Algorithmic Fairness in Sequential Decision Making

by

Yi Sun

Submitted to the Institute for Data, Systems and Society
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Social and Engineering Systems

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2023

Author . . . . . .
Institute for Data, Systems and Society
October, 2022

Certified by . . . . . .
Kalyan Veeramachaneni
Principal Research Scientist
Thesis Supervisor

Accepted by . . . . . .
Fotini Christia
Program Chair, Social and Engineering Systems Doctoral Program

Algorithmic Fairness in Sequential Decision Making

by

Yi Sun

Submitted to the Institute for Data, Systems and Society
on October, 2022, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Social and Engineering Systems

Abstract

Machine learning algorithms have been used in a wide range of applications, and there are growing concerns about the potential biases of those algorithms. While many solutions have been proposed for addressing biases in predictions from an algorithm, there is still a gap in translating predictions to a justified decision. Moreover, even a justified and fair decision could lead to undesirable consequences when decisions create a feedback effect. While numerous solutions have been proposed for achieving fairness in one-shot decision-making, there is a gap in investigating the long-term effects of sequential algorithmic decisions. In this thesis, we focus on studying algorithmic fairness in a sequential decision-making setting.

We first study how to translate model predictions to fair decisions. In particular, given predictions from black-box models (machine learning models or human experts), we propose an algorithm based on the classical learning-from-experts scheme to combine predictions and generate a fair and accurate decision. Our theoretical results show that approximate equalized odds can be achieved without sacrificing much regret. We also demonstrate the performance of the algorithm on real data sets commonly used by the fairness community.

In the second part of the thesis, we study if enforcing static fair decisions in the sequential setting could lead to long-term equality and improvement of disadvantaged groups under a feedback loop. In particular, we model the interaction between algorithmic decisions and underlying distribution using Markov Decision Model with general transition functions. We propose a new metric that measures the distributional impact of algorithmic decisions as measured by the change in distribution's center, spread and shape. This metric categorizes the impact into within-group impact and between-group impact, where within-group impact measures how policies impact the distribution within a group, and between-group impact how policies impact the distributions of two population groups differently. Our results show that there is generally a trade-off between utility and between-group impact for threshold policies, and common fairness constraints could lead to "backfire effects" where the impact on groups could be disparate.

Thesis Supervisor: Kalyan Veeramachaneni

Title: Principal Research Scientist

Acknowledgments

I am very fortunate to have received valuable guidance and support throughout this journey.

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Kalyan Veeramachanenni, for his invaluable guidance and support during my Ph.D. study. You have always motivated me to think big and think outside the box. I've learned not only research skills but also many life lessons from you.

Next, I would like to thank my thesis committee members, Professor Alberto Abadie, Professor Caroline Uhler, and Professor Alfredo Cuesta-Infante for their words of encouragement and valuable feedback on the thesis.

I would also like to thank all my collaborators and labmates, whose knowledge, insight, and generosity has always been a source of inspiration for me. I would like to thank all members of my lab: Dongyu, Ivan, Micah, Ola, Sarah, Nassim, Sara, and many others. It has been a true pleasure to brainstorm and collaborate with many of you on various projects.

I would also like to appreciate the support from IDSS. I would like to thank Munther Dahleh, Ali Jadbabaie, John N. Tsitsiklis, and Fotini Christia for taking the lead in creating a collaborative and intellectually stimulating environment. I am also thankful to the MIT staff: Elizabeth Miles, Michaela Henry, Gracie Gao, and Brian Jones.

Next, I would like to thank all my friends at MIT and in Boston, old and new, whose paths crossed with mine in the last few years. Ph.D. is a long journey and thank you for always being there for me throughout the ups and downs.

Last I would like to express my immense gratitude to my parents for their unwavering support and unconditional love. Thank you for always believing in me and encouraging me to pursue my passions and dreams.

This doctoral thesis has been examined by a Committee of the Institute of Data, Systems and Society as follows:

Professor Alberto Abadie . . . . . .
Chair, Thesis Committee
Professor of Economics

Professor Caroline Uhler . . . . . .
Member, Thesis Committee
Professor of Electrical Engineering and Computer Science

Professor Alfredo Cuesta-Infante . . . . . .
Member, Thesis Committee
Professor of Computer Science

Contents

List of Figures

List of Tables

Notation	Meaning
Z	Protected group attribute such as gender or race. In binary case, we will refer to the groups as group A and group B, where A represents the advantaged group and B represents the disadvantaged group.
p_z	$\mathbb{P}(Z = z)$. Probability that a sampled example belongs to group z.
X	Feature variables other than group attribute.
f_X	Ground-truth function that maps from group attribute Z to features X.
Y	Target variable. In the first part of the thesis the target variable is binary, where $Y \in \{0, 1\}$. In the second part of the thesis the target variable is a probability where $Y \in [0, 1]$.
O	Outcome variable. If the target variable a probability, O is a realized binary outcome sampled from the probability.
f_Y	Ground-truth function that maps from feature X to target variable Y.
$\mathcal{D}$	$\mathbb{P}(X, Y, Z)$. Ground-truth distribution where the dataset is sampled from.
(x, y, z)	$(x, y, z) \sim \mathcal{D}$. An individual sampled from the distribution is a tuple of the group attribute, feature variables, and target variable.
$\hat{Y}$	Model prediction.
ℓ	Loss function for measuring the loss between prediction $\hat{Y}$ and ground truth target Y.
FPR	$FPR = \mathbb{P}(\hat{Y} = 1 \mid Y = 0)$. False positive rate.
FNR	$FNR = \mathbb{P}(\hat{Y} = 0 \mid Y = 1)$. False negative rate.
EqFPR	$\mathbb{P}(\hat{Y} = 1 \mid Y = 0, Z = A) = \mathbb{P}(\hat{Y} = 1 \mid Y = 0, Z = B)$. Equalized false positive rate.
EqFNR	$\mathbb{P}(\hat{Y} = 1 \mid Y = 0, Z = A) = \mathbb{P}(\hat{Y} = 0 \mid Y = 1, Z = B)$. Equalized false negative rate.
DemoPar	$\mathbb{P}(\hat{Y} = 1 \mid Z = A) = \mathbb{P}(\hat{Y} = 1 \mid Z = B)$. Demographic parity.
EqOdds	Equalized odds requires both EqFPR and EqFNR.
D	Action or decision made on an individual. This is used inreplace of $\hat{Y}$ in sequential decision making.
τ	$D = \mathbb{1}(Y \geq \tau)$. Threshold on target variable at which a positive decision is issued.
T	Total number of time steps in sequential decision making.
t	A single time step t.

Table 1: Notation table of the terms used throughout the thesis.

Chapter 1

Introduction

1.1 Algorithmic Fairness

The past decade has witnessed tremendous advancements in machine learning models. In image classification, machine learning models based on deep neural network first surpasses human-level accuracy [He et al., 2016]. The list of advancements continues to grow, as these models become able to perform more tasks such as text classification [Johnson et al., 2017] and game-playing [Silver et al., 2016]. Machine learning models are able to learn and extract patterns from much larger amounts of data than humans can. Recently, machine learning models have been put to use in fields that are considered to be high-stakes and are traditionally left to humans, such as predicting healthcare needs [Obermeyer et al., 2019], accessing creditworthiness for loan applications, and predicting criminal recidivism for law enforcement [Dressel and Farid, 2018].

Recently, there have been growing concerns about potential bias and discrimination in these machine learning models. Models could propagate stereotypical and historical biases reflected in the training data. For example, image searches for professions such as CEO produce fewer images of women [Kay et al., 2015], and word embeddings used in natural language processing could encode gender biases [Caliskan et al., 2017].

The consequences are especially alarming when machine learning models are used in high-stakes applications. For example, Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) is a commercial algorithm used by U.S. courts to predict

Age	Age or generation
Race	Caste, race, color, ethnicity, national origin
Gender	Gender, gender expression, sexual orientation
Religion	Religion, ideology, politic preferences, membership to guilds-unions-political parties

Table 1.1: A list of protected attributes.

the likelihood of a defendant becoming a recidivist. While COMPAS's overall accuracy is similar for white and black defendants, it has been shown that black defendants were more likely to be misclassified as being at high risk for violent recidivism [Angwin et al., 2016]. In particular, black defendants who did not recidivate were nonetheless incorrectly predicted to re-offend at a rate of 44.9%, while white defendants were only incorrectly predicted to re-offend at a rate of 23.5%.

In health care, a widely used algorithm [Obermeyer et al., 2019] for predicting risk scores of extra healthcare needs has been shown to underestimate risks for black patients. Specifically, the algorithm assigns risk scores to patients, and patients at the 97th percentile of the risk score are enrolled in the extra healthcare program. However, at this percentile, black patients have 26.3% more chronic illnesses than white patients. This biased prediction would lead to sick black patients not receiving the extra care they need. This happens because the model uses health costs as a proxy for healthcare needs. Because black patients tend to spend less money than white patients at the same level of healthcare needs, the model underestimates their health needs.

Broadly speaking, algorithmic decisions based on machine learning models shouldn't recommend disparate treatment or predict disparate impact based on people's protected attributes [Verma and Rubin, 2018], which include age, race, color, religion, national origin, sex, marital status, and political preferences. A comprehensive list is shown in table 1.1. As shown in the previous two examples, even when demographic information is not directly used in the decision-making process, biases can still manifest because of proxies in the dataset. When algorithmic decision-making is put into practice, many different factors must be carefully considered throughout the design and deployment of a machine learning model.

Term	Meaning
State of the world	The state of the world refers to the underlying ground-truth distribution.
Sampling	Sampling is the process of taking measurements from the state of the world.
Dataset	A dataset is a collections of data points sampled.
Training	Training is the process of learning patterns from the dataset.
Model	A model takes the training data and optimize for some objective function.
Inference	Inference is the process of using a model to make predictions on new data points.
Prediction	Prediction is some scores or labels generated by the model during the inference process.
Decision-making	Decision-making is the process of transferring model predictions to an actionable decision.
Individual	An individual is a data point where the decision is made on. In the context of fairness, an individual refers to a person.
Feedback	Feedback is the process where decisions could impact the state of the world.

Table 1.2: A glossary of the terms used in the machine learning cycle. The bold terms are the nodes in the cycle, and the italic terms are the edges in the cycle.

In the next section, we show that biases can manifest in any part of the typical deployment cycle of machine learning models. These range from biases caused by using an unbalanced dataset, to biases that come from spurious correlations between demographic information and predictions in model representations to biases that occur during the process of transforming a model prediction into a decision.

1.2 A Full Taxonomy of Biases in Machine Learning

We next showcase how biases can arise through different development stages of a machine learning model. We use Figure 1-1 to illustrate a typical development cycle of machine learning models, and use this to show how biases could arise and be mitigated in each stage. The terms used in the figure are explained in Table 1.2.

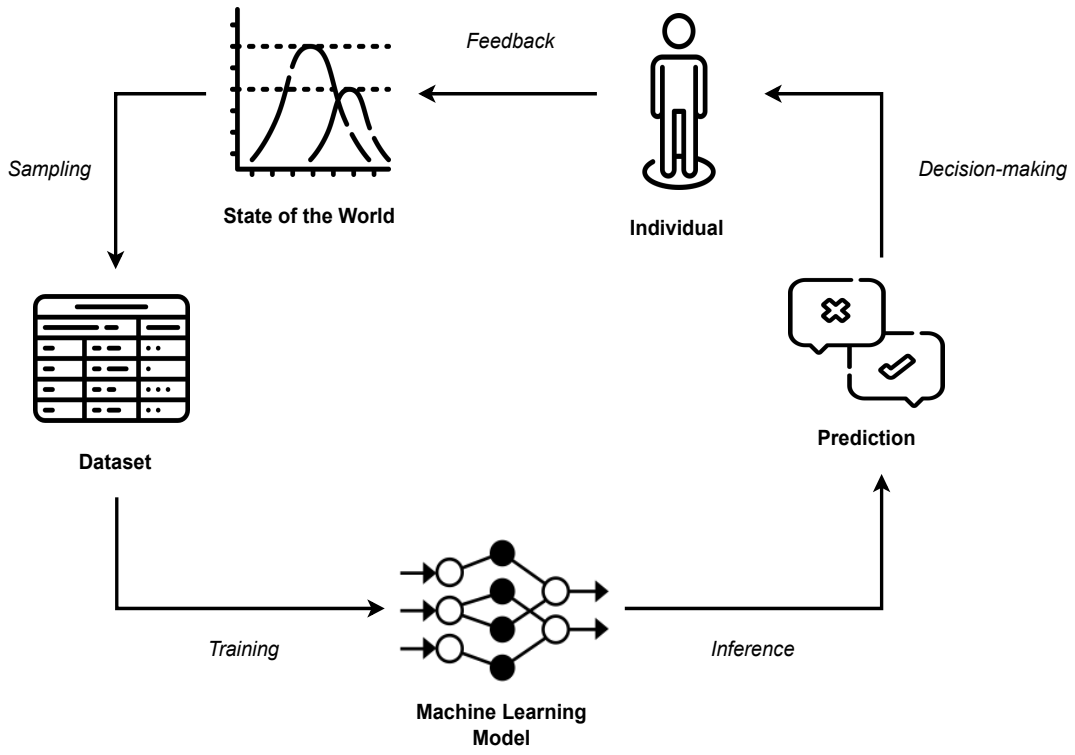


Figure 1-1: A typical machine learning cycle contains five stages: the *state of the world* describes the true underlying distribution; data is sampled from the state of the world; a *machine learning model* learns patterns from the training dataset; the model makes *predictions* on new instances; these predictions are transformed into decisions about an *individual*; an individual could take actions and change the state of the world.

1.2.1 Dataset Collection Bias

Machine learning models often need to deal with large messy datasets that are not collected under clear guidelines. As mentioned in the White House "Big Data" report, [White-House, 2016], selection bias – where data input to the model does not represent the actual population – is a main source of discrimination.

Unbalanced Dataset First, a large dataset is not always a *diverse* one. In fact, widely used machine learning datasets often suffer from a lack of diversity. For example, many facial recognition datasets have been collected through Flickr, and mostly consist of faces of white people [Kärkkäinen and Joo, 2021]. Such datasets have been widely used in different applications including image up-sampling, where the goal is to construct high-resolution images from corresponding low-resolution inputs. Recently, it was discovered that an image up-sampling model [Menon et al., 2020] trained on this dataset outputs a white face when given Barack Obama's low-resolution picture. This shows that using unbalanced data can make the model output collapse into the majority class.

Historical and systematic biases In addition to the unbalanced dataset problem, historical and systematic biases are also prevalent in collected datasets. Learning from this data puts the model in danger of repeating those systematic biases – discrimination against certain social groups and reinforcement of prevailing cultural stereotypes and existing demographic inequalities. For example, word embedding, a popular framework to represent text data as vectors, is often the first step in training a large language model. A recent study shows that word embeddings trained on Google News article [Bolukbasi et al., 2016] exhibit female/male gender stereotypes, such that males are more likely to be associated with computer programmers and females are more likely to be associated with babysitters. This shows that machine learning models can amplify historical biases that exist in data.

Much work has been done to try to address biases at this stage, and these approaches are referred to as *pre-processing techniques* [Zemel et al., 2013, Louizos et al., 2016, du Pin Calmon et al., 2017]. One possible solution is to collect more data from under-represented groups, but this can be difficult to achieve due to self-selection biases. For in-

stance, if certain jobs historically employ more males than females, those positions might attract more males to apply, worsening the data bias. In some cases, it can also be difficult or unethical to collect more data from under-represented groups. There is also work that tries to transform the dataset such that the underlying biases are removed [Bellamy et al., 2018]. The idea is to learn a new representation of the dataset, removing information correlated to the sensitive attribute and preserving the information of features as much as possible.

However, in many real-world scenarios, a pre-collected large dataset might not be available. For instance, if a new company wants to recruit employees, it often needs to start with limited data and make decisions sequentially. This requires new approaches for mitigating biases in sequential decision-making.

1.2.2 Model Bias

After data collection is over, a machine learning model needs to digest and learn from the collected dataset (training examples). Unfortunately, a machine learning model sees the training examples differently than humans do. The sole goal of most machine learning models is to learn a mapping from input to output to minimize empirical risk (training errors). Unfortunately, minimizing training errors leads models to recklessly absorb all the correlations found in the training data. Many of the extracted correlations are spurious, in the sense that although they reduce the training errors, they appear completely random and uninformative to humans.

