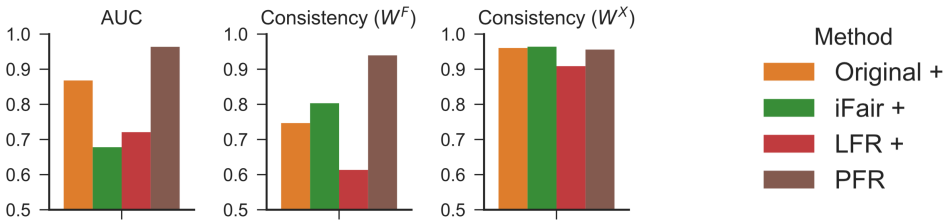
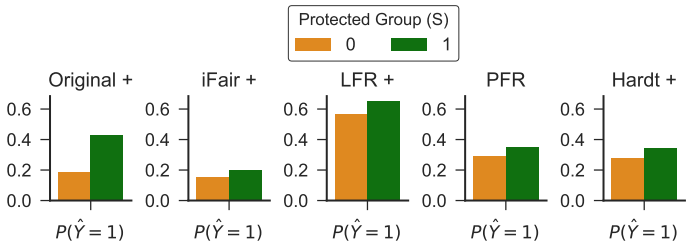


Figure 5.2: Results for Synthetic low dimension dataset: Comparison of utility vs individual fairness trade-off across methods. Higher values are better.

(a) Group-wise error rates (FPR and FNR)



(b) Group-wise fraction of positive prediction

Figure 5.3: Results for Synthetic low dimension dataset: (a) Equal Odds: Difference in error rates between protected and non-protected groups and (b) Statistical Parity: Difference in fraction of positive predictions.

model in the comparison here, as it is widely viewed as the state-of-the-art method for group fairness. Figure 5.3a shows the per-group error rates, and Figure 5.3b shows the per-group positive prediction rates. The smaller the difference in the values of the two groups, the higher the group fairness. We make the following interesting observations:

Equal Odds (Figure 5.3a): We observe that *Original+* model has high difference in error rates (aka. Equal Odds). *iFair+* and *LFR+* balance the error rates across groups fairly well, but still have fairly high error rates, indicating their loss on utility. *PFR* and *Hardt* have well balanced error rates and generally lower error. For *Hardt*, this is the expected effect, as it is optimized for the very goal of Equal Odds. *PFR* achieves the best balance and lowest

error rates, which is remarkable as its objective function does not directly consider group fairness. Again, the effect is explained by *PFR* succeeding in mapping equally deserving individuals from both groups to close proximity in its latent space.

Statistical Parity (Figure 5.3b): The *Original+* approach exhibits a substantial difference in the per-group positive predictions rates of the two groups. In contrast, *iFair+*, *LFR+*, and *PFR* representation have the *orange* and *green* data points well-mixed, and this way achieve nearly equal rates of positive predictions for both groups. Likewise *Hardt+* has the same desired effect.

5.4.2.3 Results on Synthetic High Dimension Dataset

The results for the high-dimensional synthetic data are largely consistent with the results for the low-dimensional case of Subsection 5.4.2.2. Therefore, we discuss them only briefly. Figure 5.4 shows results for *AUC*, *consistency*(W^F), and *consistency*(W^X). Figure 5.5 shows results on group fairness measures.

Utility vs. Individual fairness regarding W^F: On first glance, *LFR+* seems to perform best on consistency with regard to W^F. However, this is trivially achieved by giving the same prediction to almost all datapoints: the classifier using the learned *LFR+* representation accepts virtually all individuals, hence its very poor *AUC* of around 0.55. In essence, *LFR+* fails to learn how to cope with the utility-fairness trade-off. Therefore, we consider this method as degenerated (for this dataset) and dismiss it as a real baseline. Among the other methods, *PFR* significantly outperforms all competitors by achieving the best performance on *consistency* (W^F), similar performance as other approaches on *consistency* (W^X), but for a significantly better performance on *AUC*, as shown in Figure 5.4.

Group Fairness: Once again, *PFR* clearly outperforms all other methods on group fairness. It achieves near-equal error rates across groups, and near-equal rates of positive predictions as shown in Figures 5.5a and 5.5b. Again, *PFR*'s performance on group fairness is as good as that of *Hardt* which is solely designed for equalizing error rates by post-processing the classifier's outcomes. *LFR+* seems to achieve good results as well, but this is again due to accepting virtually all individuals (see above).

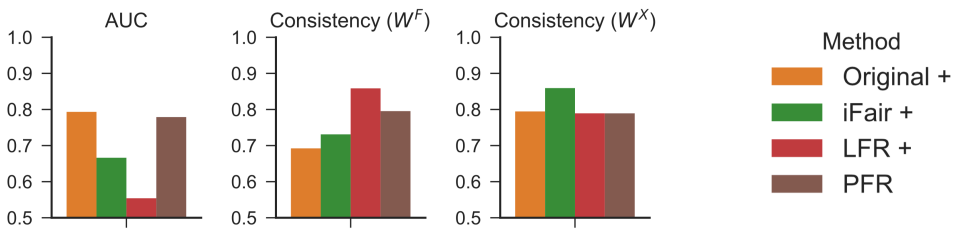
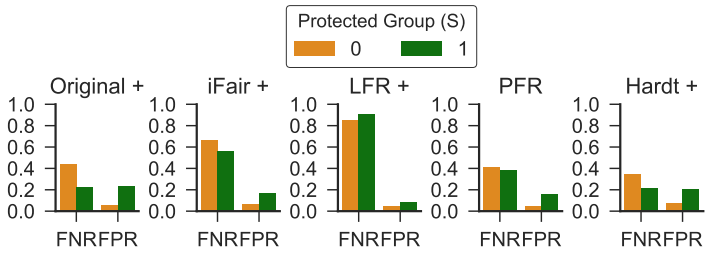
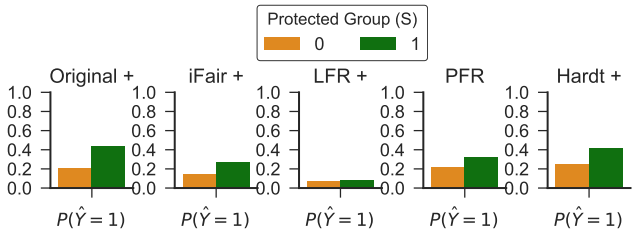


Figure 5.4: Results for Synthetic high dimension dataset: Comparison of utility vs individual fairness trade-off across methods. Higher values are better.



(a) Group-wise error rates (FPR and FNR)



(b) Group-wise fraction of positive predictions

Figure 5.5: Results for Synthetic high dimension dataset: (a) Equal Odds: Difference in error rates between protected and non-protected groups and (b) Statistical Parity: Difference in fraction of positive predictions.

5.4.3 Real-World-Data Experiments

5.4.3.1 Dataset and Setup

We evaluate the performance of *PFR* on the following two real world datasets:

Crime & Communities [M. 2009] is a dataset consisting of socio-economic (e.g., income), demographic (e.g., race), and law/policing data (e.g., patrolling) records for neighborhoods in the US. We set *isViolent* as target variable for a binary classification task. We consider the communities with majority population white as non-protected group and the rest as protected group.

Fairness Graph W^F: We need to elicit pairwise judgments of similarity that model whether two neighborhoods are similar in terms of crime and safety. To this end, we collected human reviews on crime and safety for neighborhoods in the US from `http://niche.com`. The judgments are given in the form of 1-star to 5-star ratings by current and past residents of these neighborhoods. We aggregate the judgments and compute mean ratings for all neighborhoods. We were able to collect reviews for about 1500 (out of 2000) communities. W^F is then constructed by the technique of Subsection 5.3.2.1. Although this kind of human input is subjective, the aggregation over many reviews lifts it to a level of inter-subjective side-information reflecting social consensus by first-hand experience of people. Nevertheless, the fairness graph may be biased in favor of the African-American neighborhoods, since residents tend to have positive perception of their neighborhood's safety.

COMPAS data collected by ProPublica [Angwin et al. 2016] contains criminal records comprising offenders' criminal histories and demographic features (gender, race, age etc.). We use the information on whether the offender was re-arrested as the target variable for binary classification. As protected attribute $s \in \{0, 1\}$ we use race: African-American (1) vs. others (0).

Fairness Graph W^F: We need to elicit pairwise judgments of similarity that model whether two individuals are similar in terms of deserving to be granted parole and not becoming re-arrested later. However, it is virtually impossible for a human judge to fairly compare people from the groups of *African-Americans* vs. *Others*, without imparting the historic bias. So this is a case, where we need to elicit pairwise judgments between diverse and incomparable groups.

We posit that it is fair, though, to elicit *within-group* rankings of risk assessment for each of the two groups, to create edges between individuals who belong to the same risk quantile of their respective group. To this end, we use Northpointe's Compas decile scores [Brennan et al. 2009] as background information about within-in group ranking. These *decile scores* are computed by an undisclosed commercial algorithm which takes as input official criminal history and interview/questionnaire answers to a variety of behavioral, social and economic questions (e.g., substance abuse, school history, family background etc.). The decile scores assigned by this algorithm are *within-group* scores and are not meant to be compared across groups. We sort these decile scores for each group seprately to simulate per-group ranking fairness judgments. We then use these per-group rankings as the fairness judgment to construct the fairness graphs for incomparable individuals as discussed in Subsection 5.3.2.2. Specifically, we compute k quantiles over the ranking as per Definition 9 and then, construct W^F as described in Definition 10. Note that this fairness graph has an implicit anti-subordination assumption. That is, it assumes that individuals in k-th risk quantile of one group are similar to the individuals in k-th quantile of other group - irrespective of their true risk.

Augmenting Baselines: We give our baselines access to the elicited fairness labels by adding them as numerical features to the rows of the input matrix X. For the Crime and Communities data, we added the elicited ratings (1 to 5 stars) as numerical features, with missing values set to -1. For the Compas data, where the fairness labels are per-group rankings, we added the ranking position of each individual within its respective group as a numerical feature.

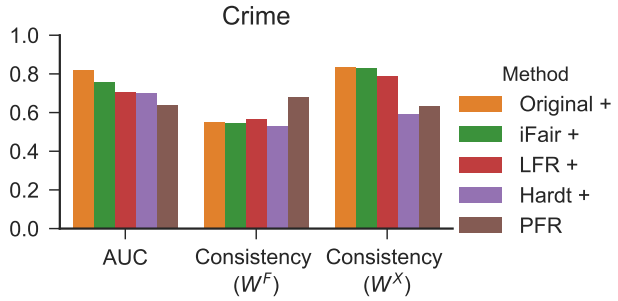
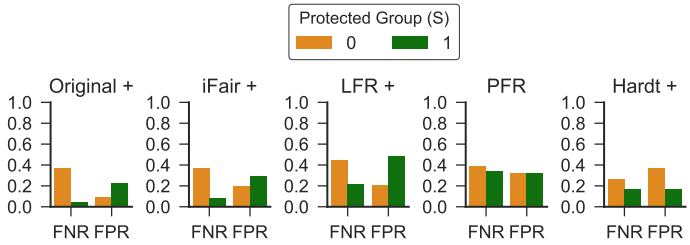
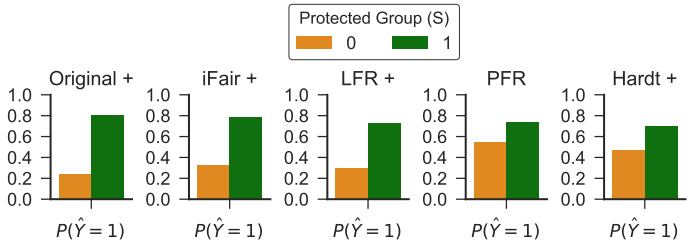


Figure 5.6: Crime & Communities data: utility vs. individual fairness.



(a) Group-wise error rates (FPR and FNR)



(b) Group-wise fraction of positive prediction

Figure 5.7: Crime & Communities: (a) Equal Odds: Difference in group-wise error rates and (b) Statistical Parity: Difference in fraction of positive predictions.

5.4.3.2 Results on Crime & Communities Dataset

Utility vs Individual Fairness: Results on individual fairness and utility are given in Figure 5.6. We observe that even though all the methods have access to the same fairness information, only *PFR* shows an improvement in consistency W^F over the baseline. *PFR* outperforms all other methods on individual fairness (consistency W^F). However, this gain for W^F comes at the cost of losing in consistency as per W^X. So in this case, the pairwise input from human judges exhibits pronounced tension with the data-attributes input. Deciding which of these sources should take priority is a matter of application design.

The higher performance of *PFR* on individual fairness regarding W^F comes with a drop in utility as shown by the AUC bars in Figure 5.6. This is because, unlike the case of the synthetic data in Subsection 5.4.2, the side-information for the fairness graph W^F is not strongly aligned with the ground-truth for the classifier. In terms of relative comparison, we observe that only *PFR* shows an improvement in consistency W^F over the baseline, the other approaches show no improvement. The performance of *iFair+* and *LFR+* on consistency on W^F and consistency on W^X is same as that of *Original+*, however for a lower *AUC*. *Hardt+* underperforms on all the three measures.

Group Fairness Figure 5.7a shows the per-group error rates, and 5.7b shows the per-group positive prediction rates. Smaller differences in the values between the two groups are preferable. The following observations are notable:

Equal Odds: *PFR* significantly outperforms all other methods on balancing the error rates of the two groups. Furthermore, it achieves nearly equal error rates comparable to the *Hardt+* model, whose sole goal is to achieve equal error rates between groups via post-processing.

Statistical parity: PFR outperforms all the methods by achieving near perfect balance (i.e., near-equal rates of positive predictions).

5.4.3.3 Results on Compas Dataset

The results for the Compas dataset are mostly in line with the results for the synthetic data and Crime & Communities datasets. Therefore, we report only briefly on them.

Utility vs. Individual Fairness: *PFR* performs similarly as the other representation learning methods in terms of utility and individual fairness on W^F, as shown in Figure 5.8.

Group Fairness: However, *PFR* clearly outperforms all other methods on group fairness. It achieves near-equal rates of positive predictions as shown in Figure 5.9b, and near-equal error rates across groups as shown in Figure 5.9a. Again, *PFR*'s performance on group fairness is as good as that of *Hardt+* which is solely designed for equalizing error rates by post-processing the classifier's outcomes.

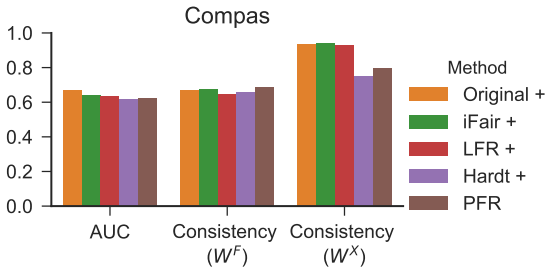
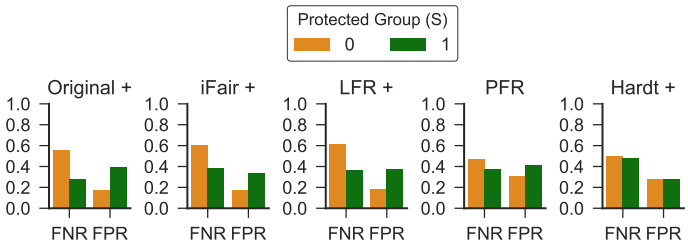
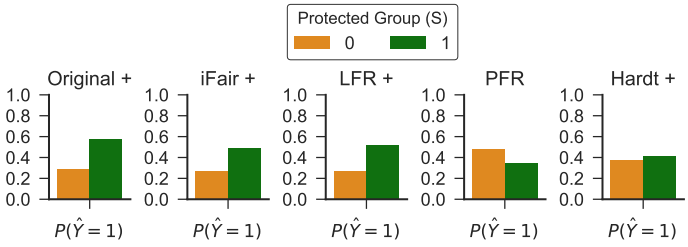


Figure 5.8: Compas data: utility vs. individual fairness.



