Algorithmic Fairness in the Real World: Challenges and Considerations

A Dissertation Presented
by

Avijit Ghosh

to

Khoury College of Computer Sciences

in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in

Computer Science

Northeastern University
Boston, Massachusetts

June 2023

NORTHEASTERN UNIVERSITY

Khoury College of Computer Science

Dissertation Signature Page

Dissertation Title:	Algorithmic Fairness in the Real World: Challenges and Considerations		
Author:	Avijit Ghosh	**NUID:**	001402322
College:	Computer Sciences		

Approved for Dissertation Requirements of the Doctor of Philosophy Degree

Dissertation Advisor
Dr. Christo Wilson _____ _____
Signature Date

Dissertation Committee Member
Dr. Tina Eliassi-Rad _____ _____
Signature Date

Dissertation Committee Member
Dr. Alina Oprea _____ _____
Signature Date

Dissertation Committee Member
Dr. Kristian Lum _____ _____
Signature Date

Department Chair
Dr. Elizabeth Mynatt _____ _____
Signature Date

Dean of Graduate School:
Dr. Amal Ahmed _____ _____
Signature Date

To all those who I call my family.

Contents

Acknowledgments

Working towards my Ph.D. has been an incredible journey. There have been ups and downs. Happy moments and sad moments. Many, many cups of hot chocolate and late nights. The work I've done, the papers I've written, and the people and organizations I've collaborated with have all helped to get me through this process, and I am grateful for every experience that I have had on this journey.

Foremost among those to whom I owe my thanks is my advisor, Christo Wilson. His guidance, support, and mentorship have given me the tools necessary to achieve most of the things I had listed in my Ph.D. statement of purpose.

I've also been privileged enough to work with amazing individuals in the industry. Aalok, Mary, Lea, and Josh at Fiddler; Kristian, Tomo, and Rumman at Twitter; Your guidance, support and friendship have helped me jump start my career in the field, and I count myself incredibly lucky to have met and collaborated with you all.

These acknowledgments would be incomplete without mentioning my friends and lab-mates in academia, both within and outside Northeastern. In no particular order, Ritam, Rajarshi, Geno, Alan, Giri, Ali, Jeffrey, Shan, Paul, Matthew, Giorgio, Aaron, Levi, and Piotr, bouncing ideas off of you and/or collaborating with you all has helped me to grow immensely, and I hope that the connections I've formed with the faculty, staff and fellow students at my time here at Khoury College will last throughout my life.

To my wonderful partner Ian, thank you. I couldn't have asked for anyone better to be by my side. You may not fully understand what it is that I do, but your love and support mean the world to me.

Lastly, I'd like to thank my family. Ma, Baba, and Bhai, without you, I wouldn't be here. You raised me, guided me, and made me the man I am today. Your love and support mean more to me than words can describe. Thank you.

Abstract of the Dissertation

Algorithmic Fairness in the Real World: Challenges and Considerations

by

Avijit Ghosh

Doctor of Philosophy in Computer Science

Northeastern University, June 2023

Dr. Christo Wilson, Advisor

The existence of social bias in Machine Learning (ML) algorithms is a pervasive, well-known problem. Strides have been made in identifying, intervening in, and regulating algorithms like facial recognition and sensitive classification problems like credit lending and recidivism prediction in criminal justice. In the light of such discoveries, the research community has shown significant theoretical progress in proposing different metrics to measure bias, and a suite of different algorithmic interventions to mitigate such biases. However, there are under-explored challenges in translating such theoretical fairness work into real world, practical ML systems. Noisy demographic information, adversarial vulnerabilities, policy restrictions, and the complex interplay of human decision makers with fair algorithmic interventions all play a significant role in the real world outcomes of these systems. In my thesis, I attempt to outline these problems in fair ML systems, with the aim to gain a more complete understanding of the challenges involved and to be able to provide technical and policy recommendations to overcome their real world implementation challenges.

Chapter 1

Introduction

1.1 Background

Algorithmic decision making has permeated every aspect of modern life, including high-stakes decisions like credit lending, bail granting, hiring, etc. While these ML models are great at scaling up processes with human bottlenecks, they also have the unintended consequences of picking up historical social biases like racism, sexism, homophobia, ableism, ageism, or religious intolerance [15].

In response to this, there is a growing body of academic work on ways to detect algorithmic bias. Journalists and academics have conducted audit studies of existing systems like Propublica's audit of the COMPAS software [11], studies about biases in Google's and Facebook's recommendation algorithms *cite? – Christo*, and several studies of biases in two-sided marketplace systems (Amazon, Uber, Lyft, etc.) [53, 102]. Acknowledging the growing criticism of the adverse social impacts of opaque systems, companies are starting to agree to independent cooperative third party audits, notable instances of which include audits of hiring software like pymetrics [211] and HireVue[1]. Different frameworks have been suggested for the best ways to conduct such audits [165].

On the other side of the spectrum, there exist various mitigation measures for biases introduced by algorithms [140]. There is now extensive literature documenting techniques for training fair classification [83, 98, 107, 143] and ranking [44, 190, 216] models. Companies are adopting and deploying fair ML systems in many real-world contexts [6, 16, 211].

[1] `https://www.hirevue.com/press-release/independent-audit-affirms-the-scientific-foundation-of-hirevue-assessments`

As with most burgeoning disciplines, however, "fair ML" research has the problem of having a number of theories and tools that work *in situ* [14], but may fail during implementations *in vivo*. To name just a few: notions of fairness in the research community are western-centric [175], intersectionality is only beginning to be implemented in fairness metrics [79], practitioners may do the bare minimum necessary to avoid legal liability while not significantly mitigating issues of bias [177], and the complex interplay between biased human decision makers and machine learning models [197]. Addressing these challenges requires a concerted effort from researchers, practitioners, and policymakers, and more importantly, moving away from benchmark based theoretical fairness studies to real-world harm measurement and mitigation.

In my work, I focus on implementation time problems with fair ML—their behavior in the presence of none or partial sensitive attributes, fairness in the presence of adversarial actors, and model bias over time. My goal is to demonstrate the severity of these issues in practice, to heighten awareness among researchers and practitioners, and to present solutions to these real world implementation challenges.

1.2 Research Questions

While a lot of progress has been made in the fields of fairness, accountability, transparency, and ethics of ML algorithms, there is still a considerable amount of practical challenges involved before this work can translate into real world social improvements for historically underrepresented minorities. My thesis aims to identify and tackle some these challenges. These are the specific research questions I aim to answer:

- **RQ1:** *Does noise in demographic information as an input to a fair ML algorithm impact the intended fairness of the outcomes for different subgroups? If so, what groups are impacted and how?*
- **RQ2:** *Can fair ML models be attacked by adversarial actors to create even more unfairness? If so, what groups are impacted and how?*
- **RQ3:** *In fair ML techniques that deliberately do not use protected attributes, how do their theoretical guarantees hold up in real life when compared against actual ground truth?*

- **RQ4:** *Do fair ML models, once deployed in a production system, continue to remain fair in the face of changing data and feature-output relationships? If so, can such unfairness be measured and mitigated?*

I discuss more details about my RQs in 1.3, 1.4, 1.5, and 1.6, respectively.

1.3 When Fair Ranking Meets Uncertain Inference

In cases where people are the data subjects being input to classification or ranking algorithms, the vast majority of existing work assumes that ground-truth demographic information will be available to mitigate sexism, racism, ageism, and other social biases [60]. This demographic data is crucial as it is used to measure and control for unfair biases, thus enabling fair outcomes.

Unfortunately, this assumption about the availability of ground-truth demographic data is often violated in practice. For example, in real-world contexts like assessing job applicants or credit seekers, social and legal barriers may prevent algorithm operators from collecting peoples' demographic information [8, 31].

The unavailability of ground-truth demographic data has led some system developers to adopt an alternative approach: infer protected class information from data and then supply it to the fair algorithm as input. One example of this is the Bayesian Improved Surname Geocoding (BISG) inference algorithm that is used by lenders and health insurers in the U.S. to infer people's race and ethnicity [3, 37]. This demographic data is used to ensure that lenders are making race-neutral lending decisions and that health insurers are not discriminating based on race. Given the high-stakes of these use cases, it is clear that accurate demographic information is critical, lest unchecked discrimination lead to serious harms.

The use of inferred data raises the issue that errors in inference may subvert the fairness objectives that a fair algorithm is attempting to optimize for. Intuitively, a fair algorithm cannot be expected to control for social biases if those biases are not represented in data due to errors. To the best of my knowledge, this problem has not been explored systematically in the literature, despite the fact that consequential real-world systems like BISG have adopted the practice.

In Chapter 3, I investigate how uncertainty in demographic inference impact fairness guarantees in the context of ranking algorithms. I approach this question using two complementary techniques. *First*, I use simulations to explore the relationship between population demographics, fairness metrics, and inference errors under controlled conditions. *Second*, to address the issue of ecological validity, I examine three case studies based on real-world datasets. Each of these datasets includes ground-truth demographic data, which enables us to generate a baseline unfair ranking and an "optimal" fair ranking. I compare these lower and upper bounds against rankings generated by a fair ranking algorithm when using erroneous demographic inferences as input. I present results using demographic inference error rates drawn from different real-world algorithms.

These were my main findings:

- In the simulation study, the fairness metrics (both representation-based and exposure-based) of the final ranked list increased monotonically with the increase in the accuracy of the prediction of the protected attributes. This is not a surprising result—as noise in the protected demographic attributes increases, the effectiveness of the fairness intervention reduces. I also observed that despite the fairness metrics varying widely based on the accuracy of demographic label prediction, the relevance of the ranked list, as measured by NDCG, barely changed, signifying that it is possible to perform noticeable fairness interventions without noticeably affecting the quality of rankings.

- In the case studies, I observed that the different rate of mispredictions for different demographic groups led to not only less fair rankings than if there were no noisy labels, but for certain demographic subgroups, the "fair" rankings were actually even more unfair than if no fairness intervention was performed. This is an alarming finding, showing that if not operationalized correctly, a fair algorithm can selectively perpetuate unfairness.

The results suggest that developers should not use inferred demographic data as input to fair ranking algorithms, unless the inferences are extremely accurate.

1.4 Subverting Fair Image Search with Generative Adversarial Perturbations

In the previous chapter, I demonstrate how *unintentional errors* in demographic data can dramatically undermine the objectives of fair ranking algorithms [77].

Another serious concern in the ML community is model *robustness*, especially in the face of clever and dedicated adversaries. The field of adversarial ML has demonstrated that seemingly accurate models display surprising brittleness when presented with maliciously crafted inputs [40, 200], and that these attacks impact models across a wide-variety of contexts [17, 51, 92, 200]. The existence of adversarial ML challenges the use of models in real-world deployments, particularly deep learning models.

In Chapter 4, I explore the intersection of these two concerns—fairness and robustness—in the context of ranking: *when a ranking model has been carefully calibrated to achieve some definition of fairness, is it possible for an external adversary to make the ranking model behave unfairly without having access to the model or training data?* In other words, can attackers *intentionally* weaponize demographic markers in data to subvert fairness guarantees?

To investigate this question, I present a case study in which I develop and then attack a fairness-aware image search engine using images that have been maliciously modified with *adversarial perturbations*. I chose this case study because image retrieval based on text queries is a popular, real-world use case for neural models (e.g., Google Image Search, iStock, Getty Images, etc.), and because prior work has shown that these models can potentially be fooled using adversarial perturbations [221] (although not in the context of fairness). To strengthen my case study, I adopt a strict threat model under which the adversary cannot *poison* training data [99] for the ranking model, and has no knowledge of the ranking model or fairness algorithm used by the victim search engine. Instead, the adversary can only add images into the query database, *after* the image retrieval model is trained.

For my experiments, I develop an image search engine that uses a state-of-the-art Multi-Modal Transformer (MMT) [73] retrieval model and a fair re-ranking algorithm (FMMR [108]) that aims to achieve demographic group fairness on the ranked list of image query results without ever explicitly using demographic labels. Under normal circumstances, where the images are unperturbed, my search engine returns demographically balanced sets of images in response to free text queries. I then train a generative adversarial perturbation (GAP) model [163] that learns from pretrained demographic classifiers to strategically insert human-

imperceptible perturbations into images. These perturbations attempt to cause FMMR to unfairly boost the rank of images containing people from an adversary-selected subpopulation (e.g., light-skinned men).

I observe the following findings from extensive experiments:

- My attacks can successfully confer significant unfair advantage to people from the majority class (light-skinned men, in my case study)—in terms of their overall representation and position in search results—relative to fairly-ranked baseline search results.
- My attack is robust across a number of variables, including the length of search result lists, the fraction of images that the adversary is able to perturb, the fairness algorithm used by the search engine, the image embedding algorithm used by the search engine, the demographic inference algorithm used to train the GAP models, and the training objective of the GAP models.
- My attacks are *stealthy*, i.e., they have close to zero impact on the relevance of search results.

In summary, I show that GAPs can be used to subvert fairness guarantees in the context of fair image retrieval. Further, I demonstrate that my attack is successful under a highly restricted threat model, which suggests that more powerful adversaries will also be able to implement successful attacks. I hypothesize that similar attacks may be possible against other classes of ML-based systems that (1) rely on highly parameterized models and (2) make fairness decisions for inputs that are based on data controlled by adversaries.

1.5 When Fair Classification Meets Noisy Protected Attributes

Similar to challenges in implementing fair ranking and search as I discuss in the previous chapters, hurdles also remain to the adoption of fair classifiers in real world scenarios—chief among them being questions about demographic data itself. Many *classical fair classifiers* assume that protected attributes are available at training time and/or testing time [60] and that this data is accurate. But as I already discuss in the previous two chapters, demographic data may be noisy, for reasons such as reliance on imperfect demographic-inference algorithms to generate protected attributes (1.3), imprecision in human-generated labels [55], or the

presence of an adversary that is intentionally poisoning demographic data (1.4). To attempt to deal with these issues, researchers have proposed *noise-tolerant fair classifiers* that aim to achieve distributional fairness by incorporating the error rate of demographic attributes in the fair classifier optimization process itself [41, 149, 209].

In other instances, demographic data may not be available at all, which violates the assumptions of both classical and noise-tolerant fair classifiers. This may occur when demographic data is unobtainable (e.g., laws or social norms impede collection [9, 31]), prohibitively expensive to generate (e.g., when large datasets are scraped from the web [56, 110, 128]), or when laws disallow the use of protected attributes to train classifiers (e.g., direct discrimination [211]). For cases such as these, researchers have proposed *demographic-blind fair classifiers* that use the latent representations in the feature space of the training data to reduce gaps in classification errors between protected groups, either via assigning higher weights to groups of training examples that are misclassified [90], or by training an auxiliary adversarial model to computationally identify regions of misclassification [121].

Motivated by this explosion of fundamentally different fair classifiers, I present an empirical, head-to-head evaluation of the performance of 14 classifiers in Chapter 5, spread across four classes: two *unconstrained classifiers*, seven classical fair classifiers, three noise-tolerant fair classifiers, and two demographic-blind classifiers. Drawing on the methodological approach used by [67] in their comparative study of classical fair classifiers, I evaluate the accuracy, stability, and fairness guarantees (defined as the equal odds difference) of these 14 classifiers across four datasets as I vary noise in the protected attribute (sex). To help explain the performance differences that I observe, I calculate and compare the feature importance vectors for our various trained classifiers. This methodological approach enables me to compare the performance of these 14 algorithms under controlled, naturalistic circumstances in an apples-to-apples manner.

Based on my head-to-head evaluation, I make the following key observations:

- Two classical fair classifiers, one noise-tolerant fair classifier, and one demographic-blind fair classifier performed consistently well across all metrics on our experiments.
- The best classifier for each case study showed some variability, confirming that the choice of dataset is an important factor when selecting a model.
- One demographic-blind fair classifier was able to achieve equal odds for males and

females under a variety of ecological conditions, confirming that demographics are not always necessary at training or testing time to achieve fairness.

I argue that large-scale, head-to-head evaluations such as the one I conducted in this study are critical for researchers and ML practitioners. My results act as a checkpoint, informing the community about the relative performance characteristics of classifiers within and between classes. For researchers, this can highlight gaps where novel algorithms are still needed (e.g. noise-tolerant and demographic-blind classifiers, based on my findings) and provide a framework for rigorously evaluating them. For practitioners, my results highlight the importance of thoroughly evaluating many classifiers from many classes before adopting one in practice, and I provide a roadmap for choosing the best classifiers for a given real-world scenario, depending on the availability and quality of demographic data.

1.6 FairCanary: Rapid Continuous Explainable Fairness

ML models that are deployed into the field cannot guarantee consistent performance over time [179]. One of the reasons for this might be that the underlying data has changed stochastically. This phenomenon, called *drift*, has been well-studied in the literature, from sudden [158] to gradual drifts [194]. Drifts may also be caused by true shifts in the relationship between the underlying variables (e.g., due to changes in the population over time), sampling issues [174], or even bugs that impact downstream data collection.

In scenarios where a deployed ML model is making sensitive decisions, I argue that analyzing the impact of drift on the *fairness* of the model is equally, if not more, important than assessing the impact of drift on traditional performance metrics like accuracy and recall. Regulators are also concerned about this issue: for example, the European Commission's recently proposed Artificial Intelligence Act states *"[ML] providers should be able to process... special categories of personal data, as a matter of substantial public interest, in order to ensure the bias monitoring, detection and correction in relation to high-risk AI systems"* as part of a *"robust post-market monitoring system."* [47] Similar regulations have been proposed in New Zealand [114], Canada [159], the US [2], and the UK [64].

The recognition that drift can negatively impact model performance, coupled with looming regulations, has spurred the creation of many commercial systems that offer *continuous model monitoring* [50]. In general, these systems track live model predictions over time, alert

the operator if performance metrics change substantively, and compute feature importance (a.k.a. *explanations*) for each prediction using methods like LIME [169] or the Shapley Value [54,135,144]. Some of these monitoring systems incorporate fairness metrics in addition to traditional performance metrics [157].

In Chapter 6, I present FairCanary, a continuous model monitoring system that offers two significant capabilities versus state-of-the-art commercial systems that help ensure model fairness over time. *First*, FairCanary incorporates a novel model bias quantification metric called Quantile Demographic Disparity (QDD) that uses quantile binning to measure differences in the overall prediction distributions over subgroups. Because QDD is measured over continuous distributions it does not require developers to choose specific (and often ad hoc) thresholds for measuring fairness, unlike most conventional fairness metrics. Additionally, QDD does not require outcome labels, which may not be available at runtime. *Second*, FairCanary reuses explanations computed for each individual prediction to quickly compute explanations for its bias metrics. This optimization makes FairCanary an order of magnitude faster than previous work that has tried to generate feature-level bias explanations [145]. To illustrate the effectiveness of FairCanary, I present a synthetic case study in Chapter 6 in which a data integrity error is artificially introduced for women on a particular day, showing how FairCanary is able to successfully detect and explain this fairness violation.

Regardless of whether ML models are regulated to mandate audits and continuous monitoring, I argue that ML practitioners have a professional and moral obligation to ensure that the systems they deploy do not misbehave. Given that issues like drift are known to occur, and that these issues may cause unfairness and bias, I argue that monitoring systems should become a standard component of most, if not all, deployed ML-based systems.

I hope that FairCanary (or other monitoring systems that incorporate its capabilities) will equip companies and institutions with improved tools to monitor, understand, and mitigate problems in their deployed ML systems, in real time. In turn, I hope that these capabilities will bring more equity and justice to the individual stakeholders impacted by deployed models.

1.7 Conclusion

In conclusion, my thesis highlights the need for a holistic approach to implementing fair machine learning algorithms that takes into account the unique characteristics of the

real-world context in which they will be deployed. My findings have practical implications for practitioners who wish to adopt fair machine learning algorithms. I emphasize the need for practitioners to consider the quality and representativeness of the training data, use appropriate fairness metrics and unbiased methods to evaluate their performance, be aware of adversarial actors in their deployment, and continuously monitor and mitigate potential issues that may arise over time. By adopting a more practice-oriented approach to implementing fair machine learning, practitioners can help ensure that these algorithms achieve their intended goals of promoting fairness and equity for all individuals.

Furthermore, in the conclusion of the thesis (Chapter 7), I discuss potential avenues for future research, namely machine unlearning to remove problematic training data – especially in the area of copyright violating generative models, investigation of the impact of human decision makers on ML bias and fairness, and developing actionable policy to ensure fair and ethical implementation of ML.

Chapter 2

Background

2.1 Algorithmic Fairness

The use of machine learning algorithms is ubiquitous in the developed world. It has become an integral part of society, affecting the lives of millions of people. Algorithmic decisions vary from low-stakes determinations, like product or film recommendations, to high-impact like loan or credit approval [150], hiring recommendations [30], facial recognition [205] and prison recidivism [48]. With this direct impact on people's lives, the need for fair and unbiased algorithms is paramount. It is critical that algorithms do not replicate and enhance existing societal biases, including those rooted in differences of race, gender, or sexual orientation.

Anti-discrimination legislation exists in various jurisdictions around the world. In the US, anti-discrimination laws exist under the Civil Rights Act [22], and under specific areas like credit lending [1] and housing[2]. There have also been efforts to introduce legislation combating algorithmic bias[3]. In the European Union, the General Data Protection Regulation (GDPR) provides for regulations regarding digital profiling, data collection, and a right to explanation [85]. Under Indian law, quotas for *scheduled castes*, *scheduled tribes* and *other backward classes* are mandated in public education and government employment.[4]

[1] `https://www.justice.gov/crt/equal-credit-opportunity-act-3`
[2] `https://www.justice.gov/crt/fair-housing-act-1`
[3] `https://www.congress.gov/bill/116th-congress/house-bill/2231/all-info`
[4] `http://www.legalservicesindia.com/article/1145/Reservations-In-India.html`

2.2 Fair Machine Learning Techniques

In the research community, there is a growing body of work on ways to detect algorithmic bias [15, 79] and develop techniques to build machine learning algorithms that incorporate fairness in the system design of models itself.

These techniques can be applied at different stages of the machine learning pipeline, including preprocessing, in-processing, and post-processing techniques [19]. Preprocessing techniques involve modifying the training data to reduce bias and ensure that the resulting model is fair. This can include techniques such as oversampling or undersampling to balance the representation of different groups in the data. In-processing techniques involve changing the machine learning algorithm itself to ensure that it produces fair outcomes. This can include adding fairness constraints to the optimization problem, or using techniques such as adversarial training to reduce bias. Post-processing techniques involve modifying the output of the machine learning algorithm to ensure that it is fair, such as by adjusting decision thresholds to ensure that decisions are not based on sensitive features.

Fair machine learning techniques can be applied to a variety of machine learning tasks, for instance classification [83, 98, 107, 143], causal inference [133, 152], word embeddings [32, 35], regression [5, 23], and retrieval/ranking [44, 190, 216]. In this thesis, I have specifically looked into the real world challenges of implementing fair classification and ranking.

2.2.1 Fair Classification

In 2012, [60] introduced the idea of fairness in machine learning classifiers by incorporating protected attributes directly into the model and jointly optimizing for accuracy and fairness. This led to the development of what I call *classical fair classifiers*, which incorporate fairness constraints into classical machine learning algorithms like decision trees and SVMs [140]. Classical fair classifiers are now widely available to practitioners [19, 129, 172], and have been adopted by real-world systems [62]. However, these classifiers rely on the assumption that data about protected attributes is accurate, which may not always be the case in practice [77].

In response to these limitations, researchers have developed what I call *noise-tolerant fair classifiers* and *demographic-blind fair classifiers*. Noise-tolerant fair classifiers jointly optimize for accuracy and fairness in the presence of uncertainty in the protected attribute data, and have been developed using approaches such as robust optimization [209], adjusting the "fairness tolerance" value [122], using noisy attributes to post-process the outputs for fairness

under conditional independence assumptions [13], estimating de-noised constraints that allow for near optimal fairness [41], or a combination of approaches [149]. A different approach for achieving fairness through awareness that is amenable to these strong constraints is embodied by what I refer to as demographic-blind fair classifiers. These algorithms do not take protected attributes as input, but they attempt to achieve demographic fairness anyway by relying on the latent representations of the training data [90, 121]. Thus, this approach to classification still incorporates a general awareness of unfair discrimination and historical inequity without being directly aware of demographics.

2.2.2 Fair Ranking

Fair Information Retrieval (IR) algorithms have received comparatively less attention than classification algorithms in the literature. Initial studies that examined fair IR proposed to solve this in a binary context, i.e., make a ranked list fair between two groups [44, 216]. Subsequent work uses constrained learning to solve ranking problems using classic optimization methods [190]. There are also methods that use pairwise comparisons [24] and describe methods to achieve fairness in learning-to-rank contexts [148, 217].

In industrial settings, researchers at LinkedIn have proposed an algorithm that uses re-ranking in post-processing to achieve representational parity [74]. However, recent work by [77] shows how uncertainty due to incorrect inference of protected demographic attributes can undermine fairness guarantees in IR contexts. Fairness methods that do not require explicit demographic labels at runtime are an emerging area of focus in classification [121] and ranking [108, 171]. One example that has been studied at large-scale is Shopify's Fair Maximal Marginal Relevance (FMMR) algorithm [108].

2.3 Practical Challenges of Fair Algorithms

2.3.1 Shortcomings of Conventional Fairness Metrics

Several conceptual definitions of fairness have been discussed in the literature that, according to [48], fall into three general classes: (1) *anti-classification*, where protected features and their proxies are not used to make decisions, (2) *classification parity*, where measures of model predictive performance are equal across protected groups, and (3) *calibration*, where the outcomes, conditional on priors, are independent of protected features. Corbett-Davies

Metric/Framework	Related Terms	CO?	E?
Demographic parity [60]	mean difference, demographic parity, disparate treatment	✗	✗
Conditional statistical parity [49]	statistical parity, conditional procedure accuracy, disparate treatment	✗	✗
Equalized odds [89]	equalized odds, false positive/negative parity, disparate treatment	✗	✗
Equal opportunity [89]	equality of opportunity, individual fairness, disparate treatment	✗	✗
Counterfactual fairness [29, 120]	counterfactual fairness, disparate treatment, fliptest	✗	✗
Statistical independence [87]	HGR coefficient, independence	✓	✗
Distributional difference [145]	KL divergence, JS Divergence, Wasserstein distance	✓	✓

Table 2.1: Summary showing whether conventional classes of fairness metrics support Continuous Output (CO) and feature-level Explanations (E). Metric families are inspired by [140] and the related terminology is from [52].

and Goel dissect fairness metrics that implement these definitions, claiming that they have "deep statistical limitations" [48], with several metrics at odds with one another.

Table 2.1 shows an overview of the terminology and limitations of different classes of fairness metrics in the literature. I refer to the first five frameworks (demographic parity, conditional statistical parity, equalized odds, equal opportunity, and counterfactual fairness) as "conventional" fairness metrics because of their prevalence in algorithmic fairness literature [140] and in the industry [18]. The last two classes, statistical independence and distributional difference, are relatively niche and new to the discussion.

Conventional fairness metrics have impossibility results [155]. Prior work [48, 113, 145] points out that it is impossible to satisfy both *classification parity* and *calibration* metrics at the same time in general, and therefore context becomes key when picking a metric [14, 183].

These statistical limitations extend to group membership limitations. Conventional fairness metrics require groups and subgroups to be discrete variables and cannot work with continuous variables [79]. Similarly, "confusion matrix based-metrics" [155] do not support continuous outputs (which is often the case in problems like regression and recommendation). This limitation necessitates that practitioners choose thresholds for determining if the given fairness metric has been violated, but the process for choosing these thresholds is ad hoc and may lead to wildly different conclusions about the fairness of a model. I show an example of this phenomenon in Figure 2.1.

The fairness metric I developed for FairCanary (introduced in 1.6) is called Quantile Demographic Drift (QDD). It is a quantile-based optimized version of the Wasserstein-1 distance metric [206]. It falls under the distributional difference family (see Table 2.1) and thus lends itself to continuous measurement and explainability.

2.3.2 Noise in Inferred Attributes

A built-in assumption in many fair algorithms is the presence of accurate demographic labels. Unfortunately, this may not be true in practice. In contexts like finance and employment candidate screening, demographic data may not be available due to legal constraints or social norms [31, 211], yet the need to fairly classify or rank people remains paramount. To bridge this gap, practitioners may infer peoples' protected attributes using human labelers [16] or algorithms that take names, locations, photos, etc. as input [1].

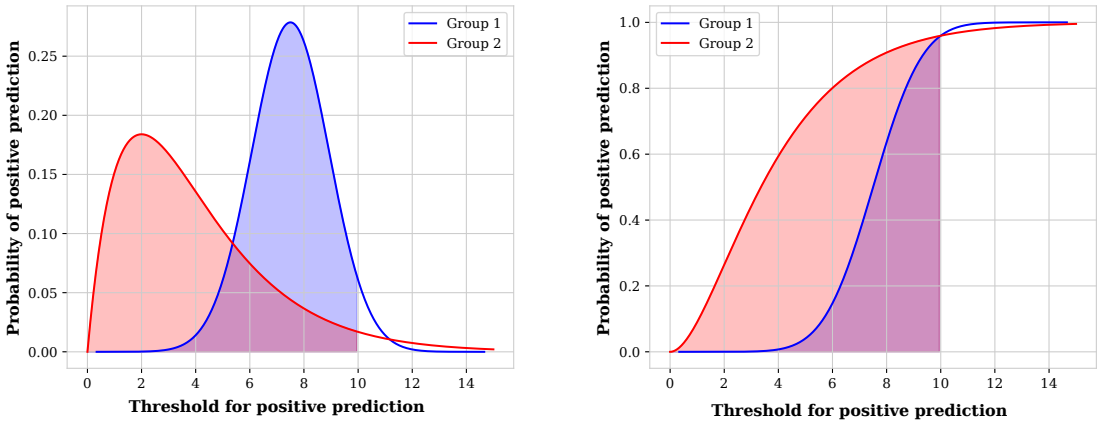
There are examples in the literature that highlight accuracy problems with demographic inference algorithms, perhaps most notably when [36] showed how the accuracy of facial analysis systems at predicting gender fell when presented with images of darker-skinned people. [72] note that leveraging crowd workers to produce demographic data is also problematic, and work on the best ways to collect such data is only beginning to emerge [103].