As a thought experiment, consider the problem of classifying profile pictures of males and females. If males are more likely to wear ties in the training data set, a naive model would pick up this spurious correlation between gender and tie. Later, if that model encountered a picture of a female with a tie, the model would fail unexpectedly. As humans, we can realize that tie-wearing does not definitionally mean a male picture, yet it would be difficult for an algorithm to differentiate correlation from causation.

Mitigating biases during the model training stage is perhaps the most well-studied solution for addressing biases. These methods can be grouped into *in processing* techniques and

post processing techniques. *In processing* approaches usually involve enforcing some fairness constraints at model training time [Corbett-Davies et al., 2017] through constrained optimization or modification of the objective function [Berk et al., 2017]. Though easy to implement, these methods can lead to drops in model performance. In addition, many machine learning models are black boxes, and it is sometimes impossible to change the training paradigm of the models. *Post-processing* techniques treat the model as a black box and adjust the model predictions to remove biases [Hardt et al., 2016].

1.2.3 Decision Bias

Even with a perfectly fair machine learning model, things can still go wrong during decision-making time. Machine learning models only generate *predictions*, not actionable *decisions*. In many high-stake systems with humans in the loop, there are gaps in translating a model prediction into a justified decision in practice. For example, suppose there is a machine learning model that predicts crime rates in a community, and police officers make decisions based on these predictions. If a model predicts that certain communities are more likely to have a higher crime rate, law enforcement in that area may tend to lower their threshold and arrest more people.

Very few works have been proposed to address the unfairness that arises in this stage. This gap motivates us to design algorithms that can deliver fair decisions when working along predictions from humans or black-box models.

1.2.4 Feedback Loop

It is also important to bear in mind that delivering a fair decision is not always the end of the story. In many cases, decisions carry big and profound consequences. Unlike common machine learning applications such as image classification, here the data comes from people, and people could react to decisions made about them. Decisions that affect individual people often create a feedback loop and change the state of the world from which the data is sampled.

Decisions could change the population distribution First, decisions could change the population distribution. Consider again the example of predicting crime rates in a community. If innocent people living in this community know that a model has predicted high crime rates there, they may move out. This leads to a self-fulfilling feedback loop that changes the future population distribution, such that the crime rate could further increase in this area.

Decisions could change the distribution of the features and outcomes Decisions could also change the outcome distribution. For example, say a bank takes affirmative action to approve loans at a lower threshold for people coming from less advantaged socioeconomic groups. And then say those people might have trouble paying back the loan later, which decreases their credit scores and credentials in the future. In this case, even well-intended actions may create an accidental feedback loop.

Strategic classification Algorithmic decisions could change distributions unintentionally. Individuals could also strategically react to the decision-making rule [Hardt et al., 2016, Milli et al., 2019, Ghalme et al., 2021]. For instance, if applicants know which features are used in a loan approval decision, they might be incentivized to manipulate those features to get approved. This might lead to strategic behaviors such as holding multiple credit cards or moving to a different zip code, changing their loan eligibility without necessarily affecting their ability to repay it. Such tension between decision-makers and individuals can be modeled in a game-theoretic setting [Zhang et al., 2022, Keswani and Celis, 2022].

Most of the fairness solutions focus on one-shot classification or regression problems, and there is a gap in addressing fairness in a sequential and dynamic environment. This motivates us to address fairness concerns under the sequential dynamic environment and investigate the long-term impact of particular solutions.

1.3 Thesis Summary and Contribution

In many real-life situations, including job and loan applications, decision-makers must make justified and fair real-time decisions about a person's fitness for a particular opportunity. In this thesis, we focus on studying algorithmic fairness in sequential decision-making settings where the data comes on the fly. Within the cycle of machine learning, many solutions have been proposed for auditing and mitigating model unfairness in terms of predictions. However, there is still a gap in addressing the biases that arise after the prediction stage in the machine learning cycle. We focus on the last two stages of the machine learning cycle and study fairness beyond prediction time.

We first study if it is possible to translate model predictions to fair decisions. In particular, given predictions from black-box models (machine learning models or human experts), we propose an algorithm based on the classical learning-from-experts scheme to combine the predictions and generate a fair and accurate decision. We measure the accuracy of the algorithm using regret, which measures the difference in the algorithm's accuracy compared to the best expert. For fairness, we adopt the equalized odds metric, which requires equalized false positive and false negative rates. Our theoretical results show that approximate fairness can be achieved without sacrificing much regret. We also demonstrate the performance of the algorithm on real data sets commonly used by the fairness community.

In the second part of the thesis, we investigate how decisions made on individuals could change the state of the world. Can enforcing fair decisions in a sequential setting lead to long-term improvement of welfare when the feedback loop is taken into account? In particular, we study the long-term impact of repeatedly enforcing different fairness constraints at each decision time on shaping the underlying population under Markov Decision Models. We propose a metric to measure the distributional impact of algorithmic decisions on the target variable distributions in terms of within-group and between-group impact. Our results show that fairness constraints could lead to "backfire effects" which further entrench distributional disparities between population groups.

1.3.1 Summary of Contributions

In the first part of the thesis, we study how to translate model predictions into fair decisions. We make the following contributions:

- We propose a meta-algorithm, G-FORCE (Group-Fair, Optimal, Randomized Combination of Experts), which combines black-box predictions into fair and accurate decisions in an online setting. We measure fairness using the strictest metrics based on classification parity (equalized odds), which require both equalized false positive and false negative rates among population groups.

- The algorithm re-weights experts based on their past performance in terms of accuracy and fairness. Under this framework, we show that the algorithm's performance on regret and fairness can be upper bounded. We demonstrate the performance of the algorithm on real data sets commonly used by the fairness community, as well as on synthetic datasets to test its performance under extreme scenarios.

- We also extend the theoretical analysis for the delayed setting, where the true label is not instantly revealed at each time step. We demonstrate how the previous algorithm can be adapted in this setting.

In the second part of the thesis, we study how decisions made on people could change the state of the world through the feedback effect. We make the following contributions:

- We first propose a new metric that measures the distributional impact of algorithmic decisions as measured by the change in distribution's center, spread and shape. Unlike previous work that has focused on the disparity of the group mean, this metric allows us to characterize the change of target distribution shape in a more fine-grained way. This metric categorizes the impact into within-group impact and between-group impact, where within-group impact measures how policies impact the distribution of a group, and between-group impact how policies impact the distributions of two population groups differently.

- We conduct experiments with general a set of well-used group fairness constraints on synthetic Gaussian distribution and real-world datasets. We demonstrate that previ-

ous work measuring disparity in group mean could be insufficient, and using a more fine-grained metric could lead to different conclusions from the previous simulation works. In particular, our results show that there is generally a trade-off between utility and between- group impact for threshold policies.

This thesis is based on the following papers published:

- Towards Reducing Biases in Combining Multiple Experts Online. [Sun et al., 2021] Preliminary version appeared at *Neurips 2019 AI for Social Good*. The final version appeared at *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI 21)*. This was joint work with Dr. Ivan Ramirez who was an equal contributor to this work.

- The Backfire Effects of Fairness Constraints. [Sun et al., 2021] The preliminary version appeared at *ICML 2022 Responsible Decision Making in Dynamic Environment*.

1.4 Thesis Outline

The thesis in organized as follows:

- In chapter 2, we introduce the background. We introduce some commonly used metrics for fairness and briefly introduce related work on achieving fairness in sequential settings.

- In chapter 3, we formally describe the setting of online learning with fairness. We then introduce our method of combining black-box classifiers' predictions to deliver fair decisions.

- In chapter 4, we study the long-term impacts of algorithmic decisions. In particular, we study under which scenarios would algorithmic decisions lead to a further disparity between population groups.

- In the last chapter, we conclude with potential future directions.

Chapter 2

Background

In this chapter, we will first introduce some preliminaries and backgrounds of fairness in machine learning. We will also introduce some of the most well-used definitions fairness, which will be referred throughout the thesis. We will then introduce related literature in the context of algorithmic fairness in a sequential setting.

2.1 Preliminaries

Throughout this thesis, we assume the underlying state of the world is represented by the joint distribution $(X, Y, Z) \sim \mathcal{D}$, where $Z \in \{A, B\}$ corresponds to the partition on sensitive/protected attributes such as gender or race, $X \in \mathcal{X}$ corresponds to the feature vectors, and $Y \in \mathcal{Y}$ denotes the ground-truth labels.

In supervised learning, the goal of the model is to learn a parametrized function f_θ that minimizes the expected risk with respect to the loss function ℓ:

$$\min_{\theta} \ \mathbb{E}_{(X,Y,Z)\sim\mathcal{D}}[\ell(f_\theta(X), Y)] \tag{2.1}$$

where θ is the parameter. We use $\hat{Y} = f_\theta(X)$ to denote the predicted label from the model. The original goal of the learning problem doesn't contain the sensitive attribute in the objective function.

When this classical risk minimization framework is applied to a dataset involving peo-

ple, some assumptions in classical machine learning might not hold:

- Average performance: The goal of minimizing average loss incentives the model to find the best parameters that fit the majority group well. This often leads to disparate performances in the majority group and the minority group.
- Static data distribution: In one shot prediction problem, it is common to assume the data distribution is independent of the predictions. However, predictions could have the power to change the underlying distribution $\mathcal{D}$ due to the feedback loop. This often leads to problems when the decision maker needs to make repetitive predictions.

The research community has come up with many ways to evaluate the biases as a result of model predictions. In the next section, we briefly introduce a few most well-adopted metrics for fairness that are based on model predictions.

2.2 Fairness Metrics

Fairness can be considered on an individual level or a group level. At the individual level, fairness can be intuitively defined as "similar individuals should be treated similarly" [Dwork et al., 2012a]. As discussed in Dwork et al. [2012a], one challenge of working with individual fairness is that the distance metric is difficult to specify. Group fairness can be defined as balancing some metrics across different demographic groups (such as gender groups, racial groups, etc.).

At the group level, the definitions can be roughly grouped into three categories: (1) statistical metrics that are solely based on the statistical relationship between the predictions and outcome variables, (2) causal metrics that involve all variables, and (3) others that are beyond prediction fairness. We introduce these metrics under a binary classification setting, although extensions are often available.

2.2.1 Statistical Fairness Metrics

We will next introduce four mostly well-used statistical fairness metrics for fairness: (1) Demographic Parity (DemoPar) requires a predictor that is independent of the sensitive

attribute (2) Equalized Opportunity (EqOpp) [Hardt et al., 2016] requires a predictor that is independent of the sensitive attribute given that the label is positive, and (3) Equalized Odds (EqOdd) requires a predictor is independent of the sensitive attribute given the true label, and(4) Calibration [Verma and Rubin, 2018] (test fairness) requires that outcomes should be independent of protected attributes conditional on the risk score.

Demographic Parity The most intuitive definition is demographic parity, which requires the probability of positive prediction should be equalized for different groups. In other words, the prediction $\hat{Y}$ should be independent of the sensitive attribute Z. This metric could be achieved when the model ignores the group attribute.

Definition 2.2.1 (Demographic Parity (DemoPar))**.** *A predictor $\hat{Y}$ satisfies demographic parity if*

$$\mathbb{P}(\hat{Y} = 1|Z = A) = \mathbb{P}(\hat{Y} = 1|Z = B)$$

One issue with demographic parity is that it ensures the acceptance rate is equal regardless of whether an individual is qualified or not. If the target variable Y is correlated with group attribute Z, demographic will rule out the perfect predictor $\hat{Y} = Y$ [Hardt et al., 2016].

Equalized Error Rates Accuracy parity, or equalized error rates, improve on demographic parity by bringing the true qualification target variable Y into the definition.

Definition 2.2.2 (Equalized Error Rates (EqERR))**.** *A predictor $\hat{Y}$ satisfies equalized error rates if*

$$\mathbb{P}(\hat{Y} \neq Y|Z = A) = \mathbb{P}(\hat{Y} \neq Y|Z = B)$$

However, equalized error rates don't distinguish between the error types, and the cost of false positives and false negatives could be very different in many applications.

Equalized Opportunity Equalized opportunity requires that the predictor $\hat{Y}$ is independent of the group attribute Z conditional on the positive outcome $Y = 1$. For example, in the loan application example, this requires that among all people who could pay back

their loan ($Y = 1$), they should have an equal probability of getting the loan regardless of their group. In the context of the confusion matrix, this metric could be defined in terms of equalized false positive rate.

Definition 2.2.3 (Equalized FPR (EqFPR)/Equalized FNR (EqFNR))**.** *Let* $\mathrm{FPR}_z = \mathbb{P}(\hat{Y} = 1|Z = z, Y = 0)$ *and* $\mathrm{FNR}_z = \mathbb{P}(\hat{Y} = 0|Z = z, Y = 1)$ *be the False Positive Rate (FPR) and the False Negative rate (FNR) for group* z *respectively. A predictor/classifier is said to satisfy Equalized FPR and Equalized FNR on group A and group B respectively if* $\mathrm{FPR}_A = \mathrm{FPR}_B$ *and* $\mathrm{FNR}_A = \mathrm{FNR}_B$.

Definition 2.2.4 (Equalized Opportunity (EqOpp))**.** *A predictor exhibits equalized opportunity if* $\mathbb{P}(\hat{Y} = 1|Z = A, Y = 1) = \mathbb{P}(\hat{Y} = 1|Z = B, Y = 1)$. *In other words, it satisfies eqFPR.*

Equalized Odds A stronger notion of fairness that is defined based on the confusion matrix is Equalized odds. Equalized odds require that the predictor achieves both equalized FPR and FNR.

Definition 2.2.5 (Equalized Odds (EqOdd))**.** *A predictor exhibits equalized odds if it achieves both an equalized FPR and an equalized FNR.*

As seen from the above, equalized odds is the most strict metric among those that are based on statistical parity of outcomes and predictions.

Definition 2.2.6 (Test Fairness)**.** *A classifier* f *is perfectly calibrated if for any score* $s \in [0, 1]$, $\mathbb{P}(Y = 1|f(X) = s] = s$.

Calibration is well-used in practice, which requires that when conditioning on scores or risk estimates, the true label should be independent of the group attribute. Essentially this requires the scores from a classifier should carry the same meaning for both groups.

First, there are often contentions and trade-offs between them. For example, previous work has shown that equalized odds and calibration can not be achieved at the same time [Chouldechova, 2017, Kleinberg et al., 2017]. How to synthesize or characterize the trade-offs of these incompatible metrics in real applications remains an open research problem.

The statistical parity metrics are often oblivious to the underlying risk distribution, and in some cases could cause harm to the group they are trying to protect [Corbett-Davies et al., 2017]. Another drawback of statistical definitions is that they largely ignore all attributes of the classified subject except for the sensitive attribute Z. We next turn our attention to causal fairness metrics, which consider the relationship between all variables.

2.2.2 Causal Fairness Metrics

Before diving into causal fairness metrics, we first briefly introduce the definition of a causal model. A causal model is a triple (U, V, F) such that:

- U is the set of exogenous variables determined by factors, not in the model.
- V is the set of endogenous variables $\{V_1, ..., V_n\}$.
- F is a set of functions $\{f_1, ..., f_n\}$ called *structural equations*, one for each $V_i \in V$, such that $V_i = f_i(pa_i, U_{pa_i})$, $pa_i \subseteq V \setminus V_i$ and $U_{pa_i} \in U$, where pa_i refers to parents of V_i in the causal graph.

Let $\hat{Y}_{Z \leftarrow z}(U = u) = y$ denotes the value of $\hat{Y}$ for given $U = u$ if Z had taken the value of z.

Definition 2.2.7 (Counterfactual Fairness)**.** *Let Z, X, Y represent the protected attributes, remaining attributes, and true label respectively. Predictor $\hat{Y}$ is counter-factually fair if*

$$\mathbb{P}(\hat{Y}_{Z \leftarrow z}(U = u) = y | X = x, Z = z) = \mathbb{P}(\hat{Y}_{Z \leftarrow z'}(U = u) = y | X = x, Z = z)$$

for all y and z.

Specifically, counterfactual fairness [Kusner et al., 2017] requires that changing the sensitive attribute Z while holding things that are not causally dependent on Z constant will not change the distribution of the prediction $\hat{Y}$. In order words, a causal graph is counterfactually fair if the predicted outcome $\hat{Y}$ in the graph does not depend on a descendant of the protected attribute Z.

2.3 Related Work

2.3.1 Fairness Metrics

Individual Fairness Fairness can be considered individually or collectively. At the individual level, fairness can be defined as "similar individuals should be treated similarly" Dwork et al. [2012a]. yet it is often challenging to specify suitable distance metrics to measure similarity between individuals.