(a) Group-wise error rates (FPR and FNR)



(b) Group-wise fraction of positive prediction

Figure 5.9: Compas data: (a) Equal Odds: Difference in error rates between protected and non-protected groups and (b) Statistical Parity: Difference in fraction of positive predictions.

5.5 Analysis

In this section, we conduct further analysis to gain insights into *PFR*. First, we investigate the sensitivity of our propsed PFR model to the number of labels in the fairness graph. Next, we investigate the robustness of our model to the PFR hyper-parameter γ. Finally, we present a discussion on key insights and lessons learned.

5.5.1 Sensitivity to Sparseness of Fairness Graph

Our goal is to study the sensitivity of *PFR* to the sparseness of the labeled pairs in the fairness graph W^F. To this end, we fix all hyper-parameters to their best values in the main experiments, and systematically vary the fraction of datapoints for which we use pairwise fairness labels. The results are shown in Figure 5.10. All results reported are on out-of-sample withheld test set of fairness graph W^F. Recall that *PFR* accesses fairness labels only for training data. For test data, it solely has the data attributes available in X.

Setup: For the synthetic data, we uniformly at random sampled fractions of $[log_2 N, \frac{N}{5}, \cdots, N, N\,log_2 N, N^2]$ pairs from the training data, which for this data translates into $[9, 120, \cdots 600, 5537, 360000]$ pairs. For the Crime data, we varied the percentage of training samples for which use equivalence labels, in steps of 10% from 10% to 100%. For the Compas data, we varied the percentage of training data points for which we elicit per-group rankings, in steps of 10% from 10% to 100%. We observe the following trends:

Results: Increasing the fraction of fairness labels improve the results on individual fairness (consistency for W^F), while hurting utility (AUC) only mildly (or even improving it in certain cases). For the synthetic data, even with as little as 0.17% of the fairness labels, the results are already fairly close to the best possible: consistency for W^F is already 90%, and AUC reaches 95%. For the Crime data, we need about 30 to 40% to get close to the best results for the full fairness graph. However, even with sparseness as low as 10%, *PFR* degrades smoothly: consistency W^F is 59% compared to 68% for the full graph, and AUC is affected only mildly by the sparseness. For the Compas data, we observe similar trends: even with very sparse W^F we stay within a few percent of the best possible consistency, and AUC varies only mildly with changing sparseness of the fairness graph.

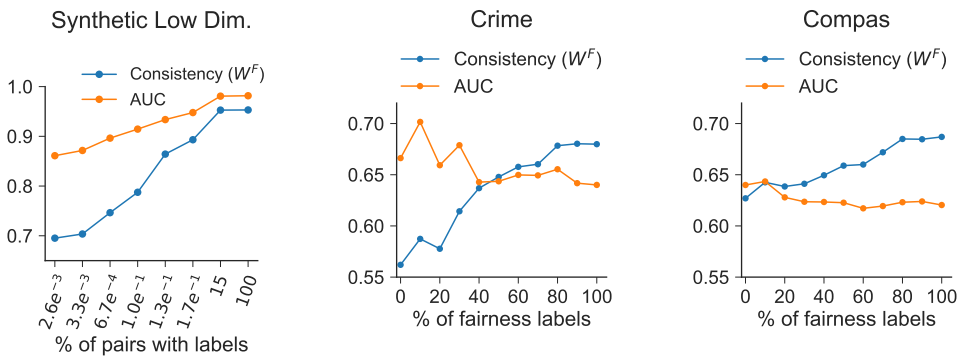


Figure 5.10: Sensitivity to sparseness of the fairness-graph.

These observations indicate that the *PFR* model yields benefits already with a small amount of human judgements of equally deserving individuals.

5.5.2 Influence of PFR Hyper-Parameter γ

Our goal is to analyze the influence of γ on the trade-off between individual fairness (consistency W^F) and utility (AUC) of the downstream classifiers. To this end, we keep all other hyper-parameters set to their values for the best result in the main experiments, and systematically vary the value of hyper-parameter γ in [0,1].

Recall that *PFR* aims to preserve local neighborhoods in the input space X (given by W^X), as well as the similarity given by the fairness graph W^F, where the hyper-parameter γ controls the relative influence of W^X and W^F. Figure 5.11 shows the influence of γ on individual fairness and utility for (a) low-dimensional synthetic, (b) Crime and (c) Compas data, respectively. We make the following key observations.

Individual Fairness: We observe that with increasing γ the consistency with regard to W^F increases. This is in line with our expectation: as γ increases the influence of W^F on the objective function, the performance of the model on individual fairness (consistency W^F) improves. This trend holds for all the datasets. It is worth highlighting that the improvement in individual fairness is for newly seen test samples that were unknown at the time when the fairness graph W^F was constructed and the *PFR* model was learned. This demonstrates the ability of *PFR* to generalize individual fairness to unseen data.

Utility: The influence of γ on the utility is more nuanced. We observe that the extent of the trade-off between individual fairness in W^F and utility depends on the degree of conflict between the pairwise W^F, and the classifier's ground-truth labels.

- If W^F indicates equal deservingness for data points that have different ground-truth labels, there is a natural conflict between individual fairness and utility. We observe this case for the real-world datasets Crime and Compas where W^F is in tension with ground-truth labels – presumably due to implicit anti-subordination embedded in graph or equivalently, due to historic discrimination in the classification ground-truth. With increasing γ, there is a slight drop in the utility *AUC* for the non-protected group. However, there is an improvement in *AUC* for the protected group. The overall AUC drops by a few percentage points, but stays at a high level even for very high γ. So we trade off a substantial gain in individual fairness for a small loss in utility. This is a clear case of how incorporating side-information on pairwise judgments can help to improve algorithmic decision making for historically disadvantaged groups.
- In contrast, if W^F pairs of equal deservingness are compatible with the classifier's ground-truth labels, there is no trade-off between utility and individual fairness. In such cases, W^F may even help to improve the utility by better learning a similarity manifold in the input space. We observe this case for the synthetic data where W^F is consistent with the ground-truth labels. As γ increases, the AUC of a classifier trained on *PFR* is enhanced. The improvement in *AUC* holds for both protected and non-protected groups.

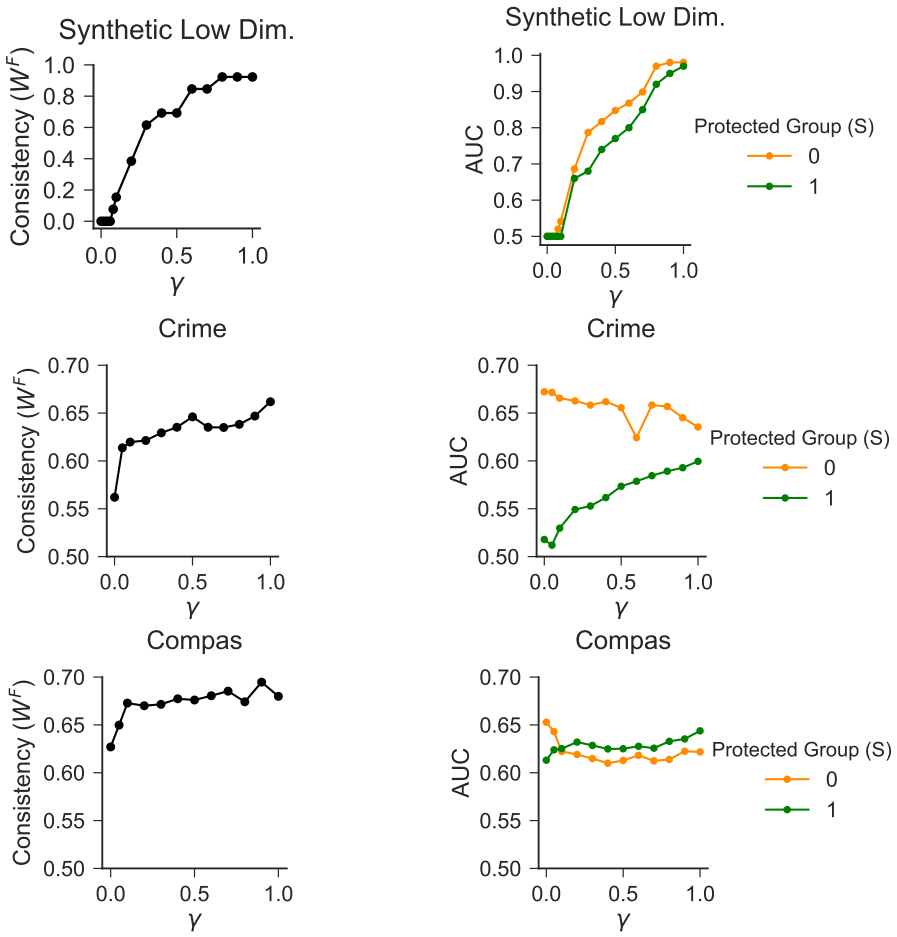


Figure 5.11: Influence of γ on individual fairness and utility.

5.5.3 Discussion

PFR outperforms all other methods on individual fairness regarding W^F for an acceptable performance in AUC, even when these baselines are given the same side-information for their augmented version (suffixed +). The improvement in individual fairness in W^F comes at the expense of reducing individual fairness for W^X, an unavoidable trade-off if the two views of fairness – data attributes (W^X) and pairwise judgements (W^F) – exhibit inherent tension. As for group fairness, *PFR* clearly outperforms all other representation learning methods, with group-fairness metrics as good as those of *Hardt+* whose sole optimization goal is to equalize the error rates. This strong behavior of PFR on group fairness measures is remarkable as PFR is not explicitly designed for this goal. It underlines, however, the point that pairwise fairness judgments are highly beneficial side-information for incorporating fairness into models, especially when comparing individuals from a-priori incomparable groups. The flexibility to incorporate a variety of fairness judgments, such as equivalence class judgments, per-group rankings) is a salient advantage of PFR, missing in prior works for fair representation learning. In this work however we make an assumption that there is consensus amongst the fairness judges. Further work is needed to address this assumption.

The experimental results suggest several key findings.

- *Individual Fairness - Utility Trade-off:* The extent of this trade-off depends on the degree of conflict between the fairness graph and the classifier's ground-truth labels. When edges in the fairness graph connect data points (for equally deserving individuals) that have different ground-truth labels, there is an inherent tension between individual fairness and utility.

  For datasets where some compromise is unavoidable, *PFR* turns out to perform best in balancing the different goals. It is consistently best with regard to individual fairness, by a substantial margin over the other methods. On utility, its *AUC* is competitive and always close to the best performing method on this metric, typically within 2 percentage points of the best *AUC* result.
- *Balancing Individual Fairness and Group Fairness:* The human judgements cast into the fairness graph help *PFR* to perform well also on group fairness criteria. On these measures, *PFR* is almost as good as the method by Hardt et al., which is specifically geared for group fairness (but disregards individual fairness). To a large extent, this is because the pairwise fairness judgments address historical subordination of groups. Eliciting human judgements is a crucial asset for fair machine learning in a wider sense.
- *Data Representation:* The graph-embedding approach of *PFR* appears to the best way of incorporating the pairwise human judgements. Alternative representations of the same raw information such as additional features in the input dataset, as leveraged by the augmented baselines (*LFR+*, *iFair+*), perform considerably worse than *PFR* on consistency (W^F).

  The W^F input is needed solely for the training data; previously unseen test data (at deployment of the learned representation and downstream classifier) does not have any pairwise judgments at all. This underlines the practical viability of *PFR*.
- *Graph Sparseness:* Even a small amount of pairwise fairness judgments helps *PFR* in improving fairness. At some point of extreme sparseness, *PFR* loses this advantage, but its performance degrades quite gracefully.
- *Robustness:* PFR is fairly robust to the dimensionality of the dataset. As the dimensionality of the input data increases, the performance of *PFR* drops a bit, but still outperforms other approaches in terms of balancing fairness and utility. Furthermore, *PFR* is quite insensitive to the choice of hyper-parameters. Its performance remains stable across a wide range of values.
- *Limitations:* When the data exhibits a strong conflict between fairness and utility goals, even *PFR* will fail to counter such tension and will have to prioritize either one of the two criteria while degrading on the other. The human judgements serve to mitigate exactly such cases of historical subordination and discrimination, but if they are too sparse or too noisy, their influence will be marginal. For the datasets in our experiments, we assumed that the information on equally deserving indidivuals would reflect high consensus among human judges. When this assumption is invalid for certain datasets, *PFR* will lose its advantages and perform as poorly as (but no worse than) other methods.

5.6 Conclusions

This chapter proposes a new departure for the hot topic of how to incorporate fairness in algorithmic decision making. Building on the paradigm of individual fairness, we devised a new method, called *PFR*, for operationalizing this line of models, by eliciting and leveraging side-information on pairs of individuals who are equally deserving and, thus, should be treated similarly for a given task. We developed a representation learning model to learn Pairwise Fair Representations (*PFR*), as a fairness-enhanced input to downstream machine-learning tasks. Comprehensive experiments, with synthetic and real-life datasets, indicate that the pairwise judgements are beneficial for members of the protected group, resulting in high individual fairness with reasonably low loss in utility.

Chapter 6

Responsible Deployment and Risk Mitigation

Contents

Reliably predicting potential failure risks of machine learning (ML) systems when deployed with production data is a crucial aspect of trustworthy AI. This chapter introduces *Risk Advisor*, a novel post-hoc *meta-learner* for predicting failure risks and estimating uncertainties of *any already-trained* black-box classification model. In addition to providing a *risk score*, the *Risk Advisor* decomposes the uncertainty estimates into aleatoric and epistemic uncertainty components, thus giving informative insights into the sources of uncertainty inducing the failures. Consequently, *Risk Advisor* can distinguish between failures caused by data variability, data shifts and model limitations and advise on mitigation actions (e.g., collecting more data to counter data shift). Extensive experiments on various families of black-box classification models and on real-world and synthetic datasets covering common ML failure scenarios show that the *Risk Advisor* reliably predicts deployment-time failure risks in all the scenarios, and outperforms strong baselines.

6.1 Introduction

Motivation and Problem: Machine learning (ML) systems have found wide adoption in mission-critical applications. Their success crucially hinges on the amount and quality of training data, and also on the assumption that the data distribution for the deployed system stays the same and is well covered by the training samples. However, this cannot be taken for granted. Saria and Subbaswamy [2019] categorize limitations and failures of ML systems into several regimes, including data shifts (between training-time and deployment-time distributions), high data variability (such as overlapping class labels near decision boundaries) and model limitations (such as log-linear decision boundaries vs. neural ML). Trustworthy ML needs models and tools for detecting such failure risks and analyzing the underlying sources of uncertainty. Unfortunately, systems often fail silently without any warning, despite showing high confidence in their predictions [Nguyen et al. 2015; Jiang et al. 2018; Goodfellow et al. 2015].

This chapter addresses the challenge of predicting, analyzing and mitigating failure risks for classifier systems. The goal is to provide the system with *uncertainty scores* for its predictions, so as to (a) reliably predict test-time inputs for which the system is likely to fail, and (b) detect the *type of uncertainty* that induces the risk, so that (c) appropriate *mitigation actions* can be pursued. Equipped with different kinds of uncertainty scores, a deployed system could improve its robustness in handling new data points that pose difficult situations. For instance, if an introspection component indicates that the ML system's output has a non-negligible likelihood of being erroneous, the system could abstain and defer the decision to a human expert (rather than risking adverse effect on human lives). When many production data points are out-of-distribution, collecting additional training samples and retraining the system would be a remedy. The challenge here is to determine which action is advised under which conditions. This is the problem addressed in this chapter: determine the amount and type of uncertainty in deployment-time inputs, so as to decide if and which kind of mitigation is needed.