The interactions between noise in protected attribute data and algorithms trying to ensure fairness is sparsely studied despite its potentially far-reaching consequences. There have been studies on the stability of classification algorithms with noisy data [173]. [67] note that classifiers may not be stable in the face of variations in the training dataset. [46] analyzed disparity under unobserved protected attributes using demographic inference, but they do not study the impact of inference on the fairness providing algorithm itself, thus providing an avenue for closer inspection for the real world outcomes of noise introduced due to imperfect inference in fair algorithms.

2.3.3 Adversarial Attacks

Adversarial Machine Learning is a growing field of research that aims to develop methods and tools that can subvert the objectives of ML algorithms. For example, prior research has highlighted that deep learning models are often not robust when presented with inputs that have been intentionally, maliciously crafted [40, 84, 147, 160, 187, 207, 212].

Several proposed defenses against state-of-the-art adversarial ML attacks have been defeated [12, 202], and *adversarial examples* (i.e., maliciously crafted inputs) have been shown to transfer across models performing similar tasks [131, 203]. The most promising defense method, adversarial training, is computationally expensive and imperfect—it results in decreased standard accuracy while still having a somewhat low adversarial accuracy [70,



(a) Probability Density Plot

(b) Cumulative Distribution Plot

Figure 2.1: Probability distribution plots for two hypothetical demographic groups. As demonstrated by the CDF plot on the right, at a threshold of $x = 10$ the positive prediction probability for both groups is about 0.95, thereby satisfying Demographic Parity $[P(Y^*)|D_1 = P(Y^*)|D_2]$, but this is misleading: the Wasserstein distance is nonzero since the two distributions have markedly different shapes. In contrast, moving the threshold to $x = 8$ immediately disadvantages one group, since the positive prediction probability for group 1 falls to 0.6 while for group 2 it only falls to 0.9, thereby violating Demographic Parity.

119, 186]. As such, adversarial ML presents a significant hurdle to deploying neural models in sensitive real-world systems.

My work considers adversarial ML attacks on IR systems. Previous work has demonstrated successful attacks on image-to-image search systems [132, 221], allowing an adversary to control the results of targeted image queries using localized patches [34] or *universal adversarial perturbations*[5] [125]. Other work has demonstrated attacks on text-to-text retrieval systems [167] and personalized ranking systems [92]. Work by [221] hypothesized that targeted attacks on selected items in a ranked list might be possible using universal adversarial perturbations. None of these works consider compromising text-to-image models or group fairness objectives, as I do in this study.

Prior work has demonstrated adversarial ML attacks against fairness objectives of ML systems at *training time*. In these attacks, the adversary supplies *poisoned* training data, which then results in models that either compromise the accuracy of targeted subgroups [99, 141] or exacerbate pre-existing unfairness between subgroups [45, 192]. Specific to classification,

[5]A Universal Adversarial Perturbation (UAP) is a an adversarial perturbation that generalizes to all classes of a target classifier, i.e., one patch to untargeted attack as many classes as possible.

there exists theoretical work that shows how to learn fair models when sensitive attributes are noisy [41] or corrupted in a poisoning attack [43], but they do not consider ranking.

Adversarial ML attacks at *test time*—i.e., after training a model using non-malicious data—that I consider in this work are relatively unexplored in fairness settings. [153] show that adversarial ML attacks can harm certain subpopulations more than others in classification tasks. However, while this is an important observation, the harms suggested by this work may be difficult to realize in practice, as they only involve disparity between examples that are adversarially corrupted. By contrast, my work shows that test-time attacks can harm fairness for benign data when launched in a ranking setting.

2.3.4 Drift

Even though a ML model may pass quality control in terms of performance during training, once deployed on live data the model may encounter issues over time that degrade or destroy its performance [181]. One of the issues that can arise in deployment is *drift*, which is caused by divergence between the data and context under which the model was trained, and the real-world context into which it is deployed. *Data drift* occurs when the runtime data is significantly different from the training data, by virtue of the constant changing of real world data [33]. *Concept drift*, in contrast, occurs when the relationship between the model output and the feature variables change [158, 174, 194].

Scholars have noted that model performance issues caused by drift extend to questions of algorithmic fairness [14], i.e., the removal of unfair and unjustified biases from ML and AI systems. For example, a temporal analysis by [130] showed how the changing of fairness metrics over time, due to data drift, concept drift, or otherwise, could actually harm sensitive groups.

The most popular methods for detecting concept drift [27, 69] assume that the labels for the predicted variable are immediately available. This may not be feasible in practice, however, especially if the labels correspond to sensitive features of human beings. Furthermore, even if labels are immediately available, concept drift may have rendered them unreliable, thus defeating the purpose of using them to detect concept drift. Given these issues, prior work [59, 75, 161, 222] has measured the drift of prediction distributions as a proxy for concept drift.

Prediction distribution drift is a new method that FairCanary uses to measure temporal

unfairness. Instead of measuring the drift of the production prediction distribution against training prediction distribution, like in [59, 75, 161, 222], I measure the shift in the prediction distributions between different protected groups. If the prediction distributions for two protected groups start diverging over time, that is an indication of unfairness.

The primary mitigation against drift is retraining models on fresher data. Retraining may be expensive, however, so determining when to retrain models is crucial: retraining too frequently wastes (potentially substantial) resources [20], while waiting too long runs the risk of performance degradation.

2.3.5 Model Monitoring and Explanations

Continuous model monitoring systems are designed to help developers ensure that deployed models—either classifiers or rankers—perform as expected over time in the face of problems like drift. A number of commercial tools are available that offer model monitoring [50]. In general, these systems offer the following features:

- Continuously record model inputs and model predictions.
- Measure and report traditional performance metrics over time, like precision, recall, and accuracy. Some systems also measure bias/fairness metrics.
- Calculate and record feature-level explanations using techniques like LIME [169] or SHAP [54, 135, 144], which are useful for post-mortem analysis if problems are observed.
- Generate alarms if particular metrics fall below an operator-specified threshold.

Continuous model monitoring systems are useful for uncovering a variety of issues with models at deployment time, including issues caused by drift. Once the developer has identified an issue they can apply mitigations, such as model retraining.

While virtually all of the commercially available monitoring systems explain predictions in terms of constituent features, none of them (to the best of my knowledge) offer explanations for measures of unfairness. I argue that it is equally important to understand which particular input features are responsible for causing unfairness to the model over time, especially given the "right to explanation" that is increasingly being enshrined in regulation [182].

Unfortunately, the interpretation of fairness metrics in terms of the input features to the model has not been studied extensively so far. Explaining conventional fairness

metrics (see Table 2.1) that rely on ground truth labels using Shapley values is possible by making the assumption that the perturbed values retain the original output label. This approach can be misleading, however, because the perturbations change the nature of the instance, and can even create Out-of-Distribution (OOD) points [117]. Another approach, proposed by [188], explains differentiable distance metrics using integrated gradients, but this technique only applies to differentiable models, which limits its practical applications. Finally, [145] developed methods to explain the Wasserstein-1 distance using a Shapley value formulation. However, this approach also suffers from practical challenges: (1) it requires that explanations be computed for every possible pair of protected groups, and (2) it is computationally challenging to compute Shapley values over large samples.

Chapter 3

When Fair Ranking Meets Uncertain Inference

3.1 Research Problem

One particular area of concern while exploring the real-world challenges of implementing fair algorithms is the potential for biases to be introduced when demographic information is used as an input. This issue has far-reaching implications, as algorithms that produce unfair or biased outcomes can have a significant impact on individuals and communities.

In this chapter, I will explore the question of how noise in demographic information can affect the intended fairness of machine learning outcomes. To do so, I will conduct empirical experiments using simulations and case studies. My goal is to evaluate the effectiveness of one or more fair ranking algorithms in achieving their stated fairness objectives when given input data that includes both ground-truth and inferred demographic information.

To accomplish this, I will use a range of fair ranking metrics that encompass different definitions of fairness. Additionally, I will use error rates drawn from inference algorithms to assess the impact of noisy demographic information on algorithmic fairness. By evaluating algorithms and datasets drawn from real-world deployments, I aim to ensure the relevance and real-world applicability of my findings.

Through these experiments, I hope to gain a better understanding of how demographic noise can affect the fairness of machine learning outcomes, and to identify strategies for mitigating these effects. With these goals and guiding principles in mind, I now move on to

selecting algorithms and metrics.

3.2 Algorithms and Metrics

3.2.1 Fair Ranking Algorithm

The fair ranking algorithm I chose for this study was developed by [74] from LinkedIn. Their paper presents four different re-ranking algorithms with varying stability but with one central goal: to achieve the desired distribution of population in the top-ranked results with respect to one or more protected attributes. At a high-level, the algorithm takes an unfairly arranged list and an integer K then generates a fairness-aware list of the top K candidates such that the fraction of candidates in each subgroup matches their fraction in the underlying population. While other algorithms from prior work [44, 190, 216] have similar goals, this algorithm was extensively tested and deployed in LinkedIn's Talent Search system. The authors of the paper claim that the deployment led to *"tremendous improvement in the fairness metrics (nearly three-fold increase in the number of search queries with representative results) without affecting the business metrics, which paved the way for deployment to 100% of LinkedIn Recruiter users worldwide"* [74]. Since my work focuses on the possible breakdown of fair ranking algorithms in real-world, deployed scenarios, this work was the best fit for my research purposes.

Of the four algorithms presented in the [74] paper, I chose the Deterministic Constrained Sorting algorithm or *DetConstSort* as my benchmark fairness algorithm since it is theoretically proven to be feasible for protected attributes having a large number of possible attribute values, unlike the other three greedy fair ranking algorithms in the paper.

DetConstSort creates a ranked list of candidates, such that for any particular rank k and for any group attributes g_j, the attribute occurs at least $\lfloor p_{g_j}.k \rfloor$ times in the ranked list (p_{g_j} = proportion of members in the list belonging to g_j). However, unlike other fair ranking algorithms that greedily pick the best candidate for a particular rank, the *DetConstSort* algorithm also strives to improve the sorting quality by re-ranking the candidates that come above it (so that candidates with better scores are placed higher in the list), as long as the resultant list satisfies the feasibility criteria. Thus, the algorithm can be conceptualized as solving a more general interval constrained sorting problem. Since the *DetConstSort* algorithm is constrained to be feasible it optimizes the Skew and NDKL fairness metrics,

which I introduce in the next section.

3.2.2 Metrics for Ranking Evaluation

The second decision I needed to make to accomplish my study was choosing metrics for evaluating the fairness of representation in ranked lists. Fairness definitions in the fair machine learning literature include concepts such as equalized odds, equal opportunity, demographic parity, and treatment parity [60, 89]. These concepts have been adapted specifically to the domain of ranking, with metrics developed by researchers measuring the underlying population representation in the top-ranked items [213] and conceptualizing ranking fairness as an attention or exposure allocation problem to different subgroups [178,190]. Consideration of the cardinality of protected categories is also important, with conventional binary metrics [116, 216] unable to assess fairness between multiple groups. Newer metrics that compare entire population distributions over an unspecified number of subgroups [74], or attention-based metrics that deal with population distributions [26, 178, 190], are agnostic to group cardinality and lend themselves to intersectionally fair frameworks [66, 79].

I focus on metrics that (1) assess *group fairness* [60], possibly balanced against secondary objectives, and (2) are capable of dealing with multiple subgroups (i.e., not just binary protected versus unprotected classes). For my analysis, I adopted the definition of a subgroup as a Cartesian product of ≥ 2 groups, as defined in [79]. A subgroup $sg_{a_1....a_n}$ is defined as set containing the intersection of all members who belong to groups g_{a_1} through g_{a_n}, where a_1, $a_2...a_n$ are marginal protected attributes like race, gender, etc. Notation wise:

$$sg_{a_1 \times a_2 \times ... \times a_n} = g_{a_1} \cap g_{a_2} ... \cap g_{a_n}. \tag{3.1}$$

Note that if the metrics satisfy fairness for a set of subgroups they will also be fair for the constituent marginal groups [66].

3.2.2.1 Representation-based Metrics.

To get an overall sense of group fairness in a given ranked list, I chose two (slightly modified) representation-based metrics introduced by [74]. These metrics do not incorporate attention, i.e., they assess the representation of people from different groups based solely on how many of those people appear in the list relative to the underlying population. The first metric is computed per group, while the second is aggregated across groups.

Skew. Given a ranked list τ, the Skew for attribute value sg_i at position k is defined as

$$\text{Skew}_{sg_i}@k(\tau) = \frac{p_{\tau^k, sg_i}}{p_{q, sg_i}} \tag{3.2}$$

where p_{τ^k, sg_i} represents the proportion of members belonging to subgroup sg_i within the top k items in the ranked list τ, and p_{q,sg_i} represents the proportion of members belonging to subgroup sg_i in the overall population q. Ideally, $\text{Skew}_{sg_i}@k$ should be close to one for each sg_i and k, indicating that people from sg_i are represented in τ proportionally relative to the underlying population. $Skew_{sg_i}@k > 1$ denotes that the subgroup sg_i is over-represented among the top k candidates, and vice versa when the $Skew_{sg_i}@k < 1$.

Divergence. Given a ranked list τ, the Normalized Discounted Kullback–Leibler (NDKL) Divergence is defined as

$$\text{NDKL}(\tau) = \frac{1}{Z} \sum_{i=1}^{|\tau|} \frac{1}{log_2(i+1)} d_{KL}(D_{\tau^i} || D_r) \tag{3.3}$$

where $d_{\text{KL}}(D_1||D_2) = \sum_j D_1(j) log_2 \frac{D_1(j)}{D_2(j)}$ is the KL divergence score of distribution D_1 with respect to distribution D_2 and $Z = \sum_{i=1}^{|\tau|} \frac{1}{log_2(i+1)}$. NDKL can be interpreted as a weighted average of the logarithm of the Skew scores for all the groups in a ranked list. NDKL values close to zero indicate that people from all subgroups are represented proportionally in a given ranked list, since the KL-Divergence of the population between the top k candidates and the underlying population will be zero. A large difference in the distributions of the different groups in the top k ranked candidates leads to a higher NDKL score.

3.2.2.2 Attention-based Metrics.

Studies have repeatedly shown that people do not pay equal attention to all items in ranked lists [151, 156]; rather, peoples' attention decays as they progress down the list, eventually abandoning the task entirely. This observation suggests that using overall representation to assess fairness is misleading, since (1) people may not look at all available items and (2) they pay more attention and are thus more likely to act on higher ranking items. To take attention into account, I computed attention per group and in aggregate across groups like in 3.2.2.1.

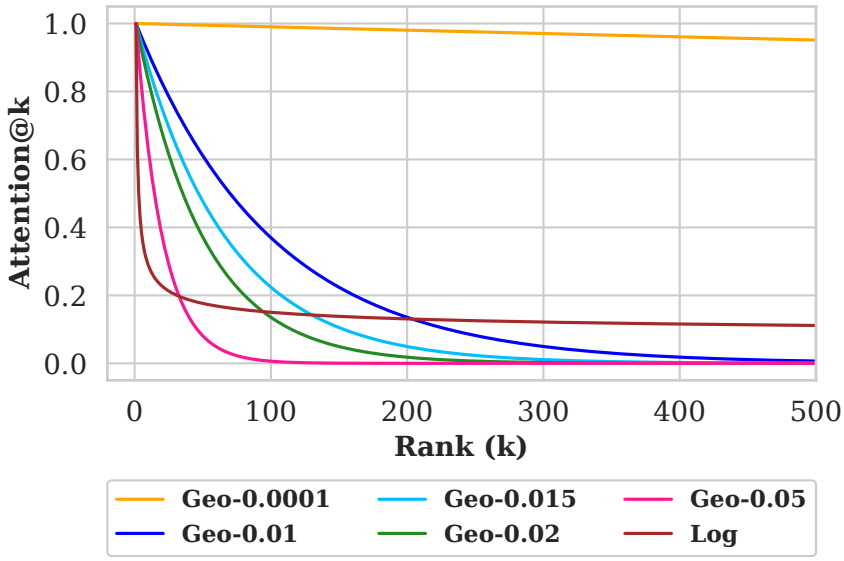


Figure 3.1: Attention versus rank for six attention functions.

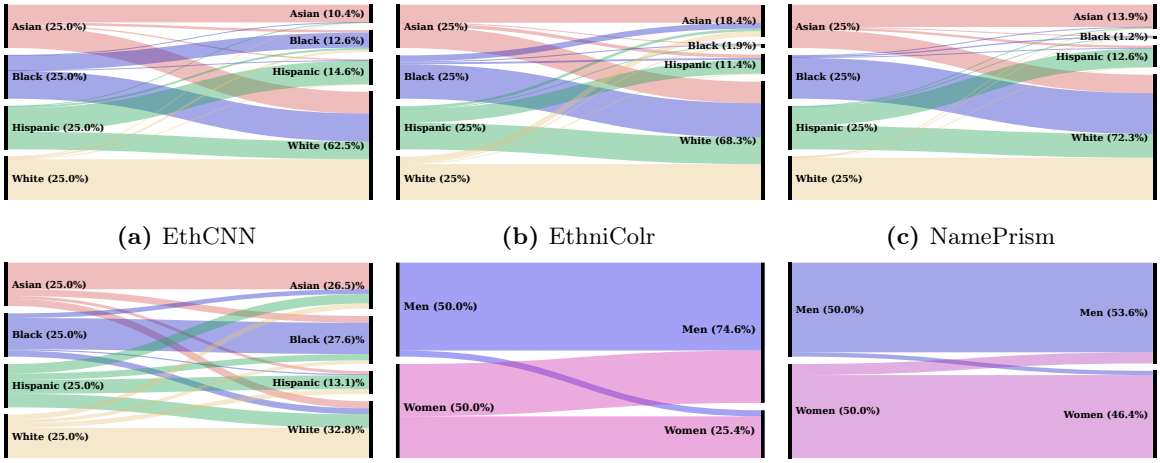
Attention. In this study, I adopted the geometric distribution to model decay in attention, similar to prior work by [178]. I compute attention at k as

$$\text{Attention}_p@k(\tau) = 100 \times (1-p)^{k-1} \times (p) \tag{3.4}$$

where τ is the ranked list and p represents the proportion of attention provided to the first result. For my experiments I set $p = 0.015$ because at this value attention decays to zero at $k = 300$, which is the value of k I fix for my experiments. Although most prior work in the Information Retrieval (IR) literature uses logarithmic decay to model attention [190, 213], I did not adopt it because it models attention decay at an unrealistically slow rate [170] and its shape flattens out at low ranks. Figure 3.1 shows how attention decays as a function of rank for a variety of values of p, as well as under logarithmic decay.

The i^{th} element in τ has an associated score, denoted s_i^τ, that corresponds to the utility or relevance of the item, and a subgroup-attribute value, denoted by sg_i^τ. The elements in the ranked list are arranged in decreasing order of score such that $s_i^\tau \geq s_j^\tau \; \forall \; i \leq j$. I define

$$\eta_{sg_j,\tau} = \frac{1}{|sg_j|} \sum_{i=1}^{|\tau|} \text{Attention}_p@i \; \forall \; sg_i^\tau \in sg_j \tag{3.5}$$



(a) EthCNN **(b)** EthniColr **(c)** NamePrism

(d) DeepFace: Race/Ethnicity **(e)** DeepFace: Gender **(f)** Genderize

Figure 3.2: Sankey plots showing the distribution of ground-truth (left) and inferred (right) demographic traits for five algorithms. The algorithms tend to mis-classify minorities as Whites. DeepFace tends to mis-classify Women as Men.

where $\eta_{sg_j,\tau}$ denotes the mean attention score of the sg_j protected attribute for τ and

$$\text{ABR}_\tau = \frac{\min_{sg_j}(\eta_{sg_j,\tau})}{\max_{sg_j}(\eta_{sg_j,\tau})} \tag{3.6}$$

where ABR_τ or the Attention Bias Ratio for the ranking τ quantifies the disparity between the groups with the lowest and highest mean attention score ($\eta_{sg_j,\tau}$). $\text{ABR}_\tau = 1$ is the ideal score, i.e., all subgroups (and thereby all groups) receive equal attention.

3.2.2.3 Ranking Quality Metrics.

Classic IR literature has proposed several evaluation metrics to measure the ranking quality of an IR system [138]. I measure two different metrics in my study: a cumulative gain based metric, and a rank change metric to measure loss in ranking utility.

Normalized Discounted Cumulative Gain. NDCG is a widely used measure to evaluate search rankings [100].

$$\text{NDCG}(\tau) = \frac{1}{Z}\sum_{i=1}^{|\tau|}\frac{s_i^\tau}{log_2(i+1)} \tag{3.7}$$

where s_i^τ is the utility score of the i^{th} element in the ranked list τ and $Z = \sum_{i=1}^{|\tau|}\frac{1}{log_2(i+1)}$.

Rank Change. This metric is a measure of the amount of itemwise distortion from the original list to the fairness-aware re-ranked list, much like the ranking utility loss measure in [216]. I define Rank Boost as the boost in the rank of an item due to re-ranking. For a candidate C_A,

$$\text{Rank Boost}_{C_A} = \tau_{org}[C_A] - \tau_{new}[C_A] \tag{3.8}$$

where τ_{org} and τ_{new} denotes the original and re-ranked list respectively. A positive value indicates that a candidate was assigned a higher rank after re-ranking.

For a subgroup sg_j, the Average Rank Change (ARC) is defined as the average of the absolute rank boosts over all candidates in that subgroup:

$$\text{ARC}_{sg_j,\tau} = \frac{1}{|sg_j|}\sum_{i=1}^{|\tau|}|\text{Rank Boost}_{C_i}|;\ \text{where } C_i \in sg_j. \tag{3.9}$$

Finally, I define Maximum Absolute Rank Change (MARC) for a particular list as the maximum value of the ARC over all subgroups in that list:

$$\text{MARC}_{\tau} = max(ARC_{sg_i,\tau});\ \forall sg_i \in |sg|. \tag{3.10}$$

3.2.3 Demographic Inference Algorithms

The final decision I needed to make for this study was selecting demographic inference algorithms. My intent is to compare the fair rankings generated by the *DetConstSort* algorithm when given ground-truth and inferred demographic information, using the metrics introduced in 3.2.2, so as to quantify the impact (if any) of mis-classifications.

I chose five diverse inference algorithms that rely on different features and machine learning techniques. For each algorithm, I computed its confusion matrix when predicting peoples' ethnicity/race and gender (in one case) using ground-truth data with known demographics. I evaluated the four algorithms in 3.2.3.1 using voter records from the state of North Carolina,[1] which are publicly available records that have the name, address, race, gender, and other personal information of each registered voter in the state. I evaluated the facial analysis algorithm in 3.2.3.2 using the FairFace dataset [110].

Using these five algorithms I predicted race/ethnicity (Asian, Black, Hispanic, and White) and gender (man and woman). I fully acknowledge that these categories are problematic,

[1] `https://www.ncsbe.gov/results-data/voter-registration-data`

however, I adopted them because they are the categories supported by the inference algorithms from prior work. I discuss the problems and limitations that derive from these categories in 4.5.

Figure 3.2 shows the results of demographic inference using these five algorithms. I used these confusion matrices in my experiments to intentionally mis-classify data, so as to observe the effect on fair ranking performance.

3.2.3.1 Name-based Inference.

I chose three algorithms that attempt to predict peoples' race/ethnicity based on their name, and I choose one algorithm that does so for gender prediction.

EthCNN. I employed a Convolutional Neural Network (CNN) architecture similar to [112] to infer peoples' ethnicity from their names, where the name is represented as a sequence of characters.

EthniColr. Inspired by the work of [94], I used Ethnicolr,[2] the publicly available library from [193], to predict an individual's race/ethnicity from their full name. Ethnicolr employs a neural architecture to model the relationship between the characters in a name and race/ethnicity.

NamePrism. I used the NamePrism API[3] by [214] for race/ethnicity classification. Motivated by the observation that individuals frequently communicated with peers of similar age, language, and location [124], Nameprism exploits the homophily phenomena in email contact lists to create name embeddings that can be used to predict race/ethnicity.

Genderize. To infer binary gender from names I used a service called genderize.[4] As of 2021, the dataset underlying genderize consists of 114,541,298 names collected from 242 countries and territories. While the sources of the names are not revealed [176], the site claims that the API has been used for data analysis in articles from the Guardian, the Washington Post, and other outlets.

3.2.3.2 Facial Analysis-based Inference.

I selected one algorithm that infer demographics from images of faces.

[2] `https://github.com/appeler/ethnicolr`

[3] `http://www.name-prism.com/`

[4] `https://genderize.io/`

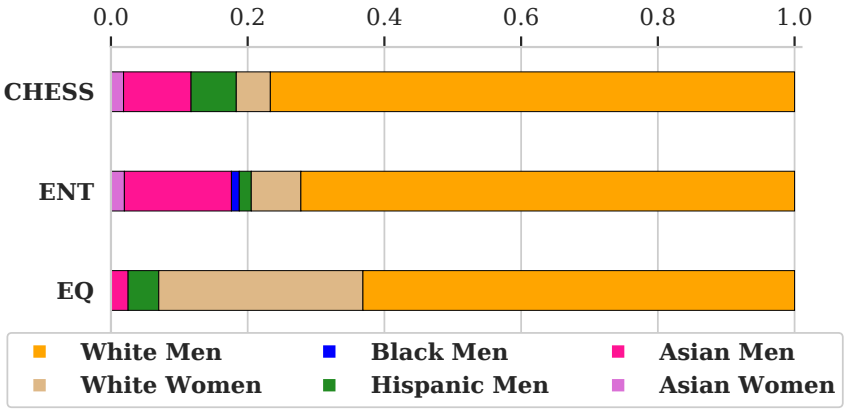


Figure 3.3: Population subgroups in my Chess players, Entrepreneurs, and Equestrians datasets.

DeepFace. I used the public wrapper [184] for DeepFace by Facebook [201] to obtain DeepFace's error rates when classifying race/ethnicity and gender from the FairFace dataset [110].

3.3 Experiments

In this section I outline the experiments that I performed to examine the relationship between inferred demographics and fair ranking.

3.3.1 Simulations

In my first experiment, I examined the relationship between demographic mis-classification and fair ranking guarantees under controlled conditions by performing simulations using synthetic data. I used a modified version of the synthetic ranked list generation method discussed in [74], as follows:

1. I manually crafted six ground-truth probability distributions P for the protected attributes of the simulated people. The distributions, labeled A through F and shown in Table 3.1, each contained three or four groups. These were the target distributions of my fairly re-ranked lists.

2. For each probability distribution P, I generated 1,000 people per group $g_i \in P$ and assigned each a random utility score $s_i \in [0, 1]$. I then sorted the combined list of people in decreasing order of s_i to generate the ranking τ.

Distribution	NDKL	ABR
Dist A (W: 0.33, B: 0.33, A: 0.33)	0.08	0.66
Dist B (W: 0.2, B: 0.3, A: 0.5)	0.08	0.71
Dist C (W: 0.1, B: 0.3, A: 0.6)	0.30	0.86
Dist D (W: 0.1, B: 0.2, A: 0.7)	0.37	0.91
Dist E (W: 0.25, B: 0.25, A: 0.25, H: 0.25)	0.11	0.60
Dist F (W: 0.1, B: 0.2, A: 0.6, H: 0.1)	0.42	0.70

Table 3.1: Fairness metrics computed between the target distribution on the left (*A*sian, *B*lack, *H*ispanic, and *W*hite) and randomly generated unfair distributions. NDCG and MARC for the unfair lists are 1.0 and 0 in all cases.

3. I ran the *DetConstSort* algorithm discussed in 3.2.1 with the desired distribution P and τ as inputs to produce the fairness-aware re-ranked list τ_f. $|\tau_f| = 300$.

4. I calculated NDKL, ABR, NDCG, and MARC on τ and τ_f.

5. I repeated steps 2–4 100 times and computed the mean values for my metrics.

6. I repeated steps 2–5 for demographic prediction accuracies varying from 0.1 to 1.0. For instance, an accuracy of 0.1 meant that the attribute g_i was predicted correctly 10% of the time and therefore, in my simulation, I mis-classify g_i as any g_j where $j \neq i$ 10% of the time.

Table 3.1 shows the mean fairness metrics for my empirical distributions before running *DetConstSort*.

3.3.2 Case studies

To establish the ecological validity of my study, I perform detailed case studies on three ranked lists obtained from the real world, to measure the potential harms of algorithmic demographic inference on fairness-aware re-ranking tasks.

3.3.2.1 Datasets.

For my case studies, I required datasets that had both names and images for my inference algorithms discussed in 3.2.3. I collected these datasets from three publicly available sources on the internet, as discussed below. All three datasets were downloaded in January 2021.[5]

[5]The code and datasets used in this paper can be found at `https://github.com/evijit/SIGIR_FairRanking_UncertainInference`.