Group Fairness At the group level, fairness can be defined as balancing some statistical metrics approximately across different demographic groups (such as gender groups, racial groups, etc.). Equalized odds [Zafar et al., 2017], or "disparate mistreatment," requires that no error type is disproportionate for any one or more groups. This could be achieved by equalizing false positive rates, commonly referred to as equal opportunity [Hardt et al., 2016], or equalizing classification errors. In addition to statistical parity, another line of research focuses on defining fairness from a causal perspective. Kusner et al. [2017] first defines counterfactual fairness as requiring a decision to be the same in the counterfactual world where the individual belongs to a different group. For a more comprehensive list of the fairness definitions, we refer the readers to the survey paper [Verma and Rubin, 2018].

Lastly, people have also proposed some metrics that are beyond prediction problems. These approaches often are rooted in other domains such as economics, law, and psychology. For one, Heidari et al. [2019] proposes an effort-based measure of fairness and quantify how algorithmic policies would reshape the underlying populations.

The incompatibility of fairness metrics Despite the numerous definition of fairness, many of them could be inherently incompatible both from a mathematical perspective and also from a conceptual perspective. First, individual fairness and group fairness could be conflicting where satisfying group fairness can yield harm for people belonging to those groups [Dwork et al., 2012b, Corbett-Davies et al., 2017, Green, 2020].

Even within group fairness metrics, recent work shows that it is impossible to simultaneously achieve equalized odds [Chouldechova, 2017, Kleinberg et al., 2017] with other

notions of fairness such as calibration, which requires that outcomes are independent of protected attributes conditional on estimates. It is also generally accepted that there is often a trade-off between predictive accuracy and fairness [Corbett-Davies et al., 2017]. Kearns et al. [2018] argue that statistical parity constraints could lead to fairness gerrymandering, where a classifier that satisfies fairness in each group could violate the constraint on groups that are combined with combinations of protected attribute values.

2.3.2 Biases mitigation in machine learning

Fair classification Classification is an important task in supervised machine learning and is used in various applications that directly impact humans such as loan applications and college admission. As we talked about in the last chapter, biases for classification problems can be addressed during model training (in-processing) or after model training (post-processing). Zafar et al. [2017] incorporate equalized odds as a constraint while solving optimization problems, while Hardt et al. [2016] remove discrimination at post-processing steps.

Fairness from a causal perspective There are also some concurrent works studying long-term fairness from a causal inference perspective. Chiappa [2019] consider the case where a sensitive attribute affects the decision through both fair and unfair pathways. They propose to use the latent inference-projection method to disregard effects along the unfair pathways. Creager et al. [2020] frame the dynamic process as a changing causal structural model to evaluate different policies. Algorithmic recourse [Karimi et al., 2021] uses counterfactual analysis to propose the set of actions resulting in the desired output from the model. A recent line of work [Karimi et al., 2021] explores providing favorable outcomes to individuals from the disadvantaged group through minimal intervention on the features.

2.3.3 Fairness in Sequential Decision Making

Fairness in online learning setting There has been recent interest in studying fairness in an online setting, particularly the bandit setting. Gillen et al. [2018] consider a bandit

setting that learns from the feedback of a fairness oracle and returns all pairs of individuals for which the individual fairness constraint is violated. Joseph et al. [2016] study fair online classification in the contextual bandit setting, where fairness is defined as a worse candidate is never favored over a better candidate by the algorithm. Liu et al. [2017] consider satisfying calibrated fairness in a bandit setting. Bechavod et al. [2019] consider the problem of enforcing the equalized opportunity constraint at every round under a partial feedback stochastic setting where only true labels of positively classified instances are observed. Blum et al. [2018] specifically shows that it is impossible to achieve equalized odds under an adversarial setting when an adaptive adversary can choose the label for an instance.

Long-term fairness in interactive and dynamic environment Several works have studied the dynamics between algorithmic decisions and long-term population qualifications. One of the first works that touch on this topic is Liu et al. [2018], which considers the one-step feedback model and shows that enforcing common static fairness metrics in constrained optimization does not in general promote average group scores. Later, D'Amour et al. [2020a] extends the previous one-step analysis to multiple-step using simulation and argues that long-term dynamics may lead to different conclusions from the one-step analysis. Mouzannar et al. [2019] study whether enforcing demographic parity could lead to equality of qualifications. Wen et al. [2021] model the feedback effects as Markov decision process and proposes learning fair decision-making policies through cross-entropy optimization. There are also some works studying fairness in multi-agent systems [Jiang and Lu, 2019].

Most related to our work, Zhang et al. [2020] study the problem under a partially observed Markov decision problem setting and characterize the impacts of fairness constraints can have on the equilibrium of group qualification rates. One thing that has been missing from previous work is that the analysis only focuses on the average qualifications of groups, yet an algorithm or policy could have a more profound impact on the shape of the population beyond the group mean. In addition, the simulation setting is often too stylish and often ignores the nuances in the data generation process such as causal relationships, con-

founders, etc. They also focus on studying the dynamic system's equilibrium behavior but do not support counterfactual inference.

Chapter 3

G-FORCE : Achieving Fairness in Online Decision Making

In this section, we investigate how to combine predictions from models to produce fair decisions for individuals. This chapter will be structured as follows: we first use college admission as a motivating example to describe the our setting. We then introduce unique properties of this problem setting and notations. We then map this setting to the mathematical framework of online classification with fairness constraints. Next, we briefly introduce the current progress and the motivation for our work. In the next section, we present our algorithm, G-FORCE , and its theoretical guarantees. Finally we present G-FORCE 's performances on synthetic and real datasets.

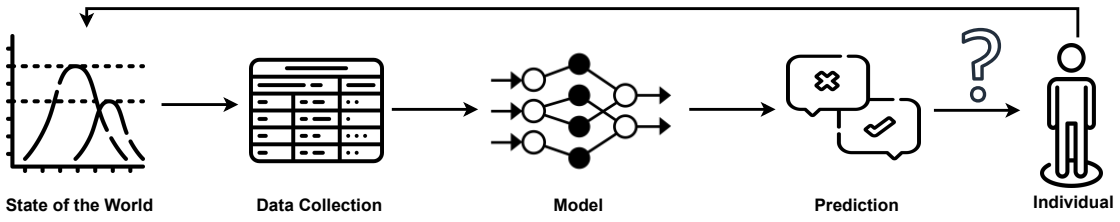


Figure 3-1: From predictions to fair decisions.

We first use an example on college admissions to illustrate the nuances in sequential decision making with fairness concerns. Suppose a college is trying to admit students for a program on a rolling basis where the admission committee has to make decisions as they go. In addition to the applicant's qualification, the admission committee also aims

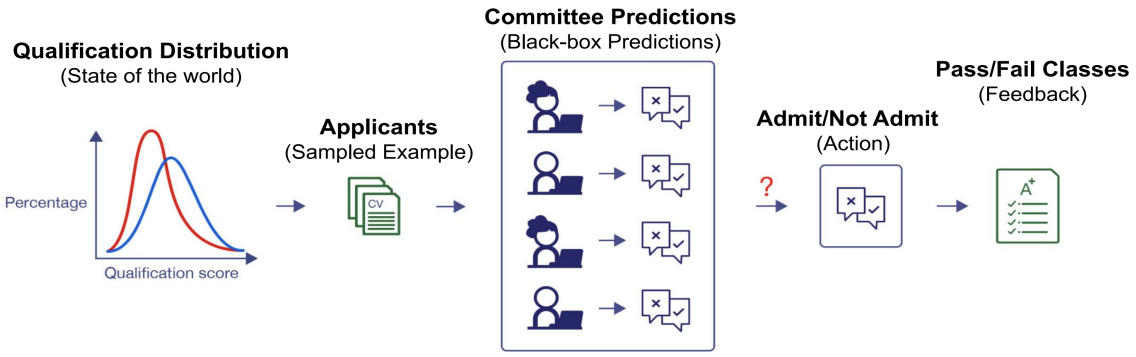


Figure 3-2: An example on college admission with experts.

to making the decision process fair to people of different races, genders, socioeconomic groups. Specifically, for all the applicants that are qualified, the committee should offer admissions to the same percentage of people regardless of their population groups. In figure 3-2, we illustrate the process and demonstrate how it fits into the machine learning cycle..

State of the world The state of the world consists of a joint distribution over group attribute Z (race, gender etc) and observed features X (SAT score, GPA, etc). In particular, students coming from different groups might have a different distribution of features.

Sampled Student At each round, a student sampled from the population applied for the program. For this motivating example, we assume the population can be split into group A (orange) and group B (blue).

Committee Predictions For each incoming applicant, each member on the admission committee will evaluate this person's qualification, and predict whether the applicant will succeed in the first semester of study. We will call the prediction of committee member i on applicant 1 as $\hat{Y}_i^1$.

Decision (Admit/Not admit) Admission decision is made based on committee member's predictions. In order to aggregate predictions from admission officers with different experience levels, the head of the admission committee need an aggregation algorithm . The

algorithm should deliver a accurate and fair decision D^1 (admit/not admit) to applicant 1.

Observe Feedback After applicant 1 has been admitted, the true label Y^1 (whether this applicant succeed in the first semester of study) would be revealed at the end of the academic semester. Meanwhile, the admission committee needs to keep evaluating the next applicant.

This is a typical setup for many human-in-the-loop decision making process. While many fairness solutions attempt to optimize a classifier, the goal here is to find a fair and accurate algorithm to combine the decisions from multiple experts or classifiers.

3.1 Online Binary Classification

3.1.1 Unique properties about the problem setting

online setting In this problem, we study the online setting where examples arrive sequentially. This is in contrast to batch learning setting where the entire dataset is available all at one. The online setting is useful when decision makers need to make decisions when data becomes available in a sequential order. The algorithms in online setting are usually adaptive to new data points and is usually for making decisions based on limited amount of data.

black box experts We assume access to a set of black box experts, which could either be machine learning models or human experts. Many applications with fairness concerns are high-stakes, and usually have human decision makers in the loop. The assumption of black box experts allows us to deliver fair decisions without any assumptions on the performances of the black box experts.

aggregation algorithm In this setting, we study online binary classification where the black box experts makes binary predictions on the example. In the case that the predictions from black box experts are continuous, an additional threshold might be learned to convert the continuous predictions to a binary prediction, and could be left as an interesting future

Notation	Meaning
$\mathcal{D}$	Underlying distribution where the dataset is sampled from
Z	Protected group attribute such as gender or race
X	Feature attributes the other than protected attribute
Y	Ground truth target variable
S	State S consists of (Z, X, Y)
O	$O \sim Bernoulli(Y)$. An instantiation of the target variable.
(z, x, y)	An individual sampled from the distribution is a tuple of the protected attribute, feature attribute, and ground-truth label
Y	The CDF distribution of target variable Y.
$\mathcal{G}$	A DAG representing the dependency between state variables
$f_v(\cdot)$	Structural equations for node v
$\nabla f_v(x)$	Derivative of structural equations f_v evaluated at x
S^t	State at time t consists of (Z, X^t, Y^t)
D^t	Decision at time t
$\mathcal{U}^t$	Utility for the decision maker at time t
π_z	Policy function for group z
τ_z^t	Threshold used for group z at time t
x_{tp}	Feature value increase for a true positive
x_{fp}	Feature value decrease for a false positive
x_{tn}	Feature value increase for a true negative
x_{fn}	Feature value decrease for a false negative
u_{tp}	Utility increase for a true positive
u_{fp}	Utility decrease for a false positive
u_{tn}	Utility increase for a true negative
u_{fn}	Utility decrease for a false negative
g	Distance metric
δ_z^t	Within-group impact for group z at time t
Δ_{AB}^t	Between-group impact of group A and B at time t

Table 3.1: Notation table for the terms used in this chapter.

Setting	Description	Examples
online setting	An online setting where data becomes available sequentially.	Admit students on rolling basis.
black box experts	black box experts where the function that generates predictions cannot be modified.	Committee Members.
aggregation algorithm	An algorithm that combines predictions from black box experts	Head of the committee.

Table 3.2: Unique properties of the setting.

work. In this section, we focus on designing an aggregation algorithm that combines the binary predictions from black box experts in order to produce a fair and accurate decision.

3.1.2 Notations

We start with binary classification problems, with a positive and a negative class, i.e., $Y \in \{+, -\}$. Each example (also referred to as *individual*) in the data set consists a pair $(x, z) \in \mathbb{R}^n$, where $x \in X$ is a vector of features attributes and $z \in Z$ is the group attribute . We also assume that the group attribute is binary can be partitioned into $Z \in \{A, B\}$. Let $\mathcal{F} = \{f_1, \ldots, f_d\}$ be a finite set of black box experts; and let $\hat{y} = f(x, z)$ be the prediction of an expert on an example (x, z). We denote the group rate p_z as the probability that an individual comes from group attribute z, where $p_z = \mathbb{P}(Z = z)$. We denote the base rate $\mu_{z,y}$ as the probability that an example comes from group attribute z has label y,where $\mu_{z,+} = \mathbb{P}(Y = y|Z = z)$.

We use superscript t to denote the time index or *round* t; for instance, y^t is the true label associated to the individual arrives at round t, i.e. (x^t, z^t). Superscript $*$ denotes optimality; for instance $f^*(z, y)$ represents the best expert on group attribute z with label class y.

Throughout this thesis, it is often necessary refer to an expert f, to the group attribute z, to the true label y or a combination of them. We indicate such a combination with a list of subscripts at the right of the variable. Thus, for instance, $w_{f,z}$ denotes the weight associ-

ated to a given expert, restricted to samples from group z, while $\ell_{f,z,y}$ represents the loss function with the same information as before but also restricted to samples from label class y. These subscripts are substituted with a specific value when needed. For instance, $\ell_{f,z,-}$ represents the same as before but specifying that all samples with negative labels. The lack of subscripts represents the generic variable.

In this section, we formally describe the setting in the language of online binary classification. As in the typical online learning setting, the algorithm runs through round $t = 1, \ldots, T$. We assume access to a set of black box experts $\mathcal{F} = \{f_1, \ldots, f_d\}$, which could be human experts or machine learning algorithms. At each round t, one expert $f^t \in \mathcal{F}$ is selected to estimate the label for the input example, $\hat{y}^t = f^t(x^t, z^t)$. Then, at the end of the round, the true label y^t is observed, producing a loss $\ell(\hat{y}^t, y^t)$.

Instant Feedback We first assume that we can instantly observe the true label after a decision has been made. In the college admission example, this means that whether a student would succeed in the first semester is instantly known after the student is admitted. Below, we relax this assumption and try to tackle the problem where the feedback of the true labels is delayed. The decision making process runs through rounds $t = 1, \ldots, T$. At each round t:

- A single individual $(x^t, z^t) \in \mathbb{R}^n$ arrives, where $x^t \in \mathcal{X}$ is a set of features and $z^t \in \mathcal{Z}$ is the group attribute .
- Each expert i makes a prediction $\hat{y}_i^t = f_i(x^t, z^t)$. According to the aggregation algorithm , a final decision $\hat{y}^t = f^t(x^t, z^t)$ is assigned to the individual .
- The true label y^t is revealed after the decision is made.

The goal here is to find an **algorithm** that combines the experts' predictions accurately and fairly.

Delayed Feedback Next we also formally describe the setting when the feedback is delayed: As before, the decision making process runs through rounds $t = 1, \ldots, T$. At each round t:

- A single individual $(x^t, z^t) \in \mathbb{R}^n$ arrives, where $x^t \in \mathcal{X}$ is a set of features and

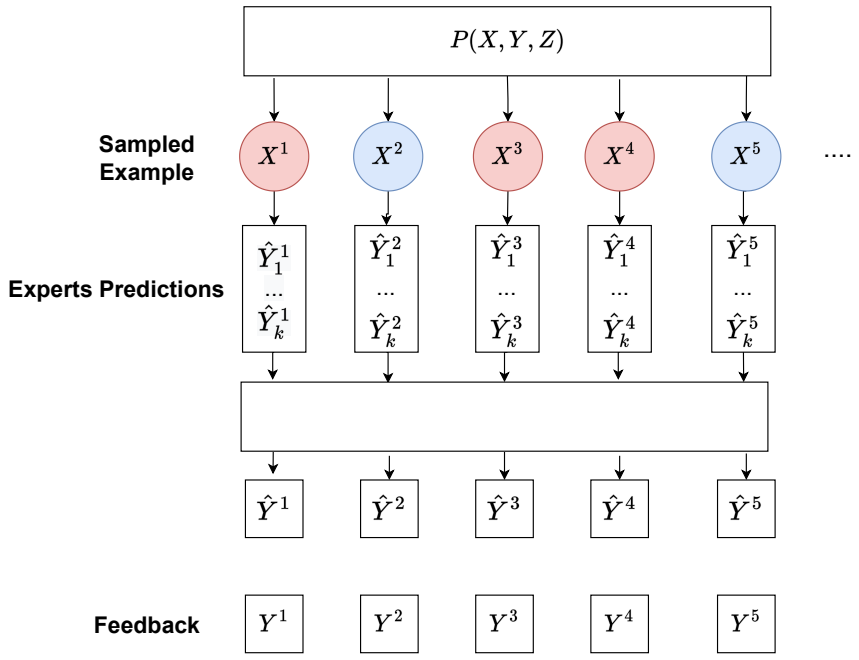


Figure 3-3: A figure depicting the online learning process.