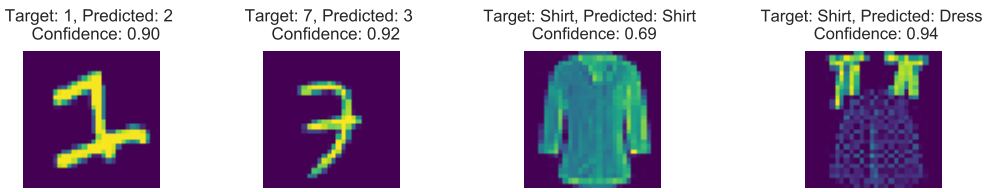


Figure 6.1: Examples of ground-truth (target) and predicted labels where (a,b) a CNN fails despite high confidence (MNIST dataset [LeCun et al. 2010]), and (c,d) a CNN assigns higher confidence to a misclassified sample than to a correct one (Fashion MNIST dataset [Xiao et al. 2017])

State of the Art and its Limitations: The standard approach for deciding whether an ML system's predictions are trustworthy is based on confidence scores computed over

predictive probabilities, such as max class probability (MCP) in neural ML [Hendrycks and Gimpel 2017] or distance from the decision boundary for SVM. However, predictive probabilities are not reliable estimates of a model's uncertainty [Gal and Ghahramani 2016; Jiang et al. 2018; Nguyen et al. 2015; Goodfellow et al. 2015]. Figure 6.1(a,b) shows two examples where a CNN model misclassifies handwritten digits (from the MNIST benchmark) while giving high scores for its (self-) confidence. Even if the confidence scores are calibrated , they may still not be trustworthy as the ordering of the confidence scores can itself be unreliable. This is because most calibration techniques (e.g., [Platt et al. 1999; Guo et al. 2017]), are concerned with scaling of the scores, i.e., they perform monotonic transformations with respect to prediction scores, which do not alter the ranking of confident vs. uncertain example. Fig. 6.1(c,d) shows two examples where a CNN model gives higher score to a misclassified sample than to a correct one.

More importantly, confidence scores do not reflect what the model *does not know*. In Fig. 6.1(c,d), the Fashion MNIST dataset has many positive training examples of shirts similar to (c) while hardly any examples that resemble (d) – a case where the training distribution does not sufficiently reflect the test-time data. Yet, the CNN model makes a prediction with high confidence of 0.94 (see Fig. 6.1d). This limitation holds even for the state-of-the-art model *Trust Score* [Jiang et al. 2018], which serves as a major baseline for this chapter.

Moreover and most critically, *confidence scores* are "one-dimensional" and do not provide any insight on which type of uncertainty is the problematic issue. Thus, confidence scores from prior works are limited in their support for identifying different types of appropriate risk mitigation actions.

A common line of work for uncertainty estimation builds on Bayesian methods [Denker and LeCun 1990; Barber and Bishop 1998], or making specialized changes to the learning algorithm (e.g.,[Gal and Ghahramani 2016; Depeweg et al. 2018; Lakshminarayanan et al. 2017; Shaker and Hüllermeier 2020; Malinin et al. 2021]). However, these are tightly coupled to the choice of the underlying classification model and thus involve making specialized modifications to the ML pipeline. Therefore, such techniques are unsuitable for dealing with a broad variety of black-box ML systems.

Proposed Approach: This chapter presents *Risk Advisor*, a generic and versatile framework for reliably estimating failure risks of any already-trained black-box classification model. The *Risk Advisor* consists of a post-hoc *meta-learner* for uncertainty estimation that is separate from the underlying ML system, and can be incorporated without any code changes in the underlying ML pipeline. The *meta-learner* is model-agnostic: it can be applied to any family of black-box classifiers (e.g., deep neural networks, decision-trees, etc). Fig. 6.2 gives a schematic overview of our framework.

In addition to providing a *risk score* that is more reliable than those of prior works, the *Risk Advisor* provides a refined analysis of the underlying types of uncertainty inducing the risks. To this end, we make use of the information-theoretic notions of *model uncertainty*, *aleatoric uncertainty* and *epistemic uncertainty* [Hora 1996; Der Kiureghian and Ditlevsen 2009; Senge et al. 2014]. These concepts are fairly old, but to the best of our knowledge, have not been considered for risk analysis of black-box ML systems. Our *Risk Advisor* quantifies each of the three risk types and thus enables judicious advise on risk mitigation action,

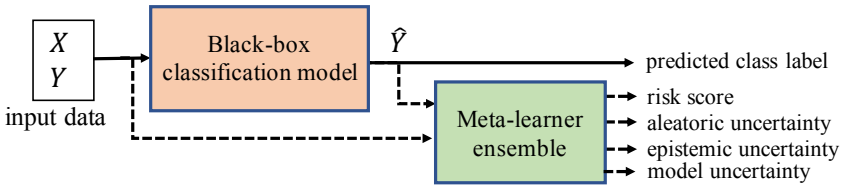


Figure 6.2: Schematic overview of the Risk Advisor framework.

depending on the type of uncertainty inducing the risks:

- *Aleatoric uncertainty* reflects the variability of data points and the resulting noise around the classifier's decision boundary. A high value indicates that it is inherently difficult to distinguish the output classes, and an appropriate mitigation then is to equip the deployed system with the option to *abstain* rather than forcing an output label. Fig. 6.1(a,b) is a case of high aleotoric uncertainty.

- *Epistemic uncertainty* captures systematic gaps in the training samples, like regions where training samples are sparse but have a substantial population of test points after deployment. This situation can only be countered by obtaining *more training data* for the underrepresented critical regions. Fig. 6.1(c,d) is a case of high epistemic uncertainty.

- *Model uncertainty* is an indicator that the black-box ML system uses models with insufficient learning capacity. In this situation, the proper action is to re-build the ML system with higher model capacity or a more expressive learning model, for example, a deep neural network instead of a log-linear model.

The proposed *meta-learner* for estimating the different types of uncertainty in the *Risk Advisor* framework is implemented as an ensemble of M stochastic gradient-boosted decision trees (E-SGBT). Each stochastic gradient boosted tree (SGBT) operates on the input-output pairs of training samples and an indicator variable stating whether the trained black-box ML system misclassified the training point. The *Risk Advisor's* analysis of uncertainty is based on the ensemble's ability to compute aleatoric and epistemic uncertainty. All of the uncertainty scores are computed on the training data, and also at deployment time for test data alone to identify slowly evolving risks.

Contributions: The state-of-the-art method to which we compare our approach is *Trust Score* [Jiang et al. 2018], which also operates in a model-agnostic post-hoc way. Trust Score is based on the distance between a test point and its nearest neighbors in the training data. Its output is a single value, which does not provide guidance on identifying the type of risk. n contrast, Risk Advisor yields refined scores for different kinds of uncertainty, being more informative towards risk analysis and mitigation. To the best of our knowledge, no prior work has come up with a model-agnostic approach to estimate uncertainty scores, distinguishing between model uncertainty, aleatoric uncertainy and epistemic uncertainty in a unified way.

This chapter's novel contributions are as follows:

- We introduce the *Risk Advisor* framework, the first *model-agnostic* method to detect and mitigate deployment-time failure risks, requiring access only to the base classifier's training data and its predictions and coping with any kind of underlying Black-box ML model.
- The *Risk Advisor* is the first method that leverages the information-theoretic notions of *aleatoric* and *epistemic* uncertainty to distinguish between ML model failures caused by distribution shifts between training data and deployment data, inherent data variability, and model limitations.
- Experiments with synthetic and real-world datasets show that our approach successfully detects uncertainty and failure risks for many families of ML classifiers, including deep neural models, and does so better than prior baselines including the Trust Score method by Jiang et al. [2018].
- We demonstrate the *Risk Advisor*'s practical utility by three kinds of risk mitigation: (i) selectively abstaining from making predictions under uncertainty (ii) detecting out-of-distribution test-examples (iii) countering risks due to data shift by collecting more training samples in a judicious way.

6.2 Related Work

The standard approach for predicting failure risks of ML systems is to rely on the system's native (self-) *confidence scores*. An implicit assumption is that that most uncertain data points lie near the decision boundary, and confidence increases when moving away from the boundary. While this is reasonable to capture *aleatoric* uncertainty, this kind of confidence score fails to capture *epistemic* and *model* uncertainty [Gal and Ghahramani 2016].

A related line of work is techniques for confidence calibration such as platt scaling [Platt et al. 1999], as well as modern neural network calibration approaches such as temperature scaling [Guo et al. 2017]. However, calibration approaches are concerned with rescaling the confidence scores to produce calibrarted proper scores. Hence, they cannot capture model uncertainty arising due to the model's own inductive bias. hus, these approaches cannot detect uncertainties that are not captured by the model's confidence in the first place. In other words, calibration approaches cannot detect model uncertainty arising due to the model's own inductive bias. Furter, like all single-dimensional notions of confidence, this is insufficient to distinguish different types of uncertainty and resulting risks. In particular, there is no awareness of epistemic uncertainty due to data shifts [Snoek et al. 2019].

Bayesian methods are a common approach to capture uncertainty in ML [Denker and LeCun 1990; Barber and Bishop 1998]. Recently, a number of non-Bayesian specialized learning algorithms were proposed to approximate Bayesian methods. For instance, variational learning [Honkela and Valpola 2004; Kendall and Gal 2017], drop-out [Gal and Ghahramani 2016], and ensembles of deep neural networks [Lakshminarayanan et al. 2017]. However, these models tend to be computationally expensive (by increasing network size and model parameters), and are not always practically viable. Moreover, they require changes to the

architecture and code of the underlying ML system. n contrast, *Risk Advisor* is a post-hoc model-agnostic approach that is uncoupled from the underlying ML system, and can be incorporated without changes to the ML pipeline.

The concepts of aleatoric and epistemic uncertainty are rooted in statistics and information theory [Hora 1996; Der Kiureghian and Ditlevsen 2009] ([Hüllermeier and Waegeman 2021] is a recent overview). [Senge et al. 2014] has incorporated these measures into a Bayesian classifier with fuzzy preference modeling. [Shaker and Hüllermeier 2020] integrated the distinction between aleatoric and epistemic uncertainty into random-forest classifiers to enhance its robustness. Both of these works are focused on one specific ML model and do not work outside these design points, whereas Risk Advisor is model-agnostic and as such universally applicable. [Shaker and Hüllermeier 2020] is included in the baselines for our experimental comparisons.

Several post-hoc approaches were proposed for estimating reliability scores and predicting test-time failures of already trained classifiers. Schulam and Saria [2019] proposed a post-hoc auditor to learn pointwise reliability scores. However, it is not model-agnostic as it relies on using gradients and the Hessian of the underlying ML model. Further, it does not differentiate between different types of uncertainty. Schelter et al. [2020] proposed a *model-agnostic* validation approach to detect *data-related* errors at serving time. However, this work focuses on errors arising from data-processing issues, such as missing values or incorrectly entered values, and relies on programmatic specification of typical data errors.

The closest approach to ours is *Trust Score* [Jiang et al. 2018], a model-agnostic method that can be applied post-hoc to any ML system. Trust Score measures the agreement between a classifier's predictions and the predictions of a modified nearest-neighbour classifier which accounts for density distribution. More precisely, the *trust score* for a new test-time data point is defined as the ratio between (a) the distance from the test sample to its nearest α-high density set with a *different* class and (b) the distance from the test sample to its nearest α-high density set with the *same* class. A crucial limitation of this approach is that it is highly sensitive to the choice of the distance metric for defining neighborhoods, and can degrade for high-dimensional data. Also, it does not provide any guidance on the type of uncertainty.

Classification with reject option [Bartlett and Wegkamp 2008] and selective abstention [El-Yaniv et al. 2010] are related problems, where the model can defer decisions (e.g., to to a human expert) when it has low confidence. However, these methods still rely on their own *confidence* scores to determine when to abstain, and thus share the limitations and pitfalls of a single-dimensional self-confidence. Similarly, the problem of detecting data shifts has been widely studied e.g., for detecting and countering covariate and label shift[Schneider et al. 2020] and for anomaly detection [Ben-Gal 2010; Steinwart et al. 2005]. These methods address data shifts, but they do not consider failure risks arising from *aleatoric* and *model* uncertainty.

6.3 Risk Advisor Model

6.3.1 Basic Concepts

Black-box Classifier's Task: We are given a training dataset $\mathcal{D} = \{(x_i, y_i) \cdots (x_n, y_n)\} \subset \mathcal{X} \times \mathcal{Y}$ drawn from an unknown data generating distribution $\mathcal{P} \sim \mathcal{X} \times \mathcal{Y}$. The goal of the *black-box classifier* is to learn a hypothesis h that minimizes the expected empirical risk over observed training distribution $\mathcal{D}$.

$$h^* = \arg\min_h \mathbb{E}_{(x,y)\in\mathcal{D}} \ell(h(x), y) \tag{6.1}$$

where $\ell(\cdot)$ is classification loss function (e.g., cross-entropy between predicted and ground-truth labels), and $\hat{y} = h(x)$ is the corresponding predicted class label.

Black-box Classifier's Uncertainty: The degree of uncertainty in a prediction can be measured by the Shannon entropy over the outcomes for any given test point. Higher entropy corresponds to higher uncertainty.

$$H[Y|X] = -\sum_{y\in\mathcal{Y}} P(y|x, \mathcal{D}) \log_2 P(y|x, \mathcal{D}) \tag{6.2}$$

The overall uncertainty corresponding to the predictive task denoted as $H[Y|X]$ encompasses uncertainty due to aleatoric, epistemic and model uncertainty [Hora 1996; Der Kiureghian and Ditlevsen 2009; Senge et al. 2014].

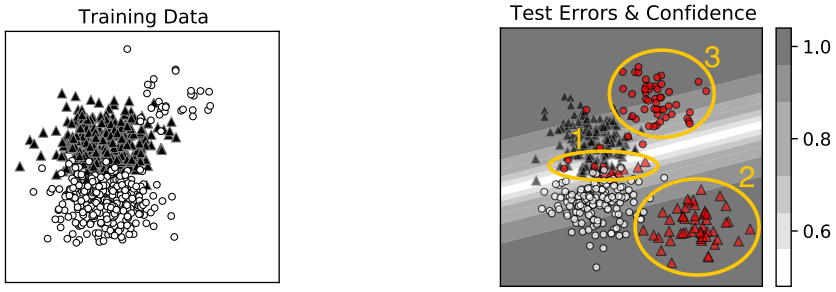
6.3.2 Mapping Failure Scenarios to Uncertainties

Next, we give a brief introduction to the types of uncertainties – aleatoric, epistemic and model uncertainty – in a predictive task, and draw a connection between predictive uncertainties and common sources of failures in ML system.