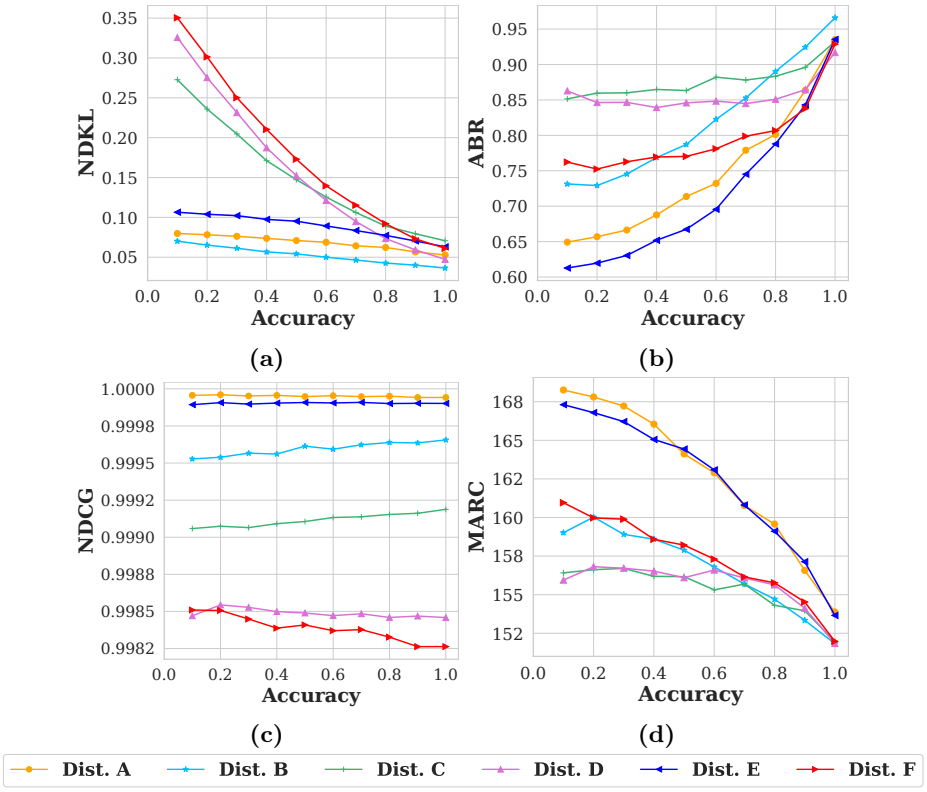


Figure 3.4: Distributions of NDKL, ABR, NDCG and MARC scores for fairly-ranked lists as demographic inference accuracy was varied, based on simulations using synthetic data. For details about the ground-truth population distributions, refer to Table 3.1.

Chess Rankings. I downloaded a ranked list of chess players sorted by their World Chess Federation (French: Fédération Internationale des Échecs or FIDE) ratings from the official FIDE website.[6] Along with the ratings, the website also provided the full name, image, and self-identified binary gender of the players.

Crunchbase Entrepreneurs Ranking. I downloaded a list from Crunchbase[7] of me startup founders who received Series A funding in the last 5 years. I summed the Series A funds of founders who were part of multiple Series A funding rounds during this time frame. I collected the name, image, and self-identified binary gender of the founders.

Equestrian Rankings. I downloaded a ranked list of equestrian athletes arranged by their Fédération Equestre Internationale (FEI) ratings from the official FEI website.[8] Along

[6] `https://ratings.fide.com/`
[7] `https://crunchbase.com/`
[8] `https://www.fei.org/jumping/rankings`

Algorithm	Inference Type	Race	Gender	Fair
BASE (Baseline)	None	Perceived	Ground Truth	No
ORCL (Oracle)	None	Perceived	Ground Truth	Yes
CNNG (EthCnn_Gen)	Name	EthCNN	Genderize	Yes
ECLG (Ethnicolr_Gen)	Name	Ethnicolr	Genderize	Yes
NPMG (Nameprism_Gen)	Name	Nameprism	Genderize	Yes
DPFC (Deepface)	Face image	DeepFace	DeepFace	Yes

Table 3.2: The algorithms and sources of demographic data (ground-truth, perceived, inferred) used in my case studies.

with the ratings, I also collected the full name, image, and self-identified binary gender of the athletes.

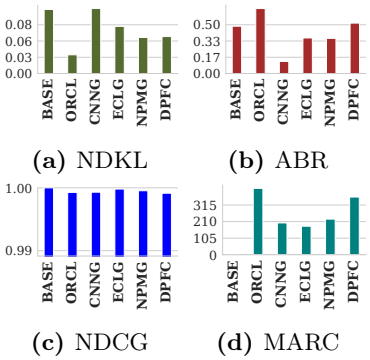
3.3.2.2 Data Annotation and Cleaning.

The datasets I collected in 3.3.2.1 contain ground-truth gender information[9] for each individual, but not race/ethnicity. To obtain race/ethnicity information, I followed a similar process as the annotation method for the occupations dataset in [42]. I asked workers from Amazon Mechanical Turk to label the images of faces that I collected. For each image, I asked workers to choose from among the following races/ethnicities based on their best judgment: White/Caucasian (Non Hispanic), Hispanic/Latino, Black/African, Asian (Far East, Southeast Asia, and the Indian subcontinent), or Other/Not sure. Each image was labeled by three independent workers and I accepted the label with majority support. After labeling I dropped 6%, 4%, and 2% of people from my lists, respectively, because they lacked majority consensus. I restricted my task to workers with $\geq 90\%$ approval ratings and my task paid roughly \$12/hour.

I do not refer to the race/ethnicity labels that I obtained from crowd sourcing as "ground-truth" because there are no phenotypical determinants of race or ethnicity. Instead, I refer to these labels as "perceived" because they correspond to the perceptions of race and ethnicity held by my workers, as informed and filtered through their own cultural lenses.

For each ranked list, I cleaned the dataset by removing all entries that did not have a picture, and then by removing subgroups that had a population less than 1% of the total length of the list. The final datasets consisted of 3,251 chess players, 3,308 startup founders, and 1,115 equestrian athletes, respectively. Figure 3.3 shows the population summary statistics for my three datasets, broken down into intersectional subgroups.

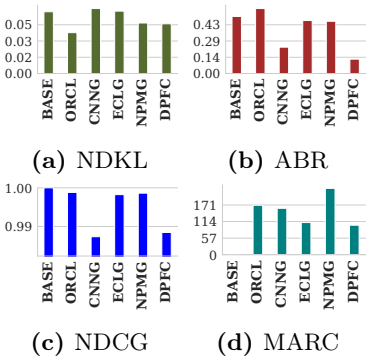
[9]I consider these gender labels to be ground-truth because they are self-identified. Unfortunately, these organizations force individuals to identify with a binary gender.



(a) NDKL (b) ABR

(c) NDCG (d) MARC

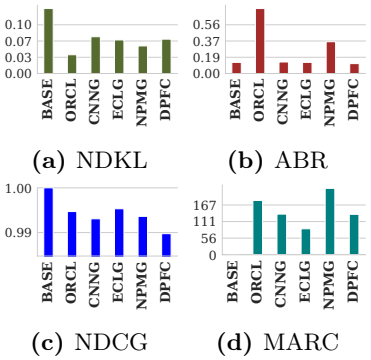
Figure 3.5: Chess Dataset.



(a) NDKL (b) ABR

(c) NDCG (d) MARC

Figure 3.6: Entrepreneurs Dataset.



(a) NDKL (b) ABR

(c) NDCG (d) MARC

Figure 3.7: Equestrians Dataset.

3.3.2.3 Measurement Approach.

To analyze the impact of demographic inference on fairness guarantees in my case studies, I used the following approach. First, I computed NDKL, ABR, NDCG and MARC on the original ranked lists that I crawled and the fair re-rankings produced by *DetConstSort* given ground-truth gender and perceived race/ethnicity data. I refer to these as "Baseline" and "Oracle," respectively, with the latter serving as my best-case fairness benchmark. Next, I reran *DetConstSort* after introducing demographic mis-classifications (while using the ground-truth gender and perceived race/ethnicity data to compute the fairness metrics NDKL and ABR). The breakdown of the inference algorithms used to produced these re-ranked lists are described in Table 3.2.

3.4 Results

Having discussed my methods and the structure of my experiments, I now present my results.

3.4.1 Simulations

I present the results of my simulations in Figure 3.4, from which I make five observations. *First*, both of the fairness metrics suffer in proportion to the error rate of demographic inference. As shown in Figure 3.4, NDKL falls (i.e., approaches representational fairness), and ABR rises (i.e., approaches attention parity) as the accuracy of demographic inference

increases. This result is intuitive: I cannot expect *DetConstSort* to perform at its best when the underlying demographic data is inaccurate.

Second, I observe that fair ranking performance varies with respect to my six ground-truth population distributions. *DetConstSort* was able to achieve low NDKL scores for relatively-uniform distributions, like A and B, regardless of inference accuracy, but struggled to achieve high ABR scores at lower accuracies. Conversely, *DetConstSort* achieves relatively high ABR scores but low NDKL scores for three-group distributions that had an overwhelming majority group, like C and D. Distribution E appears to be a worst-case scenario, combining a clear majority group with three other, much smaller minority groups. These findings demonstrate that there are complex interactions between the composition of the underlying population, accuracy of inference, and fairness guarantees.

Third, by comparing the baseline NDKL and ABR values for non-fairness aware rankings in Table 3.1 to the fairness-aware results in Figure 3.4, I observe that there are cases where the former has better fairness scores than the latter, depending on the accuracy of demographic inference. This finding shows that the use of a fair ranking algorithm is not categorically better than a non fairness-aware algorithm, depending on the accuracy of the underlying demographic data used for fair re-ranking.

Fourth, I observe no significant drop in the NDCG scores in Figure 3.4c for the fair ranked lists. This agrees with the findings of [74] and demonstrates that utility need not be sacrificed to produce fair rankings. The decrease in NDCG scores is greater for population distributions like D and F that have greater skew, in contrast to more uniform distributions like A and E.

Fifth, I observe in Figure 3.4d that the MARC values of the list decreases as the inference accuracy increases. Lower MARC values signal smaller departures in ranking from the original list, highlighting again the pitfalls of imperfect inference. The decrease in MARC is less evident for skewed distributions.

3.4.2 Chess Ranking Dataset

I present the results of my first case study using the chess players dataset in Figure 3.5 and Figure 3.8a–3.8c. The former figure focuses on aggregate metrics, while the latter presents per-group metrics.

From the NDKL and ABR scores in Figure 3.5a and Figure 3.5b, respectively, I observe

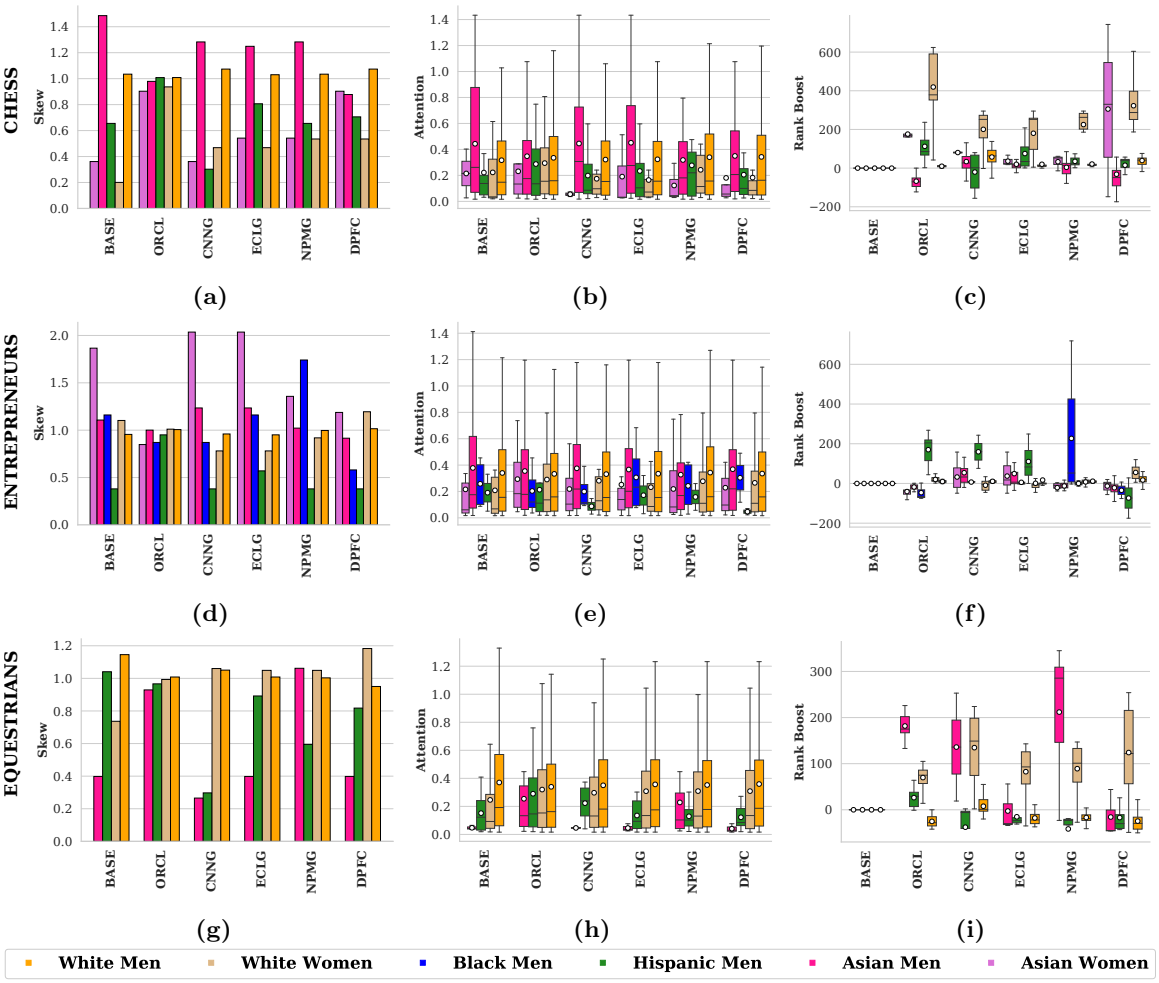


Figure 3.8: Scores for individual groups in the Chess, Entrepreneurs and Equestrians datasets.

that the "fair" re-rankings that used inferred demographic data as input offer much worse fairness than the oracle; CNNG even has worse fairness than the unfair baseline. In contrast, the NDCG scores in Figure 3.5c demonstrate that the fair re-rankings have no impact on the utility of the results, irrespective of whether fairness is actually achieved.

The MARC scores in Figure 3.5d reveal that *DetConstSort* had to move candidates farther to achieve the fair (oracle) ranking for the chess case than in my other case studies (Figure 3.6d and Figure 3.7d). Further, I observe that *DetConstSort* moved candidates shorter distances when it was supplied with erroneous inferences, which helps explain why it failed to achieve oracle-level fairness.

To delve deeper, I look into the performance of the individual inference algorithms

across the different sub-groups.[10] In the baseline unfair ranking, I observe that White and Asian males have high skew (Figure 3.8a), and that White males in particular receive a disproportionate amount of attention (Figure 3.8b). By design, when *DetConstSort* is given accurate demographic data it is able to produce a ranking with skew close to 1 for all groups, and attention is more uniform across the groups than in the baseline. Notably, *DetConstSort* had to dramatically increase the ranks of White women to achieve fairness (Figure 3.8c) because they are underrepresented in the population overall and especially within the top ranked players.

Conversely, I observe a variety of pathologies when *DetConstSort* is given inferred demographic data as input. Overall, I see that the advantaged groups (White and Asian men) retain their advantage, while non-White non-men rise or fall depending on the specific error characteristics of the inference algorithms. For example, the EthnCNN (CNNG) race inference algorithm mis-classified 15 White men and two Asian men as Hispanic men, and mis-classified three Hispanic men as White men. These errors have the pernicious consequence of causing actual Hispanic men to be under-represented in the ranking—even below the baseline representation (Figure 3.8a).

Another example: Asian men appeared frequently at high ranks in the baseline ranking and thus *DetConstSort* attempts to decrease their representation. However, the name-based inference algorithms incorrectly label high-scoring Asian men, e.g., CNNG incorrectly labels Asian *men* as Asian (5) or White *women* (3). This increases the skew of Asian men at the expense of other groups (Figure 3.8a). Further, by mislabeling high-scoring men as women, this causes *DetConstSort* to provide lower rank boosts to women than in the oracle re-ranking (Figure 3.8c), and thus women receive lower attention than in the baseline and oracle rankings (Figure 3.8b).

3.4.3 Crunchbase Entrepreneurs Dataset

Of my three case studies, the entrepreneurs dataset is the fairest, i.e., the baseline and oracle NDKL and ABR scores are closest (Figure 3.6a and Figure 3.6b). Only Asian women and Hispanic men are over- and under-represented in the baseline ranking relative to the

[10]The box plots follow standard statistical notation [196]. The box is bounded at the first and third quartile, and the central line represents the median. The upper whisker denotes the maximum point within the 3rd quartile + 1.5IQR (Inter Quartile Range), while the lower whisker denotes the minimum point within 1st quartile - 1.5IQR. The white dot denotes the mean.

underlying population (Figure 3.8d). As expected, when given accurate demographic data, *DetConstSort* is able to equalize skew, although attention remains somewhat low for Black and Hispanic men relative to the other groups (Figure 3.8e).

I observe several notable artifacts in Figure 3.8d. Skew for Asian women increases when using EthCNN and EthniColor (Figure 3.8d) due to them being mislabeled as White women or Asian men. Further, being mislabeled into relatively larger groups causes the Asian women to receive less attention in the re-ranked lists (Figure 3.8e). Likewise, the low skew and low attention for Hispanic men can be attributed to the fact that all five of the people predicted to be Hispanic men by the inference algorithms were actually White or Asian men. Black men are over-represented for Nameprism (NPMG) because it mislabeled three high-scoring Black men as Asian/White men. To compensate, *DetConstSort* then moved two low-scoring Black men to very high ranks (as evinced by the large rank boosts in Figure 3.8f).

3.4.4 Equestrian Ranking Dataset

Our equestrian athlete case study contrasts my chess case study: in both instances I observe a large fairness disparity between the baseline and oracle (Figure 3.7a and Figure 3.7b), yet in the equestrian case the inference algorithms result in re-rankings that are closer to the oracle in terms of NDKL and ABR scores, whereas in chess the inference-driven re-rankings are closer to the baseline.

However, just because the inference-driven re-rankings are relatively fair on average does not mean individual groups are not being stigmatized. As shown in Figure 3.8g and 3.8h, Asian men have low skew and low attention. The oracle mitigates this issue, but the inference algorithms routinely mislabel Asian men and White men, thus resulting in Asian men having low skew and attention in the "fair" re-rankings as well. Nameprism is the exception: it predicted six out of seven Asian men correctly. Conversely, I observe cases where White women were mislabeled as Asian men, causing *DetConstSort* to dramatically increase their rank (Figure 3.8i), leading to cases where White women become over-represented.

3.5 Discussion

In my study, I investigated the impact of inferred demographic data on algorithmic fairness in fair ranking algorithms. Through empirical experiments using real-world datasets and fair ranking metrics, I found that using inferred demographic data can harm vulnerable

groups and invalidate fairness guarantees. This highlights the need for caution when using such data in fairness-aware algorithms and for further research to identify effective strategies for mitigating its impact.

You study noise aware and demographic free algos later, so point at that chapter rather than being hypothetical. – Christo

I suggest the possibility of using uncertainty-aware algorithms as a solution, which I further discuss in Chapter 5. Another potential solution is to intentionally collect demographic data, but this must be done with care and consideration to avoid reifying oppressive structures and respecting individuals' autonomy. Overall, my study emphasizes the challenges of achieving algorithmic fairness in the presence of demographic inference and the need for further research to identify effective strategies for achieving fairness in machine learning algorithms.

3.6 Limitations

The primary limitation of my work concerns how I operationalize gender, race, and ethnicity. Gender is not binary, but the sources of data I rely on (ground-truth and inferred) only support binary labels. Similarly, my work is constrained by the race and ethnicity categories that are supported by available inference algorithms. These categories lack nuance and reify problematic political hierarchies. Future work in this space should broaden the space of gender, racial, and ethnic categories that are critically examined [88, 97], as well as examine other marginalized communities.

Chapter 4

Subverting Fair Image Search with Generative Adversarial Perturbations

4.1 Research Problem

Fairness and adversarial robustness are two important concerns in the field of machine learning. While fair machine learning techniques have gained popularity in recent years as a way to mitigate biases and ensure equitable outcomes, the vulnerability of fair machine learning models to adversarial attacks remains an open question. The field of adversarial machine learning has shown that seemingly accurate models can display surprising brittleness when presented with maliciously crafted inputs, known as adversarial examples, which can cause models to make incorrect predictions or behave in unexpected ways. The intersection of these two concerns has received relatively little attention, and it is unclear whether fair machine learning techniques are vulnerable to adversarial attacks that could subvert their fairness guarantees.

In this chapter, I investigate the research problem of whether external adversaries can make fair ranking models behave unfairly without having access to the model or training data. To address this problem, I present a case study in which I develop and attack a fairness-aware image search engine using images that have been maliciously modified with adversarial perturbations. Through extensive experiments, I demonstrate that my attacks can successfully subvert fairness guarantees in the context of fair image retrieval, even under a highly restricted threat model. The results of this chapter highlight the potential vulnerability

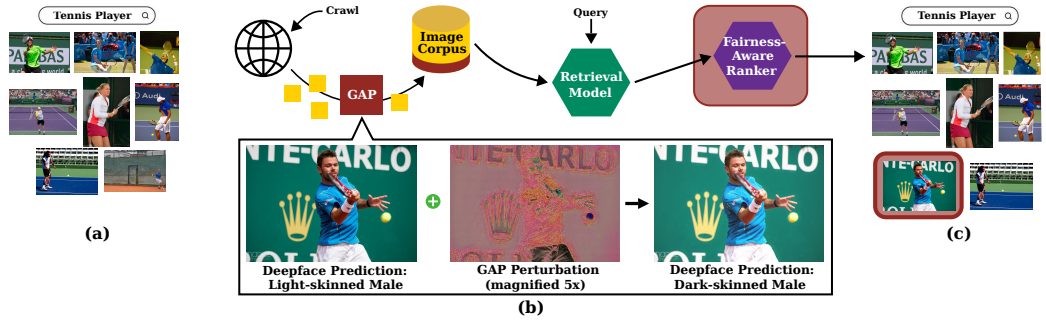


Figure 4.1: A diagram showing my attack approach. **(a)** shows example search results from an image search engine for the query "tennis player". This search engine attempts to provide demographically-fair results, and at this point no images in the corpus have been adversarially perturbed. **(b)** as this search engine crawls and indexes new images from the web, it collects images that have been adversarially perturbed using a GAP model. I show a real example of one image before and after applying the generated perturbation, which causes the Deepface model [201] to misclassify this person's skin tone. **(c)** in response to a future query for "tennis player", the retrieval model will identify relevant images, some of which are perturbed. The fairness-aware ranker (the target of the attack, highlighted in red) mistakenly elevates the rank of an image containing a light-skinned male (also highlighted in red) because it misclassifies them as dark-skinned due to the perturbations.

of fair machine learning techniques to adversarial attacks and the need for further research on the robustness of these techniques. By shedding light on these vulnerabilities, I aim to motivate the development of mitigation strategies to make fair machine learning models more robust against adversarial attacks.

4.2 Methodology

I now present the plan for my study. First, I introduce my application context and the threat model under which an attacker will attempt to compromise the application. Second, I discuss the IR models and algorithms underlying my fairness-aware image search engine. Third, I discuss my strategy for attacking this search engine using GAPs.

4.2.1 Context and Threat Model

In this study, I consider the security of a fairness-aware image search engine. This search engine indexes images from around the web (either automatically via a crawler or from user-provided submissions) and provides a free-text interface to query the image database. Examples of image search engines include Google Image Search, iStock and Getty Images, and Giphy. In my case, the image search engine attempts to produce results that are both

relevant to a given query and fair, according to some fairness objective. One example fairness objective is demographic representativeness, i.e., for search results that contain images of people.

I consider a malicious image curator (e.g., Imgur, 4chan, or similar) with a large database of *perturbed* images that are eventually scraped or uploaded into the victim image search engine's index.[1] My adversarial image curator's goal is to perturb the images in their database to subvert the fairness guarantees of the downstream retrieval system. I assume that the adversary does not have any knowledge of the internals of the ranking system (e.g., what retrieval model is used, other images in the index, or which fairness algorithm is used).

This threat model constitutes a strict, but realistic, limitation on my adversary. Notice that this threat model would also apply if the image search engine was compromised, giving the adversary access to underlying models and the entire dataset of images. I consider both adversaries in my experiments. I also note that, if the adversary only seeks to target a small set of queries, they need only control a fraction of the images matching each query, rather than a fraction of the entire image database. This is useful for the adversary in the case that not all queries are equally sensitive.

4.2.2 Building an Image Search Engine

I now turn my attention to building a realistic image search engine that will serve as the victim for my attacks.

4.2.2.1 Image Retrieval from Text Queries

The first choice I make for this study is to select an image retrieval model. There are several frameworks for image retrieval in the literature, starting from tag-based matching [123] to state-of-the-art vision-language transformers [126, 134]. For the purpose of this paper, I used a MultiModal Transformer (MMT) [73] based text-image retrieval model. This model consists of two components: a fast (although somewhat lower quality) retrieval step that identifies a large set of relevant images, followed by a re-ranking step that selects the best images from the retrieved set. Concretely, the user provides a string q that queries into a database D of n images. For the retrieval step, the query string is encoded with an embedding

[1] An adversarial image curator is also the threat model assumed for clean label poisoning attacks [185, 204]. This adversarial image curator may perturb copies of images taken from the web or original images that they author. This setup is also used by [187] as a defensive method against unauthorized models.

function f_q to produce an embedding v_q, and all images in D have pre-computed embeddings from an embedding function f_I. The cosine distance between v_q and all embeddings of D are computed to collect some large set D_q of size $n' \leq n$ plausible image matches. These images are then ranked according to a joint model f_j that takes both the query and an image as input, returning scores $\{s_i\}_{i=1}^{n'}$ indicating how well each image $D_q[i]$ matches the query. These scores are used to produce the final ranking.

Note that the MMT model is not designed to be "fair" in any normative sense. To achieve fairness, results from the model must be re-ranked, which I describe in the next section. Thus, the MMT model is not the target of my attacks, since it is not responsible for implementing any fairness objectives.

4.2.2.2 Fairness-aware Re-ranking

The second choice I make for this study is selecting an algorithm that takes the output of the image retrieval model as input and produces a fair re-ranking of the items. In fairness-aware re-ranking, a ranking function $f_r(D, q)$ is post-processed to achieve fairness according to some subgroup labels on the dataset $D = \{s_i, x_i\}_{i=1}^{n}$, where s_i denotes the score of the i^{th} item (the heuristic score according to which the list is sorted) and x_i denotes the item to be ranked.

The re-ranking algorithm I adopt is Fair Maximal Marginal Relevance (FMMR) [108], which was developed and used at Shopify for representative ranking of images. FMMR builds on the Maximal Marginal Relevance [39] technique in IR that seeks to maximize the information in a ranked list by choosing the next retrieved item in the list to be as dissimilar to the current items present in the list as possible. MMR introduces a hyperparameter that allows the operator to choose the trade-off between similarity and relevance.

FMMR modifies the "similarity" heuristic from MMR to encode for similarity in terms of demographics, with the idea being that the next relevant item chosen to be placed in the re-ranked list will be as demographically different from the existing images as possible. Similarity is calculated using image embeddings, for which I examine three models: Faster R-CNN [168], InceptionV3 [199], and ResNet18 [91]. I fix the trade-off parameter λ at 0.14 as that is the value used by [108] in their FMMR paper.

It is notable that FMMR does not require demographic labels of people in images to perform fair re-ranking, since it uses a heuristic that only relies on embeddings. Indeed,

FMMR comes from a class of fair ranking algorithms that all use the inherent latent representations of the objects for their re-ranking strategy [108, 171]. That said, since FMMR attempts to maximize the distance from the centroids of the embeddings of different demographic groups, it can be thought of as performing indirect demographic inference on individuals in images.

Additionally, I also evaluated my attacks against a second fairness-aware re-ranking algorithm, DetConstSort [74], developed by and deployed at LinkedIn in their talent search system. Unlike FMMR, DetConstSort explicitly requires demographic labels for the items it is trying to fairly re-rank. However, prior work [77] shows that DetConstSort has significant limitations when demographic inference is used rather than ground-truth demographic labels, making it unfair even without perturbed images. As a result, evaluating an attack against DetConstSort is not meaningful, and I defer my discussion of DetConstSort to 4.5.4.

4.2.3 Attack Construction

Having described my search engine, I are ready to turn my attention to my attack. First, I introduce the demographic inference models (Deepface [201] and FairFace [110]) that I use to train my attack. Next, I describe how I generate adversarial perturbations from a demographic inference model, modifying images in a way that is imperceptible to human eyes, yet significant enough to fool the fair re-ranking algorithm of my search engine.

4.2.3.1 Demographic Inference Algorithms

For large-scale datasets such as images scraped from the web, demographic meta-data for people in the images is (1) not readily available and (2) prohibitively expensive to collect through manual annotation [10, 31]. Pipelines using demographic inference are commonly used in practice when demographic labels are not available. For example, the Bayesian Improved Surname Geocoding (BISG) tool is used to measure fairness violations in lending decisions [3, 37], and it relies on inferred demographic information. This makes attacks on demographic inference models a natural candidate for adversely affecting ranking fairness.