$z^t \in \mathcal{Z}$ is the group attribute .

- Each expert i makes a prediction $\hat{y}_i^t = f_i(x^t, z^t)$. According to the aggregation algorithm , a final decision $\hat{y}^t = f^t(x^t, z^t)$ is assigned to the individual .
- At time t, a set of labels $\mathcal{D}_{z,y}^t = \{y^{t'} : t' + \tau_{z,y} = t\}$ is revealed, where $\tau_{z,y} > 0$ is the delay duration for an individual from group z with label y. Here the examples arrive at time t' will be revealed at time t, where $t = t' + \tau_{z,y}$.

3.1.3 Metric for evaluating accuracy

A frequent performance metric in online learning is *Regret*, which compares the performance of the algorithm with respect to the best fixed expert in hindsight.

Definition 3.1.1 (Regret)**.** *After T rounds, regret is formally expressed as*

$$\text{Regret}(T) = \sum_{t=1}^{T} \ell(f^t(x^t, z^t), y^t) - \inf_{f \in \mathcal{F}} \sum_{t=1}^{T} \ell(f(x^t, z^t), y^t) \tag{3.1}$$

The typical goal of online learning is to design a training algorithm that achieves sublinear regret compared with the best fixed experts in hindsight over the T rounds; i.e.

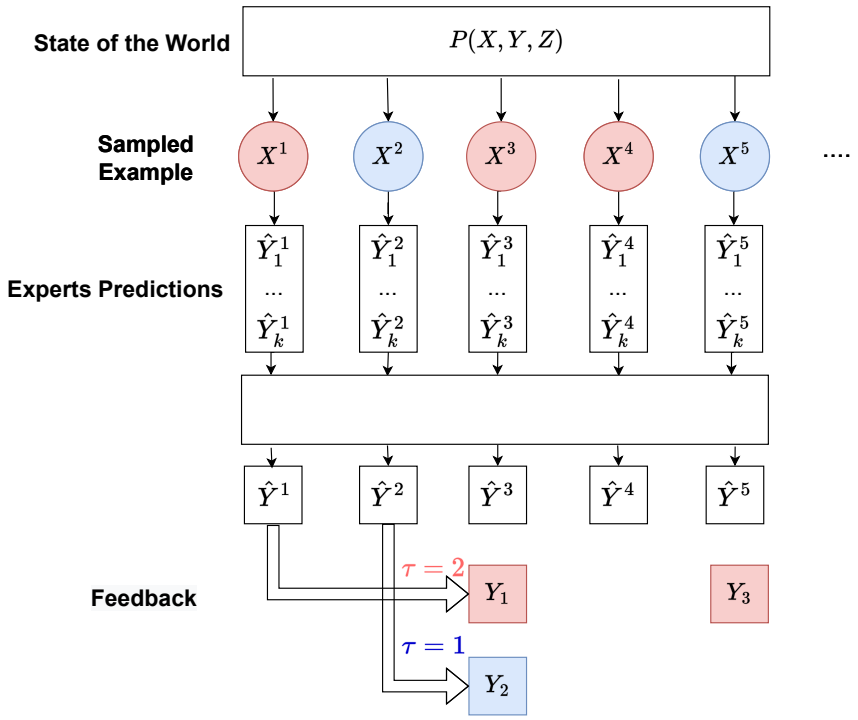


Figure 3-4: A figure depicting online learning with constant delay with $\tau_A = 2$ and $\tau_B = 1$.

$\lim_{T\to\infty} \text{Regret}(T)/T = 0$. This means that as round goes on, the average regret goes to zero and the algorithm converges to the best expert in hindsight.

3.1.4 Metrics for evaluating fairness

In addition to regret, we also evaluate the fairness on the online algorithm. We introduce two metrics: Equalized error rates (EqERR) and Equalized Odds (EqOdds).

Definition 3.1.2 (EqERR and ϵ-ERR)**.** *A randomized algorithm satisfies EqERR if:*

$$\mathbb{E}[\hat{Y} \neq Y | Z = A] = \mathbb{E}[\hat{Y} \neq Y | Z = B]$$

A randomized algorithm satisfies ϵ-ERR if:

$$|\mathbb{E}[\hat{Y} \neq Y | Z = A] - \mathbb{E}[\hat{Y} \neq Y | Z = B]| \leq \epsilon$$

EqERR requires that the algorithm makes equal percentage of errors (equal accuracy rate) for all groups. In this metric, different types of errors (false positives and false negatives) are not distinguished.

Definition 3.1.3 (EqOdds). *Let* $\mathrm{FPR}_z = \mathbb{P}(\hat{Y} = 1 | Z = z, Y = 0)$ *and* $\mathrm{FNR}_z = \mathbb{P}(\hat{Y} = 0 | Z = z, Y = 1)$ *be the False Positive Rate (FPR) and the False Negative rate (FNR) for group* z *respectively. An algorithm is said to satisfy Equalized FPR (EqFPR) and Equalized FNR (EqFNR) on group A and group B respectively if* $\mathrm{FPR}_A = \mathrm{FPR}_B$ *and* $\mathrm{FNR}_A = \mathrm{FNR}_B$. *A randomized algorithm satisfies EqOdds if it satisfies EqFPR and EqFNR.*

In EqOdds metric, the algorithm requires the algorithm has equal false positive and false negative rates for all groups.

3.2 Online Algorithms

3.2.1 Multiplicative weights algorithm (MW)

The *Multiplicative Weights* (MW), proposed by Arora et al. [2012], is a frequently used aggregation algorithm for achieving sub-linear regret. In the MW algorithm, a decision maker has a choice of d experts. The main idea is that the algorithm maintains weights w_f^t on the an expert f based on its performance up to the current round t.

- Prediction step: At prediction step, expert f is selected with probability $\pi_f^t = \frac{w_f^t}{\sum_f w_f^t}$ and it's prediction is adopted for this round.

- Update Step: At update step, suppose an expert f incurs loss l_f^t. The weight of each expert according to exponential rule:

$$w_f^{t+1} = w_f^t (1 - \eta)^{l_f^t}$$

The original MW algorithm (Arora et al. [2012]) provides a bound for the total expected loss of the algorithm by the total loss of the best experts with the following theorem:

Theorem 1. *MW Regret Bound ([Arora et al., 2012] Assume that the loss* ℓ_f^t *is bounded in*

[0,1] and $\eta < \frac{1}{2}$. Then after T rounds, for any expert f among the d experts we have:

$$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell_f^t + \frac{\ln d}{\eta},$$

$$\text{Regret}(T) \leq O(\sqrt{T \ln d}) \text{ if } \eta = \sqrt{\frac{\ln d}{T}}$$

where π^t is the selection distribution over the set of experts at time t.

This first equation shows that the expected cumulative loss achieved by the MW algorithm is upper bounded by the cumulative loss of the best fixed expert in hindsight plus a constant term $\frac{\ln d}{\eta}$. The constant term scales with the number of experts d. If we set η to be $\sqrt{\frac{\ln d}{T}}$, the first equation can be rearranged into $Regret(T) = \sum_{t=1}^{T} \pi^t \ell^t - (1+\eta) \sum_{t=1}^{T} \ell_f^t \leq O(\sqrt{T \ln d})$. In other words, this powerful theorem shows that MW algorithm achieves sub-linear regret.

3.2.2 Group-aware MW algorithm

Blum et al. [2018] first proposed a group-aware version of the MW algorithm for achieving fairness in online adversarial setting, where the examples are not i.i.d sampled from the distribution. The fairness metric they use is equalized error rates. The idea is to maintain separate set of weights for each group attribute z. They demonstrated that this is necessary to achieve equalized error rates across groups.

They presented the regret and equalized error rate achieved by the algorithm.

Theorem 2. *GroupAware Regret Bound ([Blum et al., 2018])*

Assume that the loss ℓ_f^t is bounded in [0,1] and $\eta < \frac{1}{2}$.

$$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell_f^t + 2\frac{\ln d}{\eta}$$

where π^t is the selection distribution over the set of experts at time t.

The regret of the group-aware version of the MW is almost the same as the original regret bound, except for a multiplier of 2 on the constant term $\frac{\ln d}{\eta}$.

Theorem 3. *GroupAware Equalized Error Rate (EqERR) ([Blum et al., 2018])*
Let ERR_z be the error rates on group z, and z^* be the group with the lowest error rates. Let $\eta < 1/2$,

$$|ERR_A - ERR_B| \leq 5\eta ERR_{f^*(z^*)} + \frac{\ln d}{\eta}$$

where $f^*(g^*)$ is the best expert on the group with the lowest error rate.

The above theorem shows that the equalized error rates of the algorithm is also upper bounded by the equalized error rates of the best expert.

3.3 Motivation for our work

3.3.1 Need to use distinguish error types

One potential drawback of the group-aware algorithm is that it only bounds the performance of the overall algorithm errors for each group, without a guarantee of how the errors will distribute across the label classes. In many real life applications, false positive rates and false negative rates could have very different implications and costs. In the COMPAS example Angwin et al. [2016] shown in the first chapter, the algorithm has approximate the same accuracy (error rates) for black and white defendants. The algorithm is biased towards black defendants since it has a much higher false positive rate for black defendants where they could be mis-classified as being high risk and arrested. This showcases that equalized error rates is not a suitable metric to measure fairness if FPR and FNR have different implications. In this work, we use equalized odds as the fairness metric, which balances both FPR and FNR across groups.

3.3.2 Need to care about label imbalance

Beyond the scope of fairness, unbalanced label class is a fairly common phenomenon in many machine learning applications. For a highly imbalanced distribution, even if the aggregation algorithm performs badly on the minority label class, and the regret or accuracy

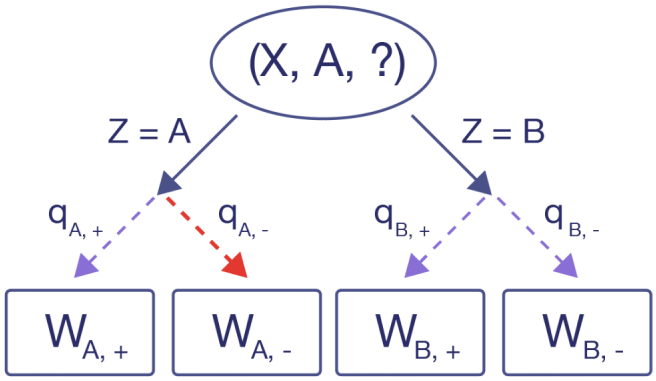


Figure 3-5: An example of predicting one example in G-FORCE .

might still look decent and this concerning problem gets swept under the rug. It's important for the aggregation algorithm to perform well on both label classes.

3.3.3 Need to consider delayed feedback

In many real world applications, true labels or outcomes are not instantly revealed and an algorithm often needs to work with delayed feedback. One example of constant delay is college admission process we described. During the rolling admissions process, the performance of a student is generally evaluated at the end of the semester, while colleges typically need to offer admission in a rolling basis. There is a constant gap between decision time (college offers admission) and feedback time (the admitted students' performances are evaluated).

3.4 G-FORCE algorithm

We propose a novel randomized MW algorithm that achieves EqOdds in an online stochastic setting. In order to satisfy EqOdds, we also need a provable bound on the number of false positives and false negatives made by the algorithm on each group. The idea is to run separate MW instances not only for groups but also for label classes, where each MW instance has a separate set of weights for the experts. Throughout the chapter, we uses tuple (z, y) to refer to a MW instance trained for subset of data with group z and label y. Each

MW instance associates a weight to a classifier f for group z and label y; e.g. the weight of classifier f for group A and negative label examples is denoted as $w_{f,A,-}$.

In figure-3-5, we illustrate how G-FORCE selects a MW instance to use. For an example from group A, G-FORCE follows the path $Z = A$ and goes to the right branch. At this point, the label is not known but G-FORCE needs to select between $(A, +)$ and $(A, -)$. For this purpose, G-FORCE constructs a **meta selection probability** q, which it uses to select instance (z, y) with probability $q_{z,y}$ at each round . The figure illustrates the case where $(A, -)$ gets selected.

We show that it is possible to bound regret, FPR and FNR as a function of the meta selection probability q. Moreover, the bounds can be further optimized by choosing the optimal meta selection probability that balance between regret, FPR and FNR. For the sake of clarity, the rest of the proposal we consider binary classification with two sensitive groups, though the algorithm can be easily extended to multi-group and multi-class problems.

3.4.1 G-FORCE mechanism

We use an example to illustrate the mechanism of G-FORCE for one round. The mechanism of G-FORCE is explained in Figure 3-6. At each round, G-FORCE takes in an example (x, z). G-FORCE works in three steps: **optimization step**, **prediction step** and the **update step**. The pseudo code for the algorithm is presented in 1.

Optimization Step G-FORCE first selects an appropriate MW instance to use. While group attribute z is known, at this point G-FORCE doesn't know the label yet, and has to choose between instance $(z, +)$ and instance $(z, -)$. G-FORCE constructs a meta selection probability q to select between the two instances, where $q_{z,+}$ and $q_{z,-}$ are the probability of selecting $(z, +)$, and $(z, -)$ respectively.

In the case that G-FORCE selects the wrong instance (for example, true label is $-$ but $(z, +)$ is selected), we refer to the additional losses as **cross-instances cost** $\alpha_{z,+}$ (formal definition in next section). This meta selection probability allows us to explicitly construct a upper bound on regret, FPR, and FNR as three functions of q. We later show $q_{z,+}$ and $q_{z,-}$ can be explicitly set to tighten this bound by solving an optimization problem that balances

Algorithm 1 GFORCE Algorithm

Initialize $w^1_{f,z,y} = 1$ for each $f \in \mathcal{F}, z \in \{A, B\}, y \in \{+, -\}$.
Initialize $q^1_{z,y} = \frac{1}{2}$ for each $z \in \{A, B\}, y \in \{+, -\}$.
Initialize $\eta < \frac{1}{2}$.

for $t \leftarrow 1, ..., T$ **do**
A new example (x^t, z^t) comes in

Obtain $\hat{y}^t_f = f(x^t, z^t)$, for each $f \in \mathcal{F}$

Optimization step:

Obtain the optimal meta selection probability $\mathbf{q}^*$

Selection step:

Select expert f with $\pi^t_{f,z} = \begin{cases} \pi^t_{f,z,+} = \frac{w^t_{f,z,+}}{\sum_{f \in \mathcal{F}} w^t_{f,z,+}} & \text{with probability } q_{z,+} \\ \pi^t_{f,z,-} = \frac{w^t_{f,z,-}}{\sum_{f \in \mathcal{F}} w^t_{f,z,-}} & \text{with probability } q_{z,-} \end{cases}$

Obtain loss $\ell^t_f = \ell(\hat{y}^t_f, y^t)$ for each classifier $f \in \mathcal{F}$

Update step: Update the weights table according to the exponential rule:

$$w^{t+1}_{f,z,y} = w^t_{f,z,y}(1-\eta)^{\ell^t_f \mathbb{1}\{Z=z\}\mathbb{1}\{Y=y\}}$$

end for

the three functions. The parameters of the three functions depend on statistics $p_z, \mu_{z,y}, \alpha_{z,y}$, which can all be estimated on the fly. We refer to these statistics as G-FORCE Statistics, and the optimal solution of the optimization problem as $\mathbf{q}^*$.

Prediction Step Suppose instance $(z, +)$ is selected, G-FORCE uses normalized weights $\pi_{f,z,+} = \frac{w_{f,z,+}}{\sum_f w_{f,z,+}}$ to sample an expert f, and adopts f's prediction for this round.