Example: These different kinds of uncertainty are illustrated via a synthetic example in Fig. 6.3. We will use this as running example to motivate the proposed approach. Consider the classification task dataset in Fig. 6.3. The position on x-axis and y-axis represents input features. The markers (black triangles and white circles) represent binary class labels. A linear SVM classifier, for example, would learn a decision boundary that best discriminates the two classes as shown in Fig. 6.3b. The test-time errors made by the model are highlighted in red. The model's errors can be mapped to different types of uncercetainties as follows:

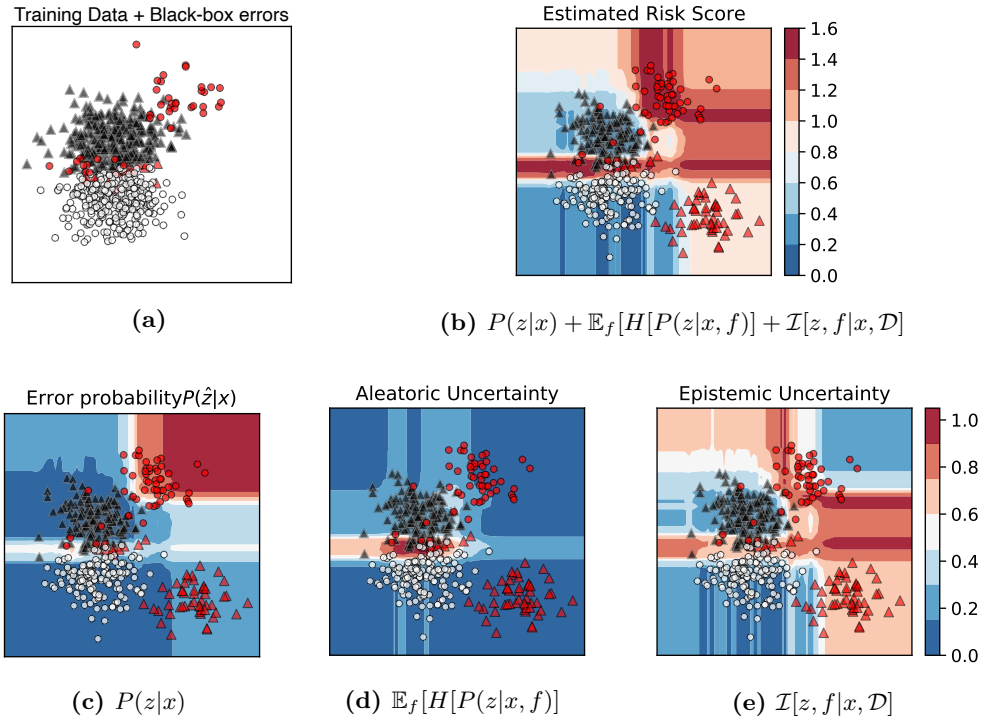
Firstly, in many predictive tasks Y can rarely be estimated deterministically from X due to inherent stochasticity in the dataset, a.k.a *aleatoric* uncertainty. For instance, errors arising due to inherent data variability and noise, marked as Region 1 in Fig. 6.3b). Such errors are inherently *irreducible* (unless additional features are collected). Additionally, there is uncertainty arising due to "lack of knowledge" about the true data generating process. For instance, consider the test errors caused by shifts in the data distribution, marked as Region 2. Such errors due to *epistemic* uncertainty can in principle be mitigated by collecting additional training data and retraining the model. Further, ML models have additional uncertainty in estimating the true model parameters given limited training data.

For instance, consider *systematic* errors arising due to fitting a linear model to non-linear data, marked as Region 3. Errors due to *model* uncertainty can in principle be addressed (e.g., by training a model from a different model class).



(a) **(b)**

Figure 6.3: Example: (a) Training data for classification task (b) learned decision boundary of an SVM classifier and different types of test-time errors, e.g., due to (1) data variability and noise (2) data shift, and (3) model limitations.



(a) **(b)** $P(z|x) + \mathbb{E}_f[H[P(z|x,f)] + \mathcal{I}[z,f|x,\mathcal{D}]$

(c) $P(z|x)$ **(d)** $\mathbb{E}_f[H[P(z|x,f)]$ **(e)** $\mathcal{I}[z,f|x,\mathcal{D}]$

Figure 6.4: Meta-learner: (a) Training input to *meta-learner* (b) *meta-learner's* estimated overall *risk score* (c, d, e) decomposition of the overall risk score into its various constituting components, i.e., (c) model, (d) aleatoric and (e) epistemic uncertainty, that capture errors due to (c) model limitations, (d) data variability and noise, and (e) data shift, respectively.

6.3.3 Design Rationale

We draw inspiration from *Fano's Inequality* [Fano 1961; Cover 1999], a classic information-theoretic inequality which when viewed from a ML perspective draws a connection between predictive uncertainty $H[Y|X]$, uncertainty in error prediction $H[Z|X]$, and probability of error $P(Z|X)$ of a Bayes optimal classifier, where Z is a random variable indicating prediction error $Z := \mathbb{I}(Y \neq \hat{Y})$.

Fano's Inequality [Fano 1961; Cover 1999]: *Consider random variables X and Y, where Y is related to X by the joint distribution $P(x, y)$. Let $\hat{Y} = h(X)$ be an estimate of Y, with the random variable Z representing an occurrence of error, i.e., $Z := \mathbb{I}(Y \neq \hat{Y})$. Fano's inequality states that*

$$H[Y|X] \leq H[Z|X] + P(Z|X) \cdot \log_2(|\mathcal{Y}| - 1) \tag{6.3}$$

where $|\mathcal{Y}|$ is the number of classes, H is Shannon entropy, and $P(Z|X)$ is probability of error.

Key Idea: The conditional entropy $H[Z|X]$ and the error probability $P(Z|X)$ in Eq. 6.3 are not known, but we can approximate them by computing empirical estimates of conditional entropy $H_f[Z|X]$ and error probability $P_f(Z|X)$ of a separate *meta-learner* $f : X \to Z$ whose goal is to predict errors Z made by the underlying black-box classifier h with respect to the original classification task.

Given such a meta-learner f, we argue that a black-box model's classification errors on unseen data, which relate to the uncertainty $H[Y|X]$, can be estimated by combining f's predicted probability of error $P_f(Z|X)$ and f's own uncertainty corresponding to predicting errors $H_f[Z|X]$.

Example: Let us revisit the synthetic example of Fig. 6.3, looking at it from a meta-learner's perspective. Fig. 6.4 shows different perspectives on this setting.

Fig. 6.4a visualizes the input to the meta-learner, which consists of training datapoints X and the black-box model's training errors Z (highlighted in red). Observe that errors due to *model limitations* (top right red points) appear as systematic errors in the input space, and are *predictable*. We argue that by training a *meta-learner* to predict black-box classification model's errors, we can capture these systematic errors due to model limitations with meta-learner's predicted *error probabilities* $P_f(Z|X)$, as shown in Fig. 6.4c.

Further, recall that both aleatoric and epistemic uncertainties are related to the underlying training data. We posit that the meta-learner, which is trained on the same data samples as the black-box classifier, inherits these data-induced uncertainties, and this is reflected in the meta-learner's *aleatoric* and *epistemic* uncertainties, as shown in 6.4d and Fig. 6.4e.

The intuition is as follows. Consider the region near the decision boundary in Fig. 6.4a. As the meta-learner sees both *failure* and *success* cases of the black-box classifier in this region, the meta-learner, too, has *aleatoric uncertainty* in this region of inherent noise. Similarly, consider the test points situated far away from the training data. The meta-learner would also have significant epistemic uncertainty in its error prediction, as it has not seen any training data in this region. Thus, by estimating the meta-learner's *own* aleatoric and epistemic uncertainty, we can indirectly capture the black-box classifier's aleatoric and

epistemic uncertainty, as shown in Fig. 6.4d and Fig. 6.4e, respectively. In our experiments, we will present empirical evidence of these insights.

Putting these three insights together, we propose the combined notion of *risk score*, as shown in Fig. 6.4b. The estimated risk score (background color) is high in the regions of actual test-time errors.

In the following, Subsection 6.3.4 formalizes the meta-learner's task, and presents our proposed meta-learner ensemble for the Risk Advisor. Subsection 6.3.5 discusses how to refine the overall uncertainty into informative components for different kinds of uncertainty, and compute overall *risk score*.

6.3.4 Meta-learner Ensemble

Meta-learner's Task: Given input training samples $x \in X$, predicted class labels $\hat{y} := h(x)$ of a fully trained *black-box classifier* h, and a random variable $Z := \mathbb{I}(Y \neq \hat{Y})$ indicating errors of the *black-box classifier* h with respect to the original classification task. Our goal is to learn an meta-learner $f : X \to Z$ trained to predict errors of the *black-box classifier* with respect to the original task given by

$$f = \arg\min_{f \in \mathcal{F}} \mathbb{E}_{(x,z)\in\mathcal{D}} \ell(f(x), z) \tag{6.4}$$

where z is a random variable indicating errors of the base classifier predictor given by $z = \mathbb{I}(y \neq \hat{y})$, ℓ is a classification loss function. Given a newly seen test point x^*, the *meta-learner's* predicted probability of error is given by $P(z|f, x^*)$.

However, the probability of error $P(z|f, x^*)$ estimated by a single meta-learner f can be biased due to its own uncertainty in the model parameters $P(f|\mathcal{D})$. Next, we show how we can obtain a reliable estimate of the black-box model's error probability by training an ensemble of M independent stochastic gradient boosted trees $\mathcal{F} = \{P(z|x^*, f^m)\}_{m=1}^M$, and computing their expectation.

Ensemble of Stochastic Gradient Boosted Trees (E-SGBT): We consider an ensemble of M independent models $\mathcal{F} = \{f^m\}_{m=1}^M$ such that each of the individual models f^m is a stochastic gradient boosted tree (SGBT) [Friedman 2002]. Note that the proposed *E-SGBT* is an ensemble of ensembles, i.e., each of the M SGBT's in the ensemble is itself an ensemble of T weak learners trained iteratively via bootstrap aggregation. To ensure minimum correlation between the M individual models in our ensemble, we introduce randomization in two ways. First, each of the SGBTs in the ensemble is initialized with a different random seed. Second, each of the individual SGBTs is itself an ensemble of T weak learners trained iteratively via bootstrap aggregation. Specifically, for each SGBT in the *E-SGBT* ensemble, at each iteration, a subsample of training data of size $\tilde{N} < N$ is drawn at random, without replacement, from the full training dataset. The fraction $\frac{\tilde{N}}{N}$ is called the sample rate. The smaller the sample rate, the higher the difference between successive iterations of the weak learners, thereby introducing randomness into the learning process.

Given M error probability estimates $\{P(z|x, f^m)\}_{m=1}^M$ by each of the models in the ensemble, an estimate of the probability of error $P(z|x, \mathcal{D})$ can be computed by taking the

expectation over all the models in the ensemble:

$$P(z|x,\mathcal{D}) := \mathbb{E}_{f\in\mathcal{F}}[P(z|x,f,\mathcal{D})] \approx \frac{1}{M}\sum_{m=1}^{M} P(z|x,f^m,\mathcal{D}) \tag{6.5}$$

The total uncertainty in the error prediction $H[P(z|x,\mathcal{D})]$ can be computed as the Shannon entropy corresponding to the estimated probability of error

$$H[P(z|x,\mathcal{D})] = -\sum_{z\in\mathcal{Z}} P(z|x,\mathcal{D})\log_2 P(z|x,\mathcal{D}) \tag{6.6}$$

6.3.5 Identifying Sources of Uncertainty

To distinguish between different sources of uncertainty – data variariability/noise vs. data shifts between training and deployment data – we compute estimates of the *aleatoric* and *epistemic* uncertainty given an ensemble of M independent stochastic gradient boosted trees $\mathcal{F} = \{f^m\}_{m=1}^M$. This approach was originally developed in the context of neural networks [Depeweg et al. 2018], but the idea is more general and has recently been applied using ensembles of gradient boosted trees and random forests [Malinin et al. 2021; Shaker and Hüllermeier 2020].

Decomposing Aleatoric and Epistemic Uncertainty: The main idea is that in the case of data points with epistemic uncertainty (e.g., out-of-distribution points), the M independent models in the ensemble given $\mathcal{F} := \{f^m\}_{m=1}^M$ are likely to yield a diverse set of predictions (i.e., different output labels) for similar inputs. In contrast, for data points with low epistemic uncertainty (e.g., in-distribution points in dense regions), they are likely to agree in their predictions. Hence, by fixing f, the *epistemic* uncertainty can be removed, and the *aleatoric* uncertainty can be computed by taking the expectation over all models $f \in \mathcal{F}$.

$$\mathbb{E}_{p(f|\mathcal{D})}H[P(z|x,f)] = \int_{\mathcal{F}} P(f|D)\cdot H[P(z|f,x)]df \tag{6.7}$$

Aleatoric Uncertainty: Given M predicted probability estimates $\{P(z|x,f^m)\}_{m=1}^M$ for each of the models in the ensemble, an estimate of *aleatoric uncertainty* in Eq. 6.7 can be empirically approximated by averaging over individual models $f^m \in \mathcal{F}$ in our *E-SGBT* ensemble.

$$\mathbb{E}_{f\in\mathcal{F}}[H[P(z|x,f)]] \approx \frac{1}{M}\sum_{m=1}^{M} H[P(z|x,f^m)] \tag{6.8}$$

Epistemic uncertainty: Finally, *epistemic uncertainty* can be computed as the difference between *total uncertainty* and *aleatoric uncertainty*.

$$\underbrace{\mathcal{I}[z,f|x,\mathcal{D}]}_{\text{Epistemic Uncertainty}} = \underbrace{H[P(z|x,\mathcal{D})]}_{\text{Total Uncertainty}} - \underbrace{\mathbb{E}_{f\in\mathcal{F}}[H[P(z|x,f)]]}_{\text{Aleatoric Uncertainty}} \tag{6.9}$$

where *total uncertainty* is the entropy corresponding to the estimated probability of error $P(z|x,\mathcal{D})$ given in Eq. 6.6.

Risk Score: Putting it all together, our proposed *Risk Score*, which captures black-box model errors arising due to all sources of uncertainty, can be computed as the sum of (i)

predicted probability of error assigned by the meta-learner, i.e., model uncertainty, (ii) epistemic uncertainty and (iii) aleatoric uncertainty.

$$\begin{aligned} \text{Risk Score} := & \underbrace{P(z|x,\mathcal{D})}_{\text{Error probability}} + \underbrace{H[P(z|x)]}_{\text{Total uncertainty}} \\ = & \underbrace{P(z|x,\mathcal{D})}_{\text{Model Uncertainty}} + \underbrace{\mathcal{I}[z,f|x,\mathcal{D}]}_{\text{Epistemic uncertainty}} + \underbrace{\mathbb{E}_f[H[P(z|x,f)]}_{\text{Aleatoric uncertainty}} \end{aligned} \tag{6.10}$$

Note that this *risk score* is neither a probability nor an entropy measure, but it proves to be a very useful indicator for failure risks in our experiments. One could consider a weighted sum of each of the components to account for associated *risk costs* for each type of error. For instance, if a system designer had expert knowledge that errors due to distribution shift (i.e., epistemic uncertainty) are more harmful, she could assign more weight to the *epistemic uncertainty* component. In our experiments we assign equal weights.

Inference: The meta-learner is trained on the underlying base-classifier's training data. Given a newly seen test point x^* at deployment-time, the *Risk Advisor* computes predicted error probabilities for each of the M models in the E-SGBT ensemble $\{P(z|x^*, f^m)\}_{m=1}^{M}$. These values are fed into the *Risk Advisor's* estimated *error probability* in Eq. 6.5, aleatoric uncertainty in Eq. 6.8, epistemic uncertainty in Eq. 6.9 and risk score in Eq. 6.10. Note that at deployment-time, we only expect the unseen data point x^*, and the trained meta-learner.

6.4 Synthetic-Data Experiments

In this section, we evaluate *Risk Advisor's* ability to detect the sources of uncertainty inducing the failure risk. To this end, we systematically generate synthetic datasets covering a variety of ML failure scenarios including errors due (i) black-box classifier's model limitations (e.g., applying a linear model to non-linear decision boundary) (ii) data shift and (iii) inherent data variability and noise. We then evaluate if the *Risk Advisor's* estimates for *model*, *epistemic*, and *aleatoric* uncertainty can correctly capture the corresponding test-time errors made by the black-box classification model.