I consider two image demographic inference models to train my attacks:

1. Deepface [201] is a face recognition model for gender and race inference developed by Facebook. I use its public wrapper [184], which includes models fine tuned on roughly 22,000 samples for race and gender classification.

2. FairFace [110] is a model designed for race and gender inference, trained on a diverse set of 108,000 images.

Since both of these models infer race/ethnicity, I used a mapping to infer skin tone, since I could not find commercially available algorithms to infer skin tones from human images.[2] I also use these models to infer demographics as input to the DetConstSort algorithm, matching the pipeline of [77], which I discuss in 4.5.4.

4.2.3.2 Subpopulation Generative Adversarial Perturbations

Recall my adversarial image curator's goal: to produce a database of malicious images that, when indexed by my image search engine, undermine its purported fairness guarantees. Concretely, this means fooling the fair re-ranker such that it believes a given set of search results is fair across two or more subgroups, when in fact the results are unfair because some subgroups are under- or over-represented. Additionally, these malicious images must (1) retain their relevance to a given query and (2) not be perceived as "manipulated" to human users of the search engine.

Prior work (see 2.3.3) has demonstrated that neural image classification models can be fooled by adding *adversarial perturbations* to images. At a high-level, the adversary's goal is to train a model that can add noise to images such that specific latent characteristics of the images are altered. In my case, these altered characteristics should impact the image embeddings calculated by the image embedding model (e.g., InceptionV3) that FMMR relies upon to do fair re-ranking.

Running an adversarial perturbation algorithm on each of the images in the adversary's database would be prohibitive, as these algorithms involve computationally expensive optimization algorithms that are not practical at the scale of an entire database. I avoid this limitation by training a Generative Adversarial Perturbation (GAP) model [163]. A GAP model f_{GAP} takes a clean image as input and returns a perturbed image that is misclassified by some target model f_{targ}. This replaces the per-image optimization problem with a much less expensive forward pass of f_{GAP}. Training the GAP is a one time expense for the adversary, amortized over the large number of image perturbations done later. Universal Adversarial Perturbations (UAPs) [146] are another approach to amortizing runtime, but

[2]The mapping I used is: White, East Asian, Middle Eastern → Light, and Black, South Asian, Hispanic → Dark. I acknowledge that this is a crude mapping, but it enabled me to train a successful attack.

require all images to be the same dimensions—an unrealistic assumption for real-world image databases.[3]

Having motivated the choice of a GAP model for my attack, I now consider the problem of impacting fairness by attacking the fair re-ranking algorithm used by a victim search engine. I choose to design a GAP to target a demographic inference model f_{DI}.[4] This will produce perturbations that, to a deep image model, make an image of a person from one demographic group appear to be from a different demographic group. This attack would heavily impact a demographic-aware re-ranking algorithm such as DetConstSort [74] (see 4.5.4) if it used an accurate demographic inference algorithm to produce annotations.

Although FMMR does not use annotations, I show in 5.4 that my attack is still successful at compromising FMMR's fairness guarantees. My attack can be seen as an application of the *transferability property* of adversarial examples. Additionally, training my GAP against a demographic inference model causes my attack to be independent of the ranking algorithm and image corpus used by the victim search engine, both of which are strong adversarial assumptions.

In designing my GAP to compromise fairness, I first note that an attack that simply forces a f_{DI} to make arbitrarily many errors may not impact fairness. For example, suppose the image database contained two subpopulations, the advantaged class A and the disadvantaged class B. Suppose the attack causes f_{DI} to misclassify all members of B as A and all members of A as B. This is the best possible result of an attack on the demographic inference algorithm, but results in no changes to a fair ranking algorithm—it will simply consider A to be the disadvantaged class, and thus produce the same ranking! For this reason, my adversary must incorporate subpopulations into the attack. To do so, I propose the Class-Targeted Generative Adversarial Perturbation (CGAP):

Definition 1 (CGAP) *I consider a loss function ℓ, target model f_{targ}, distribution $\mathcal{D}$ over inputs x and outputs y. The adversary provides a source class y_s and target class y_t. Then the CGAP model f_{CGAP} is a model that takes as input an image x and returns an image x',*

[3] A UAP can also be seen as a GAP, where $f_{\text{GAP}}(x) = x + \delta$ for a fixed δ. Therefore, I expect that a GAP will perform strictly better than a UAP.

[4] Recall that, per my threat model in 4.2.1, the attacker does not know what fair re-ranking algorithm is used by the victim and thus cannot train against it directly.

Search Queries	Attack Training	Embedding	Training Objective	Attack Probability	Top k
"Tennis Player" "Person eating pizza" "Person at Table"	Deepface FairFace	F-RCNN InceptionV3 ResNet	Any→Light Men Light Men→Any Dark Men→Light Men Light Men→Dark Men	0.2, 0.5, 0.7, 1.0	10, 15, 20..., 45, 50

Table 4.1: Variables and hyperparameters I used for evaluating my attack.

minimizing the following loss functions:

$$\ell^{s}_{CGAP}(\mathcal{D}) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f_{CGAP}(x), y_t; f_{targ})|y = y_s],$$
$$\ell^{r}_{CGAP}(\mathcal{D}) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(f_{CGAP}(x), y; f_{targ})|y \neq y_s].$$

That is, the CGAP should force the demographic inference model to misclassify samples of class y_s to class y_t, while maintaining its performance for samples not from class y_s.

I also consider two extensions of this definition. First, I permit the adversary to target multiple classes at once. In the extreme, an adversary may want all samples to be classified to the same class (this approach is proposed by [163]). For a demographic inference algorithm, all samples having the same demographic label will cause the fair re-ranking system to have similar performance to an unfair ranking system, as all points will appear to fall into the same subpopulation. The second extension is the untargeted attack, where the CGAP simply increases loss for points from class y_s, inducing arbitrary misclassifications. Simultaneously making both relaxations recovers the original untargeted GAP approach. I experiment with both relaxations independently, as well as multiple instantiations of CGAP as defined above.

4.3 Experiments

In this section I introduce the dataset I used for my evaluation, describe the setup for my experiments, and define the metrics I use to evaluate my attacks.

4.3.1 Dataset, Annotation, and Preprocessing

I use Microsoft's Common Objects in Context (MS-COCO) [128] as my retrieval dataset, since it contains a variety of images with variable dimensions and depths. This closely mimics what a real-world image search dataset might contain.

To specifically measure for demographic bias, I filter the dataset, keeping only images that contain people. I also need the images to have demographic annotations for fair ranking, so I use an annotated subset of the COCO 2014 dataset, constructed by [220]. Similar to prior work [42, 77], Zhao et al. crowdsource skin color (on the Fitzpatrick Skin Type Scale, which the authors simplified to Light and Dark) and binary perceived gender expression for 15,762 images. For the purposes of my experiments I only considered the 8,692 images that contain one person. After filtering, my final dataset consisted of 5,216 Light Men, 2,536 Light Women, 714 Dark Men, and 226 Dark Women.

4.3.2 Experimental Setup

As a starting point for my experiments, I need to collect ranked lists from my baseline, **unfair** retrieval system, as described in 4.2.2.1. To do so, I run three different search queries on the retrieval system: "Tennis Player", "Person eating Pizza", and "Person at table". I chose these queries because they all reference a human being, are ethnicity and gender neutral, and are well-supported in the COCO dataset (I picked popular object tags, see 4.3.2.1). I set the upper bound in the baseline retrieval system to be 200 images. The three queries return 131, 75, and 124 images, respectively, along with their relevance scores.

I show the distribution of the relevance scores and the skin color/gender distributions of the images within the top 40 search results for each query in Figure 4.2. As also shown by [220], Light Men comprise the overwhelming majority in all three lists, and they also have high relevance scores across the board, meaning that the retrieval system places Light Men near the top of the search results. I call these lists the *baseline* lists.

I also need to produce fair versions of the baseline lists. To do so, I pass the baseline lists for each of my three queries through *FMMR* with the three embedding algorithms, without any adversarial perturbations. I refer to the nine lists obtained via the fair re-ranker (three queries times three image embedding models) as the *oracle* lists.

To train my adversarial attacks, I first remove the 330 images in my *oracle* lists from the original dataset, leaving 8,362 images. These 8,362 images were then split randomly into training and testing sets in an 8:2 ratio to train CGAP models for all possible combinations of training objectives and demographic inference algorithms f_{DI} (described in detail below).

[5] Dark-skinned women do appear in the search results for the query "female tennis player". This seems to reflect stereotypical bias [71] within the learned-word representations in the MMT model.

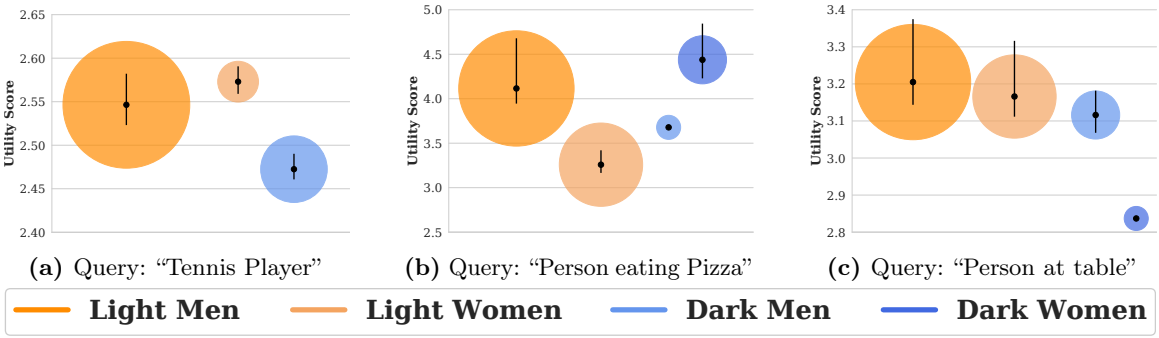


Figure 4.2: Utility/Relevance score and group size distribution within the top 40 baseline search results for three queries. The black dots represent the average utility score for that group, while the circle size represents the group size. No dark-skinned women appear in the top 40 baseline results for the "tennis player" query.[5]

I ran my experiments on PyTorch with a CUDA backend on two NVIDIA RTX-A6000 GPUs, and trained CGAP models for 10 epochs each, with the L_∞ norm[6] bound set to 10.

I describe my different training and inference combinations below. Table 4.1 shows a summary of the different settings involved during the training and testing of my CGAP attacks.

4.3.2.1 Choice of Queries

To facilitate my experiments, I chose to select search query terms that would provide a sizeable list of images. To do so, I looked at the list of terms in the COCO image captions (excluding English stop words and words related to ethnicity or gender). The following table shows some top terms. From this information, I composed my three queries given that "sitting", "tennis", "table", "person", "pizza", etc. were among the most popular terms.

4.3.2.2 Embedding Algorithm

As I discuss in 4.2.2.2, FMMR requires image embeddings. The authors of the original paper used a pretrained InceptionV3 model, which I also adopt. Additionally, I test the performance of FMMR using embeddings generated by pretrained Faster R-CNN and ResNet models. These models are trained for standard image classification tasks and have no inherent concept of demographic groups.

[6] L_∞ is the absolute distance in pixel space any one pixel is changed, i.e. a pixel can at most change by a value of 10 in each color channel.

Term	Count
sitting	55084
standing	44121
people	42133
holding	29055
large	25305
person	25123
street	21609
table	20775
small	20661
tennis	19718
riding	18809
train	18287
young	17767
red	17522
baseball	15362
pizza	11163

Table 4.2: The most common (gender or race unrelated) caption terms in the evaluation dataset.

4.3.2.3 Attack Training Algorithm

As detailed in 4.2.3, I train CGAP models to induce adversary-selected misclassifications in two target demographic inference models, denoted as f_{DI}: Deepface [201] and FairFace [184]. These models are trained for demographic inference, and so do not overlap in training objective with the image embedding models for FMMR. The only similarity in architecture between the demographic inference and FMMR embedding models is that FairFace uses a ResNet architecture.

4.3.2.4 Training Objectives

As discussed in 4.2.3.2, I select certain subpopulations to be systematically misclassified by the two f_{DI} described above. The fmy CGAPs I train induce misclassifications with the following source-target pairs: Any→Light Men, where every subgroup was perturbed to be predicted as Light Men; Light Men→Any, where only Light Men are arbitrarily misclassified; Dark Men→Light Men, where only Dark Men are misclassified as Light Men; and Light Men→Dark Men, where only Light Men are misclassified as Dark Men.

4.3.2.5 Attack Probability *pr*

It is a strong assumption that an adversary can perturb the entire image database of a victim search engine. This is only possible if the search engine itself is malicious or it is utterly compromised. Instead, I measure the effect of my attack when the attacker may perturb $pr = 20\%$, 50%, 70% and 100% of the image database relevant to each query. If a small number of queries are targeted, only few images are required to run the attack.

4.3.2.6 Top *k*

Ranking is very sensitive to position bias [77, 178], so I measure with different lengths k of the top list, ranging from top 10 to top 50, to gauge my attack's impact on the fair ranking algorithms as final list sizes vary.

4.3.3 Evaluation Metrics

To evaluate the impact of my attacks, I use three metrics that aim to measure (1) representation bias, (2) attention or exposure bias, and (3) loss in ranking utility due to re-ranking. Additionally, I introduce a summarizing meta-metric that enables me to clearly present the impact of my attacks with respect to each metric.

4.3.3.1 Skew

The metric I use to measure the bias in representation is called Skew [74, 77]. For a ranked list τ, the Skew for attribute value a_i at position k is defined as:

$$\text{Skew}_{a_i}@k(\tau) = \frac{p_{\tau^k,a_i}}{p_{q,a_i}}. \tag{4.1}$$

p_{τ^k,a_i} represents the fraction of members having the attribute a_i among the top k items in τ, and p_{q,a_i} represents the fraction of members from subgroup a_i in the overall population q. In an ideal, fair representation, the skew value for all subgroups is equal to 1, indicating that their representation among the top k items exactly matches their proportion in the overall population.

4.3.3.2 Attention

Even if all subgroups were fairly represented in the top k ranked items of a list, the relative position of the ranked items adds another dimension of bias—unequal exposure. Previous studies [151, 156] have shown that people's attention rapidly decreases as they scan down a list, with more attention given to the higher ranking items, ultimately dropping to zero attention.

In this study, I model attention decay using the geometric distribution as done in prior work by [178]. I compute attention at the k^{th} rank as:

$$\text{Attention}_p@k(\tau) = 100 \times (1-p)^{k-1} \times (p) \tag{4.2}$$

where p is the fraction of total attention given to the top search result. The choice of p is application specific—for this paper I fixed p to be 0.36, based on a study [86] that reported that the top result on Google Search receives 36.4% of the total clicks. I then calculate the average attention per subgroup:

$$\text{Average attention}_{a_i,\tau} = \frac{1}{|a_i|}\sum_{k=1}^{|\tau|}\text{Att(k) where } a_k^\tau = a_i. \tag{4.3}$$

Ideally, in a perfectly fair ranked list, all subgroups should receive equal average attention.

4.3.3.3 Normalized Discounted Cumulative Gain.

NDCG is a widely used measure in IR to evaluate the quality of search rankings [100]. It is defined as

$$\text{NDCG}(\tau) = \frac{1}{Z}\sum_{i=1}^{|\tau|}\frac{s_i^\tau}{log_2(i+1)} \tag{4.4}$$

where s_i^τ is the utility score from the MMT retrieval model of the i^{th} element in the ranked list τ and $Z = \sum_{i=1}^{|\tau|}\frac{1}{log_2(i+1)}$. NDCG scores range from 0 to 1, with the latter capturing ideal search results.

4.3.3.4 Summarizing Metric

For the purpose of quantifying how much unfair advantage my attacks confer on members of the majority class relative to all other classes, I define a new meta-metric called Attack

Effectiveness η. For a given metric $m \in \{$ Skew, Attention $\}$ and a subgroup g, it is defined as:

$$\eta(m, g) = \text{ \% change in } m \text{ for subgroup } g - \\ \text{minimum \% change in } m \text{ over other subgroups.} \tag{4.5}$$

I chose this formulation of η for two reasons. First, comparing percentage changes makes the metric scale invariant, which is useful since group sizes vary. Second, comparing to the group that gets the minimum boost ensures that the metric presents the widest fairness disparity, regardless of the total number of groups.

For the purposes of this paper, I set g as Light Men, because they are socially and historically the most advantaged group, and a large η for Light Men indicates that the attack causes their ranking to be unfairly boosted relative to the least privileged group. To make sure that the fairness impacts I observe are due to the effectiveness of my attack on the re-ranking algorithms only, the η values and the % change in NDCG are all measured against the *oracle* (i.e., fairly re-ranked) lists. Because I compare against the oracle list, all results with attack probability $pr = 0$ will have $\eta = 0$.

4.4 Results

In this section, I evaluate the impact of my attacks on the fairness guarantees of FMMR. For each set of results I examine how attack effectiveness varies for one particular variable (e.g., top k, image embedding model, etc.) as the attack probability pr (i.e., the fraction of images under adversarial control) varies. When focusing on a particular variable, I present results that are averaged across all other variables and all three of my queries.

4.4.1 Top k and pr

I begin by evaluating the impact of my attacks as I vary the length of the top list k and the fraction of images in the query list under adversarial control pr, plotted in Figure 4.3.

Varying pr has the expected effect: as the adversary has more control over the image database, attacks become more effective, i.e., η for skew and attention increase. When the adversary is able to control 100% of images in the query list, attacks are especially

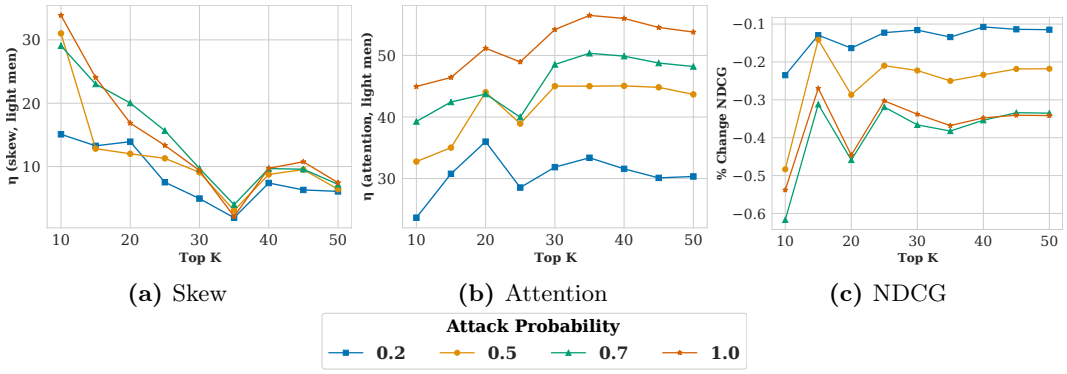


Figure 4.3: Attack effectiveness as a function of attack probability pr and list length k. Higher η is a more effective attack, i.e., the search results are more favorable to light-skinned men. Unfairness increases as pr increases, yet there is almost no impact on ranking quality (NDCG). As k increases skew is less impacted but attention is impacted somewhat more.

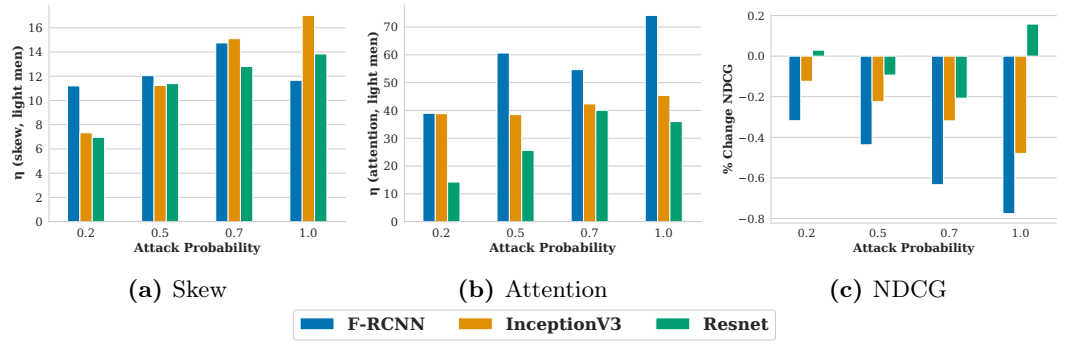
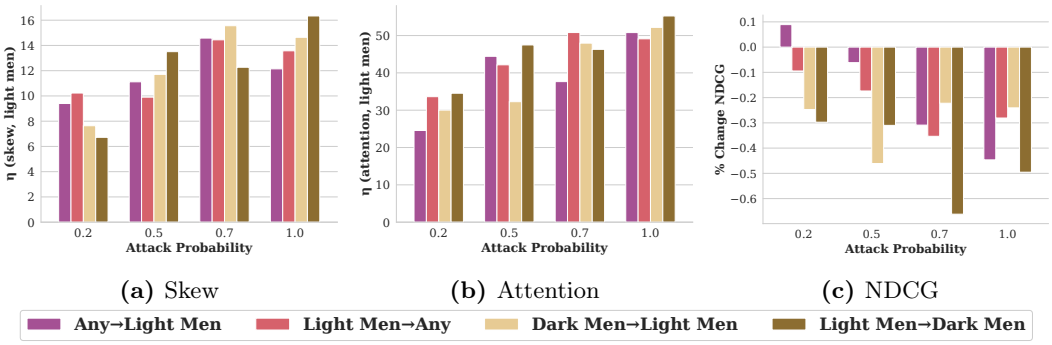


Figure 4.4: Attack effectiveness is stable when the model used for the FMMR embedding is changed. ResNet embeddings are slightly more robust to attack and F-RCNN are slightly less robust. Interestingly, the ResNet's robustness is in spite of it having the most similar model architecture to FairFace.

strong—increasing attention unfairness by over 50% for some values of k. Even with only 20% control, the adversary can increase attention unfairness by ∼30%. Recall that pr measures the fraction of each query list that is compromised, so as few as 35 images can be compromised at $pr = 0.5$ (for the "Person eating Pizza" query).

Varying k also impacts ranking fairness. As k increases, attention unfairness increases modestly and skew unfairness decreases. That skew unfairness decreases with k indicates that the composition of items in the search results becomes fairer as the length of the list grows. However, my attack is able to cause FMMR to reorder the list such the top-most items remain unfair regardless of k, which is why attention unfairness exhibits less dependency on k.



(a) Skew **(b)** Attention **(c)** NDCG

4.5 Limitations

The primary limitation of my work concerns how I operationalize gender, race, and ethnicity. Gender is not binary, but the sources of data I rely on (ground-truth and inferred) only support binary labels. Similarly, my work is constrained by the race and ethnicity categories that are supported by available inference algorithms. These categories lack nuance and reify problematic political hierarchies. Future work in this space should broaden the space of gender, racial, and ethnic categories that are critically examined [88, 97], as well as examine other marginalized communities.

Figure 4.5: Attack effectiveness is relatively stable when the GAP training objective is changed.

Lastly, I observe that my attacks are stealthy. Regardless of k or pr, NDCG never changes more than 0.7%, meaning that my attack had effectively zero impact on search result relevance.

4.5.1 Choice of Training Objective

I evaluate my attack's impact on fairness with fmy CGAP models: one that misclassifies Dark Men as Light Men, one for misclassifying Light Men as Dark Men, and relaxed CGAP models that misclassify all people as Light Men and all Light Men as other groups. I show these attacks' effectiveness in Figure 4.5.

Each of these attacks performs similarly well at harming fairness in terms of skew and attention, and remaining stealthy in terms of NDCG. One surprising observation is that misclassifying Dark Men as Light Men performs similarly to the exact opposite attack: in both cases, Light Men end up with an significant, unfair advantage. I explain this seeming contradiction with an example in Figure 4.6. In essence, using a GAP to misclassify people from a minority group into the majority group reduces the minority group's overall share

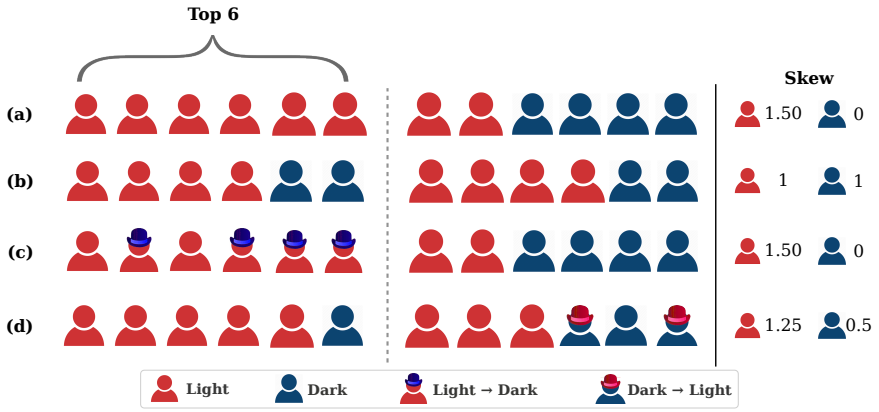


Figure 4.6: An example showing how incorrect group allocation in any direction always harms the minority group members in fair ranking. **(a)** shows a *baseline* unfair list, with all people sorted by relevance to the query and no dark people in the top 6. **(b)** shows the fair ranking produced by FMMR, with the same proportion of light and dark people in the top 6 as the overall population. In **(c)**, light people's images are perturbed using a GAP so that half of them are grouped with dark people. FMMR moves the most relevant dark people into the top 6 to make the list fair, but in this case the most relevant "dark" people are really light skinned. In **(d)**, half of the dark people are perturbed using a GAP to be grouped as light people. To FMMR, this appears to reduce the overall population of dark people, so it only needs to move one dark person into the top 6 to make the list proportionally fair. Note that if all light people were grouped as dark or all dark people were grouped as light, the ranking would remain the unfair baseline shown in **(a)**.

of the population. Since group fairness in this case is based on the overall population distribution, this causes FMMR to rerank fewer minority group members into the top of the search results.

Based on the results in Figure 4.5, it appears that there is no way to advantage a minority group with my attacks.

4.5.2 Choice of Attack Training Algorithm

I measure my attacks' effectiveness when the GAP models are trained on Deepface and FairFace demographic inference models. I observe that attack effectiveness is largely independent of the choice of inference model, and all attacks remain stealthy. I defer a plot of the results to the supplementary material, in Figure 4.8.

4.5.3 Choice of Query

Lastly, I examine the effectiveness of my attacks against three different queries and plot the results in Figure 4.7. I observe that all attacks were successful, but that effectiveness

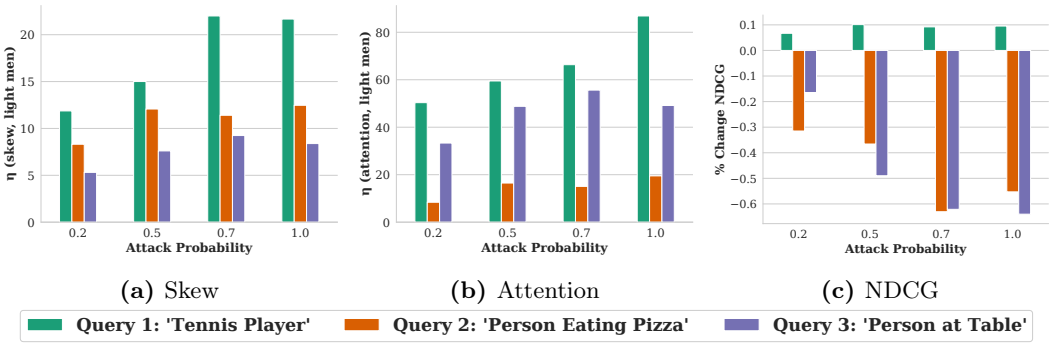


Figure 4.7: Attacks are effective against all three of my queries, but the effectiveness varies in relation to the underlying population and utility score distributions (see Figure 4.2).

varies by query. The differences in attack effectiveness are explained by the underlying distributions of population and utility scores (see Figure 4.2). The "tennis" results exhibit the most unfairness post-attack because they were most unfair to begin with, i.e., the difference in utility scores between Light and Dark skinned people was greatest in the "tennis" results as compared to the other queries. In contrast, the "pizza" results exhibit the most robustness to attack in terms of attention because these were the only results among the queries where minority people had higher utility scores than majority people in the baseline results (Figure 4.2).

4.5.4 Additional results: comparison between DetConstSort and FMMR

In this section I compare the performance of two fair re-rankers in the presence of my GAP attack. I have already described the details of the first algorithm, FMMR, in 4.2.2.2.

4.5.4.1 DetConstSort

The second algorithm, DetConstSort [74], was developed by and is currently deployed at LinkedIn in their talent search system. Unlike FMMR, DetConstSort requires access to the demographic labels of the items it is trying to fairly re-rank. DetConstSort rearranges a given list of items, such that for any particular rank k and for any attribute a_j, the attribute is present at least $\lfloor p_{a_j}.k \rfloor$ times in the ranked list, where p_{a_j} is the proportion of items in the list that have the attribute a_j. DetConstSort also re-sorts the items within the relevance criteria so that items with better utility scores are placed higher in the ranked list as much as

possible, while maintaining the desired attribute ratio. It thus aims to solve a deterministic interval constrained sorting problem.

If ground-truth demographic labels are unavailable, DetConstSort may instead utilize labels sourced from a demographic inference model. Recent work, however, has shown that DetConstSort is sensitive to errors in demographic labels, with one example of such errors being inaccurate inferences [77].

4.5.4.2 Evaluation Results

I present the results of my GAP attacks against my search engine when it uses DetConstSort and FMMR as the fair re-ranker, respectively, in Figure 4.9. As in 5.4, these results are averaged across three queries, multiple values of k, etc.