Update Step After the prediction, the true label y is observed and each expert f produces loss $\ell^t_{f,z,y} = \ell(f(x,z), y)$. G-FORCE only updates the weights for instance (z, y) with the exponential rule

$$w^{t+1}_{f,z,y} = w^t_{f,z,y}\,(1-\eta)^{\ell^t_{f,z,y}}\,.$$

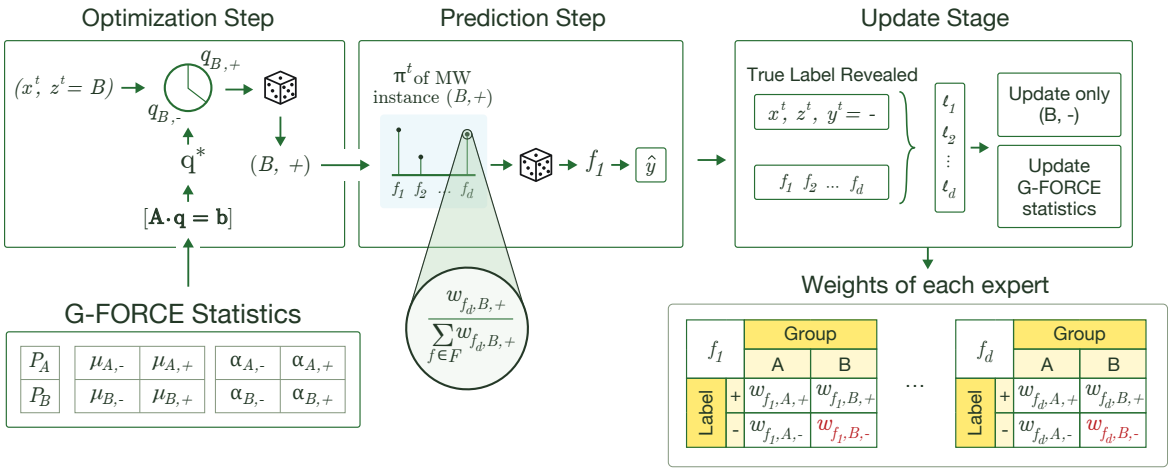


Figure 3-6: This figure shows how G-FORCE process an input pair (x, z), where z assumed to be B. In the optimization step, G-FORCE samples from PMF $[q_{B,+}, q_{B,-}]$ constructed from G-FORCE statistics and selects MW instance (B,+) to use. In prediction step, instance (B,+) samples a classifier f_1 to predict. In the update stage, the true label revealed to be $-$, indicating that G-FORCE selected the wrong instance to use in the first stage. G-FORCE only updates the weights for the correct instance (B,-), as well as the G-FORCE statistics.

Multiplicative Weights	$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell^t_{f^*} + \frac{\ln d}{\eta}$
GroupAware	$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell^t_{f^*} + 2\frac{\ln d}{\eta}$
G-FORCE	$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell^t_{f^*} + 4\frac{\ln d}{\eta} + h_{REG}(\mathbf{q})$
G-FORCE (delayed)	$\sum_{t=1}^{T} \pi^t \ell^t \leq (1+\eta) \sum_{t=1}^{T} \ell^t_{f^*} + 4\frac{\ln d}{\eta} + D^{max} h_{REG}(\mathbf{q})$

Table 3.3: Comparison on regret bound for the three algorithms.

where η can be interpreted as the learning rate. When η is large, the weight decay is faster. In addition, we also update the G-FORCE statistics used to compute $\mathbf{q}^*$. Note that although we recalculate $\mathbf{q}^*$ at early rounds since the estimation of G-FORCE statistics has not converged, As time goes on, the estimation of G-FORCE statistics converge to the true value, and $\mathbf{q}^*$ would also converge.

3.4.2 Theoretical Analysis of G-FORCE

One key contribution of this thesis is to show that: (1) the fairness loss in G-FORCE can be asymptotically upper bounded as a function of $q_{z,+}$ and $q_{z,-}$,and (2) the function values can be reduced to zeros by solving for $q_{z,+}$ and $q_{z,-}$, which further minimizes the upper bound.

Let $\mathbf{q} = [q_{A,-}, q_{B,-}, q_{A,+}, q_{B,+}]^T$ be the vector of meta selection probability . Specifically,

$$|FPR_A - FPR_B| \leq |c_{FPR} + h_{FPR}(\mathbf{q})|$$

$$|FNR_A - FNR_B| \leq |c_{FNR} + h_{FNR}(\mathbf{q})|$$

where c_{FPR}, c_{FNR} are constants that depend on the factors intrinsic to the problem (data distribution and the underlying metrics of the experts), and h_{FPR}, h_{FNR} are functions of meta selection probability $\mathbf{q}$. A formal version of the theorem is stated in Theorem- 6.

In this section, we aim to develop an upper bound on EqOdds for G-FORCE . We start by first providing an upper bound on regret for the worst cases scenarios, as well as a lower bound on regret for the best case scenarios (we leave the proof to the appendix).

Since there is randomness involved in the selection of MW instances, we define the costs of using a sub-optimal instance as cross-instances cost.

Definition 3.4.1 (Cross-instances cost)**.** *Let $\pi^t_{f,z,y} = \frac{w^t_{f,z,y}}{\sum_f w^t_{f,z,y}}$ denote the probability of choosing expert f when using instance (z,y). We define the cross-instances cost at round t as the difference in expected loss between selecting right instance (z, y) and wrong instance (z, y'):*

$$\alpha^t_{z,y'} = \underbrace{\sum_{f \in \mathcal{F}} \pi^t_{f,z,y'} \cdot \ell^t_{f,z,y}}_{\textit{expected losses with wrong instance (z,y')}} - \underbrace{\sum_{f \in \mathcal{F}} \pi^t_{f,z,y} \cdot \ell^t_{f,z,y}}_{\textit{expected losses with instances (z,y)}}$$

For example, $\alpha_{z,-}$ is the cross-instances cost of selecting the wrong MW instance $(z, -)$ when the actual example has $y = +$. The cross-instances cost is non-negative since the expected losses using the wrong instance would be larger than the expected losses using the correct instance. The cross-instances cost is larger when the weight vector learned by the wrong MW instance and the weight vector learned by the right MW instance are more disparate.

Implication of cross-instances cost Note that how large cross-instances cost is depends on the performance of black box experts and is not known in advance. In practice, since G-FORCE keeps track of weights $\pi_{f,z,y}$, cross-instances cost can be estimated on the fly. At the end of each round, the true label is revealed and the weights are updated. The estimation for α is updated at each round after the MW weights are updated. Let us rearrange the terms of cross-instances cost as following:

$$\alpha^t_{z,y'} = \sum_{f \in \mathcal{F}} (\pi^t_{f,z,y'} - \pi^t_{f,z,y}) \cdot \ell^t_{f,z,y}$$

This rearrangement enables us to analyze this cross-instances cost in detail. Here we explain each component in the definition of cross-instances cost :

- For a single expert f, $(\pi^t_{f,z,y'} - \pi^t_{f,z,y})$ is the difference in probability, where the first term is the probability of choosing the expert f when algorithm picks the wrong instance (z, y') and the second term is the probability when the algorithm picks the correct instance (z, y). After revelation of the label at the end of the round t this can be calculated. We calculate this after updating the weights.

- $\ell^t_{f,z,y} = \ell^t_f \mathbb{1}\{Z^t = z\}\mathbb{1}\{Y^t = y\}$ is the loss of an expert f at round t when the example comes from group z with label y. If expert f is a good expert for instance (z, y), $l_{f,z,y}$ would be small (equal to zero in binary classification). On the other hand, if the expert f is a bad expert for instance (z, y), $l_{f,z,y}$ would be large (equal to one for binary classification).

3.4.2.1 Regret Bound

Since we have a separate MW instance for each combination of group and label class (z, y), we can first develop regret bound for each MW instance separately. We use $\mathbb{E}[L_{z,y}]$ to indicate G-FORCE 's cumulative expected loss on MW instance (z, y).

Theorem 4 (Regret Upper Bound)**.** *Let f^* be the best expert in hindsight.*

$$\mathbb{E}[L_{z,y}] \leq (1+\eta)L_{f^*,z,y} + \frac{\ln d}{\eta} + \sum_t q^t_{z,y} \cdot \alpha^t_{z,y} \tag{3.2}$$

The overall cumulative expected loss $\mathbb{E}[L]$ of G-FORCE can be bounded by:

$$\mathbb{E}[L] \leq (1+\eta)L_{f^*} + 4\frac{\ln d}{\eta} + h_{REG}(\mathbf{q}) \tag{3.3}$$

where $h_{REG}(\mathbf{q}) = \sum_{z\in\{A,B\},y\in\{+,-\}} \sum_t q^t_{z,y} \cdot \alpha^t_{z,y}$.

Implication of the regret bound This upper bound shows that the expected cumulative loss is upper bounded by the summation of three terms: (1) the cumulative loss of the best expert in hindsight, (2) the constant term $4\frac{\ln d}{\eta}$, (3) the function h_{REG} of meta selection probability q. Here we breakdown the components in the function h_{REG}:

- The cross-instances cost at round t is the difference in expected loss between selecting right instance (z, y) and wrong instance (z, y'). For MW instance (z, y), $q^t_{z,y}\alpha^t_{z,y}$ is the cross-instances cost of instance (z, y) weighted by the meta selection probability of choosing instance (z, y).

- The function h_{REG} is the cross-instances cost summed over all MW instances. The value of h_{REG} can be minimized by choosing proper values of meta selection probability q. For instance with higher cross-instances cost , we might want to assign a lower meta selection probability q.

In order to show the bound for differences in FPR across groups (i.e. for EqOdds), we also provide a lower bound on the expected cumulative loss of G-FORCE.

Lemma 5 (Lower Bound)**.** *Let f^* be the best expert in hindsight. Then, G-FORCE's expected cumulative loss is lower bounded by:*

$$\mathbb{E}[L] \geq \gamma(\eta) \cdot L_{f^*} + h_{REG}(\mathbf{q}). \tag{3.4}$$

where $\gamma(\eta)$ is defined as $\gamma(\eta) = \dfrac{\ln(1-\eta)}{\ln\left(1-\eta(1+\eta)\right)}$.

3.4.2.2 Fairness bound

For the bound on fairness, we assume each expert $f \in \mathcal{F}$ satisfies ϵ-EqOdds with respect to data distribution $\mathbb{P}_{x,y,z}$ for some unknown ϵ, i.e., for $y \in \{+,-\}$;

$$\left| \mathbb{E}_{x,y,z}\left[\frac{L_{f,A,y}}{C_{A,y}}\right] - \mathbb{E}_{x,y,z}\left[\frac{L_{f,B,y}}{C_{B,y}}\right] \right| \leq \epsilon, \tag{3.5}$$

where $C_{z,y}$ is the cardinality of group z and label y. Here ϵ represents the maximum absolute difference of FPR and FNR between two groups, and we don't put restriction on the value of ϵ.

Theorem 6 (Fairness Bound)**.** *Let z^* be the group that with the lowest FPR (FNR), and let $f^*(z^*)$ be the expert with lowest FPR (FNR) for group z^*, where group $z*$ is the group with lower FPR (FNR). For G-FORCE , and for $\boldsymbol{q} = [q_{A,-}, q_{B,-}, q_{A,+}, q_{B,+}]^T$, we have:*

$$|FPR_A - FPR_B| \leq |\left(1+\eta-\gamma(\eta)\right) FPR_{f^*(z^*)} + \underbrace{\epsilon(1+\eta)}_{c_{FPR}} + \underbrace{\frac{\alpha_{A,-}q_{A,-}}{p_A\mu_{A,-}T} - \frac{\alpha_{B,-}q_{B,-}}{p_B\mu_{B,-}T}}_{h_{FPR}(\boldsymbol{q})}| \tag{3.6}$$

$$|FNR_A - FNR_B| \leq |\left(1+\eta-\gamma(\eta)\right) FNR_{f^*(z^*)} + \underbrace{\epsilon(1+\eta)}_{c_{FNR}} + \underbrace{\frac{-\alpha_{A,+}q_{A,+}}{p_A\mu_{A,+}T} + \frac{\alpha_{B,+}q_{B,+}}{p_B\mu_{B,+}T}}_{h_{FNR}(\boldsymbol{q})}| \tag{3.7}$$

Implication of fairness bound The absolute difference in FPR ($|FPR_A - FPR_B|$) can be upper bounded by the summation of three term: (1) The FPR of the best expert for the best group; (2) constant term c_{FPR}; (3) and a function $h_{FPR}(q)$ of meta selection probability q. The same analogy applies to the FNR bound ($|FNR_A - FNR_B|$). Here we give an explanation for the individual terms in the bound:

- The first term in FPR bound depends on the best expert f^*, where best is defined as the expert that achieves the lowest FPR over all groups. This is similar to the regret

bound in the sense that the bound depends on the expert with lowest FPR, which is equivalent to the best expert on instance $(z, -)$.

- The second term c_{FPR} is a constant that depends on the maximum difference of $|FPR_A - FPR_B|$ for an individual expert (ϵ) and η. We don't put any assumption on the value of ϵ and treat it like a property of the black-box experts.

- The last term is a function $h_{FPR}(\mathbf{q})$ of q. The numerator $\alpha_{A,-}q_{A,-}$ is expected cross-instances cost for instance $(A, -)$, and the denominator $p_A\mu_{A,-}T$ is the expected number of examples from group A with $-$ label. This ratio can be interpreted as the additional expected error rate for instance $(A, -)$ due to selecting a wrong instance of MW. Since errors made on instance $(A, -)$ are false positives for group A, this is equivalent to the additional FPR for group A due to selecting a wrong instance of MW. Thus $h_{FPR}(\mathbf{q})$ is the difference in FPR for group A and group B due to selecting a wrong instance of MW. Since $\alpha_{A,-}, p_A, \mu_{A,-}T$ and $\alpha_{B,-}, p_B, \mu_{B,-}T$ can all be estimated on the fly, the value of function $h_{FPR}(\mathbf{q})$ can be set to zero by choosing $q_{A,-}$ and $q_{B,-}$.

3.4.3 Implication of the theoretical result

The fairness bound shows the asymptotic result that after the optimization step converges, the absolute difference of FPR/FNR between groups can be bounded by constants c_{FPR} and c_{FNR} . In appendix, we show that these constants depend on factors intrinsic to the problem: properties of the distribution and the fairness of the base expert (ϵ,FPR_{f^*},FNR_{f^*}). In the appendix, we also compare the theoretical bound of EqOdds with the achieved value of EqOdds in experiments to get a sense of the tightness of the bound under different distributions.

3.4.3.1 Optimal balance between regret and fairness

In this section, we show that h_{FPR} and h_{FNR} can be set to zeros by solving the following set of functions:

$$\begin{bmatrix} \frac{\sum_t \alpha^t_{A,-}}{p_A \cdot \mu_{A,-} \cdot T} & \frac{-\sum_t \alpha^t_{B,-}}{p_B \cdot \mu_{B,-} \cdot T} \end{bmatrix} \begin{bmatrix} q_{A,-} \\ q_{B,-} \end{bmatrix} = 0, \tag{3.8}$$

$$\begin{bmatrix} \frac{-\sum_t \alpha^t_{A,+}}{p_A \cdot \mu_{A,+} \cdot T} & \frac{\sum_t \alpha^t_{B,+}}{p_B \cdot \mu_{B,+} \cdot T} \end{bmatrix} \begin{bmatrix} q_{A,+} \\ q_{B,+} \end{bmatrix} = 0. \tag{3.9}$$

In addition, the upper bound for regret in Eq. (4) can also be tighten by adding the following constraint:

$$\begin{bmatrix} \sum_t \alpha^t_{A,-} & \sum_t \alpha^t_{B,-} & \sum_t \alpha^t_{A,+} & \sum_t \alpha^t_{B,+} \end{bmatrix} \begin{bmatrix} q_{A,-} \\ q_{B,-} \\ q_{A,+} \\ q_{B,+} \end{bmatrix} = 0, \tag{3.10}$$

Given all these equations, constraints and inequalities we can define the following optimization step.

Optimization step At each round, we are led to solve three functions of $\mathbf{q}$ where function parameters are determined by the equations (3.8), (3.9) and (3.10) defined above.

$$\mathbf{q}^* = \lambda_1 h_{REG}(\mathbf{q}) + \lambda_2 h_{FPR}(\mathbf{q}) + \lambda_3 h_{FNR}(\mathbf{q}) \tag{3.11}$$

where $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \lambda_3]$ is a vector balancing the importance of regret, equalized FPR, equalized FNR that can be provided on a case-by-case basis for different applications. In our experiments, we solve (3.11) by using a Sequential Least Squares Programming method (SLSQP) and setting $\lambda_1 = \lambda_2 = \lambda_3 = 1$.

In practice, G-FORCE can accommodate different use cases by setting different $\boldsymbol{\lambda}$ at

each round. For example, during the early rounds, since the algorithm hasn't converged yet, we might want to set $\boldsymbol{\lambda}$ for equalized FPR and equalized FNR to be smaller to penalize the algorithm less for unfairness. Another scenario is a shifting distribution, where G-FORCE can be adaptive to the distribution with different $\boldsymbol{\lambda}$.