Errors due to Black-box Classifier's Model Limitations: In order to simulate this scenario, we construct a classification dataset with a non-linear decision boundary, i.e., two concentric circles Pedregosa et al. [2011]. We then fit a misspecified classification model to the task, i.e., a logistic regression classifier with a (log-)linear decision boundary as shown in Fig. 6.5. The contour plot in Fig. 6.5a visualizes the training data and the learned decision boundary. Fig. 6.5b visualizes the test data. Test-set errors made by the black-box model are highlighted in red.

The contour plot in Fig. 6.5c visualizes the *Risk Advisor's* predicted *Error probability* ($P(\hat{z}|x)$). Ideally, we would expect the *Risk Advisor* to assign a higher error score for regions of the input space where the black-box classifier makes errors due to its model limitations. We clearly see this trend: the *Risk Advisor* correctly identifies the regions where the black-box classifier is likely to make errors due to its incorrect linear decision boundary. This is especially remarkable given that the *Risk Advisor* has no knowledge of the model family of the underlying black-box model (e.g., whether it is log-linear model or a neural network). In

spite of having no information about the underlying model (other than its predictions), the *Risk Advisor* is able to correctly capture the *model uncertainty*.

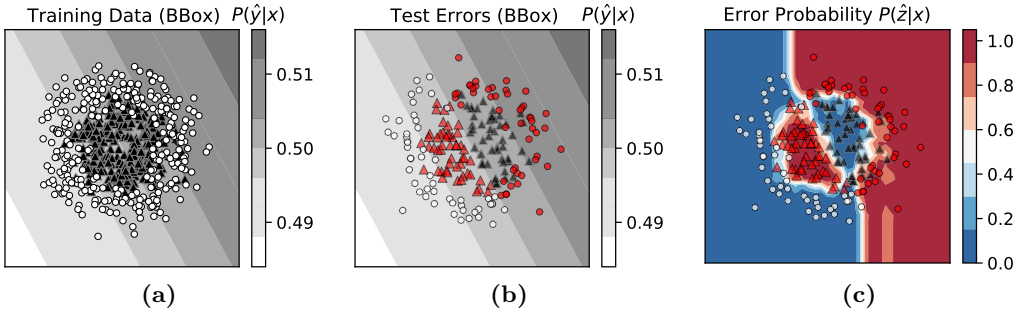


Figure 6.5: Errors from model limitations: *Risk Advisor's* estimated error probability $P(z|x)$ correctly identifies errors due to model limitation.

Errors due to Distribution Shift: In order to simulate a distribution shift scenario, we draw points from a mixture of two Gaussians. For the training points we set the mixture coefficient for one of the Gaussians to zero; for the test points both mixture components are active. This way, we are able to construct a dataset containing out-of-distribution test points as shown in Fig. 6.6. Fig. 6.6a visualizes the training data and the decision boundary learned by a 2-layer feed-forward neural network (NN). Fig. 6.6b visualizes the test data. Test errors of the NN are highlighted in red. Observe that the NN *misclassifies* out-of-distribution test points while (incorrectly) reporting high confidence. The contour plot in Fig. 6.6c visualizes *Risk Advisor's* estimated *epistemic* uncertainty.

Ideally, we would like to see that the *epistemic* uncertainty increases as we move towards the sparse regions of the training data, and that it is high for out-of-distribution regions. Despite some noise, we clearly see this trend: regions of low epistemic uncertainty (i.e., dark-blue regions) coincide with the dense in-distribution test points. *Epistemic* uncertainty increases as we move towards sparse regions, and the values are especially high for out-of-distribution regions (bottom right in Fig. 6.6c).

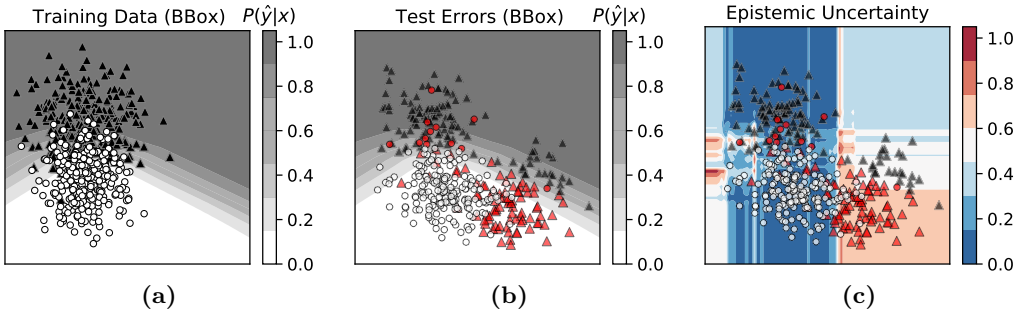


Figure 6.6: Errors from distribution shift: Risk-Advisor's epistemic uncertainty correctly identifies test points far away from training distribution.

Errors due to Data Variability and Noise: To simulate a dataset with inherent noise, we draw points from the classic two-moons dataset Pedregosa et al. [2011], and add Gaussian noise with standard deviation 0.5 to the dataset as shown in Fig. 6.7. Fig. 6.7a visualizes the training data and the decision boundary learned by a 2-layer feed-forward neural network (NN). Fig. 6.7b visualizes the test data. Test-errors are highlighted in red.

The contour plot in Fig. 6.7c visualizes estimated *aleatoric* uncertainty. Ideally, we would expect that *aleatoric* uncertainty is high for the regions with large class overlap. We clearly see this trend: the estimated *aleatoric* uncertainty is high for the test points near the decision boundary, with high class overlap.

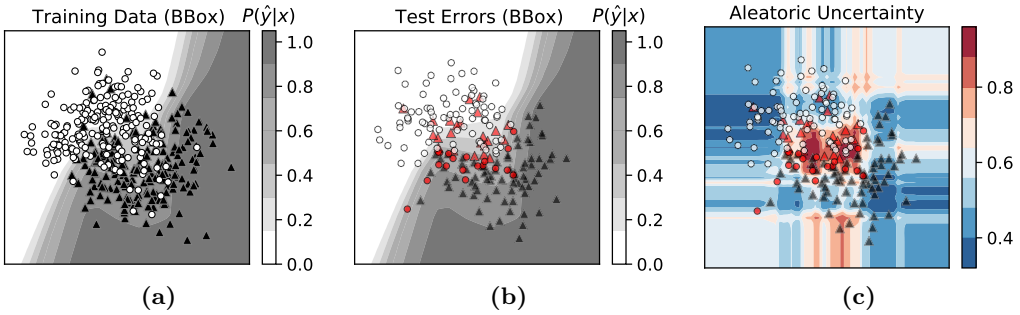


Figure 6.7: Errors from data variability and noise: Risk Advisor's aleatoric uncertainty correctly identifies test points in the regions with class overlap.

6.5 Real-World-Data Experiments

In this section, we evaluate the performance of *Risk Advisor* by performing extensive experiments on 9 real-world datasets and on 6 families of black-box classification models. First, we evaluate the *Risk Advisor's* ability to *predict failure risks* at deployment time (Subsection 6.5.2). Next, we investigate its performance on a variety of applications for *risk mitigation*, including (i) selectively abstaining under uncertainty (Subsection 6.5.3) (ii) detecting out-of-distribution test examples (Subsection 6.5.4) and (iii) mitigating risk by judiciously collecting additional samples for re-training the system (Subsection 6.5.5).

6.5.1 Experimental Setup

Datasets: We perform our evaluation on the following small and large benchmark classification datasets covering a variety of common ML failure scenarios:

High-dimensional Image Datasets:

- *CIFAR 10:* The CIFAR-10 dataset [Krizhevsky 2009] consists of 60K color images in 10 classes, including blurred and noisy images, which are specially prone to model failures.
- *MNIST:* The MNIST dataset [LeCun et al. 2010] consists of 60K grayscale images of handwritten digits in 10 classes. Due to the variability in writing style, certain images are prone to misclassification.

- *Fashion MNIST:* The fashion MNIST dataset [Xiao et al. 2017] consists of 60K images of clothing and accessories in 10 classes, including images with rare and unusual product designs, which can be prone to errors.

Mission-critical Fairness Datasets:

- *Census Income:* Recent work in ML fairness has shown that models often make more errors for underrepresented groups in training data. To simulate this setting, we consider the Adult dataset [Dua and Graff 2017], a benchmark dataset in fairness literature, consisting of 49K user records. The dataset contains underrepresented groups (e.g., Female).
- *Law School:* Similarly, we use the LSAC dataset [Wightman 1998] consisting of 28K law school admission records. The classification task is to predict whether a candidate would pass the bar exam. The dataset contains underrepresented groups (e.g., "Black").

Distribution shift, unseen demographics/regions/domain:

- *Census Income (Male $\rightarrow$ Male, Female):* To simulate distribution shift, we take the aforementioned *Census Income* dataset, and exclude *female* points from the training set. Our test set consists of both Male and Female points.
- *Law School (White $\rightarrow$ White, Black):* Similarly, we take the aforementioned *Law School* dataset, and exclude user records from the *Black* demographic group from the training set. The test set consists of both White and Black points.
- *Heart Disease:* A common ML failure scenario is when a ML model is applied to a new geographic region. To simulate this scenario we combine four different heart disease datasets available in the UCI repository [Dua and Graff 2017] by using a subset of features overlapping between them. We use the US Cleveland heart disease dataset as our training dataset, and use it to predict heart disease on a UK statlog dataset, Hungarian (HU) and Switzerland (CH) heart disease dataset.
- *Wine Quality:* Another failure scenario is when a trained model is applied to an application domain for which it has inadequate or bad training data. To simulate this scenario, we train models on white wine, and apply it to predict quality of red wine in UCI wine dataset [Dua and Graff 2017]. The classification task is to predict if the wine quality is ≥ 6.

Black-box Classification Models: To demonstrate the versatility of *Risk Advisor*, we evaluate it on classifiers from 6 different families, including deep neural models such as ResNet50 and CNN for the high dimensional image dataset, and classic ML algorithms such as SVM, Random Forests, Multi-layer Perceptron, and logistic regression for tabular datasets. Following are the implementation details:

- ResNet 50: The 50-layer deep residual network architecture [He et al. 2016] trained with batch size of 128 for 100 epochs.
- CNN: A convolutional neural net with 2 convolutional layers with 32, 64 hidden units, max pooling, and ReLu activations, trained with batch size 128 for 10 epochs.
- MLP: Multi-layer perceptron with 2 hidden layers with $32, 16$ hidden units, batch size

64, and ReLu activations.

- SVM: support vector machines with RBF kernel and Platt scaling [Platt et al. 1999] to produce probability estimates.
- RF: a random forest with 1000 decision trees, bootstrap sampling, and max-features set to 'sqrt'.
- LR: logistic regression with L2 regularization.

State-of-the-art Baselines: Our baseline comparison includes the underlying black-box classification model's own (self-) *confidence scores*. Specifically, for all deep neural models, i.e., ResNET50, MLP, and CNN, we rely on the confidence score given by max class probability (DNN-MCP), as proposed by [Hendrycks and Gimpel 2017], which is a well established strong baseline. For RF's, we rely on the *uncertainty* score, computed as per the state-of-the-art method for random forest (RF-uncertainty), as proposed by [Shaker and Hüllermeier 2020]. For SVM, we rely on the standard approach of computing confidence scores over prediction probabilities from decision values after Platt scaling (SVM-Platt) [Platt et al. 1999]. For LR, the confidence score is given by the distance from the decision boundary (LR-Confidence).

Our main comparison is with the state-of-the-art method *Trust Score* [Jiang et al. 2018]. Similar to *Risk Advisor*, Trust Score is a model-agnostic post-hoc approach, which takes as input a black-box classifier's predictions, and training data to produce point-wise trust scores for newly seen test points.

While calibrating a classifier's scores is a popular technique for producing calibrated confidence values, such techniques are rank-preserving. As all our evaluation metrics, i.e., AUROC, AUPR, and PRR (introduced later in Subsections 6.5.2, 6.5.4 and 6.5.3) are based on seeing different relative rankings of the scores rather than absolute values, there is no point in comparing against rank-preserving calibration techniques.

Implementation: The *Risk Advisor* is implemented as an ensemble of 10 SGBT classifiers, each initialized with a different random seed. Train and test sets are constructed using a 70:30 stratified split. All categorical features are one-hot encoded. Best hyper-parameters are chosen via grid-search by performing 5-fold cross validation. For *Risk Advisor's* E-SGBT model, we tune max-depth in [3,4,5,6], sample-rate in [0.25, 0.5, 0.75] and num-estimators in [100, 1000]. For *Trust Score*, we use the code shared by [Jiang et al. 2018] and perform grid search over the parameter space reported in the paper. All experiments are conducted using scikit-learn and Keras on 2 GPUs and CPUs. Results reported are mean values over 5 runs.

6.5.2 Predicting Test-time Failure Risks

First, we evaluate to what extent the *Risk Advisor* can successfully detect test points misclassified by the underlying ML system. We measure the quality of failure prediction using standard metrics used in the literature [Hendrycks and Gimpel 2017]: area under ROC curve (AUROC) and area under precision recall curve (AUPR), where misclassifications are chosen as the positive class.

Results: Tables 6.1 and 6.2 shows a comparison between the black-box models' own *confidence scores* [Hendrycks and Gimpel 2017; Platt et al. 1999; Shaker and Hüllermeier

Table 6.1: AUROC for predicting test-time failure risks: Values in the table are area under ROC curve (AUROC). Higher values are better.

Dataset	CIFAR 10	Fashion MNIST	MNIST	Census Income				Law School				Wine Quality white → white, red				Heart Disease US → US, UK, CH, HU				Census Income Male → Male, Female				Law School White → White, Black			
Black-box (BBox) classification model	ResNet50	CNN	CNN	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM
LR-Confidence	-	-	-	**0.80**	-	-	-	0.70	-	-	-	0.62	-	-	-	0.70	-	-	-	0.78	-	-	-	0.71	-	-	-
SVM-Platt [130]	-	-	-	-	-	-	0.83	-	-	-	0.75	-	-	-	0.72	-	-	-	0.76	-	-	-	0.85	-	-	-	0.75
DNN-MCP [74]	0.78	0.90	**0.98**	-	**0.80**	-	-	-	0.76	-	-	-	0.57	-	-	-	0.68	-	-	-	0.78	-	-	-	0.76	-	-
RF-uncertainty [143]	-	-	-	-	-	0.75	-	-	-	0.71	-	-	-	0.65	-	-	-	**0.75**	-	-	-	0.69	-	-	-	0.70	-
Trust score [84]	0.64	0.88	0.96	0.65	0.65	0.71	0.71	0.69	0.69	0.83	0.81	**0.67**	0.71	0.69	0.73	0.70	0.66	0.65	0.69	0.62	0.62	0.67	0.67	0.70	0.68	0.83	0.82
Riskscore (Proposed)	**0.80**	**0.92**	**0.98**	**0.80**	**0.80**	**0.87**	**0.86**	**0.83**	**0.77**	**0.86**	**0.86**	0.66	**0.75**	**0.74**	**0.75**	**0.78**	**0.72**	0.72	**0.79**	**0.79**	**0.79**	**0.87**	**0.87**	**0.83**	**0.77**	**0.86**	**0.85**

Table 6.2: AUPR for Predicting test-time failure risks: Values reported in the table are area under PR curve (AUPR). Higher values are better.