For DetConstSort, the skew and attention metrics are not impacted by my attack. This can be clearly seen by comparing the η values when $pr = 0$ (i.e., there are no perturbed images) to other values of pr: for DetConstSort, η for skew and attention starts high (unfair) when $pr = 0$, and does not change as pr increases. The correct interpretation of these results is **not** that DetConstSort is resilient to my attack. Rather, the correct interpretation is that DetConstSort starts off unfair due to the use of inaccurate, inferred demographic data [77], and my attack is unable to make the unfairness worse.

Thus, I find that a prerequisite for evaluating the success of my attacks on DetConstSort is an accurate demographic inference model. Developing such models is still an active area of research, and is out-of-scope for my work. Should a more accurate demographic inference model be designed in the future, however, it must be designed with adversarial robustness in mind to prevent my attacks.

4.6 Limitations

My study has a number of limitations. First, my analysis is limited to two discrete racial and two discrete gender categories. Although my CGAP attack could be tailored to select any group, it is unclear how well my attack would perform in situations with > 4 discrete protected groups, groups with continuous attributes, people with multiple or partial group memberships, or with population distributions that varied significantly from my dataset. Second, while my dataset is sufficiently large to demonstrate my attack, it is smaller than the databases that real-world image search engines retrieve from. Third, my proof-of-concept

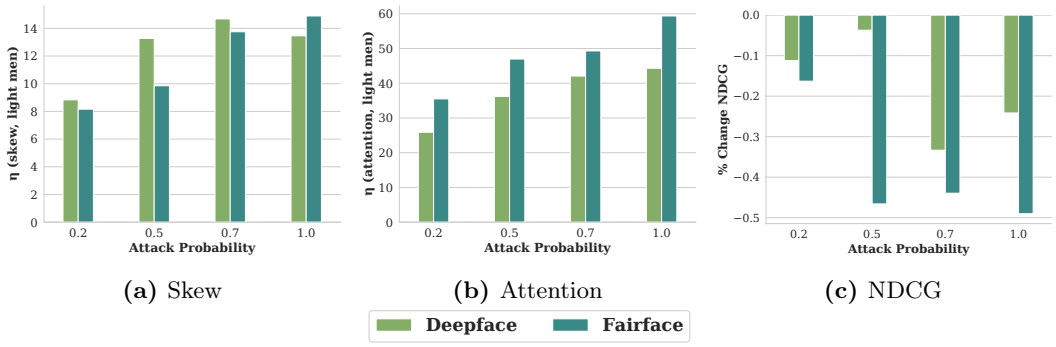


Figure 4.8: GAP models trained on different demographic inference algorithms offer similar attack effectiveness.

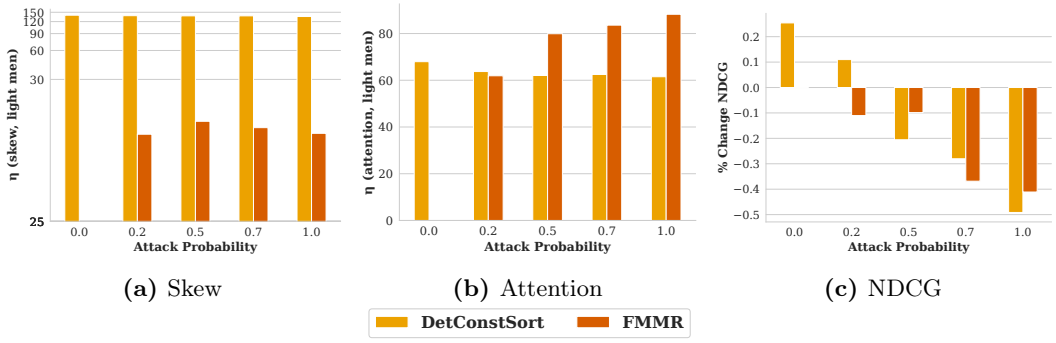


Figure 4.9: DetConstSort has poor performance even without an attack, making my results uninteresting.

was tuned to attack FairFace and Deepface. It is unclear how well my CGAP attack would generalize to other models or real-world deployed systems. Fourth, as I observe in Figure 4.5, my attack is only successful at generating unfairness in favor of already-advantaged groups. While this is a limitation, it in no way diminishes the potential real-world harm my attack could inflict on marginalized populations. Finally, as shown in Figure 4.7, my attack's effectiveness varies by query. In real world scenarios, an attacker could mitigate this to some extent by devoting more of their resources towards perturbing images that are relevant to high-value queries. It is unclear how much the attackers' effort would need to vary in practice given that I make no attempt to attack deployed search engines.

4.7 Discussion

In this study, I investigate the vulnerability of fair ranking algorithms to adversarial machine learning attacks. I use a case study of fairness-aware text-to-image retrieval to demonstrate the effectiveness of my novel adversarial attack. My results show that my attack is highly successful at subverting the fairness algorithm of the search engine across an extensive set of attack variations, while having almost no impact on search result relevance.

My analysis raises concerns regarding the use of demographic inference in fair algorithms and highlights the need for more robust fair machine learning interventions that are resilient to adversarial attacks. Achieving demographic fairness requires either high-quality demographic data or robustness against uncertainty, and in the absence of either safeguards, allowing adversaries to influence demographic metadata is the underlying flaw that enables my attack to succeed. My research also has limitations, including the dataset size and the number of discrete protected groups analyzed. However, I believe that my work will raise awareness and spur further research into vulnerabilities in fair algorithms, and I call for the development of more robust fair machine learning interventions. I also emphasize the importance of high-quality, consensual demographic data to improve ethical norms and defend against adversarial machine learning attacks, and further investigation into adversarially robust and uncertainty aware fair ranking models.

Chapter 5

When Fair Classification Meets Noisy Protected Attributes

5.1 Research Problem

Fairness is a critical component of trustworthy AI, and the adoption of fair classifiers in real-world scenarios is a formidable challenge. One of the primary challenges is the accurate collection and use of demographic data. Just as I discussed earlier for fair ranking, many classical fair classifiers also assume that protected attributes are available at training and testing time, and that this data is accurate. However, demographic data may be noisy for various reasons, including imprecision in human-generated labels, reliance on imperfect demographic-inference algorithms, or intentionally poisoning demographic data. To deal with these issues, researchers have proposed noise-tolerant fair classifiers that incorporate the error rate of demographic attributes in the fair classifier optimization process itself.

In some cases, demographic data may not be available at all, which violates the assumptions of both classical and noise-tolerant fair classifiers. This may occur when demographic data is unobtainable, prohibitively expensive to generate, or when laws disallow the use of protected attributes to train classifiers. To address this issue, researchers have proposed demographic-blind fair classifiers that use the latent representations in the feature space of the training data to reduce gaps in classification errors between protected groups.

While demographic-blind fair classifiers are an attractive solution in contexts where protected attributes are unavailable, practical questions about the efficacy of these algorithms

remain. First, because these techniques are unsupervised, it is unclear what groups are identified for fairness optimization. Under what circumstances are demographic-blind fair classifiers able to achieve fairness for social groups that have been historically marginalized or are legally protected? Conversely, are the groups constructed by demographic-blind fair classifiers arbitrary and thus divorced from salient real-world sociohistorical context? Second, assuming that demographic-blind fair classifiers do identify and act on meaningful groups of individuals, how does their performance (in terms of predictions and fairness) compare to classical and noise-tolerant fair classifiers?

Benchmark studies address this gap by focusing on the evaluation of a large set of models under expansive and carefully controlled conditions [67, 96]. These studies provide important context for the ML field, e.g., by identifying models that do not work well in practice, models that have equivalent performance characteristics under a wide range of circumstances, and areas where new models may be needed. To the best of my knowledge, existing benchmark studies focus solely on classical fair classifiers, which motivates me to update their results.

Motivated by the proliferation of fundamentally different fair classifiers, in this chapter I present an empirical, head-to-head evaluation of the performance of 14 classifiers spread across four classes: two unconstrained classifiers, seven classical fair classifiers, three noise-tolerant fair classifiers, and two demographic-blind classifiers. My study evaluates the accuracy, stability, and fairness guarantees of these classifiers across four datasets as the noise in the protected attribute varies. To help explain the performance differences observed, I calculate and compare the feature importance vectors for the various trained classifiers. This methodological approach enables me to compare the performance of these algorithms under controlled, naturalistic circumstances in an apples-to-apples manner.

5.2 Algorithms and Metrics

In this section, I introduce the 14 classifiers that I evaluated in this study and the metrics I used to evaluate them.

5.2.1 Classifiers

I group the classifiers that I evaluated in this study into four classes: (1) unconstrained classifiers that solely optimize for accuracy; (2) classical fair classifiers that require access to protected attributes at training (and sometimes testing) time, and assume that this data are

accurate; (3) noise-tolerant fair classifiers that also require access to protected attributes but account for uncertainty in the data; and (4) demographic-blind fair classifiers that jointly optimize for accuracy and fairness but without access to any protected attribute data. The set of classifiers I have selected is not exhaustive. Instead, aim to include representative classifiers from the various types of approaches that exist within each class. I discuss the classifiers from each class that I selected for my study below, with further details on related approaches in each subsection.

5.2.1.1 Unconstrained Classifiers

I chose two classifiers that do not have any fairness constraints, i.e., they only aim to maximize predictive accuracy.

- **Logistic Regression (LR)** is the simplest classifier I evaluate. While LR is demographic-aware because it takes all features (including protected attributes) as model inputs at both train and test time, it is not designed to achieve any fairness criteria.

- **Random Forest (RF)** is an ensemble method for classification built out of decision trees. Like LR, I train RF classifiers on all input features including protected attributes.

5.2.1.2 Classical Fair Classifiers

I chose seven classifiers from the literature that take protected attributes as input and attempt to achieve demographic fairness. These classifiers vary with respect to how they implement fairness, i.e., by pre-processing data, in-process during model training, or by post-processing the trained model. In particular, there exist many techniques for fairness optimization in this class, such as: reweighting of samples via group sizes [38, 63, 105] or via mutual independence of protected and unprotected features in the latent representations [218,219], adding fairness constraints during the learning process [4,5,107,215], or by changing the output labels to match some fairness criterion [106, 162]. The seven classifiers I choose below are representative of these different approaches.

- **Sample Reweighting (SREW)** is a pre-processing technique that takes each (group, label) combination in the training data and assigns rebalanced weights to them. The

goal of this procedure is to remove imbalances in the training data, with the ultimate aim of ensuring fairness before the classifier is trained [105].

- **Learned Fair Representation (LFR)** is a pre-processing technique that converts the input features into a latent encoding that is designed to represent the training data well while simultaneously hiding protected attribute information from the classifier [218].
- **Adversarial Debiasing (ADDEB)** is an in-process technique that trains a classifier to maximize accuracy while simultaneously reducing an adversarial network's ability to determine the protected attributes from the predictions [219].
- **Exponentiated Gradient Reduction (EGR)** is an in-process technique that reduces fair classification to a set of cost-sensitive classification problems, essentially treating the main classifier itself as a black box and forcing the predictions to be the most accurate under a given fairness constraint [4]. In this case, the constraint is solved as a saddle point problem using the exponentiated gradient algorithm.
- **Grid Search Reduction (GSR)** uses the same set of cost-sensitive classification problems approach as EGR, except in this case the constraints are solved using the grid search algorithm [4, 5].
- **Calibrated Equalized Odds (CALEQ)** is a post-processing technique that optimizes the calibrated classifier score output to find the probabilities that it uses to change the output labels, with an equalized odds objective [162].
- **Reject Option Classifier (ROC)** is a post-processing technique that swaps favorable and unfavorable outcomes for privileged and unprivileged groups around the decision boundaries with the highest uncertainty [106].

Note that the CALEQ and ROC algorithms have access to protected attributes at both train and test time, while the other classifiers only have access to protected attributes at training time.

5.2.1.3 Noise-tolerant Fair Classifiers

I chose three classifiers from the literature that take protected attributes as input and attempt to achieve demographic fairness even in the presence of noise. Other than the three

classifiers that I chose, I are aware of only one other approach: by [41], who suggests using de-noised constraints to achieve near-optimal fairness.[1]

- **Modified Distributionally Robust Optimization (MDRO)** [209] is an extension of the Distributionally Robust Optimization (DRO) algorithm [90] that adds a maximum total variation distance in the DRO procedure. By assuming a noise model for the protected attributes, it aims to provide tighter bounds for DRO.

- **Soft Group Assignments (SOFT)**, also by [209], is a theoretically robust approach that first performs "soft" group assignments and then performs classification, with the idea being that if an algorithm is fair in terms of those robust criteria for noisy groups, then they must also be fair for true protected groups [104].

- **Private Learning (PRIV)** is an approach by [149] that uses differential privacy techniques to learn a fair classifier while having partial access to protected attributes. The approach requires two steps. The first step is to obtain locally private versions of the protected attributes (like [122]). Second, following [13], PRIV tries to create a fair classifier based on the private attributes. For this study, I select the privacy level hyperparameter to be a medium value (zero).

5.2.1.4 Demographic-blind Fair Classifiers

I chose two classifiers from the literature that attempt to achieve fairness without taking protected attributes as input.

- **Adversarially Reweighted Learning (ARL)** harnesses non-protected attributes and labels by utilizing the computational separability of these training instances to divide them into subgroups, and then uses an adversarial reweighting approach on the subgroups to improve classification fairness [121].

- **Distributionally Robust Optimization (DRO)** is an algorithm that attempts to minimize the worst case risk of all groups that are close to the empirical distribution [90]. In the spirit of Rawlsian distributive justice, the algorithm tries to control the risk to minority groups while being oblivious to their identities.

[1] [41]'s source code only supported Statistical Parity and False Discovery constraints, not EOD, which is why I omitted their classifier from my analysis.

These two classifiers operate under similar principles: they both try to reduce the gap in errors between protected groups by reducing the classification errors between latent groups in the training set. They do however have one difference: while DRO just increases the weights of the training examples that have higher errors, ARL trains an auxillary adversarial network to identify the regions in the latent input space that lead to higher errors and tries to equalize them, a phenomenon [121] call *computational identifiability*.

5.2.2 Evaluation Metrics

To compare the above 14 classifiers head-to-head, I studied their predictive power and their ability to achieve a fairness condition. I also measured the stability of these quantities when noise in the protected attributes was and was not present (described in 5.3.2).

To assess predictive performance I computed accuracy, defined as:

$$\text{Accuracy}=\frac{\text{number of correct classifications}}{\text{test dataset size}}. \tag{5.1}$$

Accuracy is continuous between zero and one with the ideal value being one, which indicates a perfectly predictive classifier.

Many measures of fairness exist in the literature [140]. For the purposes of this study, however, I needed to choose a metric that is supported by all the 14 classifiers so that my comparison is apples-to-apples. The classical and noise-tolerant fair classifiers have support for achieving any user-specified fairness constraint, while the demographic-blind fair classifiers try to minimize the gap in utility between the protected groups. Based on this limitation, and for the sake of brevity, I choose the Average Odds Difference between two demographic groups as my fairness metric, and subsequently choose Equal Odds Difference (EOD) over both groups as my regularization constraint for the classical and noise-tolerant fair classifiers. EOD is defined as:

$$\text{EOD}=\frac{(\text{FPR}_{\text{unpriv}}-\text{FPR}_{\text{priv}})+(\text{TPR}_{\text{unpriv}}-\text{TPR}_{\text{priv}})}{2} \tag{5.2}$$

where TPR is the true positive rate and FPR is the false positive rate. Priv and Unpriv denote the privileged and unprivileged groups, respectively. The ideal value of EOD is zero, which indicates that both groups have equal odds of correct and incorrect classification by the trained classifier.

In this study, when I evaluate fairness, I do so for binary sex attributes. I adopted this

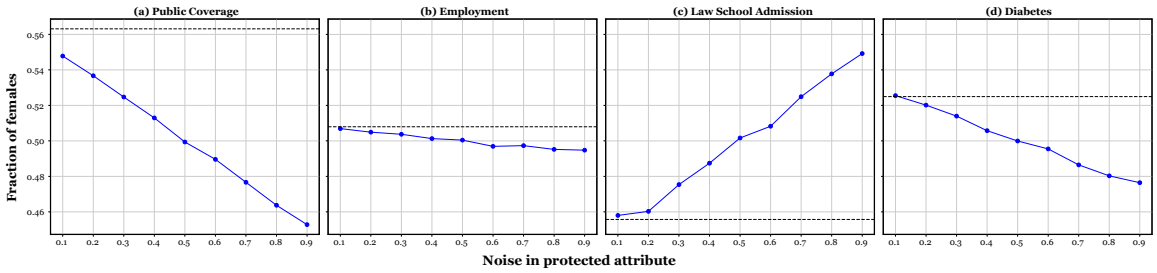


Figure 5.1: Fraction of females in the datasets after adding synthetic noise. The dashed line indicates the true fraction of females.

approach because the datasets I use in my evaluation all include this attribute (see 5.3) and four classifiers in my evaluation (e.g., CALEQ, ROC, EGR, GSR) only support fairness constraints over two groups. Whenever necessary, I consider males to be the privileged group and females to be the unprivileged group. Note that optimizing for fairness between two groups is the simplest scenario that fair classifiers will encounter in practice—if they perform poorly on this task, then they are unlikely to succeed in more complex scenarios with multiple, possibly intersectional, groups.

5.3 Methodology

In this section, I describe the approach I used to empirically evaluate the 14 classifiers that I chose for my study.

5.3.1 Case Studies

To observe how the classifiers perform on real-world data I chose four different datasets. The classification tasks are described below. Each dataset had binary sex as part of the input features.

1. **Public Coverage [58].** The task is to predict whether an individual (who is low income and not eligible for Medicare) was covered under public health insurance. I used census data from California for the year 2018.

2. **Employment [58].** The task is to predict whether an individual (between the ages of 16 and 90), is employed. For this task too, I looked at census data from California for the year 2018.

3. **Law School Admissions [210].** The task is to predict whether a student was admitted to law school.

4. **Diabetes [195].** The task is to predict whether a diabetes patient was readmitted to the hospital for treatment after 30 days.

For each of these case studies, I split the dataset into train and test sets in an 80:20 ratio, trained every classifier on the same training set, and then used the trained classifiers to generate predictions on the same testing set. I verified via two-tailed Kolmogorov–Smirnov tests [115, 191] and Mann–Whitney U tests [137] that the test set distribution for every feature was the same as the training set distribution. Finally, I calculated the metrics in 5.2.2 on these predictions and compared the results from each classifier head-to-head. I repeated this procedure ten times to assess the stability of accuracy and EOD for each classifier.

5.3.2 Synthetic Noise

While studying the performance of these classifiers on a variety of real-world datasets is important, in order to get a more thorough understanding of the theoretical fairness and predictivity limits of the classifiers I subjected them to robust synthetic stress tests. As discussed in 2.2.1, in the real world, practitioners may not have access to the protected attribute information of people in their dataset. As a result, practitioners may use inference tools to find proxies for protected attributes, which can lead to unexpected, unfair outcomes [77]. To characterize what might happen in such a scenario, I perform the following synthetic experiments:

1. For each dataset, with a given probability (ranging from 0.1 to 0.9), I randomly flip the protected attribute labels (binary sex in this case) in the dataset. I refer to this probability value as *noise*.

2. With the synthetically generated dataset from Step 1, I then proceed to split the dataset 80:20, train all 14 algorithms on the same training set, and then calculate predictions on the same test set. The noisy (flipped) labels are passed as inputs to the classifiers at this step.

3. Next, with the predicted outcomes from Step 2, I calculate accuracy and EOD. Note calculate EOD with the *true* protected attributes, i.e., I measure the output bias in terms of the original sex labels from the given dataset.

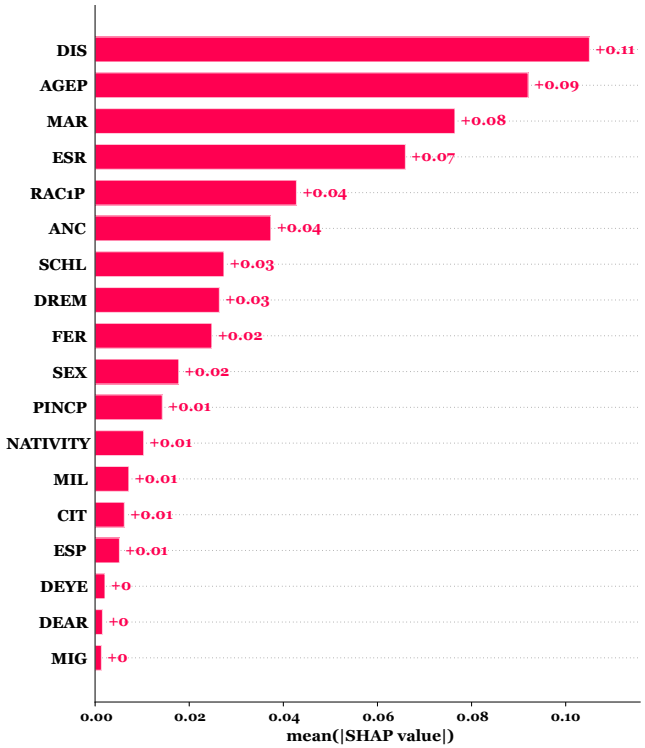


Figure 5.2: KernelShap feature explanations calculated for the Logistic Regression (LR) classifier when trained on the Public Coverage dataset with no added noise. I used the same approach to calculate feature importances for every classifier-dataset pair at different noise levels.

4. I repeat Steps 1–3 ten times for each value of noise, to ensure statistical fairness and assess the stability of my metrics per classifier.

Figure 5.1 shows the fraction of females in the noised datasets at each level of noise. The fraction of females goes up or down with noise depending on what the true fraction of females in the different datasets were to begin with.

5.3.3 Calculating Feature Importance

To help explain the variations in performance that I observed in my results, I calculated feature importance for each of my trained models. Although there are several black-box model explanation tools in the research literature—such as LIME [169], SHAP [136], and Integrated Gradients [198]—I required an explanation method that was model agnostic. The method that I settled on was KernelShap.[2] According to the documentation, KernelShap

[2] `https://shap-lrjball.readthedocs.io/en/latest/generated/shap.KernelExplainer.html`

uses a special weighted linear regression model to calculate local coefficients, to estimate the Shapley value (a game theoretic concept that estimates the individual contribution of each player towards the final outcome). As opposed to retraining the model with every combination of features as in vanilla SHAP, KernelShap uses the full model and integrates out different features one by one. It also supports any type of model, not just linear models, and was thus a good candidate for my study.

Figure 5.2 shows an example distribution of feature importances calculated for the LR algorithm when trained on the Public Coverage dataset at noise level zero (i.e., no noise). In a similar fashion, I used KernelShap to calculate feature importance values for trained classifier outputs at noise levels 0, 0.2, 0.4, 0.6 and 0.8 for all 14 models.

Research by [117] has shown that different explanation methods often do not agree with each other. I do not claim that the feature importances I calculated using KernelShap are guaranteed to agree with those produced by other tools. Nonetheless, I are specifically interested in the relative importance of the sex feature towards the final outcome as compared to the other input features. Shapley value-based explanations give us a reasonable sense of relative feature importance, as has been empirically shown in previous work [81].

5.4 Results

In this section, I present the results of my experiments. I begin by examining the baseline performance of the 14 classifiers when there is no noise, followed by their performance in the presence of synthetic noise. Finally, I delve into feature importance explanations to help explain the relative performance characteristics of the classifiers.

5.4.1 Baseline Characteristics

Figure 5.3(a–d) shows the accuracy and fairness outcomes for all 14 classifiers when there was no noise in the datasets. I executed each classifier ten times without fixing a random seed and present the resulting distributions of metrics using violin plots. I observe that most of the classifiers achieved comparable accuracy to each other on each dataset, and that most classifiers exhibited stable accuracy over the ten executions of the experiments. Learned Fair Representation (LFR), Soft Group Assignment (SOFT), and Distributed Robust Optimization (DRO) were the exceptions: the former two exhibited unstable accuracy on all four datasets, the latter on two datasets. conduct As shown in Figure 5.3(e–h), EOD

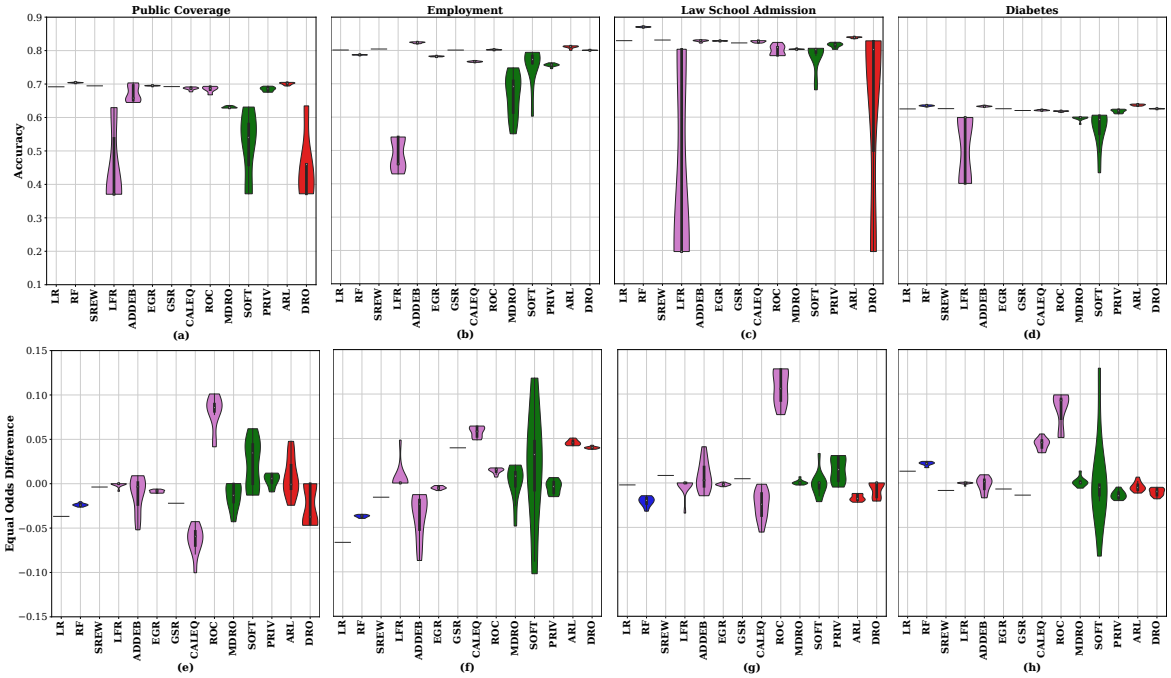


Figure 5.3: Accuracy and EOD for my 14 classifiers, calculated over four datasets with ten runs each. No noise was added to the protected attribute in these tests. Violins are color coded by class: blue for unconstrained classifiers, purple for classical fair classifiers, green for noise-tolerant fair classifiers, and red for demographic-blind fair classifiers. LR, SREW, and GSR are deterministic algorithms and therefore appear as fixed points.

was considerably more variable over runs than accuracy. The unconstrained classifiers (LR and RF) were relatively stable and, in some cases, achieved roughly equalized odds (e.g., on the Law School and Diabetes datasets). The classical fair classifier group contained the two least fair classifiers in these experiments (CALEQ and ROC), while the other pre-processing and in-process algorithms performed relatively better. Adversarial Debiasing (ADDEB) was slightly unstable but the distribution centered around zero. Among the noise-tolerant fair classifiers, Soft Group Assignment (SOFT) was unstable on three out of four dataset, while the other two classifiers (MDRO and PRIV) were relatively more stable and more fair. The two demographic-blind fair classifiers (ARL and DRO) were unstable on the Public Coverage dataset (Figure 5.3e) and did not achieve equalized odds on the Employment dataset (Figure 5.3f). However, ARL and DRO were stable and fair on the remaining two datasets.

In summary, I observe that the accuracy and fairness performance of these classifiers was dependent on the dataset that they are trained and tested on, i.e., there was no single

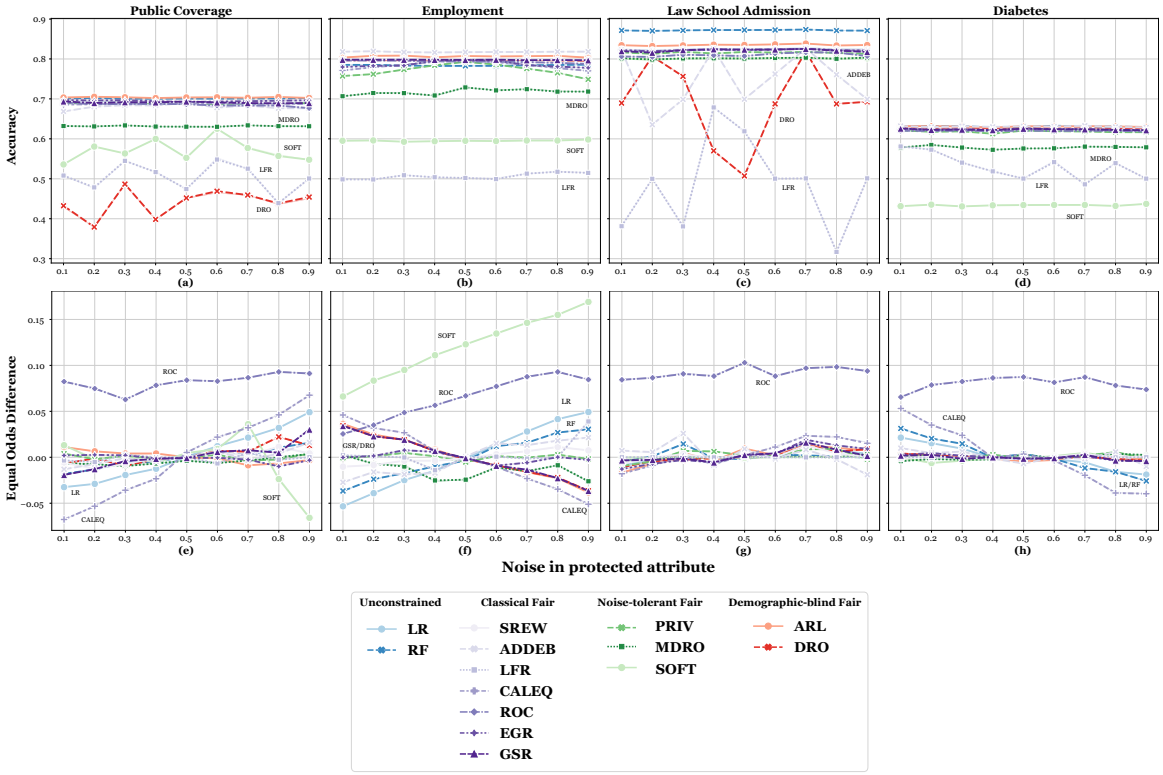


Figure 5.4: Accuracy and EOD for my 14 classifiers, calculated over four datasets as I increase noise in the protected attribute (sex). Each point is the average of ten runs for a given classifier, dataset, and noise level. Classifiers are color coded according to the legend. I highlight classifiers whose performance significantly diverges from the consensus with annotated labels.

best classifier. Additionally, I can see that several classifiers are consistently unstable, which explains some of the results that I will present in the next section.