3.5 G-FORCE for delayed feedback

In many real world applications, true labels are not instantly revealed and an algorithm often needs to work with delayed feedback. For example, during the college admissions process, the performance of a student is generally evaluated at the end of each term, while colleges typically offer admission decisions in mid-year. Similarly, when an individual applies for a loan, the bank often needs to wait for some time to know whether the applicant will default or not. The duration of the delay could be a constant, a random variable, or a function of time.

In the constant delay setting, we assume that the delay duration is a constant determined by the sensitive attribute and the true label. Therefore, for a example of group z and label y arrives at time t, the algorithm will make a prediction at time t, but the true label will be only revealed at the end of time $t + \tau_{z,y}$ (where $\tau_{z,y}$ is some delay duration). For simplicity, we assume $t + \tau_{z,y} < T$. Thus the indices of feedback at time t is a set $\mathcal{D}^t_{z,y} = \{t' : t' < t, t' + \tau_{z,y} = t\}$. In contrast, for the non-delayed setting, the indices of feedback at time t is a singleton $\mathcal{D}^t_{z,y} = \{t\}$.

We next present how G-FORCE 's theoretical bound changes when the feedback is delayed.

3.5.1 Theoretical Result Under Delayed Feedback

Under the construction of the G-FORCE , both regret and fairness bound are a small modification of the original theoretical result. We leave the details of the proof to appendix.

Theorem 7 (Regret Bound)**.** *Let f^* be the best expert in hindsight. Let $D_{max} = \max_{t,z,y} |\mathcal{D}^t_{z,y}|$ be the maximum cardinality of the feedback set of all MW instances.The cumulative ex-*

pected loss $\mathbb{E}[L]$ of G-FORCE can be bounded by:

$$\mathbb{E}[L] \leq (1+\eta)L_{f^*} + 4\frac{\ln d}{\eta} + D^{max} h_{REG}(\mathbf{q}), \tag{3.12}$$

where $h_{REG}(\mathbf{q}) = \sum_{z\in\{A,B\}, y\in\{+,-\}} \sum_t q^t_{z,y} \cdot \alpha^t_{z,y}$.

This upper bound shows that the expected cumulative loss has an additional multiplicative factor of D_{max} on the function $h_{REG}(\mathbf{q})$ compared to the non-delayed setting. Nevertheless, the order of regret is still the same. This result is consistent with Joulani et al. [2013], in the sense that delay feedback normally increases regret in an additive way for stochastic setting.

Theorem 8 (Fairness Bound)**.** *Let $FPR_{f^*}(FNR_{f^*})$ be the classifier achieving lowest expected cumulative loss on subset $\{z,-\}(\{z,+\})$, $\forall z \in \{A,B\}$. For G-FORCE , we have:*

$$\begin{aligned} &|FPR_A - FPR_B| \\ &\leq |\left(1+\eta-\gamma(\eta)\right) FPR_{f^*} + \epsilon(1+\eta) + \\ &\qquad \underbrace{\left(\frac{D^{max}_{A,-} q_{A,-} \cdot \sum_t \alpha^t_{A,-}}{p_A(1-\mu_{A,+})T} - \frac{D^{max}_{B,-} q_{B,-} \cdot \sum_t \alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)}_{h_{FPR\_delay}(\mathbf{q})}| \end{aligned} \tag{3.13}$$

where $D^{max}_{A,-} = max_t|\mathcal{D}^t_{A,-}|$ *and* $D^{max}_{B,-} = max_t|\mathcal{D}^t_{B,-}|$

For the fairness bound, compared to the original G-FORCE 's bound, the function h_{FPR} also depends on the maximum feedback cardinality $D^{max}_{A,-}, D^{max}_{B,-}$.

3.6 Empirical evaluation of G-FORCE

In this section we present G-FORCE 's performance on real and synthetic datasets. G-FORCE keeps three statistics that are necessary to compute parameters for functions h_{FPR} and h_{FNR}: (i) the probability of a sample coming from group z, denoted by p_z, (ii) the *base rates of outcomes*, denoted by $\mu_{z,y}$, and (iii) the cross-instance costs α, which is estimated as differences of expected loss between using a right instance and a wrong instance. All

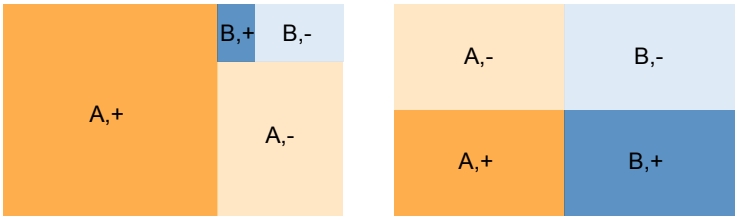
three statistics above are estimated with Bayesian and Dirichlet Prior. We use $\eta = 0.35$ in experiments.

3.6.1 Case study: Synthetic Datasets

It is important to test what can be achieved for both algorithms under extreme scenarios.

Datasets We create a synthetic data framework that allows us to control the distributions and experts with certain properties. The balance between group attribute and labels is controlled by setting parameters $p_A, \mu_{A,+}, \mu_{B,+}$. For this purpose, we create one synthetic dataset with imbalanced setting one with balanced setting. The first one is imbalanced setting where group A is the majority group with higher percentage positive labels, and group B is the minority group also with lower percentage positive labels. In particular, we have $p_A = 0.9, \mu_{A,+} = 0.7, \mu_{B,+} = 0.3$. The second one is the balanced setting where each group-label combination has the equal number of examples. We visualize these two settings in figure-3-7.

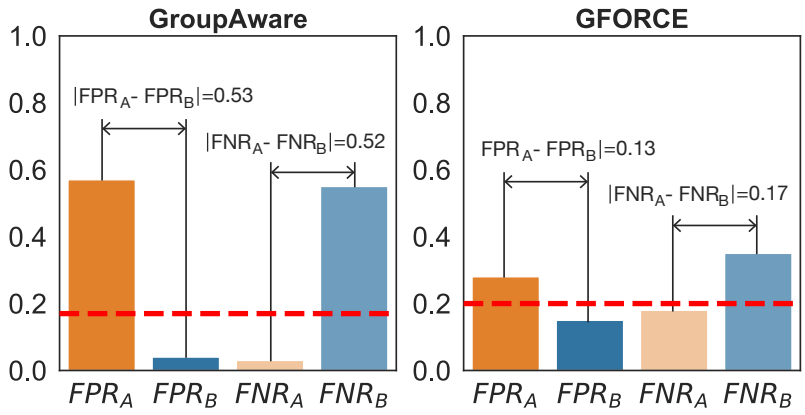
Creating Black-box Experts It is also important to test the efficacy of our approach when experts have disparate performances or are extremely biased towards different groups. For binary classification with two groups, we create four extreme expert, where each is perfect (100% accurate) for one of the group-label subsets ($\{A, +\}, \{A, -\}, \{B, +\}, \{B, -\}$), and random ($50\%$ accurate) for the other three. Thus for each group-label subset, there is at least one perfect expert/classifier.



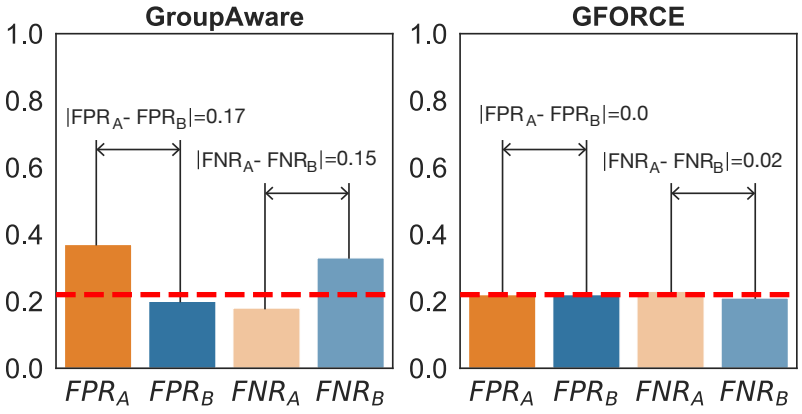
(a) Imbalanced Setting. (b) Balanced Setting.

Figure 3-7: The size of each color block is proportional to the number of examples in that group-label subset. Imbalanced setting is created with $p_A = 0.9, \mu_{A,+} = 0.7, \mu_{B,+} = 0.3$ and balanced setting is created with $p_A = 0.5, \mu_{A,+} = 0.5, \mu_{B,+} = 0.5$.

Results For each dataset, we repeat the experiments 100 times, each with 10000 samples from a specific distribution setting. For imbalanced setting, the results in Figure 3-8a shows that for GroupAware algorithm, the larger subsets $\{A, +\}$ and $\{B, -\}$ have nearly 100% accuracy while $\{A, -\}$ and $\{B, +\}$ have around 50% accuracy. The GroupAware algorithm, which runs only one MW instance per group attribute z, promotes selecting the perfect classifier for the larger group-label subset within each protected group. This leads to high error rates on the remaining subsets since their associated perfect classifiers are unlikely to be picked.



(a) Imbalanced Setting.



(b) Balanced Setting.

Figure 3-8: The achieved accuracy on group-label subsets for imbalanced setting ($p_A = 0.9, \mu_{A,+} = 0.7, \mu_{B,+} = 0.3$) and balanced Setting ($p_A = 0.5, \mu_{A,+} = 0.5, \mu_{B,+} = 0.5$). Left: GroupAware. Right: G-FORCE . The vertical black line denotes the standard deviation. The red dashed line is the overall accuracy.

Even for the perfectly balanced setting, G-FORCE achieves a more balanced accuracy in each subset and a more stable behavior compared to GroupAware as in Figure 3-8b.

	Group A	Group B	Positive label	# of rounds	p_A	$\mu_{A,+}$	$\mu_{B,+}$
Adult	White	non-White	income exceeds 50k/yr	24421	0.851	0.26	0.16
German	Male	Female	good credit score	300	0.853	0.73	0.50
COMPAS	White	non-White	low risk for recidivism	1584	0.398	0.54	0.39

Table 3.4: Summary statistics of datasets. Here p_A is the percentage of group A, $\mu_{A,+}$ is the percentage of positive labels in group A, and $\mu_{B,+}$ is the percentage of positive labels in group B.

Since the label distribution is balanced, $\{A, -\}$ and $\{A, +\}$ have the same accuracy when classifying an example from group A. GroupAware arbitrarily chooses between perfect classifier for $\{A, +\}$ or $\{A, -\}$ when classifying examples from group A, which leads to large deviations when considering errors on each more fine-grained subset (same analogy for group B). On the contrary, in both settings, G-FORCE is able to track the performance of the EqOdds on each group-label subset and compensate their differences in terms of accuracy. In the plot, the red dashed line represents the overall error rates of the algorithms. As shown in the theoretical results, compared to GroupAware G-FORCE has a slightly increase in regret, and is reflected as the slight increase error rates in the experiment.

3.6.2 Case study: Real Data sets

Datasets We use the Adult, German Credit and COMPAS datasets, all of which are commonly used by the fairness community. Adult consists of individuals' annual income measurements based on different factors, and the goal is to predict whether someone's income exceeds $50k$/yr based on census data. The group attribute is race, and the two groups are White (Group A) and non-White (Group B). In the German dataset, people applying for credit from a bank are classified as "good" or "bad" credit based on their attributes. The group attribute is gender, where the two groups are male (Group A) and female (Group B). COMPAS provides a likelihood of recidivism based on a criminal defendant's history. The group attribute is again race, where the two groups are White (Group A) and non-White (Group B).

Creating Black-box Experts The set of black box experts$\mathcal{F}$ that form the black-box experts are: Logistic Regression (LR), Linear SVM (L SVM), RBF SVM, Decision Tree

(DT) and Multi-Layer Perceptron (MLP). These classifiers are trained using 70% of the data set. The remaining 30% of the dataset is set aside to simulate the online arrival of individuals. We compare our G-FORCE algorithm with the GroupAware in terms of regret and fairness. We repeated the experiments 1000 times for `German` and `COMPAS`, as well as 10 times for `Adult`, by randomizing the arrival sequence of individuals.

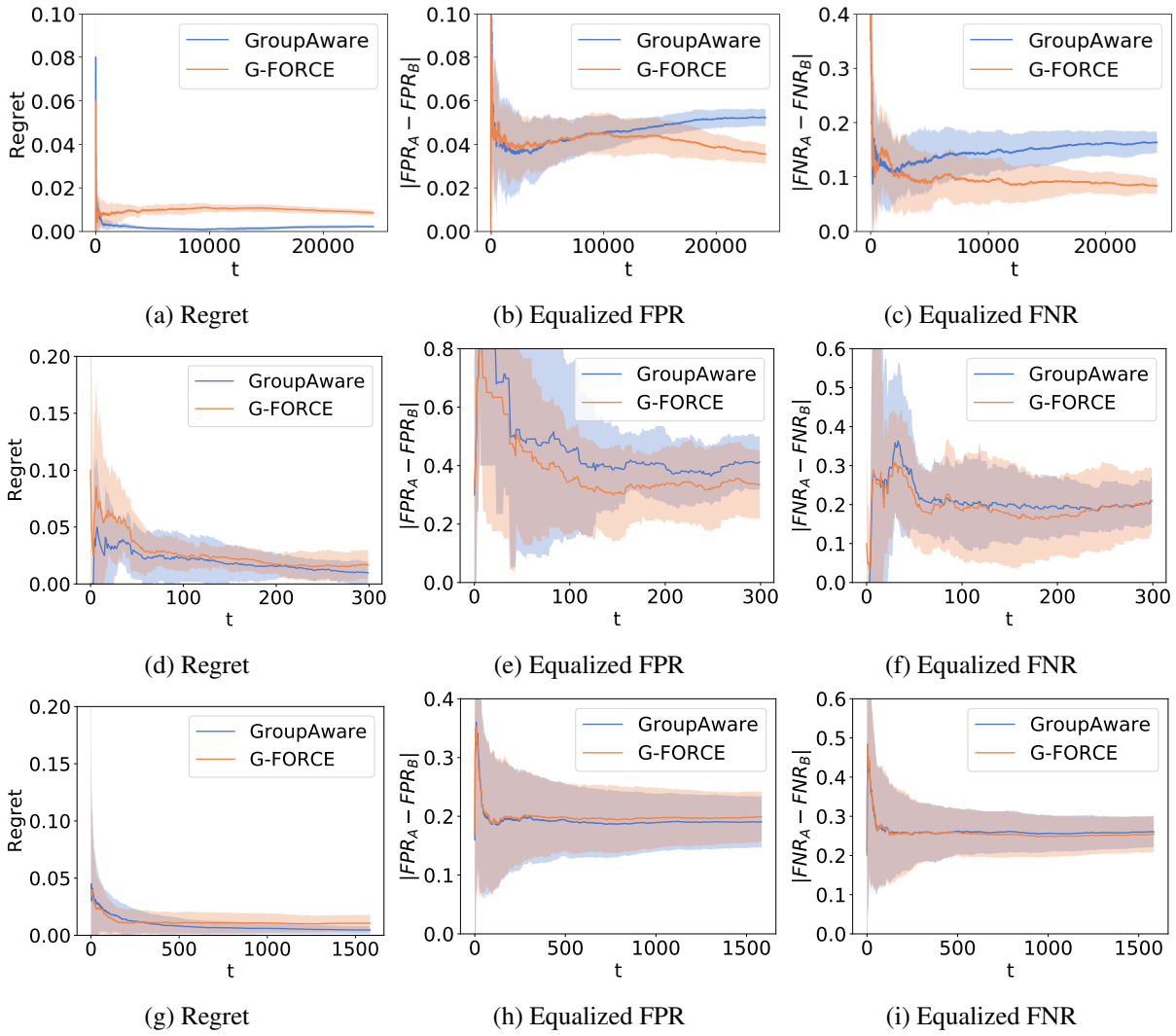
		Individual Experts					Combined Experts	
		L SVM	RBF SVM	DT	MLP	LR	Group-Aware	G-FORCE
Adult	FPR	0.022	0.046	0.043	0.047	0.047	0.052	**0.035**
	FNR	0.026	0.199	0.200	0.214	0.214	0.163	**0.083**
	EER	0.058	0.062	0.062	0.061	0.061	0.074	**0.069**
German	FPR	0.00	0.371	0.471	0.421	0.050	0.373	**0.329**
	FNR	0.000	0.320	0.770	0.680	0.650	0.207	**0.181**
	EER	0.090	0.090	0.208	0.210	0.280	**0.093**	0.098
COMPAS	FPR	0.190	0.150	0.160	0.158	0.240	0.191	**0.184**
	FNR	0.256	0.240	0.260	0.240	0.340	0.264	**0.249**
	EER	0.019	0.010	0.010	0.010	0.010	**0.016**	0.019

Table 3.5: ϵ-Fairness of base experts, GroupAware and G-FORCE .