Dataset	CIFAR 10	Fashion MNIST	MNIST	Census Income				Law School				Wine Quality white → white, red				Heart Disease US → US, UK, CH, HU				Census Income Male → Male, Female				Law School White → White, Black			
Black-box (BBox) classification model	ResNet50	CNN	CNN	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM
Random Baseline	0.30	0.11	0.01	0.16	0.16	0.22	0.22	0.19	0.15	0.28	0.31	0.31	0.37	0.35	0.37	0.28	0.20	0.19	0.25	0.18	0.19	0.26	0.28	0.21	0.17	0.31	0.34
LR confidence	-	-	-	0.39	-	-	-	0.37	-	-	-	0.42	-	-	-	0.42	-	-	-	0.40	-	-	-	0.41	-	-	-
SVM-Platt [130]	-	-	-	-	-	-	0.52	-	-	-	0.48	-	-	-	0.56	-	-	-	0.47	-	-	-	0.63	-	-	-	0.50
DNN-MCP [74]	**0.58**	0.46	0.31	-	**0.38**	-	-	-	0.38	-	-	-	0.43	-	-	-	0.39	-	-	-	0.39	-	-	-	0.40	-	-
RF uncertainty [143]	-	-	-	-	-	0.42	-	-	-	0.43	-	-	-	0.46	-	-	-	0.40	-	-	-	0.42	-	-	-	0.44	-
Trust score [84]	0.43	0.47	**0.36**	0.22	0.21	0.33	0.33	0.37	0.29	0.66	0.64	**0.49**	0.54	0.51	**0.59**	0.47	**0.39**	0.33	0.40	0.23	0.23	0.35	0.37	0.41	0.30	**0.66**	**0.67**
Riskscore (Proposed)	**0.59**	**0.52**	**0.36**	**0.40**	0.37	**0.61**	**0.60**	**0.56**	**0.44**	**0.67**	**0.66**	0.45	**0.61**	**0.56**	0.58	**0.54**	0.37	**0.45**	**0.58**	**0.42**	**0.42**	**0.68**	**0.69**	**0.58**	**0.42**	**0.66**	0.66

2020], *Trust Score* [Jiang et al. 2018] and the *Risk Advisor*'s estimated *risk score*, for all combinations of datasets and black-box models. Table 6.1 reports AUROC for detecting test-set errors of the underlying black-box classifiers. Best values are marked in bold. We make the following observations.

First, we observe that AUROC values for all the methods are higher than a random baseline (AUROC of 0.5), indicating that all the approaches are informative in detecting test errors. Second, the proposed *risk score* consistently *outperforms* black-box models' own confidence scores (barring a few exceptions). This holds true for all families of black-box classifiers including deep neural models and Random Forests, which build on DNN-MCP [Hendrycks and Gimpel 2017] and RF-uncertainty [Shaker and Hüllermeier 2020]. Finally, we observe that our Risk Advisor's *risk scores* consistently *outperform Trust Scores* by a significant margin, for all the datasets and all families of black-box classifiers. Similar trends hold for the AUPR metric as shown in Table 6.2.

6.5.3 Application: Risk Mitigation by Selective Abstention

A benefit of predicting failure risks at deployment time is that we can take meaningful *risk mitigation* actions. For instance, if we expect that a ML system is likely to misclassify certain deployment/test-points, we can ask the ML system to abstain from making predictions and instead forward these data points to a fall-back system or human expert. In this experiment, we simulate the latter scenario as follows.

Setup and Metric: We generate a ranking of all the test points by ordering them according to the scores assigned by each approach, i.e., black-box model's *confidence score* (ascending order), *trust score* (ascending order), and *Risk Advisor*'s *risk score* (descending order), respectively. We then use these rankings to choose test points to defer to an *oracle*, in which case the ML systems predictions are replaced with the *oracle's* labels. This setup allows us to compute an *Accuracy-Rejection curve* (AR curve) [Malinin 2019; Bartlett and Wegkamp 2008; El-Yaniv et al. 2010]. AR curves are summarized using *prediction rejection ratio* (PRR), a metric which measures the degree to which the uncertainty scores are informative [Malinin 2019]. The *PRR* score lies between 0.0 and 1.0, where 1.0 indicates perfect ordering, and 0.0 indicates 'random' ordering.

Results: Table 6.3 shows a comparison between the black-box models' own *confidence scores* [Hendrycks and Gimpel 2017; Platt et al. 1999; Shaker and Hüllermeier 2020], *Trust Scores* [Jiang et al. 2018] and the proposed *risk scores*. Values in the table are PRR values for all combinations of datasets and models. Best results are highlighted in bold. We make the following observations.

First, all methods under comparison have a PRR > 0, indicating that all the approaches are informative and better than a random baseline (with random abstention). Second, *risk scores* consistently yield the *best PRR* across all datasets and classification models (barring a few exceptions). There is no clear winner between Trust Scores and each of the black-box classifiers' native confidence scores.

Table 6.3: Risk mitigation by selective abstention. Values in the table are prediction rejection ratio (PRR). Higher values are better.

Dataset	CIFAR 10	Fashion MNIST	MNIST	Census Income				Law School				Wine Quality white → white, red				Heart Disease US → US, UK, CH, HU				Census Income Male → Male, Female				Law School White → White, Black			
Black-box (BBox) classification model	ResNet50	CNN	CNN	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM
LR-Confidence	-	-	-	0.59	-	-	-	0.39	-	-	-	0.23	-	-	-	0.41	-	-	-	0.56	-	-	-	0.41	-	-	-
SVM-Platt [130]	-	-	-	-	-	-	0.66	-	-	-	0.51	-	-	-	0.44	-	-	-	0.52	-	-	-	0.69	-	-	-	0.50
DNN-MCP [74]	0.57	0.80	**0.96**	-	**0.60**	-	-	-	0.52	-	-	-	0.14	-	-	-	0.36	-	-	-	0.56	-	-	-	0.53	-	-
RF-uncertainty [143]	-	-	-	-	-	0.51	-	-	-	0.41	-	-	-	0.30	-	-	-	**0.50**	-	-	-	0.38	-	-	-	0.41	-
Trust score [84]	0.29	0.77	0.91	0.31	0.30	0.41	0.42	0.38	0.38	0.67	0.63	**0.34**	0.41	0.38	0.46	0.39	0.32	0.30	0.38	0.24	0.25	0.34	0.35	0.41	0.36	0.66	0.63
Riskscore (Proposed)	**0.59**	**0.83**	**0.96**	**0.61**	**0.60**	**0.74**	**0.73**	**0.65**	**0.54**	**0.73**	**0.71**	0.32	**0.50**	**0.47**	**0.50**	**0.57**	**0.45**	0.44	**0.57**	**0.57**	**0.58**	**0.74**	**0.74**	**0.66**	**0.53**	**0.73**	**0.70**

Table 6.4: Detecting out-of-distribution (OOD) test examples: Values in the table are AUROC for OOD detection. Higher values are better.

Dataset	Wine Quality white → white, red				Heart Disease US → US, UK, CH, HU				Census Income Male → Male, Female				Law School White → White, Black			
BBox classification model	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM	LR	MLP	RF	SVM
LR-Confidence	0.33	-	-	-	0.66	-	-	-	0.45	-	-	-	0.68	-	-	-
SVM-Platt [130]	-	-	-	0.81	-	-	-	0.67	-	-	-	0.42	-	-	-	0.66
MCP [74]	-	0.25	-	-	-	0.63	-	-	-	0.42	-	-	-	0.61	-	-
RF-epistemic uncertainty [143]	-	-	**0.84**	-	-	-	0.55	-	-	-	**0.64**	-	-	-	0.62	-
Trust score [84]	0.61	0.65	0.62	0.64	0.66	**0.65**	0.62	0.64	0.42	0.42	0.42	0.42	0.66	0.68	0.67	0.62
Epistemic uncertainty	**0.81**	**0.87**	0.82	**0.91**	**0.67**	0.54	**0.74**	**0.72**	**0.51**	**0.57**	0.54	**0.48**	**0.72**	**0.70**	**0.72**	**0.68**

6.5.4 Application: Detecting Out-of-distribution Test Points

In this experiment, we evaluate how well the *Risk Advisor* can successfully detect the underlying sources of uncertainty. However, for real-world datasets and complex black-box models it is difficult to collect ground truth (for evaluation) on which errors are due to inherent data complexity or model limitations. Hence, in this section we only focus on detecting errors due to *lack of knowledge* (e.g., due to data shifts between training and deployment distributions). To this end, we narrow our focus on the four datasets on out-of-distribution (OOD) test points for which we have ground truth labels shown in Table 6.4. Our goal is to evaluate how well the *Risk Advisor's* estimated *epistemic uncertainty* can be used to detect test points coming from a different distribution than the one which the model was trained on.

Setup and Metric: Given a combined test dataset consisting of both in-distribution and out-of-distribution test points, the question at hand is to what extent the *Risk Advisor*'s estimated *epistemic uncertainty* can effectively separate in-distribution and out-of-distribution test points. As we have ground truth for out-of-distribution test points and we have ensured that there are equal numbers of in/out distribution test points, we can use the area-under-the-ROC-curve metric (AUROC) for evaluation. Intuitively, AUROC measures the degree to which each of the confidence scores ranks a randomly chosen OOD data point higher than a randomly chosen non-OOD point.

Results: Table 6.4 shows a comparison between the black-box model's own *confidence* score, *trust score*, and the *Risk Advisor*'s estimated *epistemic uncertainty*. Unlike baseline methods for DNN, SVM, and LR, the baseline for computing uncertainty of RF by [Shaker and Hüllermeier 2020] can decompose the overall uncertainty into aleatoric and epistemic components. Thus, for RF, we rely on the *epistemic* uncertainty estimates. We make the following observations.

First, observe that *epistemic uncertainty* consistently outperforms both the black-box model's own *confidence scores* and *trust scores* across all datasets and classification methods, with a significant margin. Further *Risk Advisor's* epistemic uncertainty is competitive with RF-epistemic uncertainty, which is model-specific. This supports our argument that a post-hoc *meta-learner* trained to compute uncertainties, is a viable alternative to replacing the underlying black-box ML classifier, which may not be feasible in production practice. Second, observe that for the Wine and Census Income datasets, the DNN-MCP[Hendrycks and Gimpel 2017] and LR *confidence score* has AUROC <0.5, i.e., a performance worse than the *random baseline*, implying that black-box model *incorrectly* assigns higher confidence scores for OOD points than for in-distribution points. A similar trend can be observed for *trust scores* for the Census Income dataset, thus indicating that *confidence* scores and *trust scores* are not that reliable under distribution shifts. In contrast, the the AUROC values for the *epistemic uncertainty* are always > 0.5, implying that the *Risk Advisor* always assigns higher *epistemic* uncertainty for OOD test points than for in-distribution test points. This is an important property, as it indicates that a ranked ordering of test points by *epistemic uncertainty* can be used in a deployed application to detect out-of-distribution test points (given an application-specific threshold). For these critical data points, the system could

resort to a human expert (or other fall-back option), and thus enhance trustworthiness of the ML system.

6.5.5 Application: Risk Mitigation by Sampling & Retraining

Being able to identify black-box classifier's epistemic uncertainty enables another type of mitigation action: to mitigate risks due to evolving data by judiciously collecting more training examples and re-training the ML system.

We acknowledge the large body of literature on active sampling and domain adaptation in this context. In our experiment the goal is not to compare with these existing techniques, but rather to demonstrate an application of the *Risk Advisor*'s epistemic uncertainty, which can be achieved without making any changes to the underlying black-box classification system.

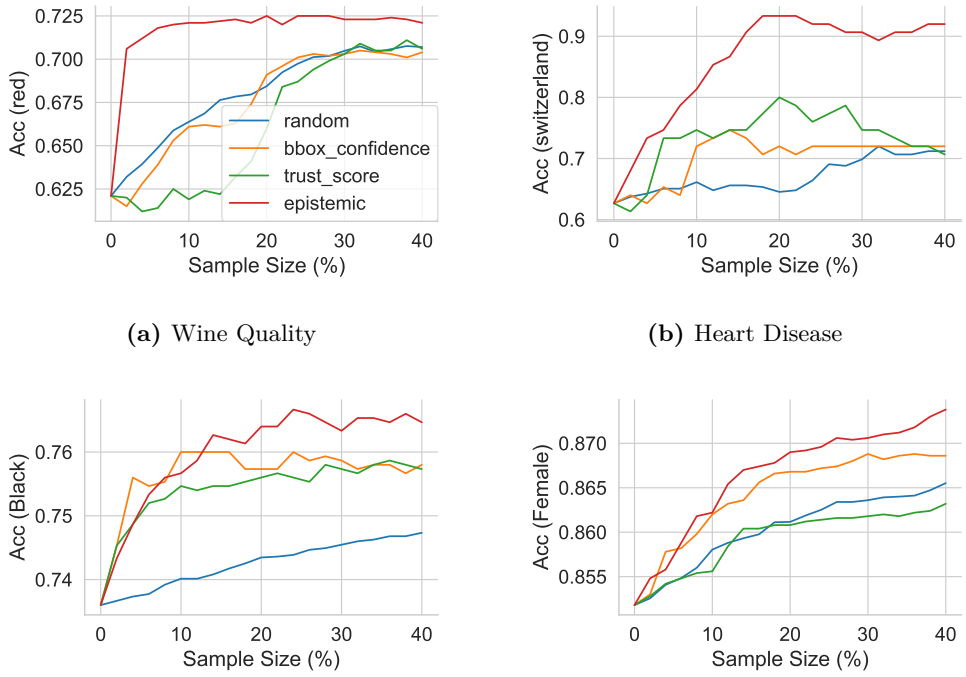
Setup and Metric: In this experiment, we fix the black-box classifier to logistic regression, and we assume that we have access to an untouched held-out set of labeled samples (different from training and test set). Our goal is to evaluate if the performance of the underlying black-box classifier can be improved for out-of-distribution test points by additional sampling and re-training the ML system on (a subset of) these held-out points. To evaluate the performance, we use the black-box classifier's improvement in accuracy for out-of-distribution test points.

We compare different sampling strategies by selecting data points from the with-held set in different orders based on three criteria: the *LR-Confidence*, *Trust score*, and the *Risk Advisor*'s *epistemic uncertainty*. For each approach, we first compute point-wise scores for all the points in the held-out set (different from training and test set, kept aside for sampling experiment). We then order the points in the held-out set according to these scores, i.e., *LR-confidence* (ascending order), *Trust score* (ascending order), and *Risk Advisor*'s *epistemic uncertainty* (descending order), respectively. Next, at each round of an iterative sampling, we select k% points from the held-out set (with replacement), and re-train the ML system.

Results: Fig. 6.8 shows results averaged over 5 independent runs. The x-axis shows the percentage of additional points sampled from the held-out set for re-training, and the y-axis shows the corresponding improved accuracy for the OOD group (e.g., accuracy on red wine for the Wine dataset). Ideally, we would expect the accuracy to rise higher with as few additional training points as possible. We make the following observations.

First, as we sample and retrain on additional points from the held-out data, the accuracy for OOD test-points increases for all the approaches on all datasets. However, the percentage of additional samples required to achieve similar performance differs across approaches. Not surprisingly, *random* sampling is the slowest improving approach for 3 out of 4 datasets, followed by *trust scores* and *confidence scores*. The Risk Advisor's sampling by *epistemic* uncertainty consistently outperforms on all datasets, by a large margin. For instance, on the Heart Disease dataset *epistemic* uncertainty achieves 30 percentage points (pp) improvement in accuracy (from 0.6 to 0.9) for an additional 20% samples from the held-out set. In contrast, all the other approaches stagnate around 0.7 even for an additional 40% samples. Similarly, on the Wine Quality dataset we see an improvement of 10 pp for an additional 10% samples, while other approaches do not reach this improvement even for additional 40% of samples.

We observe similar trends across approaches for Law School and Census Income datasets, albeit with smaller gains.