5.4.2 Characteristics Under Noise

Next, I present the results of experiments where I added noise to the protected attribute of the datasets. I added noise in increments of 0.1 starting from 0.1 and ranging up to 0.9. I added a given amount of noise to each dataset ten times and repeated the experiment, thus I plot the average values of accuracy and EOD for each classifier at each noise level.

Figure 5.4(a–d) shows the accuracy of the 14 classifiers' outputs as I varied noise. I observe that the MDRO, SOFT, and LFR classifiers had poor accuracy across all datasets and noise levels, while the DRO classifier had poor accuracy in two out of the four datasets. These observations mirror those from Figure 5.3, i.e., these classifiers exhibited poor average accuracy in the noisy experiments because they were unstable in general. The other classifiers

tended to be both accurate and stable, irrespective of noise.

As shown in Figure 5.4(e–h), the EOD results were much more complex than the accuracy results. ROC generated unfair outputs over all four datasets, at every noise level. Its companion post processing algorithm, CALEQ, exhibited rising EOD with noise for the Public Coverage dataset (Figure 5.4e) and falling EOD for the Employment and Diabetes datasets (Figure 5.4f, h).[3] The unconstrained classifiers (LR and RF) moved in the same direction for every dataset, either rising (Figure 5.4e, f) or falling (Figure 5.4h) with noise. The SOFT classifier also exhibited some variable behavior: on the Employment dataset EOD rose with noise (Figure 5.4f), and on the Public Coverage (Figure 5.4e) and Employment datasets it failed to achieve equal odds at all noise levels. The remaining classifiers tended to achieve equal odds irrespective of the noise level.

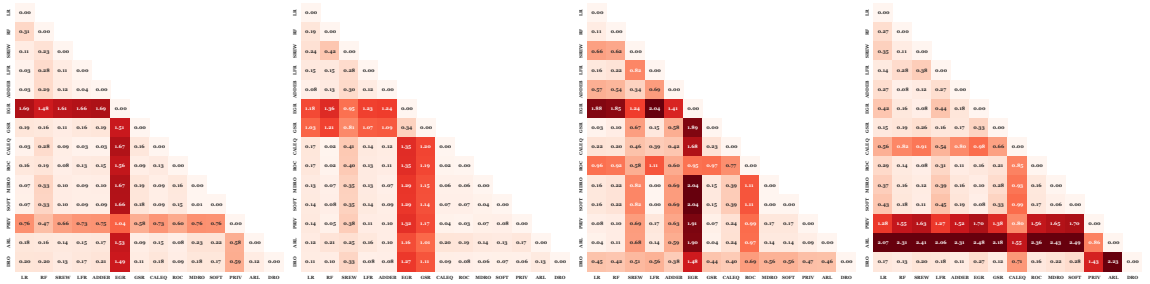
Figure 5.4 only depicts average values for accuracy and EOD, which is potentially problematic because it may hide instability in the classifiers' performance. To address this I present Figure 5.7, which shows the distribution of accuracy and EOD results for each classifier on each dataset at the 0.1, 0.5, and 0.9 noise levels. I observe that, overall, no classifier became consistently less stable as noise increased. Rather, the stability patterns for each classifier mirrored the patterns that I already observed in Figure 5.3.

In summary, the classifiers that had problematic performance in the baseline experiments (see Figure 5.3) continued to have issues in the presence of noise. Additionally, the unconstrained classifiers exhibited inconsistent fairness as noise varied. Surprisingly, the noise-tolerant classifiers did not uniformly outperform the other fair classifiers.

5.4.3 Feature Importance

Finally, I delve into model explanations as a means to further explore the root causes of the classifier performance characteristics that I observed in the previous sections. First, I calculated feature explanations using KernelShap for every classifier at five noise levels—0, 0.2, 0.4, 0.6 and 0.8—using the method I described in 5.3.3. Next, I averaged the explanation distributions for each classifier to form a feature importance vector per classifier. Finally, I repeated this process for each dataset. For each dataset, I calculated Wasserstein distances [206] between the feature explanation distributions for each algorithm pair and present the results in Figure 5.5. Additionally, I plot the rank of the sex feature in terms of

[3]Note that a higher value of EOD (Equation 5.2.2) signifies that females received more positive predictions than males.



(a) Public Coverage **(b)** Employment **(c)** Law School Admissions **(d)** Diabetes

Figure 5.5: Wasserstein distances between the average KernelShap feature importance distributions over different noise levels for the four datasets. Each square compares the average feature importances of two classifiers. Redder squares denote pairs of classifiers with more divergent feature importance distributions.

mean absolute feature importance for each classifier and present the results in Figure 5.6 (I also show the range of ranks if they vary over noise).

Figure 5.5 reveals that, with few exceptions (EGR in Public Coverage, EGR and GSR in Employment, EGR and ROC in Law school, and CALEQ, PRIV and ARL in Diabetes), most classifiers had similar feature explanation distributions. I do not observe any clear patterns among the exceptional classifiers, i.e., no classifier consistently diverged from the others across all datasets. Further, I do not observe clear correlations between accuracy, EOD, and feature distribution similarity, suggesting that different classifiers took different paths to reach the same levels of performance.

Figure 5.6 is more informative than Figure 5.5. four of the classifiers that exhibited consistently poor performance—LFR, MDRO, and SOFT (Figure 5.3a–d), and ROC (Figure 5.3e–h)—learned to weight the sex feature higher than other features, which may point to the root cause of their accuracy and fairness issues. Similarly, the unconstrained classifiers (LR and RF) exhibited changing EOD with noise levels in three out of four datasets (Figure 5.4e, f, h), but not for Law School Admissions (Figure 5.4g), and I observe that they learned a relatively low weight for sex among the available features for the Law School dataset. CALEQ also learned a relatively low weight for sex on the Law School datase and was subsequently unaffected by noise (Figure 5.4g), but showed variable trends in EOD for the other three dataset (Figure 5.4e, f, h) on which it learned a relatively higher weight for sex.

Sex was the lowest ranked feature for the two demographic-blind fair classifiers (DRO

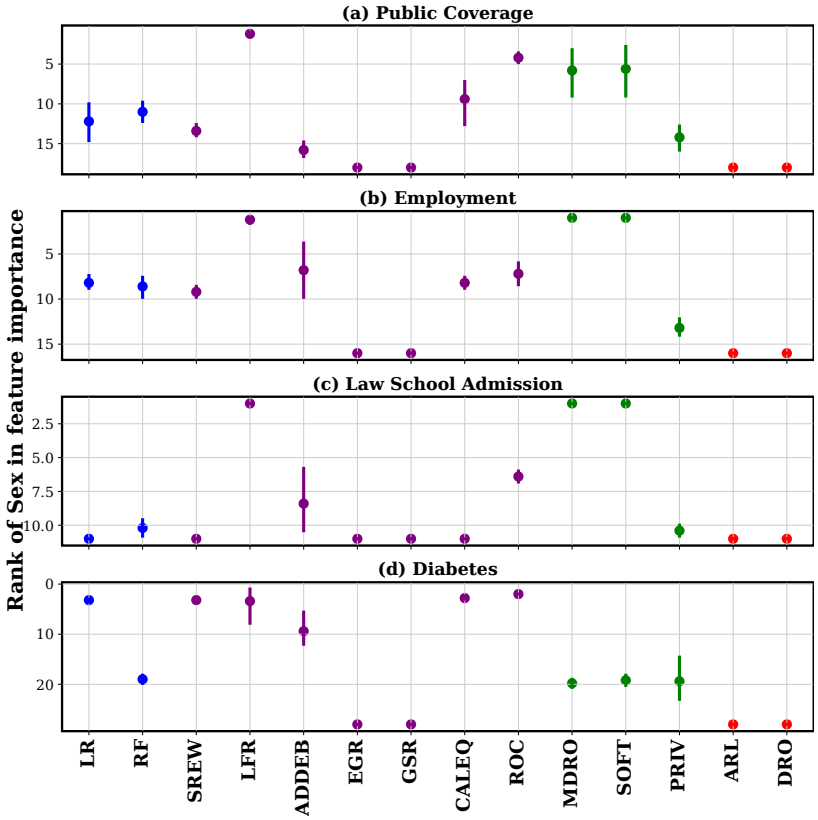


Figure 5.6: Rank of Sex in the average absolute KernelShap feature importances for the different algorithms in my case studies.

and ARL), which makes sense because they were not given these features as input. EGR and GSR also did not have access to sex while classifying the test dataset, so they also had sex as the lowest ranked feature.

5.4.4 Fairness-Accuracy Tradeoff

Three algorithms in my list - EGR, GSR, and PRIV, provide a mechanism to control the fairness-accuracy tradeoff via a hyperparameter – namely fairness violation *eps* in the case of EGR and GSR [4], and the privacy level ϵ in the case of PRIV [149]. Based on the experiments the authors of these algorithms did in their papers, I used different *eps* values between 0.01 and 0.20 and ϵ values between -2 and 2 and reran my experiments. I found that tweaking the tradeoff hyperparameter did not contribute meaningfully to the stability and noise resistance capabilities of these algorithms. Consequently I omit these results from the chapter.

5.5 Discussion

In this study, I benchmarked 14 ML classifiers divided into four classes and evaluated their accuracy, fairness, and stability across four datasets with varying levels of random noise in the protected attribute. My results suggest that classical fair classifiers like SREW and EGR may perform well in the face of noise, and that demographic-blind fair classifiers like ARL can achieve comparable fairness performance to demographic-aware fair classifiers on some datasets.

However, I urge caution with the adoption of demographic-blind fair classifiers for practical reasons. Monitoring the health of a classifier like ARL in the field requires demographic data, and determining whether a classifier will achieve acceptable performance in a given context requires thorough evaluation on a dataset that includes demographic data. My study highlights the need for further development in the areas of noise-tolerant and demographic-blind fair classifiers, and I hope to provide a foundation for evaluating these novel classifiers in the future by releasing my source code and data.

As models are deployed in the real world, they may face issues such as data or concept drift, leading to a fair model becoming unfair over time. This can result in biased decisions and unintended consequences, compromising the model's original intent of promoting fairness and equity. Thus, during monitoring the health of classifiers, it is also crucial to develop appropriate methods that can detect and correct drift to ensure that the model continues to achieve its intended objectives. In the next chapter, I discuss a system to monitor and maintain the model's performance over time and prevent it from becoming unfair in the face of drift.

5.6 Limitations

My study has several limitations. First, I only evaluate classifiers using binary protected attributes. It is unclear how their performance and consistency would change under more complex conditions. That said, I are confident that the classifiers that performed poorly will continue to do so in the presence of more complex fairness objectives. Second, my case studies and synthetic experiments, while thorough, are by no means completely representative of all real world datasets and contexts. We caution that my results should not be generalized indefinitely. Third, I did not evaluate all of the classical fair classifiers from the literature

(see [67] and [140] for more). That said, my primary focus was on adding to the literature by benchmarking noise-tolerant and demographic-blind fair classifiers. Finally, in this study I only evaluated one fairness metric—EOD—because it was the common denominator among all of the classifiers I selected. Future work could explore fairness performance more deeply by choosing other fairness metrics along with subsets of amenable classifiers.

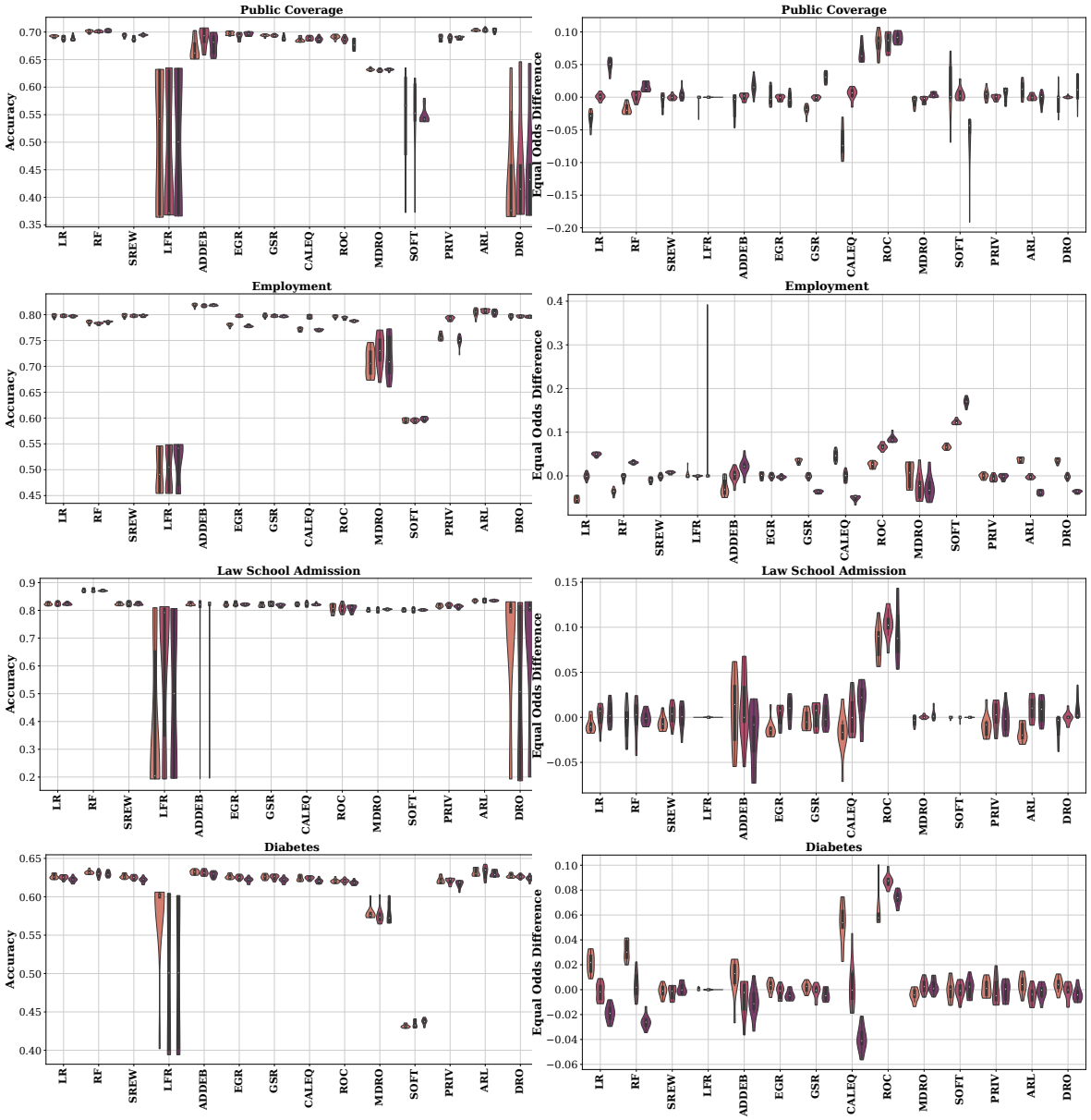


Figure 5.7: Plots showing the stability of my 14 classifiers over three different levels of noise in protected attributes (0.1, 0.5 and 0.9). For each dataset I present the stability of each classifiers' accuracy and EOD.

Chapter 6

FairCanary: Rapid Continuous Explainable Fairness

6.1 Research Problem

As ML models are increasingly deployed in high-stakes applications, the issue of concept drift has become a major concern. Concept drift refers to the phenomenon where the underlying data used to train a model changes over time, leading to a mismatch between the training data and the data used in production. This can cause the model to make errors and impact its ability to make accurate predictions. While there is a significant body of research on detecting and mitigating concept drift in ML models, little attention has been paid to the impact of drift on the fairness of these models.

To address this issue, I present FairCanary, a continuous model monitoring system that offers two significant capabilities to help ensure model fairness over time. First, FairCanary incorporates a novel model bias quantification metric called Quantile Demographic Disparity (QDD) that uses quantile binning to measure differences in the overall prediction distributions over subgroups. Second, FairCanary reuses explanations computed for each individual prediction to quickly compute explanations for its bias metrics. These optimizations make FairCanary significantly faster and more suitable for continuous monitoring than previous work on generating feature-level bias explanations.

FairCanary is closely related to the work by [145], with a couple of key differences. While [145] calculated fairness explanations from scratch using Shapley-based methods, for

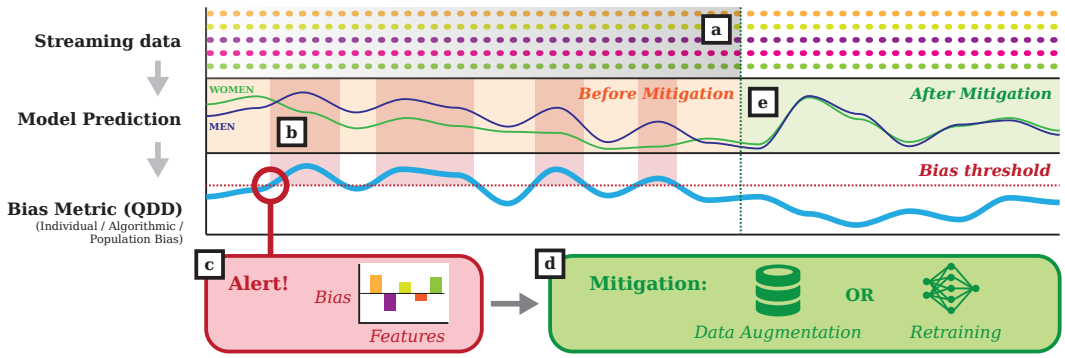


Figure 6.1: A diagram illustrating how FairCanary monitors the inputs and outputs of a trained model over time, identifies bias, alerts the developer, and assists in mitigation. See 6.2.1 for further details.

FairCanary I assume that a system that continuously generates prediction explanations, like the systems in [50], are already available. FairCanary sits on top of such a system and reuses these existing prediction explanations to generate fairness explanations in linear time (see 6.2.3).

In the following sections, I introduce FairCanary, present an overview of its operation and capabilities, formally define the QDD metric, and discuss how to obtain explanations for it by reusing existing prediction attributions. Next, I present a synthetic case study that highlights FairCanary's capabilities, demonstrating how it can detect and explain bias caused by concept drift in real-time. By providing an effective tool for monitoring and mitigating bias and unfairness in ML models, FairCanary can help bring more equity and justice to the individual stakeholders impacted by deployed models.

6.2 FairCanary System Description

I now describe FairCanary, my system for performing continuous model monitoring. First, I present the context in which FairCanary is designed to operate and describe its operations at a highlevel. Next, I discuss how FairCanary measures bias and introduce my novel Quantile Demographic Disparity (QDD) metric. Finally, I describe how FairCanary provides explanations that attribute observed biases to specific features, and the bias mitigation options provided by FairCanary.

6.2.1 Overview

FairCanary is a system for performing continuous model monitoring. It is designed to be deployed into production environments alongside a trained ML model to help the developers monitor the model's performance over time in terms of traditional and fairness performance metrics. In this paper, my focus will be on the latter, fairness metrics.

The developer of the model must configure FairCanary, a priori, by defining the (intersectional) groups for which unfairness will be monitored, identifying the feature(s) in the dataset that encode group membership, establishing base rate statistics for these groups (i.e., as ascertained from the model's underlying training data), and setting thresholds to trigger bias alerts.

Figure 6.1 illustrates FairCanary's mode of operation and some of its key capabilities. (**a**) As new data arrives it is fed into the trained model, which (**b**) produces predictions that are stored by FairCanary. Over time, FairCanary maintains a record of the predictions for each group at an operator-specified time granularity.

(**c**) Periodically, FairCanary computes the fairness metric (QDD, see 6.2.2) for the model and alerts the developers if any group performs below the preconfigured threshold. FairCanary provides explanations along with alerts that inform the developer which feature(s) are attributable to the issue (see 6.2.3). (**d**) Subsequently, the developer may mitigate the emergent unfairness using tools provided by FairCanary (see 6.2.4), which (**e**) should return the model to a state where predictions are fair across groups.

6.2.2 Quantile Demographic Disparity

In this section, I describe a new metric to measure bias in the predictions of a ML model, at both the group and individual level. The prediction tasks covered by my metric include any single dimensional output, such as regression output, or the output of any particular class in a multi-class classification model.

My metric, Quantile Demographc Disparity (QDD), falls within the distributional difference family of fairness metrics (see Table 2.1). I argue that there are two reasons for assessing the fairness of an ML model by comparing its prediction distributions over the groups of interest, versus focusing on post-threshold outcomes. The *first* reason is to ensure that I measure bias across the whole spectrum of classified individuals, as opposed to focusing solely on the individuals that are above the threshold of selection, or on group-level approximations.

Second, as groups of interest get smaller, they reveal more information about intra-group disparities that would have otherwise been lost due to aggregation [79], all the way down to groups of one, i.e., individuals. This helps remove aggregation bias from the bias measurement itself.

6.2.2.1 Desired Properties of a Bias Metric

I now discuss desirable properties of a distributional fairness metric that fit my stated objectives:

1. The metric should be in the units of the model's prediction scores. The utility of this is especially evident when dealing with continuous output models. This is desirable because it provides insight into the extent of the problem, before human intervention is applied, such as deciding and applying a threshold.

2. The metric should take the value zero only if the prediction distributions being compared are exactly the same. The benefit of this is that, when taken along with the first property, it gives the ML practitioner a mental scale to understand the extent of the bias.

3. The metric should be continuous with respect to changes in the geometry of the distribution [145]. This ensures that any distributional change is captured.

4. The metric should be non-invariant with respect to monotone transformations of the distributions [145]. For example, given two samples of points S_1 and S_2, if I multiply the value of each point in the samples by a constant k, the distance between the modified samples should now depend on k. Jensen-Shannon Divergence (JSD) [127], for example, does not satisfy this property.

5. The metric should be bias-transforming as described in [208], i.e., the metric should not be satisfied by a model that preserves the biases present in the data.

QDD satisfies all of these properties when the number of bins is equal to the number of samples.The choice of number of bins can be adjusted to satisfy these properties.

6.2.2.2 Formalization

I now describe my QDD metric, which is a function of the quantile bin that a prediction event lies in. QDD is a novel formulation of the Wasserstein-1 distance metric [206], and thus it is designed to work for continuous outputs and can be customized to provide sliced views down to the individual-level.

For two groups G_1 and G_2, let the two distributional samples of model scores be S_1 and S_2. I divide the samples into B bins of equal size N_1 and N_2, respectively. This is equivalent to segmenting by quantiles. For example, if there are 10 bins, I are essentially bucketing individuals between the 0^{th}–10^{th} percentile, 10^{th}–20^{th} percentile, and so on.

I define QDD for bin b as

$$QDD_b = \mathbb{E}_{G_{1,b}}[S_1] - \mathbb{E}_{G_{2,b}}[S_2]. \tag{6.1}$$

This can be approximated as

$$QDD_b = \frac{1}{N_1}\sum_{n=1}^{N_1} S_{1,n} - \frac{1}{N_2}\sum_{n=1}^{N_2} S_{2,n}. \tag{6.2}$$

The QDD, when conditioned on certain attributes C, becomes the Conditional Quantile Demographic Disparity.

To demonstrate the flexibility of QDD, I demonstrate how it can be used to measure three different conceptualizations of bias.

1. Intra-Group Bias is defined as the maximum QDD across the b bins of a given group of individuals. This quantity is useful to combat aggregation bias within groups.

2. Disparity with Base Rate is defined as the difference between the QDD calculated over the production data and the QDD of the training data. This quantity is most relevant when the training data is representative of the population the model is expected to encounter during deployment.

3. Individual Fairness via Alignment.

QDD is defined between two groups over a given number of bins, which determines the resolution of the metric. If the number of bins is equal to the number of instances in the

sample, QDD becomes a comparison between individuals at the same rank or percentile. This is equivalent to the concept of *alignment* proposed by [188].

Computing QDD over individual instances gives me a clean way to obtain individual fairness insights, with the counterfactual example being the same ranked counterpart in the opposite group. This method does not require me to compute complex counterfactuals, which could have their own biases and errors [111]. The principle I use to justify this insight is as follows: if there is no bias between two groups, and I have a large enough sample of both, then the distance between individuals of the same rank in the prediction space should be zero.

6.2.3 Explanation

Explainability of ML systems that are deployed in production is a very important part of the practice of responsible AI [25,139]. This especially applies to models that are contributing to decisions that can impact peoples' lives. Such decisions cannot be inscrutable, and thus the internal workings employed by the ML models must be human-verified to be logical and normatively justifiable.

FairCanary incorporates two state-of-the-art methods for explaining the output of predictions in terms of specific features: Shapley value-based methods [135] and (if the model being monitored is differentiable) Integrated Gradients (IG) [198]. I adopted these methods because they satisfy the desirable axiom of *efficiency* [135], which helps provide a precise accounting of bias.[1]

Just like explanations for individual predictions, I argue that it is vital to be able to explain measures of fairness or bias, so that the features that are responsible for the bias can be identified. FairCanary incorporates a novel method for explaining the feature importance contributing to QDD that I call Local Quantile Demographic Disparity attribution.

The Local QDD attribution for feature f, for prediction sample S_1 over S_2 in bin b, $QDDA_{b,A,f}$ is a measure of the change in QDD in bin b that can be attributed to (a.k.a. explained by) feature f using attribution method A that satisfies the efficiency axiom. r

[1] Although there are other explanation methods that satisfy efficiency [28, 189] I do not explore them in this work.

denotes the reference and t denotes the target distribution. I define

$$\text{QDDA}_{b,A,f} = \frac{1}{N_t}\sum_{n=1}^{N_t}\text{attr}_{n,A,S_1,f} - \frac{1}{N_r}\sum_{n=1}^{N_r}\text{attr}_{n,A,S_2,f} \tag{6.3}$$

where $\text{attr}_{n,A,S_i,f}$ refers to the attribution of the n^{th} data point to feature f for a prediction from bin b of distribution S_i using attribution method A. Given that the attribution method A satisfies the efficiency axiom, $\text{QDD}_b = \sum_{f=1}^{F}\text{QDDA}_{b,A,f}$.

Proof: Since the attribution method A satisfies efficiency, for each instance in the sample S_1 and S_2, $\sum_{f=1}^{F}\text{attributions}_f = \text{prediction} - \text{baseline prediction}$.

For the same baseline,

$$\frac{1}{N_t}\sum_{n=1}^{N_t}\sum_{f=1}^{F}\text{attr}_{n,A,S_1,f} - \frac{1}{N_r}\sum_{n=1}^{N_r}\sum_{f=1}^{F}\text{attr}_{n,A,S_2,f} = \frac{1}{N_1}\sum_{n=1}^{N_2}S_{1,n} - \frac{1}{N_2}\sum_{n=1}^{N_2}S_{2,n}$$

$$\therefore \text{QDD}_b = \sum_{f=1}^{F}\text{QDDA}_{b,A,f}.$$

Explaining bias in this manner enables a single attribution to be used for multiple explanations across groups. In contrast, Shapley values over a particular metric must be re-calculated for every grouping. My explanation technique therefore is much more computationally efficient than previous techniques [145] since it requires the calculation of attributions only once. To elaborate, Shapley values without approximation are exponential in the number of model features. While there exist approximation techniques, the complexity is worse than linear time. Hence, Wasserstein Shapley computation (n points times d features) is worse than calculating Shapley values for n points separately for d features. Additionally, I can re-use the Shapley values computed for a data point when calculating QDD between any combination of protected features, whereas the whole computation needs to be repeated for each combination in the case of Wasserstein Shapley.

6.2.4 Mitigation

Mitigation is a key outcome of monitoring bias, enabling corrective action to be taken. FairCanary provides an option for developers to automatically mitigate bias revealed by my QDD metric using a quantile norming approach. In essence, this approach replaces the score

Feature	Values	Distribution
Location	{'Springfield', 'Centerville'}	70:30
Education	{'GRAD', 'POST_GRAD'}	80:20
Engineer Type	{'Software', 'Hardware'}	85:15
Experience (Years)	$(0, 50)$	Normal Distribution
Relevant Experience (Years)	$(0, 50)$	Normal Distribution
Gender	{'MAN', 'WOMAN'}	50:50

Table 6.1: Features, values, and their distributions used in my synthetic case study. Note that the gender feature is only used for measuring and mitigating bias, it is not used for model training or prediction.

of the disadvantaged group with the score of the corresponding rank in the advantaged group, similar to the mitigations proposed in [101, 154]. The justification for quantile norming is that if (1) bias is known to exist, (2) bias is the only rational explanation for disparity, and (3) bias is assumed to be equal within the disadvantaged group, then normalizing across ranks is normatively justifiable.