Results in Table-3.6 show a general improvement in fairness over the GroupAware algorithm, both in terms of equalized FPR and FNR, along with a small increase in regret. For `Adult` data set, we plot the performance of the algorithm over time (Figure 3-9). Although `German` and `COMPAS` have fewer examples, and thus the standard deviation is higher to make a conclusion, there is still a slight improvement over fairness with slight increase in regret.

	Adult			Compas			German		
	FPR	FNR	Regret	FPR	FNR	Regret	FPR	FNR	Regret
GroupAware	0.05± 0.01	0.17± 0.02	**0.00**± 0.00	0.20± 0.04	0.27± 0.04	**0.01**± 0.00	0.40± 0.13	0.21± 0.08	0.01± 0.01
G-FORCE	**0.04**± 0.01	**0.08**± 0.01	0.01± 0.00	**0.18**± 0.03	**0.25**± 0.04	0.01± 0.01	**0.32**± 0.15	**0.18**± 0.01	0.01± 0.01

Table 3.6: Equalized FPR, equalized FNR and regret on real datasets. Lower numbers are better.



(a) Regret (b) Equalized FPR (c) Equalized FNR

(d) Regret (e) Equalized FPR (f) Equalized FNR

(g) Regret (h) Equalized FPR (i) Equalized FNR

Figure 3-9: G-FORCE shows a clear improvement over GroupAware on both equalized FPR (bottom left) and equalized FNR (bottom right) on adult dataset.

We also report the error rates and associated ϵ-fairness of each classifier in the appendix. The base classifiers expose similar and more mild behaviors (compared with in real datasets) which makes the task of the algorithm easier, and thus the results are less significant compared to the real dataset.

3.7 Conclusion

Many real world applications require decision makers to make decisions in a sequential setting. Multiplicative weights algorithm is a classical no-regret algorithm used in sequential

learning. In the case the samples come from population groups, could we adapt the MW algorithm to guarantee fairness as well? Blum et al. [2018] first proposed to adapt MW algorithm for online learning with fairness guarantees. The idea is to run separate instances of the MW for each group in order to equalize the error rates among groups.

However, equalized error rates is a very simplified notion of fairness. In many real world applications, the impact or cost of false positives and false negatives could be very different. For example, as discussed in the first chapter, in the COMPAS example, the model satisfies approximate equalized error rates on white and black defendants, but the model has a much higher false positive rates for black defendants. In this paper, we introduce G-FORCE , a randomized algorithm achieving approximate EqOdds, which guarantees that both false positive rate and false negative rate are equalized. We achieve this by keeping separate instances of MW instance for each sensitive group-label combination (z, y). This allows us to provide an upper bound for the number of false positives and negatives for each group. We show that, given a set of black box experts, it is possible to obtain an optimal meta selection probability for choosing between different MW instances that, in turn, will balance regret and fairness. We also show that the algorithm can work in the delayed feedback setting, where the true label is not revealed instantly after a decision is made.

G-FORCE can be applied to a wide range of applications as it could work alongside with human decision makers and correct potential biases. A user could choose the hyper-parameter λ to set a desirable trade-off between fairness and accuracy. We are also deploying the algorithm to a real world application.

Chapter 4

Study of Fairness with Feedback Loop

4.1 Introduction

The first part of this thesis considers whether it is possible to produce fair decisions from black-box predictions in an online setting. One important assumption in this study is that a past decision will not impact future distributions of features. As we saw in the first chapter, decisions could create a feedback loop that nudges feature distributions of different groups in different ways. The change in feature distributions will in turn change the target variable distribution. In classical machine learning settings, the goal is to create a model that minimizes empirical risk with respect to a dataset. In this setting, fairness constraints can be enforced through a constrained optimization. However, these predictions could lead to consequential decisions, because the predictions of the model could have long-lasting effects on target variable distribution beyond a single step.

In this chapter, we study whether enforcing fair decisions closes the gap of target variable distribution between advantaged and disadvantaged groups. The chapter is structured as follows:

- In section-4.2, we first use an example of loan application to showcase how fair decisions could shape underlying distribution undesirably.
- In section-4.3, we present our setting for modeling interactions between decision-makers and underlying distribution as Markov Decision Process. We focus on threshold policies, i.e. policies which assign positive decisions when features or target variable is above some threshold.
- In section-4.4, we first formally propose a metric to measure the distributional impact of algorithmic decisions on the target variable distributions. We identify the backfire effect – i.e. when policies result in a disproportionate impact on a protected group over the long term. Specifically, we can categorize the backfire effect into two scenarios: (1) within-group impact measures how a sequence of decisions shifts the distribution of the target variable of a group, and (2) between-group impact measures the absolute difference between two groups' within-group impact.
- In section-4.5, we investigate the impact of fair threshold policies, which are derived

from one-step constrained optimization subject to some fairness constraints. We investigate whether these fair policies could have a disparate impact on shaping the target variable distributions of different population groups.

- Lastly, in section-4.6, we conclude with key takeaways and considerations in designing policies that align with long-term fairness.

4.1.1 Related Work

Recently, a few works have studied the dynamics of algorithmic decisions and the underlying distributions [Liu et al., 2017, D'Amour et al., 2020a, Zhang et al., 2020]. Liu et al. [2017] first use loan application as an example to study how the variable of interests (credit scores) change as a result of decisions in a simple one-step feedback model. They demonstrate theoretically that under the one-step model, unconstrained optimization never decreases group-wise average credit scores while common fairness criteria could lead to a decrease in the group-wise average credit scores.

Later, D'Amour et al. [2020a] extends the one-step theoretical model to multi-step simulation using MDP. They show that multi-step simulation gives qualitatively different conclusions compared to the previous one-step analysis because of edge effects. In particular, constrained optimization could decrease group-wise average credit scores when there is a maximum cap on credit scores.

Most recently, [Zhang et al., 2020] models the dynamics under the partially observed Markov decision process (POMDP) where the hidden variable represents the binary qualification state. They study the equilibrium of qualification rates in the long term. However, in many real-world applications, the target variable could be continuous. For example, the target variable could be the probability of repayment for a loan application or the probability of re-offense in recidivism prediction. In this type of setting, the dynamics of the distribution target variable are more complex and cannot be simply captured by equilibrium analysis on the qualification rate.

In this chapter, we propose a setting that is more realistic and suitable for real-world applications. Our setting is different from the previous work in the following ways:

Name	Target Variable	Framework
Liu et al. [2018]	Continuous	MDP with linear transition function
D'Amour et al. [2020a]	Discrete	MDP with linear transition function
Zhang et al. [2020]	Binary	POMDP
Ours	Continuous	MDP with general transition functions

Table 4.1: Setting of the four frameworks.

- General transition functions: We provide a framework to model dependency between random variables using structural equations. This allows the dependency between features and variables to be a general function.

- Distributional change: In our work, we model the target variable as a continuous variable that measures the qualification probability of each individual. In this case, we can characterize the distributional change of the target variable beyond the mean.

- Metrics for disparate impact across groups: All three previous works provide an analysis of the group-wise outcome change separately. There is a lack of a clear metric to measure the disparity of outcome change between groups. We provide a new metric

4.2 Motivating Example

We next use a loan lending example to show how algorithmic decisions could further segregate distributions of different population groups. Loan lending is a classical example that has been widely used to study fairness Liu et al. [2018]. The Equal Credit Opportunity Act, a United States law enacted in 1974, makes it unlawful for any creditor to discriminate against any applicant on the basis of race, color, religion, national origin, sex, marital status, or age. Suppose a bank predicts whether approving or rejecting loan applications from a stream of applicants. To simplify the process, the only observed features are sensitive attribute and credit score. Each applicant has a group attribute $Z \in \{A, B\}$ and a discrete credit score $c \in [1, 10]$.

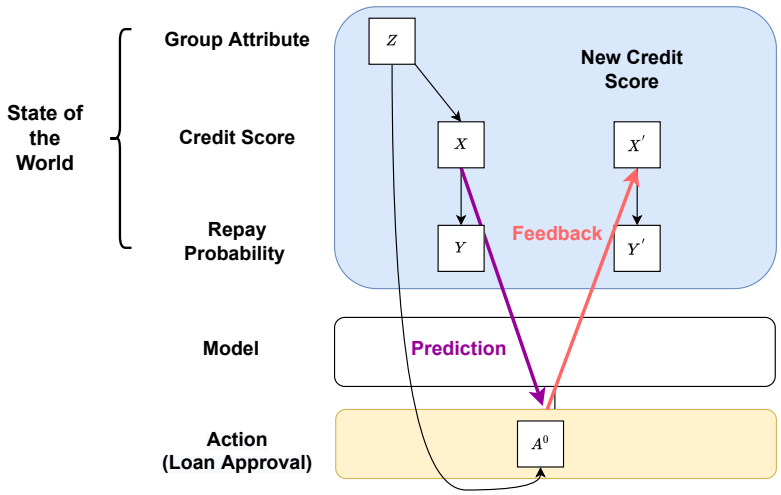


Figure 4-1: An overview of the feedback loop in the loan application example.

We first use figure-4-1 to illustrate the different components in the loan lending process as a causal graph.

- **State of the World** The state of world consists of a tuple (X, Y, Z).
  - Sensitive Attribute (Z): Each individual comes with a sensitive attribute $Z \in \{A, B\}$, such as race, gender etc. The sensitive attribute is time invariant.
  - Credit Score (X): Initially, each individual starts with a credit score X that depends on group attribute Z.
  - Repaying Probability (Y): The repaying probability Y is a function of the credit score $Y = X/10$.
- **Model**: A model takes the state of the world and generates a prediction for the repaying probability. The process is indicated by the purple link.
- **Loan Approval Decision** : Based on the prediction, a binary loan approval decision A is issued, which could potentially depend both on an applicant's credit score X and the sensitive attribute Z.
- **Feedback**: A decision will have a feedback effect (indicated by the red link) on the credit scores. In particular, if an applicant successfully repays a loan, the new credit score X' (or X^{t+1} in a multi-step process) will increase by 1 and the bank's utility

will increase by 1. If an applicant defaults, the new credit score will be decreased by 1 and the bank's utility will be decreased by 1. When an applicant's credit score decreases, so does the repaying probability.

We now illustrate what will happen in a one-step feedback loop for banks using the following policies: max profit, demographic parity, and equalized odds. Assuming there are 10 applicants, 5 from group A, and 5 from the group B, and $c_{max} = 10$. The initial credit scores X of the applicants are shown in Figure 4-2, which depends on the group membership.

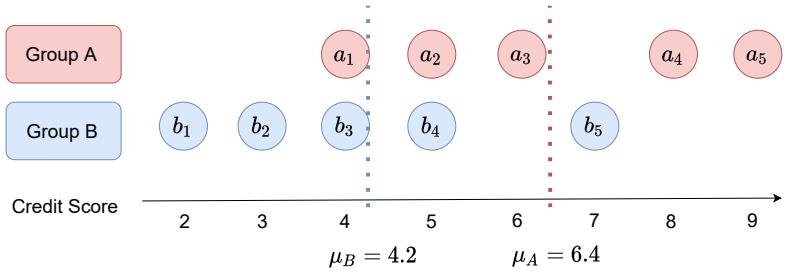


Figure 4-2: Initial credit scores distribution of group A (advantaged group) and group B (disadvantaged group).

The mean score of group A is $\mu_A = 6.4$, and the mean score of group B is $\mu_B = 4.2$, with their difference $\Delta = 2.2$. We refer to the group with higher initial mean as the advantaged group (group A).

We keep track of two metrics: the bank's utility and the group welfare disparity.

- Bank's utility: The bank's profit will be increased by 1 if an applicant repays, and will be decreased by 1 if an applicant defaults.

- Group disparity: We measure disparity as the absolute difference of group means of credit scores, i.e., $\Delta = |\mu_A - \mu_B|$.

Max Profit The first bank issues loans based on a fixed threshold on credit score regardless of group. Specially, an applicant will be approved if the credit score $c \geq 5$ since this is a break-even point for the bank. When an applicant has a credit score of 5, there are 50/50 chance that the applicant will default and the expected profit of the bank is 0.

	Group A						Group B					
Applicant	a_1	a_2	a_3	a_4	a_5	Mean	b_1	b_2	b_3	b_4	b_5	Mean
Credit Score X	4	5	6	8	9	6.4	2	3	4	5	7	4.2
Decision	0	1	1	1	1	0.8	0	0	0	1	1	0.2
New Expected X'	4	5	6.2	8.6	9.8	6.72	2	3	4	5	7.4	4.28

Table 4.2: Outcome when using a policy that has the same threshold regardless of group.

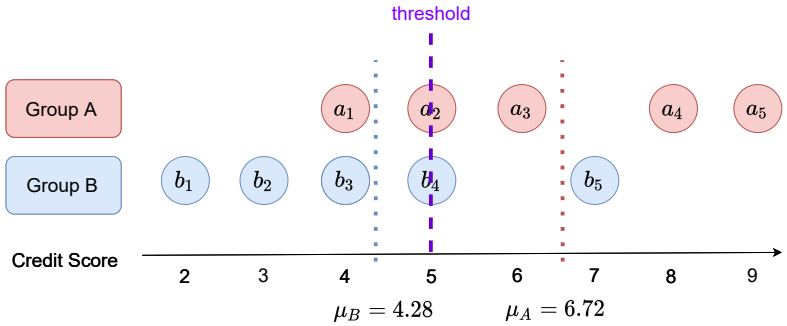


Figure 4-3: Outcome when using a policy that has the same threshold regardless of group.

In table-4.2, we computed the new average credit scores for the two groups. Both groups ameliorate with higher average credit scores, though the new score differences between the two groups $\Delta' = 2.44$ is slightly higher than the initial $\Delta = 2.2$.

Demographic Parity The second bank uses demographic parity as a fairness metric, which requires the bank to issue loans to the same percentage of people in both groups. Thus, if 4 out of 5 applicants are qualified for the loan in group A, the bank will also give out loans to 4 out 5 applicants in group B.

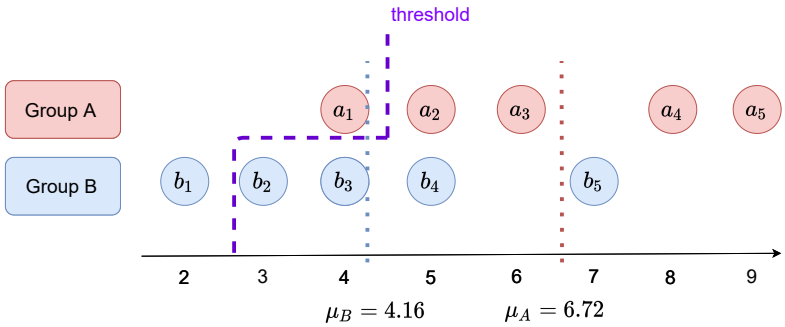


Figure 4-4: Outcome when using a demographic parity policy that issue loans to the same percentage of people in both groups.

	Group A						Group B					
Applicant	a_1	a_2	a_3	a_4	a_5	Mean	b_1	b_2	b_3	b_4	b_5	Mean
Credit Score X	4	5	6	8	9	6.4	2	3	4	5	7	4.2
Decision	0	1	1	1	1	**0.8**	0	1	1	1	1	**0.8**
New Expected X'	4	5	6.2	8.6	9.8	6.72	2	2.6	3.8	5	7.4	4.16

Table 4.3: Outcome when using a demographic parity policy. The bank issues loans to the same fraction of people (80%) in both group.

In table-4.3, the expected average score of group B will be decreased to $\mu_B = 4.16$, and the difference of averages between the two groups are increased to $\Delta' = 2.56$. Although the second bank tries to be fair, it actually further make group B's average credit scores worse and further segregates the two groups' credit score distributions.

Equalized Opportunity The third bank adopts a more constrained fairness metric called equalized opportunity. This metric requires that among those who can payback the loans, the bank should issue loans to the same percentage of people (Equalized false negative rate). For all applicants with credit score $X \geq 5$, their repaying probability is $Y \geq 0.5$. We call this set the qualified applicants, and this includes 4 individuals in group A and 2 individuals in group B. The bank decides to issue to the top 50% of qualified applicants, which will be a_4, a_5 and b_5.