(a) Wine Quality

(b) Heart Disease

(c) Law School (White → Black)

(d) Census Income (Male → Female)

Figure 6.8: Addressing distribution shift: Comparison of various sampling strategies to selectively sample data points and retrain the black-box classification model. Curves that grow higher and faster from left to right are better.

6.6 Conclusion

This chapter presented the *Risk Advisor* model for detecting and analyzing sources of uncertainty and failure risks when a trained classifier is deployed for production usage. The Risk Advisor treats the base classifier as a black-box model, and this model-agnostic approach makes it a highly versatile and easy-to-deploy tool. In contrast to the prior state-of-the-art (including the main baseline Trust Scores [Jiang et al. 2018]), the Risk Advisor goes beyond providing a single measure of uncertainty, by computing refined scores that indicate failure risks due to data variability and noise, data shifts between training and deployment, and model limitations. Extensive experiments on various families of black-box classifiers and on real-world datasets covering common ML failure scenarios show that the Risk Advisor reliably predicts deployment-time failure risks in all the scenarios, and outperforms strong baselines. Thereby, we believe the Risk advisor, with its ability to audit and identify potential regions of failure risks would be a useful asset for the trustworthy machine learning toolbox.

Chapter 7

Conclusions and Outlook

7.1 Contributions

In this thesis we developed methods for fair and responsible machine learning. The methods and models we have proposed overcome common assumptions made in fair machine learning and address challenges faced by ML practioners in operationalizing fairness in practice, thereby bringing ML fairness models closer to application.

In Chapter 3, we scrutinized one of the major assumptions in methods for mitigating unfairness in ML systems, namely that the protected demographic features are specified upfront, and the membership to the protected groups is known. However, this is rarely the case in practice. Improving model fairness without access to group memberships is a difficult and understudied challenge. To this end, we proposed ARL (adversarially reweighted learning) an adversarial optimization approach, which aims to minimize the loss for the worst-off group by focuses the objective on *computationally-identifiable* regions of errors. Extensive empirical experiments show that ARL improves AUC for worst-case protected groups across multiple datasets, and over multiple types of training data biases. Our key insight in this work is that when protected groups are unknown, it is valuable to guard against worst-case of forming a group. As a result we believe this insight and the ARL method provide a foundation for how to pursue fairness without access to protected group memberships.

In Chapter 4, we took a critical look at the prevalently pursued paradigm of *group fairness*, highlighted its limitations, and advanced the alternate paradigm of *individual fairness*. We proposed iFair, a generic and versatile, unsupervised framework to perform a probabilistic transformation of data into individually fair representations. Our approach accommodates two important criteria. *iFair* views fairness from an application-agnostic view, which allowed us to incorporate it in a wide variety of tasks, including general classifiers and regression for learning-to-rank. Second, *iFair* treats individual fairness as a property of the dataset (in some sense, like privacy), which can be achieved by pre-processing the data into a transformed individually fair representation. We demonstrated the versatility of our method by applying it to classification and learning-to-rank tasks on a variety of real-world and synthetic datasets. Our experiments showed substantial improvements over the best prior work for this setting, thereby making first steps towards operationalizing individual fairness in practice.

In Chapter 5, we further advanced the work on *individual fairness* by proposing methods to model and incorporate human intuitions on individual fairness. To this end, we devised a new method called PFR for operationalizing individual fairness by eliciting and leveraging

side-information on pairs of individuals who are equally deserving and thus should be treated similarly for a given task. We developed a representation learning model to learn Pairwise Fair Representations (PFR), as a fairness-enhanced input to downstream machine learning tasks. Comprehensive experiments with synthetic and real-life datasets indicate that the pairwise judgments are beneficial for members of the protected group, resulting in high individual fairness and high group fairness with reasonably low loss in utility. Thereby we believe our contributions are an important step towards mitigating unfairness at an individual level, and pave the path towards eliciting and modeling expert human side-information on (un)fairness.

Methods for fair model learning commonly assume that the training data is representative of the target population after model deployment. In practice, however this is rarely the case. Models are frequently deployed on new target populations (e.g., geographic locations, demographic groups, etc) on which the model was not trained. Further, models are often deployed in ways that are not inline with the purpose for which they were trained. To address this, we need fairness approaches that look beyond fair model-learning and consider methods for responsible model deployment. In Chapter 6, we contributed to responsible model deployment by proposing the *Risk Advisor*, a novel post-hoc meta-learner for predicting potential failure risks of deployed ML systems. In addition to flagging failure risks, *Risk Advisor* is constructive by providing actionable insight into the sources of uncertainties inducing the risk. Extensive experiments on various families of black-box classification models and on real-world and synthetic datasets covering common ML failure scenarios show that the *Risk Advisor* reliably predicts deployment-time failure risks in all the scenarios, and outperforms strong baselines. Thereby, we believe the *Risk Advisor*, with its ability to proactively audit and identify potential regions of failure risks, would be a useful addition to the toolkit for responsible deployment of machine learning systems in practice.

7.2 Outlook

Privacy and Fairness. Ironically, developing fair machine learning algorithms often require access to sensitive demographic data such as knowledge of membership to protected groups, placing fairness and privacy in tension. Therefore, while we think it is imperative that we develop methods for mitigating unfairness, there remains further challenges in developing complementary approaches for mitigating unfairness that do not harm privacy.

Our proposed ARL method in Chapter 3 is a first step towards overcoming this crucial challenge. Looking forward, we believe further research is needed to tease-out the relationship between privacy and fairness, and develop methods for measuring and mitigating unfairness that are privacy-friendly.

Modeling Expert Knowledge on (Un)fairness. Predominantly, work on mitigating unfairness in ML has taken a statistical approach, which often reduces the broader *fairness* goals to a narrow optimization problem that aims to equalize some evaluation metric of interest (e.g., error rate) across demographic groups. While this approach makes realization of ML fairness a tangible goal, in the process we lose out on the valuable human aspect.

Going forward, human-centered approaches for fairness are needed that can involve key

stakeholders such as policy makers and advisors in the loop through the machine learning cycle. Our proposed notion of *fairness graphs* introduced in Chapter 5 presents a first approach for eliciting and modeling expert knowledge on (un)fairness into fair machine learning methods.

Beyond Fair Model Learning A typical ML pipeline involves a series of choices from problem formulation, data collection, model learning, to evaluation, model deployment, and monitoring. Much of the existing ML fairness work focuses only on the model learning step, and a selected few works look into fair data collection and pre-processing space. However, sources of unfairness can arise at every step of the ML pipeline. For instance, models are often deployed on population demographics, geographical regions, or even for tasks for which they were not trained in the first place. In such scenarios, it is crucial to step back and ask important questions such as "Is it safe to deploy the model on this population? What kind of risk mitigation actions can we take to ensure that the model works for everyone?" Our proposed *Risk Advisor* model for responsible model deployment proposed in Chapter 6 takes a step in this direction. Going forward, comprehensive approaches are needed that address unfairness arising at all steps of the ML pipeline.

Bibliography

Ehsan Amid and Antti Ukkonen. Multiview triplet embedding: Learning attributes in multiple maps. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *Journal of Machine Learning Research, Workshop and Conference Proceedings*, pages 1472–1480, 2015.

Ed Anderson, Zhaojun Bai, Jack J. Dongarra, Anne Greenbaum, A. McKenney, Jeremy Du Croz, Sven Hammarling, James Demmel, Christian H. Bischof, and Danny C. Sorensen. LAPACK: a portable linear algebra library for high-performance computers. In Joanne L. Martin, Daniel V. Pryor, and Gary Montry, editors, *Proceedings Supercomputing '90, New York, NY, USA, November 12-16, 1990*, pages 2–11. IEEE Computer Society, 1990.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals and it's biased against blacks. *ProPublica*, 2016. `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing` (accessed on 14-10-2021).

K. Asif, W. Xing, S. Behpour, and B. D. Ziebart. Adversarial cost-sensitive classification. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, UAI 2015, July 12-16, 2015, Amsterdam, The Netherlands*, pages 92–101, 2015.

Abolfazl Asudeh, H. V. Jagadish, Julia Stoyanovich, and Gautam Das. Designing fair ranking schemes. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, pages 1259–1276, 2019.

Pranjal Awasthi, Matthäus Kleindessner, and Jamie Morgenstern. Equalized odds post-processing under imperfect group information. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 1770–1780, 2020.

Ananth Balashankar, Alyssa Lees, Chris Welty, and Lakshminarayanan Subramanian. Pareto-efficient fairness for skewed subgroup data. In *International Conference on Machine Learning AI for Social Good Workshop. Long Beach, United States*, volume 8, 2019.

David Barber and Christopher M Bishop. Ensemble learning in bayesian neural networks. *Nato ASI Series F Computer and Systems Sciences*, 168:215–238, 1998.

Solon Barocas and Andrew D Selbst. Big data's disparate impact. *California Law Review*, 104(3):671, 2016.

Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. 2018. `http://www.fairmlbook.org` (accessed on 14-10-2021).

Peter L. Bartlett and Marten H. Wegkamp. Classification with a reject option using a hinge loss. *Journal of Machine Learning Research*, 9:1823–1840, 2008.

Yahav Bechavod, Christopher Jung, and Steven Z. Wu. Metric-free individual fairness in online learning. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Irad Ben-Gal. Outlier detection. In Oded Maimon and Lior Rokach, editors, *Data Mining and Knowledge Discovery Handbook, 2nd ed*, pages 117–130. Springer, 2010.

A. Beutel, J. Chen, T. Doshi, H. Qian, A. Woodruff, C. Luu, P. Kreitmann, J. Bischof, and E. H. Chi. Putting fairness principles into practice: Challenges, metrics, and improvements. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*, pages 453–459, 2019.

Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H. Chi. Data decisions and theoretical implications when adversarially learning fair representations. *CoRR*, abs/1707.00075, 2017.

Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. Equity of attention: Amortizing individual fairness in rankings. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 405–414, 2018.

Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In Sorelle A. Friedler and Christo Wilson, editors, *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, volume 81 of *Proceedings of Machine Learning Research*, pages 149–159. PMLR, 2018.

Reuben Binns. On the apparent conflict between individual and group fairness. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 514–524, 2020.

Arpita Biswas and Suvam Mukherjee. Ensuring fairness under prior probability shifts. In *AIES '21: AAAI/ACM Conference on AI, Ethics, and Society, Virtual Event, USA, May 19-21, 2021*, pages 414–424, 2021.

A. Blum and K. Stangl. Recovering from biased data: Can fairness constraints improve accuracy? In *1st Symposium on Foundations of Responsible Computing, FORC 2020, June 1-3, 2020, Harvard University, Cambridge, MA, USA (virtual conference)*, volume 156 of *LIPIcs*, pages 3:1–3:20, 2020.

Francesco Bonchi, Sara Hajian, Bud Mishra, and Daniele Ramazzotti. Exposing the probabilistic causal structure of discrimination. *International Journal of Data Science and Analytics*, 3(1):1–21, 2017.

Tim Brennan, William Dieterich, and Beate Ehret. Evaluating the predictive validity of the compas risk and needs assessment system. *Criminal Justice and Behavior*, 36(1):21–40, 2009.

Roy L Brooks. *Rethinking the American race problem*. Univ of California Press, 1992.

J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency, FAT 2018, 23-24 February 2018, New York, NY, USA*, pages 77–91, 2018.

Toon Calders, Faisal Kamiran, and Mykola Pechenizkiy. Building classifiers with independency constraints. In *ICDM Workshops 2009, IEEE International Conference on Data Mining Workshops, Miami, Florida, USA, 6 December 2009*.

Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ArXiv preprint*, abs/2010.04053, 2020.

Jiahao Chen, Nathan Kallus, Xiaojie Mao, Geoffry Svacha, and Madeleine Udell. Fairness under unawareness: Assessing disparity when protected class is unobserved. In *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*, pages 339–348, 2019.

Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5029–5037, 2017.

Alexandra Chouldechova. Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big data*, 5(2):153–163, 2017.

Alexandra Chouldechova and Aaron Roth. A snapshot of the frontiers of fairness in machine learning. *Communications of the ACM*, 63(5):82–89, 2020.

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*, pages 797–806, 2017.

Amanda Coston, Karthikeyan Natesan Ramamurthy, Dennis Wei, Kush R Varshney, Skyler Speakman, Zairah Mustahsan, and Supriyo Chakraborty. Fair transfer learning with missing protected attributes. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*, pages 91–98, 2019.

Amanda Coston, Alan Mishler, Edward H. Kennedy, and Alexandra Chouldechova. Counterfactual risk assessments, evaluation, and fairness. In Mireille Hildebrandt, Carlos Castillo, L. Elisa Celis, Salvatore Ruggieri, Linnet Taylor, and Gabriela Zanfir-Fortuna, editors, *FAT* '20: Conference on Fairness, Accountability, and Transparency, Barcelona, Spain, January 27-30, 2020*, pages 582–593, 2020.

Thomas M Cover. *Elements of information theory*. 1999.

Kate Crawford and Ryan Calo. There is a blind spot in AI research. *Nature*, 538, 2016.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR, 2021.

Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D Sculley, and Yoni Halpern. Fairness is not static: deeper understanding of long term fairness via simulation studies. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 525–534, 2020.

J. S Denker and Y. LeCun. Transforming neural-net output levels to probability distributions. In *Advances in Neural Information Processing Systems 3, [NIPS Conference, Denver, Colorado, USA, November 26-29, 1990]*, pages 853–859, 1990.

Stefan Depeweg, José Miguel Hernández-Lobato, Finale Doshi-Velez, and Steffen Udluft. Decomposition of uncertainty in bayesian deep learning for efficient and risk-sensitive learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1192–1201, 2018.

A. Der Kiureghian and O. Ditlevsen. Aleatory or epistemic? does it matter? *Structural safety*, 31(2), 2009.

Dua Dheeru and Efi Karra Taniskidou. UCI machine learning repository, 2017.

Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, and Aaron Roth. Minimax group fairness: Algorithms and experiments. In *AIES '21: AAAI/ACM Conference on AI, Ethics, and Society, Virtual Event, USA, May 19-21, 2021*, pages 66–76, 2021.

D. Dua and C. Graff. UCI machine learning repository, 2017.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 214–226, 2012a.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 214–226, 2012b.

L. Eckhouse. Big data may be reinforcing racial bias in the criminal justice system. *The Washington Post*, 2017. `https://www.washingtonpost.com/opinions/big-data-may-be-reinforcing-racial-bias-in-the-criminal-justice-system/2017/02/10/d63de518-ee3a-11e6-9973-c5efb7ccfb0d_story.html` (accessed on 14-10-2021).

Harrison Edwards and Amos J. Storkey. Censoring representations with an adversary. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

R. El-Yaniv et al. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11(5), 2010.

Shady Elbassuoni, Sihem Amer-Yahia, Christine El Atie, Ahmad Ghizzawi, and Bilel Oualha. Exploring fairness of ranking in online job marketplaces. In *Advances in Database Technology - 22nd International Conference on Extending Database Technology, EDBT 2019, Lisbon, Portugal, March 26-29, 2019*, pages 646–649, 2019.

Danielle Ensign, Sorelle A. Friedler, Scott Neville, Carlos Eduardo Scheidegger, and Suresh Venkatasubramanian. Decision making with limited feedback. In Firdaus Janoos, Mehryar Mohri, and Karthik Sridharan, editors, *Algorithmic Learning Theory, ALT 2018, 7-9 April 2018, Lanzarote, Canary Islands, Spain*, volume 83 of *Proceedings of Machine Learning Research*, pages 359–367. PMLR, 2018.

R. M Fano. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11), 1961.

Michael Feldman, Sorelle A. Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Sydney, NSW, Australia, August 10-13, 2015*, pages 259–268, 2015.