In essence, quantile norming is a post-processing mitigation. The advantages of post-processing mitigations as opposed to pre-training debiasing are discussed by [74]. Additionally, quantile norming is a relatively computationally inexpensive approach to bias elimination.

I note that all bias mitigation approaches, including quantile norming, should only be adopted in practice after conducting a thorough examining of their consequences on the outputs of a model. [49] demonstrate several cases where mitigation may cause additional harm to individuals or to particular groups. Developers that adopt FairCanary are under no obligation to use quantile norming for mitigation, and are free to adopt other, perhaps more thorough and computationally expensive, approaches (e.g., model retraining [179], data preprocessing [105], etc.) that better suit their needs.

6.3 Case Study

In this section, I present an example of FairCanary in action via a case study on a synthetic dataset. This allows me to inject controlled drifts into the data stream to demonstrate how FairCanary, via QDD, can detect and explain the resulting bias. Additionally, I present comparisons to conventional fairness metrics.

6.3.1 Scenario

In this case study, I posit a scenario where a developer has trained a model to predict the starting salary of job seekers based on relevant features of their resume, such as education level and years of experience (see Table 6.1). Note that the output of this model is continuous. Additionally, the developer designed the model to be fair with respect to the binary gender of job seekers, i.e., the distribution of salaries predicted for men and women should be nearly identical. I assume that the model was audited and found to be fair relative to the data that was available at training time.

I assume that the model has learned the following relationship to predict an individual's salary from the features in Table 6.1:

$$\begin{aligned}\text{Salary} = 50,000 + (20,000 \times \text{location}) + (20,000 \\ \times \text{education}) + (5,000 \times \text{relevant\_experience}) \\ +(100 \times \text{experience}) + (10,000 \times \text{engineer\_type})\end{aligned}$$

In my scenario, the developer deploys this model into production along with FairCanary to continuously monitor its output. I generate 20,000 synthetic job seekers' data per day for three days that are fed into the model, using feature values drawn from the distributions given in Table 6.1 (with the added constraint that experience $\geq$ relevant experience). On Day One and Day Three I generate all of the candidate data correctly, but crucially, on Day Two, I simulate a data engineering bug that erroneously labels all women as 'GRAD' instead of 'POST_GRAD' regardless of their true educational attainment. This reduces the estimated salary for all women post-graduates by $20,000 on Day Two.

I argue that the scenario I have outlined here is realistic. ML-based resume screening and analysis tools are widely available, and given that they gate access to employment opportunities, it is crucial that these systems be fair [30, 164]. The bug I intentionally simulate on Day Two could easily occur in practice, e.g., due to the temporary malfunctioning of a resume parser that prepares data for the salary prediction model.

6.3.2 Analysis

Figure 6.2 shows how FairCanary would detect and explain the fairness problem that occurs on Day Two. The model outputs on Day One show that the prediction distributions

	Day One		Day Two	
Threshold	**SPD**	**DI**	**SPD**	**DI**
\$50000	0.00009	1.00009	-0.00556	0.99439
\$100000	0.00911	1.01749	-0.08290	0.84569
\$200000	0.00088	1.02876	-0.01049	0.65544

Table 6.2: The performance of two conventional fairness metrics, Statistical Parity Difference (SPD) and Disparate Impact (DI), against different salary thresholds for the case study. The predictions on Day One were fair, while they were unfair to women on Day Two. Only one metric catches the bias, and only at one threshold (highlighted in red).

for men and women are mostly aligned (Figure 6.2a), thereby being fair. On the second day (Figure 6.2b), due to the data integrity error discussed above, the prediction distributions differ. When I examine the running plot for QDD[2] (Figure 6.2d), I notice a sharp dip on Day Two—QDD goes from an average value of \$156 on Day One to -\$8677 on Day Two—indicating a bias against women.[3] Note that the absolute value of QDD goes up, indicating an increase in bias, and would trigger the alarm system like in Figure 6.1. Similarly, the feature explanations (generated here using Integrated Gradients) go from being distributed among the different features on Day One (Figure 6.2e) to assigning the majority of blame to the education feature on Day Two (Figure 6.2f).

FairCanary would alert the model developer of the problem on Day Two, and its explanations could help the developer perform root-cause analysis of the bias issue. Based on this information, the developer could then identify and correct the underlying data engineering bug. Once corrected, I observe that the model's predictions are again aligned for men and women on Day Three (Figure 6.2c), and the QDD values have returned to their expected range (Figure 6.2d).

To further illustrate the utility of QDD I compare it with two conventional fairness metrics—Statistical Parity Difference (SPD)[4], and Disparate Impact (DI)[5]—to see if a monitoring system using these metrics would have caught the bias against women on the

[2]For simplicity, I set the number of quantile bins as 1 for the case study. Thus, the explanations are for the entire distribution and not any particular quantile bin.

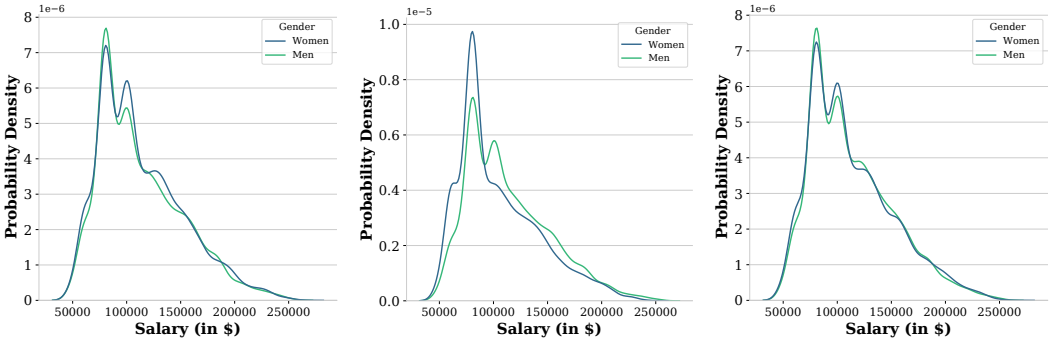
[3]Recall in 6.2.2.1 I say that one useful feature of QDD is that the metric has the same units as the predicted output. Having the QDD value in dollars clearly helps users to understand the extent of bias and thereby aids usability.

[4]Statistical or Demographic Parity Difference is the difference in the positive outcome rate between the privileged and unprivileged group. SPD = $\Pr(\hat{y} = 1|p = 1)$ - $\Pr(\hat{y} = 1|p = 0)$.

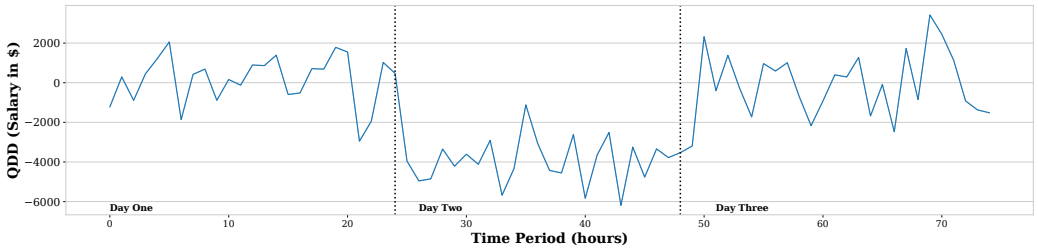
[5]Disparate Impact is the ratio of the passing rate of the the privileged and unprivileged group. DI = $\frac{Pr(\hat{y}=1|p=1)}{Pr(\hat{y}=1|p=0)}$.

second day.

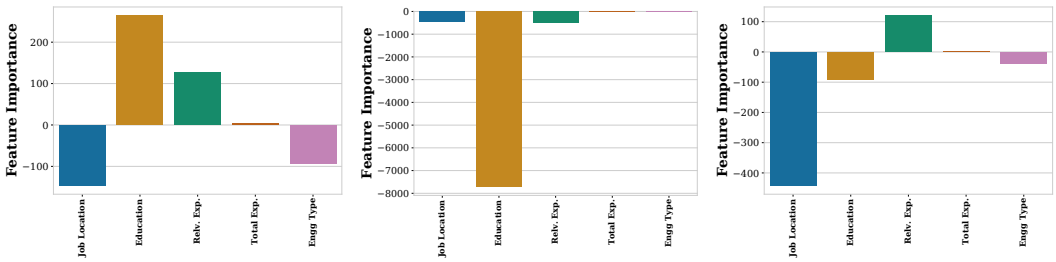
Table 6.2 shows the values of the two conventional metrics for different salary thresholds (i.e., for the positive outcome) on Day One and Day Two. I configure the alert threshold for both metrics[6] in accordance with the US UGESP $4/5^{\text{ths}}$ rule [61] that is commonly used in disparate impact analysis [211]. Alarmingly, I observe that, as configured here, SPD would not catch the bias on the second day at all, and DI would only catch it at one threshold level.



(a) Prediction distribution on Day One **(b)** Prediction distribution on Day Two **(c)** Prediction distribution on Day Three



(d) Continuous plot of the QDD metric over time. There is a clear dip on the second day.



(e) QDD Explanations for Day One **(f)** QDD Explanations for Day Two **(g)** QDD Explanations for Day Three

Figure 6.2: Plots for my case study showing how FairCanary would detect and explain the bias against women on Day Two on a continuously running salary prediction model. The explanations for Day Two clearly indicate Education as the feature responsible for the bias, which enables the practitioner to correct the data integrity issue and fix the biased predictions.

[6]For Statistical Parity Difference, since there is no conventionally accepted value, I set the threshold to 20% to be consistent with Disparate Impact.

6.3.3 Limitations

While threshold independence is one of the strengths of QDD, it is also a potential weakness: without ground truth labels, calculated disparities are, at the end of the day, best-case approximations of the discrimination that actually takes place in society. I therefore do not advocate for the elimination of conventional fairness metrics that require ground truth labels and thresholds, but instead propose using them in conjunction with QDD to obtain a fuller picture of real life harms in a context dependent manner [65].

I used Integrated Gradients as the explanation method for my case study. However, the choice of explanation method is potentially important, as recent research [95, 118] shows that different explanation methods often do not produce the same results, and ensembling them is superior than using any one of them in isolation.

Finally, FairCanary/QDD is not completely automated: there are still manual parameters that need to be set, like number of bins and alert sensitivity. Providing FairCanary users with guidance on how to tune the system for their use case and context will be crucial for real use cases. Additionally, all fairness monitoring systems should consider providing actionable recourse tips [109] through explanations to end-users via a carefully designed, accessible interface.

6.4 Discussion

In this chapter I have presented FairCanary, a continuous model monitoring system that offers a novel model bias quantification metric called Quantile Demographic Disparity (QDD) to help ensure model fairness over time. I have shown that QDD improves on conventional fairness metrics by not requiring prediction labels or threshold values and can be used in conjunction with other metrics to obtain a fuller picture of real-life harms. Additionally, FairCanary provides explanations for QDD by reusing the explanations for individual predictions, making it significantly faster and more suitable for continuous monitoring than previous work.

I have demonstrated the functionality of FairCanary and the usefulness of my QDD metric through examples and a synthetic case study. However, my work has limitations, such as the choice of explanation method and the need for manual parameter setting. I believe that providing guidance for users on how to tune the system for their specific use case and

context, as well as providing actionable recourse tips through explanations to end-users, will be crucial for real use cases.

Overall, I argue that ML practitioners have a professional and moral obligation to ensure that the systems they deploy do not misbehave, and monitoring systems like FairCanary should become a standard component of most, if not all, deployed ML-based systems. I hope that FairCanary (or other monitoring systems that incorporate its capabilities) will equip companies and institutions with improved tools to monitor, understand, and mitigate problems in their deployed ML systems, in real time. In turn, I hope that these capabilities will bring more equity and justice to the individual stakeholders impacted by deployed models.

Chapter 7

Conclusion

Throughout this thesis, I have explored the challenges that practitioners face when implementing fair machine learning algorithms in real-world settings. While there have been many academic papers on fair machine learning, there is a significant gap between these theoretical concepts and their practical implementation. My research has shown that theoretical fairness guarantees often do not hold up in practice and can even exacerbate unfairness for minority groups. Furthermore, I have found that a lack of thoughtful design considerations can make systems vulnerable to issues such as adversarial attacks and concept drift, which can erode fairness over time.

I dive into detailed conclusions for each paper below.

7.1 When Fair Ranking Meets Uncertain Inference

In this study I investigate the interactions between five demographic inference algorithms and the *DetConstSort* fair ranking algorithm. To ensure realism, I derive the error rates for the demographic inference algorithms from real-world datasets, and present results from controlled simulations and real-world case studies. The takeaway from my experiments is that using inferred demographic data as input to fair ranking algorithms can invalidate their fairness guarantees in ways that are (1) difficult to predict and (2) often harm vulnerable groups of people.

It would have been a positive, pragmatic result if my study found that fair ranking under uncertain inference was categorically fairer than non-fairness-aware ranking, even if it did not achieve optimal fairness. Unfortunately, this is not the case: in some instances, groups that

were not disadvantaged in the baseline, non-fairness aware ranking became disadvantaged under fair ranking due to errors in inference (e.g., Hispanic men in the Equestrians dataset).

One solution to the problem at hand is uncertainty-aware fair ranking algorithms. There are newer of fair ranking algorithms that explicitly deal with noisy attributes [142] and also adjusted fair ranking metrics that account for noise [76] that I did not evaluate in this paper since they were published after Chapter 3 was published, but I perform a similar analysis with uncertainty-aware fair classifiers in Chapter 5. As I highlight below in 7.3, uncertainty-aware algorithms are starting to become a potentially viable alternative to demographic-aware algorithms in the absence of reliable protected attribute information.

The other solution is to intentionally collect demographic data, thus avoiding inference entirely. However, this data must be collected with great care and consideration. *First*, the choices presented to people (e.g., binary gender or US Census race/ethnicity categories) constrain the groups that can ultimately benefit from fairness interventions. *Second*, designers must determine whether self-reported or perceived demographic attributes are more appropriate for their context. For example, AirBNB purposefully uses perceived demographics to identify patterns of discrimination by hosts against guests [16]. *Third*, when classifying people I must always consider the potential for reifying oppressive structures [93]. In a given ranking context, if people are reluctant to divulge demographic data or there is the potential for this data to be misused, then designers must seriously consider whether algorithmically ranking people is appropriate in the first place.

7.2 Subverting Fair Image Search with Generative Adversarial Perturbations

In this study, I develop a novel, adversarial ML attack against fair ranking algorithms, and use fairness-aware text-to-image retrieval as a case study to demonstrate my attack's effectiveness. Unfortunately, I find that my attack is very successful at subverting the fairness algorithm of the search engine—across an extensive set of attack variations—while having almost zero impact on search result relevance.

Although I present a single case study, I argue that my attack is likely to generalize. I adopt a strong threat model and demonstrate that my attacks succeed even when the attacker cannot poison training data, access the victim's whole image corpus, or know what

models are used by the victim. Thus, my attack is highly likely to succeed in cases where the threat model is more relaxed, e.g., when the fairness algorithm used by the victim is known.

Alarmingly, my work shows that an adversary can attack a fairness algorithm like FMMR *even when it does not explicitly rely on demographic inference*. Thus, it is highly likely that my attack will also succeed against any ranking algorithm that does rely on a demographic inference model, even if that model is highly accurate. I explore this possibility to the best of my ability in 4.5.4.

Above all, this work highlights that achieving demographic fairness requires high-quality demographic data [10]. Allowing an adversary to influence demographic meta-data is the underlying flaw that enables my attack to succeed. Demographic data may be sourced from data subjects themselves, with full knowledge and consent, or from human labelers [16], with the caveat that these labels themselves will need to be de-biased [220].

Like all adversarial attack research, my methods can potentially be misused by bad actors. However, this necessisates my research, since documenting vulnerabilities is a crucial first step in mitigating them. Currently, except for Shopify and LinkedIn, few services employ fair ranking systems, leaving a window of opportunity to identify attacks, raise awareness, and deploy mitigations. It is not my intention to hinder the adoption of fair ML techniques, but rather to demonstrate that fairness guarantees can be weaponized. This will energize the research community to develop mitigations, such as making models more robust and adopting high-quality sources of demographic data that are resistant to manipulation. To facilitate mitigation development without arming attackers, I release my code and data to researchers only upon request.

Prior work on adversarial ML attacks against fairness made their source code publicly available [153]. However, because attack tools are dual-use, I have opted to take a more conservative approach: I will only share source code with researchers from (1) research universities (e.g., as identified by taxonomies like the Carnegie Classification) and (2) companies that develop potentially vulnerable products. Given that my attack can be used for legitimate, black-box algorithm auditing purposes, I opt to restrict who may access my source code rather than the uses it may be put towards. In my opinion, this process will facilitate follow-up research, mitigation development, and algorithm auditing without supplying bad actors with ready-made attack tools.

Like many works in the computer vision field, I rely on images with crowdsourced and

inferred demographic labels. Both processes have been criticized for their lack of consent [82], the way they operationalize identity [180], and the harm they may cause through misidentification [21]. These problems reinforce the need for high-quality, consensual demographic data as a means to improve ethical norms and defend against adversarial ML attacks.

I believe that future work is needed to develop more robust fair ML interventions. I adopt a broad view of possible mitigations, spanning from value sensitive design [68] methods that help developers preemptively identify attack surface and plan defenses [57], to models that are hardened against adversarial perturbation techniques [7, 84], to auditing checklists [166] and tools that help developers notice and triage attacks.

7.3 When Fair Classification Meets Noisy Protected Attributes

In this study, I present benchmark results—in terms of accuracy, fairness, and stability—for 14 ML classifiers divided into four classes. I evaluated these classifiers across four datasets and varying levels of random noise in the protected attribute. Overall, I found that two classical fair classifiers (SREW and EGR), one noise-tolerant fair classifier (PRIV), and one demographic-blind fair classifier (ARL) performed consistently well across metrics on my experiments. In the future I recommend that ML researchers benchmark their own fair classifiers against these classifiers and that practitioners consider adopting them.

One surprising finding of my study was how well SREW and EGR performed in the face of noise in the protected attribute. Contrast this to noise-tolerant classifiers like MDRO—whose performance did not vary with noise but was inaccurate on some datasets—and SOFT—which was consistently inaccurate and had variable fairness in the face of noise. These results suggest that some classical fair classifiers may actually fare well in the face of noise, and that adopting more complex noise-tolerant fair classifiers may not always be necessary.

Another surprising finding of my study was how well ARL performed. As a demographic-blind fair classifier it did not have access to the sex feature at training or testing time, yet it achieved fairness performance that was comparable to demographic-aware fair classifiers on three of my datasets, and its fairness performance was noise invariant on three datasets as well. I fit linear regression models on each dataset with sex as the independent variable, but these models did not uncover any obvious proxy features for ARL to use in place of the sex feature. This speaks to the strength of the ARL algorithm's adversarial approach to learning.

On one hand, my results confirm that demographic-blind fair classifiers can achieve fairness for real-world disadvantaged groups under ecological conditions. This is positive news for practitioners who would like to adopt a fair classifier but lack (high-quality) demographic data. Demographic-blind fair classifiers may even be practical solutions to the problems I investigated in the previous two chapters, as discussed in 7.1 and 7.2. On the other hand, I still urge caution with respect to the adoption of demographic-blind fair classifiers for some further downstream practical reasons. First, determining whether a classifier like ARL will achieve acceptable performance in a given context requires thorough evaluation on a dataset that includes demographic data, as I have done here. Second, even if a demographic-blind fair classifier performs well in testing, its performance may degrade after deployment if the context changes or there is distribution drift [81]. Monitoring the health of a classifier like ARL in the field requires demographic data. In short, adopting a demographic-blind classifier does not completely obviate the need for at least some high-quality demographic data.

In general, the results of my study point to the need for further development in the areas of noise-tolerant and demographic-blind fair classifiers. By releasing my source code and data, I hope to provide a solid foundation for evaluating these novel classifiers in the future.

7.4 FairCanary: Rapid Continuous Explainable Fairness

In this work I present a novel metric called QDD that improves on conventional fairness metrics by not requiring prediction labels or threshold values (6.2.2). I utilize this metric in FairCanary, a system for performing continuous monitoring of deployed ML models. FairCanary includes all of the typical capabilities of ML monitoring systems [50]: it records inputs to and outputs from the model over time, calculates traditional measures of model performance (e.g., accuracy), allows operators to set configurable alerts if model performance changes dramatically, and calculates explanations for individual predictions using existing techniques [135, 198].

Additionally, FairCanary is able to provide explanations for QDD by reusing the explanations for individual predictions, which is (1) a capability not offered by conventional fairness metrics and (2) less computationally demanding than similar approaches from prior work [145] (6.2.3).

Through examples (Figure 2.1) and a synthetic case study (6.3), I demonstrate the functionality of FairCanary and the useful properties afforded by my QDD metric. I publicly

release the code [1] used to generate the plots in my case study.

Regardless of whether ML models are regulated to mandate audits and continuous monitoring, I argue that ML practitioners have a professional and moral obligation to ensure that the systems they deploy do not misbehave. Given that issues like drift are known to occur, and that these issues may cause unfairness and bias, I argue that monitoring systems should become a standard component of most, if not all, deployed ML-based systems.

7.5 Final Thoughts

My findings highlight the importance of a holistic approach to implementing fair machine learning algorithms that takes into account the unique characteristics of the real-world context in which they will be deployed. This includes considerations such as the quality and representativeness of the training data, the choice of fairness metrics used to evaluate the performance of the algorithm, and the potential for unintended consequences due to a lack of anticipated deployment time challenges, such as not accounting for malicious actors that can convert the fair algorithm into a tool of oppression, or a fair algorithm becoming unfair over time due to the changing nature of real world datasets. My hope is that by adopting a more practice oriented approach to implementing fair machine learning, practitioners can help ensure that these algorithms achieve their intended goals of promoting fairness and equity for all individuals.

7.6 Future Research

I would like to continue on my path of exposing shortfalls of real world sociotechnical systems, and designing solutions for fairer ML that works in both theory and practice.

Adversarial defenses against fairness attacks: I have already shown that it is possible to attack fair ML models via adversarial attacks to cause them to become unfair [80]. I plan to investigate solutions to such an attack, through a mix of methods such as adversarial training, cloaking defenses, and online learning with bias annotations, to find a solution that is both computationally scalable and robust against not just black box attacks, but hopefully

[1] `https://github.com/fiddler-labs/faircanary`

also stronger white box attacks.

Machine unlearning to remove problematic training data: I have taken a keen interest in the rampant phenomenon of massive models being trained on data scraped without consent—this includes both Large Language Models (LLMs) and also Text-to-Image Generation models. These models have been shown to be riddled with sexist, racist, toxic, or factually incorrect content in their outputs—that the model creators themselves believe are too difficult to correct for. Text generation models such as Github Copilot[2] and image generation models such as DALL.E[3] also presents the deeply concerning aspect of models trained on unconsenting individuals' data and then using them for profit [78]. Retraining such massive models from scratch is a prohibitively expensive task, and I believe building on early research work in machine unlearning—to build a tool for individuals to request model owners to "forget" their training data and respect their IP—is the more feasible technical solution to this predicament. Machine unlearning is largely unexplored, and I plan to work with both machine learning scientists in academia and companies who are in the business of commercializing these models to find actionable, cheaply computable, and scalable unlearning solutions.

Injection of bias via human stakeholders: Humans can impact the behavior of a ML pipeline in at least two ways: bias added by annotators in the annotation stage before model training, and personal bias of the decision makers who are in charge of converting predictions into outcomes. While creating ML datasets, it would be interesting to measure annotator perspective and cultural bias in annotations—whether the gaps in perception are different for different demographic groups. Normatively speaking, I would like to think more deeply about whether a person's self disclosed attributes are the more important factor in corrective fairness techniques than what the majority of annotators think is their label. In terms of decision making bias, I would like to delve deeper into how humans in charge of making final decisions might subvert algorithmic fairness interventions—for instance, human recruiters on the other end of a fair candidate ranking system. This would involve working with people in human computer interaction, labor economics, and psychology.

[2] `https://github.com/features/copilot`

[3] `https://openai.com/dall-e-2/`

Developing actionable policy: I would like my solution-focused research approach to inform better regulation of AI/ML practice. I plan to initiate dialogue with policymakers and regulatory agencies worldwide, such as the FTC and CFPB in the US or analogous bodies within the EU Commission, to help fine tune policy. I would also set aside time for my research group to respond during public comment periods for any new regulations proposed by regulatory agencies. My belief is that my work will be able to highlight specific technical interventions that model operators can take to implement AI responsibly, and such specific regulation will avoid cases of escaped accountability due to vague regulatory language. Ultimately, I would like my research to shape policy and have lasting positive impact on society.

Bibliography

[1] Using publicly available information to proxy for unidentified race and ethnicity: A methodology and assessment. Consumer Financial Protection Bureau, Summer 2014.

[2] 116th Congress (2019-2020). H.r.2231 - algorithmic accountability act of 2019. `https://www.congress.gov/bill/116th-congress/house-bill/2231`.

[3] D. Adjaye-Gbewonyo, R. A. Bednarczyk, R. L. Davis, and S. B. Omer. Using the bayesian improved surname geocoding method (bisg) to create a working classification of race and ethnicity in a diverse managed care population: a validation study. *Health services research*, 49(1):268–283, 2014.

[4] A. Agarwal, A. Beygelzimer, M. Dudík, J. Langford, and H. Wallach. A reductions approach to fair classification. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2018.

[5] A. Agarwal, M. Dudík, and Z. S. Wu. Fair regression: Quantitative definitions and reduction-based algorithms. In *International Conference on Machine Learning*, pages 120–129. PMLR, 2019.

[6] F. AI. How we're using Fairness Flow to help build AI that works better for everyone. Facebook AI, Mar. 2021. `https://ai.facebook.com/blog/how-were-using-fairness-flow-to-help-build-ai-that-works-better-for-everyone/`.

[7] N. Akhtar, J. Liu, and A. Mian. Defense against universal adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3389–3398, 2018.

[8] M. Andrus, E. Spitzer, J. Brown, and A. Xiang. What we can't measure, we can't understand: Challenges to demographic data procurement in the pursuit of fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 249–260, New York, NY, USA, 2021. Association for Computing Machinery.

[9] M. Andrus, E. Spitzer, J. Brown, and A. Xiang. What we can't measure, we can't understand: Challenges to demographic data procurement in the pursuit of fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 249–260, 2021.

[10] M. Andrus, E. Spitzer, J. Brown, and A. Xiang. What we can't measure, we can't understand: Challenges to demographic data procurement in the pursuit of fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 249–260, New York, NY, USA, 2021. Association for Computing Machinery.

[11] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias: There's software used across the country to predict future criminals and it's biased against blacks. 2016. *URL https://www. propublica. org/article/machine-bias-risk-assessments-in-criminal-sentencing*, 2019.

[12] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *International Conference on Machine Learning*, pages 274–283. PMLR, 2018.

[13] P. Awasthi, M. Kleindessner, and J. Morgenstern. Effectiveness of equalized odds for fair classification under imperfect group information. *arXiv preprint arXiv:1906.03284*, 2019.

[14] S. Barocas, A. Guo, E. Kamar, J. Krones, M. R. Morris, J. W. Vaughan, D. Wadsworth, and H. Wallach. Designing disaggregated evaluations of ai systems: Choices, considerations, and tradeoffs. *arXiv preprint arXiv:2103.06076*, pages 368–378, 2021.

[15] S. Barocas and A. D. Selbst. Big data's disparate impact. *Calif. L. Rev.*, 104:671, 2016.

[16] S. Basu, R. Berman, A. Bloomston, J. Campbell, A. Diaz, N. Era, B. Evans, S. Palkar, and S. Wharton. Measuring discrepancies in Airbnb guest acceptance rates using anonymized demographic data. AirBNB, Apr. 2020. `https://news.airbnb.com/wp-content/uploads/sites/4/2020/06/Project-Lighthouse-Airbnb-2020-06-12.pdf`.

[17] M. Behjati, S.-M. Moosavi-Dezfooli, M. S. Baghshah, and P. Frossard. Universal adversarial attacks on text classifiers. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7345–7349. IEEE, 2019.

[18] R. K. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, et al. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*, 2018.

[19] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, 2018.

[20] E. M. Bender, T. Gebru, A. McMillan-Major, and S. Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021.

[21] C. L. Bennett, C. Gleason, M. K. Scheuerman, J. P. Bigham, A. Guo, and A. To. "it's complicated": Negotiating accessibility and (mis)representation in image descriptions of race, gender, and disability. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. Association for Computing Machinery.

[22] R. K. Berg. Equal employment opportunity under the civil rights act of 1964. *Brook. L. Rev.*, 31:62, 1964.

[23] R. Berk, H. Heidari, S. Jabbari, M. Joseph, M. Kearns, J. Morgenstern, S. Neel, and A. Roth. A convex framework for fair regression. *arXiv preprint arXiv:1706.02409*, 2017.

[24] A. Beutel, J. Chen, T. Doshi, H. Qian, L. Wei, Y. Wu, L. Heldt, Z. Zhao, L. Hong, E. H. Chi, and C. Goodrow. Fairness in recommendation ranking through pairwise comparisons. In *KDD*, 2019.