Andres Ferraro, Xavier Serra, and Christine Bauer. Break the loop: Gender imbalance in music recommenders. In *Proceedings of the 2021 Conference on Human Information Interaction and Retrieval*, pages 249–254, 2021.

Benjamin Fish, Jeremy Kun, and Ádám Dániel Lelkes. A confidence-based approach for balancing fairness and accuracy. In *Proceedings of the 2016 SIAM International Conference on Data Mining, Miami, Florida, USA, May 5-7, 2016*, pages 144–152, 2016.

Sorelle A. Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. On the (im)possibility of fairness. *CoRR*, abs/1609.07236, 2016.

Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. pages 329–338, 2019.

J. H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38 (4), 2002.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *Journal of Machine Learning Research, Workshop and Conference Proceedings*, pages 1050–1059, 2016.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna M. Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. *ArXiv preprint*, abs/1803.09010, 2018.

Stephen Gillen, Christopher Jung, Michael J. Kearns, and Aaron Roth. Online learning with an unknown fairness metric. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2605–2614, 2018.

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Ben Green and Lily Hu. The myth in the methodology: Towards a recontextualization of fairness in machine learning. In *Proceedings of the machine learning: the debates workshop*, 2018.

Nina Grgic-Hlaca, Elissa M. Redmiles, Krishna P. Gummadi, and Adrian Weller. Human perceptions of fairness in algorithmic decision making: A case study of criminal risk prediction. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 903–912, 2018a.

Nina Grgic-Hlaca, Muhammad Bilal Zafar, Krishna P. Gummadi, and Adrian Weller. Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 51–60, 2018b.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330, 2017.

Sara Hajian and Josep Domingo-Ferrer. A methodology for direct and indirect discrimination prevention in data mining. *IEEE Transactions on Knowledge and Data Engineering*, 25 (7):1445–1459, 2013.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2), 2011.

William L. Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *IEEE Data Engineering Bulletin*, 2017.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural*

Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain, pages 3315–3323, 2016.

Tatsunori B. Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness without demographics in repeated loss minimization. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 1934–1943, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778, 2016.

U. Hébert-Johnson, M. P. Kim, O. Reingold, and G. N. Rothblum. Calibration for the (computationally-identifiable) masses. *arXiv preprint arXiv:1711.08513*, 2017.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé III, Miroslav Dudík, and Hanna M. Wallach. Improving fairness in machine learning systems: What do industry practitioners need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems, CHI 2019, Glasgow, Scotland, UK, May 04-09, 2019*, page 600, 2019.

A. Honkela and H. Valpola. Variational learning and bits-back coding: an information-theoretic view to bayesian learning. *IEEE transactions on Neural Networks*, 15(4), 2004.

Stephen C Hora. Aleatory and epistemic uncertainty in probability elicitation with an example from hazardous waste management. *Reliability Engineering & System Safety*, 54 (2-3), 1996.

Hui Hu, Yijun Liu, Zhen Wang, and Chao Lan. A distributed fair machine learning framework with private demographic data protection. In *2019 IEEE International Conference on Data Mining, ICDM 2019, Beijing, China, November 8-11, 2019*, pages 1102–1107, 2019.

E. Hüllermeier and W. Waegeman. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. *Machine Learning*, 110(3), 2021.

M. Höfler, H. Pfister, R. Lieb, and H. Wittchen. The use of weights to account for non-response and drop-out. *Social psychiatry and psychiatric epidemiology*, 40, 2005.

Christina Ilvento. Metric learning for individual fairness. In Aaron Roth, editor, *1st Symposium on Foundations of Responsible Computing, FORC 2020, June 1-3, 2020, Harvard University, Cambridge, MA, USA (virtual conference)*, volume 156, pages 2:1–2:11, 2020.

Maliha Tashfia Islam, Anna Fariha, and Alexandra Meliou. Through the data management lens: Experimental analysis and evaluation of fair classification. *CoRR*, abs/2101.07361, 2021.

Matthew Jagielski, Michael J. Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan R. Ullman. Differentially private fair learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 3000–3008, 2019.

Heinrich Jiang, Been Kim, Melody Y. Guan, and Maya R. Gupta. To trust or not to trust A classifier. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5546–5557, 2018.

Thorsten Joachims and Filip Radlinski. Search engines that learn from implicit feedback. *IEEE Computer*, 40(8), 2007.

Christopher Jung, Michael Kearns, and Seth an Neel. Eliciting and enforcing subjective individua. *ArXiv preprint*, abs/1905.10660, 2019.

Christopher Jung, Michael Kearns, Seth Neel, Aaron Roth, Logan Stapleton, and Zhiwei Steven Wu. An algorithmic framework for fairness elicitation. In *2nd Symposium on Foundations of Responsible Computing*, 2021.

H. Kahn and A. W. Marshall. Methods of reducing sample size in monte carlo computations. *Journal of the Operations Research Society of America*, 1, 1953.

N. Kallus, X. Mao, and A. Zhou. Assessing algorithmic fairness with unobserve. *ArXiv preprint*, abs/1906.00285, 2019.

Faisal Kamiran, Toon Calders, and Mykola Pechenizkiy. Discrimination aware decision tree learning. In *ICDM 2010, The 10th IEEE International Conference on Data Mining, Sydney, Australia, 14-17 December 2010*, pages 869–874, 2010.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Considerations on fairness-aware data mining. In *12th IEEE International Conference on Data Mining Workshops, ICDM Workshops, Brussels, Belgium, December 10, 2012*.

Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part II*, pages 35–50, 2012.

Michael Kearns and Aaron Roth. *The ethical algorithm: The science of socially aware algorithm design*. Oxford University Press, 2019.

Michael J. Kearns, Aaron Roth, and Zhiwei Steven Wu. Meritocratic fairness for cross-population selection. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1828–1836, 2017a.

Michael J. Kearns, Aaron Roth, and Zhiwei Steven Wu. Meritocratic fairness for cross-population selection. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1828–1836, 2017b.

Michael J. Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2569–2577, 2018.

Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5574–5584, 2017.

Niki Kilbertus, Mateo Rojas-Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 656–666, 2017.

Niki Kilbertus, Adrià Gascón, Matt J. Kusner, Michael Veale, Krishna P. Gummadi, and Adrian Weller. Blind justice: Fairness with encrypted sensitive attributes. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 2635–2644, 2018.

M. P. Kim, A. Ghorbani, and J. Zou. Multiaccuracy: Black-box post-processing for fairness in classification. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019, Honolulu, HI, USA, January 27-28, 2019*, pages 247–254, 2019.

Michael P. Kim, Omer Reingold, and Guy N. Rothblum. Fairness through computationally-bounded awareness. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 4847–4857, 2018.

Jon M. Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. In *8th Innovations in Theoretical Computer Science Conference, ITCS 2017,Berkeley, CA, USA, January 9-11, 2017*.

A Krizhevsky. Learning multiple layers of features from tiny images. *Master's thesis, University of Toronto*, 2009.

Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4066–4076, 2017.

Preethi Lahoti, Krishna Gummadi, and Gerhard Weikum. Operationalizing individual fairness with pairwise fair representations. *Proceedings of the VLDB Endowment (Proc. VLDB)*, 13(4):506–518, 2019a.

Preethi Lahoti, Krishna Gummadi, and Gerhard Weikum. iFair: Learning individually fair data representations for algorithmic decision making. In *ICDE 2019, 35th IEEE International Conference on Data Engineering*, pages 1334–1345, 2019b.

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: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Preethi Lahoti, Krishna P Gummadi, and Gerhard Weikum. Detecting and mitigating test-time failure risks via model-agnostic uncertainty learning. In *ICDM 2021, IEEE International Conference on Data Mining*, 2021.

Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6402–6413, 2017.

Yann LeCun, Corinna Cortes, and CJ Burges. Mnist handwritten digit database. *Available: http://yann.lecun.com/exdb/mnist*, 2, 2010.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

Y. Lin, T. Liu, and H. Chen. Semantic manifold learning for image retrieval. In *ACM Multimedia*, 2005.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *International conference on machine learning*, pages 3122–3130. PMLR, 2018.

Roderick J A Little and Donald B Rubin. *Statistical Analysis with Missing Data*. John Wiley & Sons, Inc., USA, 1986.

Dong C. Liu and Jorge Nocedal. On the limited memory BFGS method for large scale optimization. *Math. Program.*, 45(1-3), 1989.

Lydia T. Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 6196–6200, 2019.

Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. The variational fair autoencoder. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

Kristian Lum and William Isaac. To predict and serve? *Significance*, 13(5):14–19, 2016.

Redmond M. Communities and crime dataset, uci machine learning repository, 2009.

David Madras, Elliot Creager, Toniann Pitassi, and Richard S. Zemel. Learning adversarially fair and transferable representations. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3381–3390, 2018.

A. Malinin. Uncertainty estimation in deep learning with application to spoken language-assessment. *PhD thesis*, 2019.

Andrey Malinin, Liudmila Prokhorenkova, and Aleksei Ustimenko. Uncertainty in gradient boosting via ensembles. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021.

Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 4615–4625, 2019.

Debarghya Mukherjee, Mikhail Yurochkin, Moulinath Banerjee, and Yuekai Sun. Two simple ways to learn individual fairness metrics from data. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 7097–7107, 2020.

Anh Mai Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 427–436, 2015.

Alexandra Olteanu, Carlos Castillo, Fernando Diaz, and Emre Kiciman. Social data: Biases, methodological pitfalls, and ethical boundaries. *Frontiers Big Data*, 2:13, 2019.

Judea Pearl. *Causality: Models, Reasoning and Inference*. 2nd edition, 2009. ISBN 052189560X.

F. Pedregosa et al. Scikit-learn: Ml in Python. *Journal of Machine Learning Research*, 12, 2011.

Dino Pedreschi, Salvatore Ruggieri, and Franco Turini. Discrimination-aware data mining. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Las Vegas, Nevada, USA, August 24-27, 2008*, pages 560–568, 2008.

J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3), 1999.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon M. Kleinberg, and Kilian Q. Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5680–5689, 2017.

B. Becker R. Kohavi. UCI ml repository, 1996.

J. Rawls. *Justice as fairness: A restatement.* 2001.

Ashkan Rezaei, Rizal Fathony, Omid Memarrast, and Brian Ziebart. Fairness for robust log loss classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5511–5518, 2020.

Ashkan Rezaei, Anqi Liu, Omid Memarrast, and Brian D. Ziebart. Robust fairness under covariate shift. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 9419–9427, 2021.

Anian Ruoss, Mislav Balunovic, Marc Fischer, and Martin T. Vechev. Learning certified individually fair representations. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Babak Salimi, Luke Rodriguez, Bill Howe, and Dan Suciu. Interventional fairness: Causal database repair for algorithmic fairness. In *Proceedings of the 2019 International Conference on Management of Data, SIGMOD Conference 2019, Amsterdam, The Netherlands, June 30 - July 5, 2019*, pages 793–810, 2019.

S Saria and A. Subbaswamy. Tutorial: safe and reliable machine learning. *ArXiv preprint*, abs/1904.07204, 2019.

Sebastian Schelter, Tammo Rukat, and Felix Bießmann. Learning to validate the predictions of black box classifiers on unseen data. In *Proceedings of the 2020 International Conference on Management of Data, SIGMOD Conference 2020, online conference [Portland, OR, USA], June 14-19, 2020*, pages 1289–1299, 2020.

S. Schneider, E. Rusak, L. Eck, and O. et al. Bringmann. Improving robustness against common corruptions by covariate shift adaptation. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Peter Schulam and Suchi Saria. Can you trust this prediction? auditing pointwise reliability after learning. In *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 1022–1031, 2019.

R. Senge, S. Bösner, K. Dembczyński, J. Haasenritter, and O. et al. Hirsch. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Information Sciences*, 255, 2014.

M. H. Shaker and E. Hüllermeier. Aleatoric and epistemic uncertainty with random forests. In *Advances in Intelligent Data Analysis XVIII - 18th International Symposium on Intelligent Data Analysis, IDA 2020, Konstanz, Germany, April 27-29, 2020, Proceedings*, volume 12080, pages 444–456, 2020.

Hidetoshi Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of statistical planning and inference*, 90(2):227–244, 2000.

Ashudeep Singh and Thorsten Joachims. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 2219–2228, 2018.

Harvineet Singh, Rina Singh, Vishwali Mhasawade, and Rumi Chunara. Fairness violations and mitigation under covariate shift. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 3–13, 2021.

Jasper Snoek, Yaniv Ovadia, Emily Fertig, Balaji Lakshminarayanan, Sebastian Nowozin, D. Sculley, Joshua V. Dillon, Jie Ren, and Zachary Nado. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 13969–13980, 2019.

Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P. Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. A unified approach to quantifying algorithmic unfairness: Measuring individual &group unfairness via inequality indices. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018*, pages 2239–2248, 2018.

I. Steinwart, D. Hush, and C. Scovel. A classification framework for anomaly detection. *J. Mach. Learn. Res.*, 6(2):211–232, 2005.

Julia Stoyanovich, Ke Yang, and H. V. Jagadish. Online set selection with fairness and diversity constraints. In *Proceedings of the 21st International Conference on Extending Database Technology, EDBT 2018, Vienna, Austria, March 26-29, 2018*, pages 241–252, 2018.

M. Veale and R. Binns. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society*, 4(2):2053951717743530, 2017.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Bias preservation in machine learning: The legality of fairness metrics under eu non-discrimination law. *West Virginia Law Review, Forthcoming*, 2021.

Hanchen Wang, Nina Grgic-Hlaca, Preethi Lahoti, Krishna P. Gummadi, and Adrian Weller. An empirical study on learning fairness metrics for COMPAS data with human supervision. volume abs/1910.10255, 2019.

Serena Wang, Wenshuo Guo, Harikrishna Narasimhan, Andrew Cotter, Maya Gupta, and Michael I Jordan. Robust optimization for fairness with noisy protected groups. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

L. F Wightman. Lsac national longitudinal bar passage study. lsac research report series. 1998.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarkin. *ArXiv preprint*, abs/1708.07747, 2017.

Ke Yang and Julia Stoyanovich. Measuring fairness in ranked outputs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management, Chicago, IL, USA, June 27-29, 2017*, pages 22:1–22:6, 2017.

Mikhail Yurochkin, Amanda Bower, and Yuekai Sun. Training individually fair ML models with sensitive subspace robustness. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114, 2004.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017, 20-22 April 2017, Fort Lauderdale, FL, USA*, volume 54 of *Proceedings of Machine Learning Research*, pages 962–970, 2017a.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web, WWW 2017, Perth, Australia, April 3-7, 2017*, pages 1171–1180, 2017b.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, Krishna P. Gummadi, and Adrian Weller. From parity to preference-based notions of fairness in classification. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 229–239, 2017c.

John R Zech, Marcus A Badgeley, Manway Liu, Anthony B Costa, Joseph J Titano, and Eric Karl Oermann. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: a cross-sectional study. *PLoS medicine*, 15(11): e1002683, 2018.

Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. Fa*ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, pages 1569–1578, 2017.

Richard S. Zemel, Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *Journal of Machine Learning Research, Workshop and Conference Proceedings*, pages 325–333, 2013.

B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, AIES 2018, New Orleans, LA, USA, February 02-03, 2018*, pages 335–340, 2018.

Chongjie Zhang and Julie A. Shah. Fairness in multi-agent sequential decision-making. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2636–2644, 2014.

Junzhe Zhang and Elias Bareinboim. Fairness in decision-making - the causal explanation formula. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 2037–2045. AAAI Press, 2018.

Xueru Zhang, Ruibo Tu, Yang Liu, Mingyan Liu, Hedvig Kjellström, Kun Zhang, and Cheng Zhang. How do fair decisions fare in long-term qualification? In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.