[25] U. Bhatt, A. Xiang, S. Sharma, A. Weller, A. Taly, Y. Jia, J. Ghosh, R. Puri, J. M. Moura, and P. Eckersley. Explainable machine learning in deployment. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 648–657, 2020.

[26] A. J. Biega, K. P. Gummadi, and G. Weikum. Equity of attention: Amortizing individual fairness in rankings. In *The 41st international acm sigir conference on research & development in information retrieval*, pages 405–414, 2018.

[27] A. Bifet and R. Gavalda. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 443–448. SIAM, 2007.

[28] A. Binder, G. Montavon, S. Lapuschkin, K.-R. Müller, and W. Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*, pages 63–71. Springer, 2016.

[29] E. Black, S. Yeom, and M. Fredrikson. Fliptest: fairness testing via optimal transport. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 111–121, 2020.

[30] M. Bogen and A. Rieke. Help wanted: An examination of hiring algorithms, equity, and bias. 2018.

[31] M. Bogen, A. Rieke, and S. Ahmed. Awareness in practice: tensions in access to sensitive attribute data for antidiscrimination. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 492–500, 2020.

[32] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pages 4349–4357, 2016.

[33] E. Breck, N. Polyzotis, S. Roy, S. Whang, and M. Zinkevich. Data validation for machine learning. In *MLSys*, 2019.

[34] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer. Adversarial patch. *arXiv preprint arXiv:1712.09665*, 2017.

[35] M.-E. Brunet, C. Alkalay-Houlihan, A. Anderson, and R. Zemel. Understanding the origins of bias in word embeddings. In *International Conference on Machine Learning*, pages 803–811, 2019.

[36] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, pages 77–91. PMLR, 2018.

[37] C. F. P. Bureau. Using publicly available information to proxy for unidentified race and ethnicity. *Report available at http://files. consumerfinance. gov/f/201409_ cfpb_ report_ proxy-methodology. pdf*, 2014.

[38] F. Calmon, D. Wei, B. Vinzamuri, K. Natesan Ramamurthy, and K. R. Varshney. Optimized pre-processing for discrimination prevention. *Advances in neural information processing systems*, 30, 2017.

[39] J. Carbonell and J. Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336, 1998.

[40] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[41] L. E. Celis, L. Huang, V. Keswani, and N. K. Vishnoi. Fair classification with noisy protected attributes: A framework with provable guarantees. In *International Conference on Machine Learning*, pages 1349–1361. PMLR, 2021.

[42] L. E. Celis and V. Keswani. Implicit diversity in image summarization. *Proceedings of the ACM on Human-Computer Interaction*, 4(CSCW2):1–28, 2020.

[43] L. E. Celis, A. Mehrotra, and N. K. Vishnoi. Fair classification with adversarial perturbations. *arXiv preprint arXiv:2106.05964*, 2021.

[44] L. E. Celis, D. Straszak, and N. K. Vishnoi. Ranking with fairness constraints. In *45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

[45] H. Chang, T. D. Nguyen, S. K. Murakonda, E. Kazemi, and R. Shokri. On adversarial bias and the robustness of fair machine learning. *arXiv preprint arXiv:2006.08669*, 2020.

[46] J. Chen, N. Kallus, X. Mao, G. Svacha, and M. Udell. Fairness under unawareness: Assessing disparity when protected class is unobserved. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 339–348, 2019.

[47] E. Commission. Proposal for a regulation laying down harmonised rules on artificial intelligence (artificial intelligence act). `https://digital-strategy.ec.europa.eu/en/library/proposal-regulation-laying-down-harmonised-rules-artificial-intelligence-artificial-intelligence`.

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

[49] S. Corbett-Davies, E. Pierson, A. Feller, S. Goel, and A. Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd acm sigkdd international conference on knowledge discovery and data mining*, pages 797–806, 2017.

[50] J. Czakon. Best tools to do ml model monitoring. 2022.

[51] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song. Adversarial attack on graph structured data. In *International conference on machine learning*, pages 1115–1124. PMLR, 2018.

[52] S. Das, M. Donini, J. Gelman, K. Haas, M. Hardt, J. Katzman, K. Kenthapadi, P. Larroy, P. Yilmaz, and M. B. Zafar. Fairness measures for machine learning in finance. *The Journal of Financial Data Science*, 3(4):33–64, 2021.

[53] A. Dash, A. Chakraborty, S. Ghosh, A. Mukherjee, and K. P. Gummadi. When the umpire is also a player: Bias in private label product recommendations on e-commerce marketplaces. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 873–884, 2021.

[54] A. Datta, S. Sen, and Y. Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE symposium on security and privacy (SP)*, pages 598–617. IEEE, 2016.

[55] A. M. Davani, M. Díaz, and V. Prabhakaran. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110, 2022.

[56] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[57] T. Denning, B. Friedman, and T. Kohno. The security cards: A security threat brainstorming toolkit. University of Washington, 2013. `https://securitycards.cs.washington.edu/`.

[58] F. Ding, M. Hardt, J. Miller, and L. Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, 34:6478–6490, 2021.

[59] D. M. dos Reis, P. Flach, S. Matwin, and G. Batista. Fast unsupervised online drift detection using incremental kolmogorov-smirnov test. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1545–1554, 2016.

[60] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226, 2012.

[61] Equal Employment Opportunity Commission, Civil Service Commission, et al. Uniform guidelines on employee selection procedures. *Federal Register*, 43(166):38290–38315, 1978.

[62] EY. Assessing and mitigating unfairness in credit models with fairlearn. `https://www.ey.com/en_ca/financial-services/assessing-and-mitigating-unfairness-in-credit-models`, 2020. [Accessed: March 16th, 2023].

[63] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[64] U. O. for Artificial Intelligence. Ethics, transparency and accountability framework for automated decision-making. `https://www.gov.uk/government/publications/ethics-transparency-and-accountability-framework-for-automated-decision-making`.

[65] C. for Data Science and P. Policy. Aequitas: Fairness tree. `http://www.datasciencepublicpolicy.org/projects/aequitas/`.

[66] J. R. Foulds, R. Islam, K. N. Keya, and S. Pan. An intersectional definition of fairness. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 1918–1921. IEEE, 2020.

[67] S. A. Friedler, C. Scheidegger, S. Venkatasubramanian, S. Choudhary, E. P. Hamilton, and D. Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 329–338, 2019.

[68] B. Friedman and D. Hendry. *Value sensitive design: shaping technology with moral imagination*. MIT Press, 2019.

[69] J. Gama, R. Sebastiao, and P. P. Rodrigues. On evaluating stream learning algorithms. *Machine learning*, 90(3):317–346, 2013.

[70] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

[71] N. Garg, L. Schiebinger, D. Jurafsky, and J. Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644, 2018.

[72] R. S. Geiger, K. Yu, Y. Yang, M. Dai, J. Qiu, R. Tang, and J. Huang. Garbage in, garbage out? do machine learning application papers in social computing report where human-labeled training data comes from? In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 325–336, 2020.

[73] G. Geigle, J. Pfeiffer, N. Reimers, I. Vulić, and I. Gurevych. Retrieve fast, rerank smart: Cooperative and joint approaches for improved cross-modal retrieval. *arXiv preprint*, abs/2103.11920, 2021.

[74] S. C. Geyik, S. Ambler, and K. Kenthapadi. Fairness-aware ranking in search & recommendation systems with application to linkedin talent search. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2221–2231, 2019.

[75] S. Ghanta, S. Subramanian, L. Khermosh, S. Sundararaman, H. Shah, Y. Goldberg, D. S. Roselli, and N. Talagala. ML health: Fitness tracking for production models. *CoRR*, abs/1902.02808, 2019.

[76] A. Ghazimatin, M. Kleindessner, C. Russell, Z. Abedjan, and J. Golebiowski. Measuring fairness of rankings under noisy sensitive information. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 2263–2279, 2022.

[77] A. Ghosh, R. Dutt, and C. Wilson. When fair ranking meets uncertain inference. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pages 1033–1043, New York, NY, USA, 2021. Association for Computing Machinery.

[78] A. Ghosh and G. Fossas. Can there be art without an artist? *arXiv preprint arXiv:2209.07667*, 2022.

[79] A. Ghosh, L. Genuit, and M. Reagan. Characterizing intersectional group fairness with worst-case comparisons, 09 Feb 2021.

[80] A. Ghosh, M. Jagielski, and C. Wilson. Subverting fair image search with generative adversarial perturbations. In *FAccT '22: 2022 ACM Conference on Fairness, Accountability, and Transparency, Seoul, Republic of Korea, June 21 - 24, 2022*, pages 637–650. ACM, 2022.

[81] A. Ghosh, A. Shanbhag, and C. Wilson. Faircanary: Rapid continuous explainable fairness. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*, pages 307–316, 2022.

[82] W. Gies, J. Overby, N. Saraceno, J. Frome, E. York, and A. Salman. Restricting data sharing and collection of facial recognition data by the consent of the user: A systems analysis. In *2020 Systems and Information Engineering Design Symposium (SIEDS)*, pages 1–6, 2020.

[83] N. Goel, M. Yaghini, and B. Faltings. Non-discriminatory machine learning through convex fairness criteria. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pages 116–116, 2018.

[84] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[85] B. Goodman and S. Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

[86] D. Goodwin. Top Google Result Gets 36.4% of Clicks [Study]. Search Engine Watch, Apr. 2011. `https://www.searchenginewatch.com/2011/04/21/top-google-result-gets-36-4-of-clicks-study/`.

[87] V. Grari, B. Ruf, S. Lamprier, and M. Detyniecki. Fairness-aware neural réyni minimization for continuous features. *arXiv preprint arXiv:1911.04929*, 2019.

[88] A. Hanna, E. Denton, A. Smart, and J. Smith-Loud. Towards a critical race methodology in algorithmic fairness. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 501–512, 2020.

[89] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *arXiv preprint arXiv:1610.02413*, pages 3315–3323, 2016.

[90] T. Hashimoto, M. Srivastava, H. Namkoong, and P. Liang. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pages 1929–1938. PMLR, 2018.

[91] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[92] X. He, Z. He, X. Du, and T.-S. Chua. Adversarial personalized ranking for recommendation. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 355–364, 2018.

[93] A. L. Hoffmann. Terms of inclusion: Data, discourse, violence. *New Media & Society*, Sept. 2020.

[94] B. Hofstra, V. V. Kulkarni, S. M.-N. Galvez, B. He, D. Jurafsky, and D. A. McFarland. The diversity–innovation paradox in science. *Proceedings of the National Academy of Sciences*, 117(17):9284–9291, 2020.

[95] S. Hooker, D. Erhan, P.-J. Kindermans, and B. Kim. A benchmark for interpretability methods in deep neural networks. *Advances in neural information processing systems*, 32, 2019.

[96] M. Hort, J. M. Zhang, F. Sarro, and M. Harman. Fairea: A model behaviour mutation approach to benchmarking bias mitigation methods. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*, 2021.

[97] L. Hu and I. Kohler-Hausmann. What's sex got to do with machine learning. *arXiv preprint arXiv:2006.01770*, 2020.

[98] L. Huang and N. K. Vishnoi. Stable and fair classification. *arXiv preprint arXiv:1902.07823*, 2019.

[99] M. Jagielski, G. Severi, N. P. Harger, and A. Oprea. Subpopulation data poisoning attacks. *arXiv preprint arXiv:2006.14026*, 2020.

[100] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[101] R. Jiang, A. Pacchiano, T. Stepleton, H. Jiang, and S. Chiappa. Wasserstein fair classification. In *Uncertainty in Artificial Intelligence*, pages 862–872. PMLR, 2020.

[102] S. Jiang, L. Chen, A. Mislove, and C. Wilson. On ridesharing competition and accessibility: Evidence from uber, lyft, and taxi. In *Proc. of WWW*, 2018.

[103] E. S. Jo and T. Gebru. Lessons from archives: Strategies for collecting sociocultural data in machine learning. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 306–316, 2020.

[104] N. Kallus, X. Mao, and A. Zhou. Assessing algorithmic fairness with unobserved protected class using data combination. *Management Science*, 68(3):1959–1981, 2022.

[105] F. Kamiran and T. Calders. Data preprocessing techniques for classification without discrimination. *Knowledge and information systems*, 33(1):1–33, 2012.

[106] F. Kamiran, A. Karim, and X. Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929. IEEE, 2012.

[107] T. Kamishima, S. Akaho, H. Asoh, and J. Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 35–50. Springer, 2012.

[108] C. Karako and P. Manggala. Using image fairness representations in diversity-based re-ranking for recommendations. In *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*, pages 23–28, 2018.

[109] A.-H. Karimi, G. Barthe, B. Schölkopf, and I. Valera. A survey of algorithmic recourse: definitions, formulations, solutions, and prospects. *arXiv preprint arXiv:2010.04050*, 2020.

[110] K. Kärkkäinen and J. Joo. Fairface: Face attribute dataset for balanced race, gender, and age. *arXiv preprint arXiv:1908.04913*, pages 1548–1558, 2019.

[111] N. Kilbertus, P. J. Ball, M. J. Kusner, A. Weller, and R. Silva. The sensitivity of counterfactual fairness to unmeasured confounding. In *Uncertainty in artificial intelligence*, pages 616–626. PMLR, 2020.

[112] Y. Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, Oct. 2014. Association for Computational Linguistics.

[113] J. Kleinberg, S. Mullainathan, and M. Raghavan. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807*, 2016.

[114] A. Knott. Moving towards responsible government use of ai in new zealand). `https://digitaltechitp.nz/2021/03/22/moving-towards-responsible-government-use-of-ai-in-new-zealand/`.

[115] A. Kolmogorov. Sulla determinazione empirica di una lgge di distribuzione. *Inst. Ital. Attuari, Giorn.*, 4:83–91, 1933.

[116] C. Kuhlman, M. VanValkenburg, and E. Rundensteiner. Fare: Diagnostics for fair ranking using pairwise error metrics. In *The World Wide Web Conference*, pages 2936–2942, 2019.

[117] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler. Problems with shapley-value-based explanations as feature importance measures. In *International Conference on Machine Learning*, pages 5491–5500. PMLR, 2020.

[118] I. E. Kumar, S. Venkatasubramanian, C. Scheidegger, and S. Friedler. Problems with shapley-value-based explanations as feature importance measures. In H. D. III and A. Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5491–5500. PMLR, 13–18 Jul 2020.

[119] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016.

[120] M. J. Kusner, J. R. Loftus, C. Russell, and R. Silva. Counterfactual fairness. *arXiv preprint arXiv:1703.06856*, 2017.

[121] P. Lahoti, A. Beutel, J. Chen, K. Lee, F. Prost, N. Thain, X. Wang, and E. Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.

[122] A. Lamy, Z. Zhong, A. K. Menon, and N. Verma. Noise-tolerant fair classification. *Advances in neural information processing systems*, 32, 2019.

[123] K. Lerman, A. Plangprasopchok, and C. Wong. Personalizing image search results on flickr. *Intelligent Information Personalization*, 2007.

[124] J. Leskovec and E. Horvitz. Planetary-scale views on a large instant-messaging network. In *Proceedings of the 17th International Conference on World Wide Web*, WWW '08, page 915–924, New York, NY, USA, 2008. Association for Computing Machinery.

[125] J. Li, R. Ji, H. Liu, X. Hong, Y. Gao, and Q. Tian. Universal perturbation attack against image retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4899–4908, 2019.

[126] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, et al. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020.

[127] J. Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151, 1991.

[128] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[129] LinkedIn. Lift: A scalable framework for measuring fairness in ml applications. `https://github.com/linkedin/LiFT`, 2021. [Accessed: March 16th, 2023].

[130] L. T. Liu, S. Dean, E. Rolf, M. Simchowitz, and M. Hardt. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, pages 3150–3158. PMLR, 2018.

[131] Y. Liu, X. Chen, C. Liu, and D. Song. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770*, 2016.

[132] Z. Liu, Z. Zhao, and M. Larson. Who's afraid of adversarial queries? the impact of image modifications on content-based image retrieval. In *Proceedings of the 2019 on International Conference on Multimedia Retrieval*, ICMR '19, page 306–314, New York, NY, USA, 2019. Association for Computing Machinery.

[133] J. R. Loftus, C. Russell, M. J. Kusner, and R. Silva. Causal reasoning for algorithmic fairness. *arXiv preprint arXiv:1805.05859*, 2018.

[134] J. Lu, D. Batra, D. Parikh, and S. Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019.

[135] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

[136] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[137] H. B. Mann and D. R. Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60, 1947.

[138] C. D. Manning, P. Raghavan, and H. Schütze. *Evaluation in information retrieval*, chapter 8, pages 151–175. Cambridge University Press, 2009.

[139] A. F. Markus, J. A. Kors, and P. R. Rijnbeek. The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies. *Journal of Biomedical Informatics*, 113:103655, 2021.

[140] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. *arXiv preprint arXiv:1908.09635*, 2019.

[141] N. Mehrabi, M. Naveed, F. Morstatter, and A. Galstyan. Exacerbating algorithmic bias through fairness attacks. *arXiv preprint arXiv:2012.08723*, 2020.

[142] A. Mehrotra and N. Vishnoi. Fair ranking with noisy protected attributes. *Advances in Neural Information Processing Systems*, 35:31711–31725, 2022.

[143] A. K. Menon and R. C. Williamson. The cost of fairness in binary classification. In *Conference on Fairness, Accountability and Transparency*, pages 107–118. PMLR, 2018.

[144] L. Merrick and A. Taly. The explanation game: Explaining machine learning models using shapley values. In *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*, pages 17–38. Springer, 2020.

[145] A. Miroshnikov, K. Kotsiopoulos, R. Franks, and A. R. Kannan. Wasserstein-based fairness interpretability framework for machine learning models. *arXiv preprint arXiv:2011.03156*, 2020.

[146] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard. Universal adversarial perturbations, 2017.

[147] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016.

[148] M. Morik, A. Singh, J. Hong, and T. Joachims. Controlling fairness and bias in dynamic learning-to-rank. *arXiv preprint arXiv:2005.14713*, 2020.

[149] H. Mozannar, M. Ohannessian, and N. Srebro. Fair learning with private demographic data. In *International Conference on Machine Learning*, pages 7066–7075. PMLR, 2020.

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

[151] A. Mullick, S. Ghosh, R. Dutt, A. Ghosh, and A. Chakraborty. Public sphere 2.0: Targeted commenting in online news media. In *European Conference on Information Retrieval*, pages 180–187. Springer, 2019.

[152] R. Nabi and I. Shpitser. Fair inference on outcomes. In *Proceedings of the... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, volume 2018, page 1931. NIH Public Access, 2018.

[153] V. Nanda, S. Dooley, S. Singla, S. Feizi, and J. P. Dickerson. Fairness through robustness: Investigating robustness disparity in deep learning. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 466–477, 2021.

[154] P. Nandy, C. Diciccio, D. Venugopalan, H. Logan, K. Basu, and N. E. Karoui. Achieving fairness via post-processing in web-scale recommender systems, 2021.

[155] A. Narayanan. 21 fairness definitions and their politics. `https://fairmlbook.org/tutorial2.html`.

[156] J. Nielsen. Usability 101: introduction to usability. jakob nielsen's alertbox, 2003.

[157] D. Nigenda, Z. Karnin, M. B. Zafar, R. Ramesha, A. Tan, M. Donini, and K. Kenthapadi. Amazon sagemaker model monitor: A system for real-time insights into deployed machine learning models. *arXiv preprint arXiv:2111.13657*, 2021.

[158] K. Nishida, S. Shimada, S. Ishikawa, and K. Yamauchi. Detecting sudden concept drift with knowledge of human behavior. In *2008 IEEE International Conference on Systems, Man and Cybernetics*, pages 3261–3267, 2008.

[159] G. of Canada. Responsible use of artificial intelligence (ai). `https://www.canada.ca/en/government/system/digital-government/digital-government-innovations/responsible-use-ai.html`.

[160] N. Papernot, F. Faghri, N. Carlini, I. Goodfellow, R. Feinman, A. Kurakin, C. Xie, Y. Sharma, T. Brown, A. Roy, et al. Technical report on the cleverhans v2. 1.0 adversarial examples library. *arXiv preprint arXiv:1610.00768*, 2016.

[161] F. Pinto, M. O. Sampaio, and P. Bizarro. Automatic model monitoring for data streams. *arXiv preprint arXiv:1908.04240*, 2019.

[162] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger. On fairness and calibration. *Advances in neural information processing systems*, 30, 2017.

[163] O. Poursaeed, I. Katsman, B. Gao, and S. Belongie. Generative adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4422–4431, 2018.

[164] M. Raghavan, S. Barocas, J. Kleinberg, and K. Levy. Mitigating Bias in Algorithmic Hiring: Evaluating Claims and Practices. In *Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (FAT*)*, 2020.

[165] I. D. Raji, A. Smart, R. N. White, M. Mitchell, T. Gebru, B. Hutchinson, J. Smith-Loud, D. Theron, and P. Barnes. Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 33–44, 2020.

[166] I. D. Raji, A. Smart, R. N. White, M. Mitchell, T. Gebru, B. Hutchinson, J. Smith-Loud, D. Theron, and P. Barnes. Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. 2020.

[167] N. Raval and M. Verma. One word at a time: adversarial attacks on retrieval models. *arXiv preprint arXiv:2008.02197*, 2020.

[168] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015.

[169] M. T. Ribeiro, S. Singh, and C. Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[170] R. E. Robertson, D. Lazer, and C. Wilson. Auditing the personalization and composition of politically-related search engine results pages. In *Proc. of WWW*, volume 2, pages 1–22. ACM New York, NY, USA, 2018.

[171] A. Romanov, M. De-Arteaga, H. Wallach, J. Chayes, C. Borgs, A. Chouldechova, S. Geyik, K. Kenthapadi, A. Rumshisky, and A. T. Kalai. What's in a name? reducing bias in bios without access to protected attributes. *arXiv preprint arXiv:1904.05233*, 2019.

[172] B. S., D. M., E. R., H. D., L. R., M. V., and S. M. Fairlearn: A toolkit for assessing and improving fairness in ai. *Proceedings of Machine Learning Research*, 120:1–8, 2020.

[173] J. A. Sáez, M. Galar, J. Luengo, and F. Herrera. Tackling the problem of classification with noisy data using multiple classifier systems: Analysis of the performance and robustness. *Information Sciences*, 247:1–20, 2013.

[174] M. Salganicoff. Tolerating concept and sampling shift in lazy learning using prediction error context switching. In *Lazy learning*, pages 133–155. Springer, 1997.

[175] N. Sambasivan, E. Arnesen, B. Hutchinson, T. Doshi, and V. Prabhakaran. Re-imagining algorithmic fairness in india and beyond. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 315–328, 2021.

[176] L. Santamaría and H. Mihaljević. Comparison and benchmark of name-to-gender inference services. *PeerJ Computer Science*, 4:e156, 2018.

[177] P. Sapiezynski, A. Ghosh, L. Kaplan, A. Mislove, and A. Rieke. Algorithms that" don't see color": Comparing biases in lookalike and special ad audiences. *arXiv preprint arXiv:1912.07579*, 2019.

[178] P. Sapiezynski, W. Zeng, R. E Robertson, A. Mislove, and C. Wilson. Quantifying the impact of user attentionon fair group representation in ranked lists. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 553–562, 2019.

[179] S. Schelter, F. Biessmann, T. Januschowski, D. Salinas, S. Seufert, and G. Szarvas. On challenges in machine learning model management. 2018.

[180] M. K. Scheuerman, K. Wade, C. Lustig, and J. R. Brubaker. How we've taught algorithms to see identity: Constructing race and gender in image databases for facial analysis. 4(CSCW1), may 2020.

[181] D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, and D. Dennison. Hidden technical debt in machine learning systems. *Advances in neural information processing systems*, 28, 2015.

[182] A. Selbst and J. Powles. "meaningful information" and the right to explanation. In *Conference on Fairness, Accountability and Transparency*, pages 48–48. PMLR, 2018.

[183] A. D. Selbst, D. Boyd, S. A. Friedler, S. Venkatasubramanian, and J. Vertesi. Fairness and abstraction in sociotechnical systems. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 59–68, 2019.

[184] S. I. Serengil and A. Ozpinar. Lightface: A hybrid deep face recognition framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–5. IEEE, 2020.

[185] A. Shafahi, W. R. Huang, M. Najibi, O. Suciu, C. Studer, T. Dumitras, and T. Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *arXiv preprint arXiv:1804.00792*, 2018.

[186] A. Shafahi, M. Najibi, A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein. Adversarial training for free! *arXiv preprint arXiv:1904.12843*, 2019.

[187] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao. Fawkes: Protecting privacy against unauthorized deep learning models. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 1589–1604, 2020.

[188] A. Shanbhag, A. Ghosh, and J. Rubin. Unified shapley framework to explain prediction drift. *arXiv preprint arXiv:2102.07862*, 2021.

[189] A. Shrikumar, P. Greenside, and A. Kundaje. Learning important features through propagating activation differences. In *International conference on machine learning*, pages 3145–3153. PMLR, 2017.

[190] A. Singh and T. Joachims. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2219–2228, 2018.

[191] N. V. Smirnov. On the estimation of the discrepancy between empirical curves of distribution for two independent samples. *Bull. Math. Univ. Moscou*, 2(2):3–14, 1939.

[192] D. Solans, B. Biggio, and C. Castillo. Poisoning attacks on algorithmic fairness. *arXiv preprint arXiv:2004.07401*, 2020.

[193] G. Sood and S. Laohaprapanon. Predicting race and ethnicity from the sequence of characters in a name, 2018.

[194] K. O. Stanley. Learning concept drift with a committee of decision trees. *Informe técnico: UT-AI-TR-03-302, Department of Computer Sciences, University of Texas at Austin, USA*, 2003.

[195] B. Strack, J. P. DeShazo, C. Gennings, J. L. Olmo, S. Ventura, K. J. Cios, and J. N. Clore. Impact of hba1c measurement on hospital readmission rates: analysis of 70,000 clinical database patient records. *BioMed research international*, 2014, 2014.

[196] L. Stryjewski. 40 years of boxplots. 2010.

[197] T. Sühr, S. Hilgard, and H. Lakkaraju. Does fair ranking improve minority outcomes? understanding the interplay of human and algorithmic biases in online hiring. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 989–999, 2021.

[198] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

[199] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[200] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[201] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[202] F. Tramer, N. Carlini, W. Brendel, and A. Madry. On adaptive attacks to adversarial example defenses. *arXiv preprint arXiv:2002.08347*, 2020.

[203] F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel. The space of transferable adversarial examples. *arXiv*, 2017.

[204] A. Turner, D. Tsipras, and A. Madry. Clean-label backdoor attacks. 2018.

[205] M. A. O. Vasilescu and D. Terzopoulos. Multilinear image analysis for facial recognition. In *Object recognition supported by user interaction for service robots*, volume 2, pages 511–514. IEEE, 2002.

[206] C. Villani. The wasserstein distances. In *Optimal transport*, pages 93–111. Springer, 2009.

[207] Y. Vorobeychik and M. Kantarcioglu. Adversarial machine learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 12(3):1–169, 2018.

[208] S. Wachter, B. Mittelstadt, and C. Russell. Bias preservation in machine learning: The legality of fairness metrics under eu non-discrimination law. *West Virginia Law Review, Forthcoming*, 2021.

[209] S. Wang, W. Guo, H. Narasimhan, A. Cotter, M. Gupta, and M. Jordan. Robust optimization for fairness with noisy protected groups. *Advances in neural information processing systems*, 33:5190–5203, 2020.

[210] L. F. Wightman. Lsac national longitudinal bar passage study. lsac research report series. 1998.

[211] C. Wilson, A. Ghosh, S. Jiang, A. Mislove, L. Baker, J. Szary, K. Trindel, and F. Polli. Building and auditing fair algorithms: A case study in candidate screening. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 666–677, 2021.

[212] Z. Wu, S.-N. Lim, L. S. Davis, and T. Goldstein. Making an invisibility cloak: Real world adversarial attacks on object detectors. In *European Conference on Computer Vision*, pages 1–17. Springer, 2020.

[213] K. Yang and J. Stoyanovich. Measuring fairness in ranked outputs. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, pages 1–6, 2017.

[214] J. Ye, S. Han, Y. Hu, B. Coskun, M. Liu, H. Qin, and S. Skiena. Nationality classification using name embeddings. In *Proceedings of the 2017 ACM on Conference on Information*

and Knowledge Management, CIKM '17, page 1897–1906, New York, NY, USA, 2017. Association for Computing Machinery.

[215] M. B. Zafar, I. Valera, M. G. Rogriguez, and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pages 962–970. PMLR, 2017.

[216] M. Zehlike, F. Bonchi, C. Castillo, S. Hajian, M. Megahed, and R. Baeza-Yates. Fa* ir: A fair top-k ranking algorithm. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1569–1578, 2017.

[217] M. Zehlike and C. Castillo. Reducing disparate exposure in ranking: A learning to rank approach. In *Proceedings of The Web Conference 2020*, pages 2849–2855, 2020.

[218] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. In *International conference on machine learning*, pages 325–333. PMLR, 2013.

[219] 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*, pages 335–340, 2018.

[220] D. Zhao, A. Wang, and O. Russakovsky. Understanding and evaluating racial biases in image captioning. In *International Conference on Computer Vision (ICCV)*, 2021.

[221] M. Zhou, Z. Niu, L. Wang, Q. Zhang, and G. Hua. Adversarial ranking attack and defense. *arXiv preprint arXiv:2002.11293*, 2020.

[222] I. Žliobaite. Change with delayed labeling: When is it detectable? In *2010 IEEE International Conference on Data Mining Workshops*, pages 843–850. IEEE, 2010.

List of Figures

List of Tables