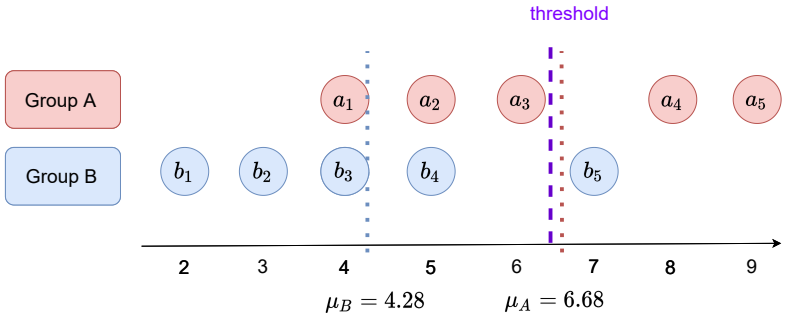


Figure 4-5: Outcome when using equalized opportunity policy.

As shown in Table-4.4, the difference between the two groups is again $\Delta' = 2.4$, which is the same as the first bank that only maximizes profit. Recall that the initial group disparity is $\Delta = 2.2$, and again the equalized opportunity bank makes the average credit scores of the two groups more disparate.

	Group A						Group B					
Applicant	a_1	a_2	a_3	a_4	a_5	Mean	b_1	b_2	b_3	b_4	b_5	Mean
Credit Score X	4	5	6	8	9	6.4	2	3	4	5	7	4.2
Decision	0	0	0	1	1	**0.4**	0	0	0	0	1	**0.2**
New Expected X'	4	5	6	8.6	9.8	6.68	2	3	4	5	7.4	4.28

Table 4.4: Outcome when using an equalized opportunity policy. The bank issues loans to the same fraction of qualified applicants (50%) in both groups.

Notation	Meaning
$\mathcal{D}$	Underlying distribution where the dataset is sampled from
Z	Protected group attribute such as gender or race
X	Feature attributes the other than protected attribute
Y	Ground truth target variable
S	State S consists of (Z, X, Y)
O	$O \sim Bernoulli(Y)$. An instantiation of the target variable.
(z, x, y)	An individual sampled from the distribution is a tuple of the protected attribute, feature attribute, and ground-truth label
P_Y	The CDF distribution of target variable Y.
$\mathcal{G}$	A DAG representing the dependency between state variables
$f_v(\cdot)$	Structural equations for node v
$\nabla f_v(x)$	Derivative of structural equations f_v evaluated at x
S^t	State at time t consists of (Z, X^t, Y^t)
D^t	Decision at time t
$\mathcal{U}^t$	Utility for the decision maker at time t
π_z	Policy function for group z
τ_z^t	Threshold used for group z at time t
x_{tp}	Feature value increase for a true positive
x_{fp}	Feature value decrease for a false positive
x_{tn}	Feature value increase for a true negative
x_{fn}	Feature value decrease for a false negative
u_{tp}	Utility increase for a true positive
u_{fp}	Utility decrease for a false positive
u_{tn}	Utility increase for a true negative
u_{fn}	Utility decrease for a false negative
g	Distance metric
δ_z^t	Within-group impact for group z at time t
Δ_{AB}^t	Between-group impact of group A and B at time t

Table 4.5: Notation table for the terms used in this chapter.

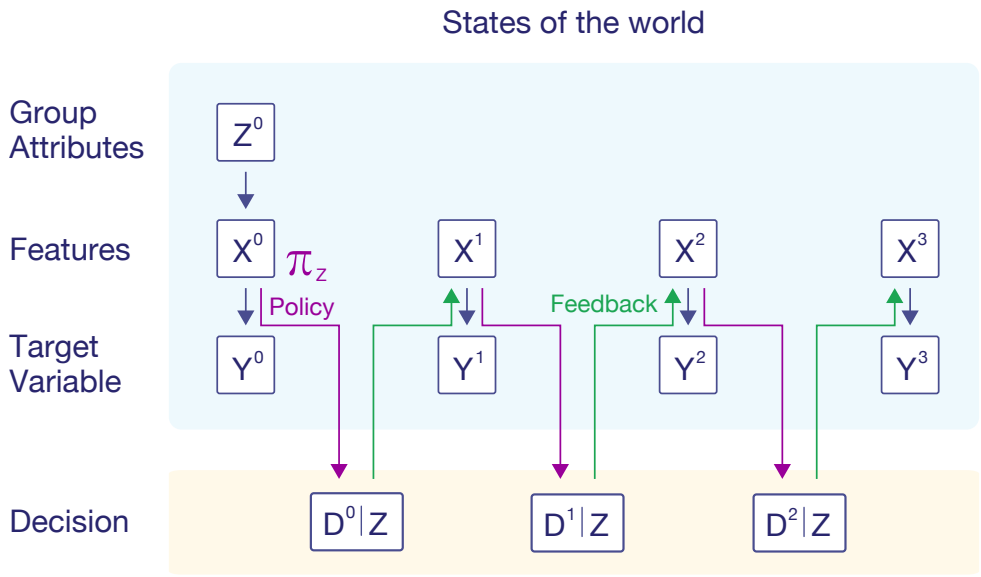


Figure 4-6: The dynamic data generation process unrolled by time. Z is the sensitive attribute, X is the features, Y is the target variable, and D is the decision applied by the agent. The purple arrow indicates a policy function that maps from the features X^t to a decision D^t, and the red arrow indicates the feedback effect from decision D^t to features X^{t+1}.

4.3 Formulation and setting

The previous motivating example showcases one-step feedback of decisions on underlying distributions. In this section, we formulate the long-term feedback of decisions through the lens of the Markov Decision Process (MDP).

4.3.1 Background: Markov decision process

We assume the target variable is a function of the features, and the target variable is a function of the features. MDP can be leveraged to characterize long-term inter-dependencies of features, the target variable, and the decisions as a graphical model. It characterizes the dependencies between variables at each state and also models the temporal transition of the underlying distribution. Although the ground truth dependencies are rarely known in real life, knowledge of the causal dependencies that generate the data could be useful when comparing different policies. This framework also naturally constructs a computation graph, where gradient flow over a long horizon can be easily computed.

Markov Decision Process A Markov decision process (MDP) is a tuple $(\mathcal{S}, \mathcal{A}, P, R)$ in which $\mathcal{S}$ is a finite set of states, $\mathcal{A}$ is a finite set of actions, P is a transition function defined as $P : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \longrightarrow [0, 1]$, and R is a reward function defined as $R : \mathcal{S} \times A \longrightarrow \mathbb{R}$.

4.3.2 Modeling the feedback loop as MDP

In this section, we show that the feedback loop of algorithmic decisions can be modeled using MDP. At each time step, the state contains three variables: $S = (X, Y, Z)$, where $X \in \mathbb{R}^d$ is a set of features, $Z \in \{0, 1\}$ is the time-invariant group attribute such as race or gender, and $Y \in [0, 1]$ is the target variable representing the probability of a positive outcome.

- **Initialization**: The process is initialized with a time-invariant group attribute Z, a set of observed features X^t, and the target variable Y^t. The initial group distribution is time-invariant and sampled from $Z \sim Bernoulli(p_0)$ where $p_0 = \mathbb{P}(Z = 0)$ is the probability that an individual comes from group $Z = 0$. The initial feature distribution X^0 is sampled from the initial distribution $\mathbb{P}(X^0|Z)$.

- **Decision and Outcome**: The target variable Y^t is a function of the features X^t, i.e., $Y^t = f_Y(X^t)$. At time step t, a binary decision $D^t \in \{0, 1\}$ is generated from a policy function π based on state S^t, i.e, $D^t = \pi(S^t)$. After applying the decision, a binary outcome is observed. We use an auxiliary variable $O^t \sim Bernoulli(Y^t)$ to indicate the outcome variable, which is sampled from a Bernoulli distribution with Y^t as the parameter.

- **Transition**: Based on the realized outcome O^t, the features X^t for each individual will be updated based on the decision and the outcome, where $X^{t+1} = f_X(X^t, D^t, O^t)$. The target variable Y^t will be updated accordingly.

- **Utility**: The decision maker's utility is a function of the decision and the realized outcome, i.e., $\mathcal{U}^t = f_{\mathcal{U}}(O^t, D^t)$.

In Figure 4-6, we illustrate the dynamic environment unrolled by time, where the purple

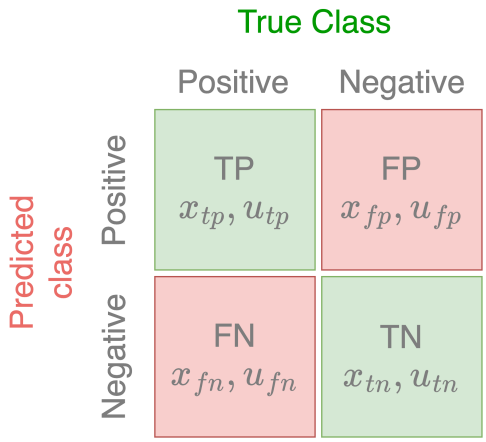


Figure 4-7: Parameters defined in terms of the confusion matrix.

arrow indicates a policy function that generates the decision, and the red arrow indicates the feedback effect of the decision.

We restrict our attention to linear utility functions and feature updates. Based on the realized outcome $O^t \sim Bernoulli(Y)$ and decision D^t, we can construct a confusion matrix containing true positive (TP), false positive (FP), true negative (TN), and false negative (FN).

Specifically, if a qualified ($O = 1$) candidate is accepted ($D = 1$), the decision-maker gains utility $u_{tp} > 0$ and the individual's feature is increased by X_{tp}; if an unqualified ($O = 0$) candidate is accepted ($D = 1$) , the decision-maker's utility is decreased by $u_{fp} > 0$ and the individual's feature is decreased by $X_{fp} > 0$. If an unqualified ($O = 0$) candidate is rejected ($D = 0$), the decision-maker gains utility $u_{tn} > 0$ and the individual's feature is increased by X_{tn}; if a qualified ($O = 1$) candidate is rejected ($D = 0$), the decision-maker's utility is decreased by $u_{fn} > 0$ and the individual's feature is decreased by $X_{fn} \geq 0$. In many cases, the utility and features won't change upon a negative action, and $u_{tn} = u_{fn} = X_{tn} = X_{fn} = 0$ could be set to 0.

4.3.3 Threshold policies

One of the most common solutions in fair machine learning is constrained optimization, where the goal is to learn a model that minimizes the expected loss with respect to loss

Notations used in MDP literature	Notations used in this framework
State S	A state S contains (Z, X, Y)
Set of actions $\mathcal{A}$	Binary decisions $D = \{0, 1\}$
Transition function P	Structure equation for feature update f_X
Reward function R	Utility function $\mathcal{U}$

Table 4.6: A mapping between notations in the literature and notations used in our framework.

Notation	Meaning
$Y^t = f_Y(X^t)$	Function links features X to target variable Y.
$D^t = \pi(S^t)$	Function links state S to decision D.
$X^{t+1} = f_X(X^t, D^t, O^t)$	Feature update function that links features new feature X^{t+1} to old feature X^t, decision D^t, and outcome O^t
$\mathcal{U} = f_{\mathcal{U}}(O^t, D^t)$	Function links utility to outcome and decision.

Table 4.7: Set of structural equations and their purpose.

function $\mathcal{L}$ and subject to some fairness constraints [Donini et al., 2018]. For example, the following constraint ensures demographic parity:

$$\begin{aligned} \min_{\theta} \;& \mathbb{E}_{(X,Y,Z)\sim\mathcal{D}}[\mathcal{L}(f_\theta(X,Z),Y)] \\ s.t. \;& \mathbb{E}_{(X,Y,Z)\sim\mathcal{D}}[f_\theta(X,Z)] = \mathbb{E}_{(X,Y,Z)\sim\mathcal{D}}[f_\theta(X,Z)] \end{aligned}$$

In sequential decision making, the decision maker uses a policy function π as guidance for sequential decisions, where decisions are repeatedly sampled from this function.

Definition 4.3.1. *A policy $\pi : \mathcal{S} \to [0, 1]$ is a function that maps from states $S \in \mathcal{S}$ to the probability distribution over decision d, i.e., $\pi(d|s) = \mathbb{P}(D^t = d|S^t = s)$.*

In sequential decision making, a decision-maker repeatedly maximizes the utility subject to the fairness constraints as in the one-step optimization. If the probability of an individual coming from group z is p_z, we can decompose the utility with respect to the group distribution:

$$\begin{aligned} \max_{\pi=(\pi_A,\pi_B)} \;& p_A\mathbb{E}_{D^t\sim\pi_A(S^t)}[\mathcal{U}(D^t,Y^t)] + p_B\mathbb{E}_{D^t\sim\pi_B(S^t)}[\mathcal{U}(D^t,Y^t)] \\ s.t. \;& \mathbb{E}_{S^t\sim\mathcal{D}_A^c}[\pi_A(S^t)] = \mathbb{E}_{S^t\sim\mathcal{D}_B^c}[\pi_B(S^t)] \end{aligned} \tag{4.1}$$

where $\mathcal{D}$ is the underlying distribution and $\mathcal{D}_z^{\mathcal{C}}$ is the distribution constrained by some fairness metric $\mathcal{C}$ for group z, and π_A, π_B are group-specific policies for group A and group B respectively.

In the rest of this chapter, we restrict our attention to threshold policy, where the policy function is a threshold function on the target variable Y. In real applications, the policy function should put a threshold on features based on implicitly learned mapping from the features to the target variable. The reason that we directly put a threshold on the target variable is to eliminate the effect of complications on learning the policy function.

Definition 4.3.2. *A threshold policy assigns positive action when $Y^t \geq \tau$ for some threshold τ, i.e., $\pi_z(D^t = 1|Y^t) = \mathbb{P}(Y^t \geq \tau|Z = z)$.*

We list a few threshold policies that are based on commonly used fairness constraints and show that they can be reduced in this form:

- A Maximum Utility (`MaxUtil`) policy maximizes the expected utility without constraint.

- A Demographic Parity (`DemoPar`) policy maximizes the expected utility subject to the demographic parity constraints, which requires that both groups have equalized positive rates on decisions, i.e., $\mathbb{E}[D^t = 1|Z = A] = \mathbb{E}[D^t = 1|Z = B]$. This is equivalent to $\mathbb{E}_{Z=A}[\pi_A(Y^t)] = \mathbb{E}_{Z=B}[\pi_B(Y^t)]$.

- An Equalized Opportunity (`EqOpp`) policy maximizes the expected utility subject to the equalized opportunity constraints, which requires that both groups have equalized false positive rates,i.e., $\mathbb{E}[D^t = 1|Y^t = 0, Z = A] = \mathbb{E}[D^t = 1|Y^t = 0, Z = B]$. This is equivalent to $\mathbb{E}_{Y^t=0,Z=A}[\pi_A(Y^t)] = \mathbb{E}_{Y^t=0,Z=B}[\pi_B(Y^t)]$.

  As specified by the order in the list, each policy requires finding optimal thresholds within a smaller search space specified by more restricted constraints. In generally we would expect $\mathcal{U}_{\texttt{MaxUtil}} \geq \mathcal{U}_{\texttt{DemoPar}} \geq \mathcal{U}_{\texttt{EqOpp}}$.

4.4 Measuring the Long-term Impact of Decisions

In this section, we propose a new metric for measuring fairness through the long-term impact of decisions. We first discuss the shortcomings of current fairness metrics in a sequential decision-making environment. These gaps motivate us to design better metrics to assess the fairness of decisions in sequential and dynamic environments.

4.4.1 Filling in the gaps for long-term fairness metrics

Decision Fairness vs Outcome Fairness Existing fairness metrics are defined for decision fairness, which ensures the decisions satisfy classification parity (accuracy, false positive rates [Hardt et al., 2016] etc.) at the time of decision-making. However, under the feedback loop, even fair decisions can potentially impact outcomes or the target variable unfairly.

Average Fairness vs Distributional Fairness In many real-world scenarios, such as with loan applications, the target variable distribution is often skewed or heavy-tailed. Conclusions drawn from decisions made using only metrics defined as oblivious to the distributions could be insufficient.

4.4.2 The distributional impact of algorithmic decisions

We introduce a novel fairness metric for measuring the impact of algorithmic decisions on the distribution of the target variable. We first categorize the impact as within-group impact and between-group impact:

- **Within-Group Impact**(WGI): Within-group impact measures how a sequence of algorithmic decisions shifts the distribution of the target variable of the group. Within-group disparity happens when decisions following a policy lead to a negative impact on the group, such as further increases in inequality or dichotomy within a population group.

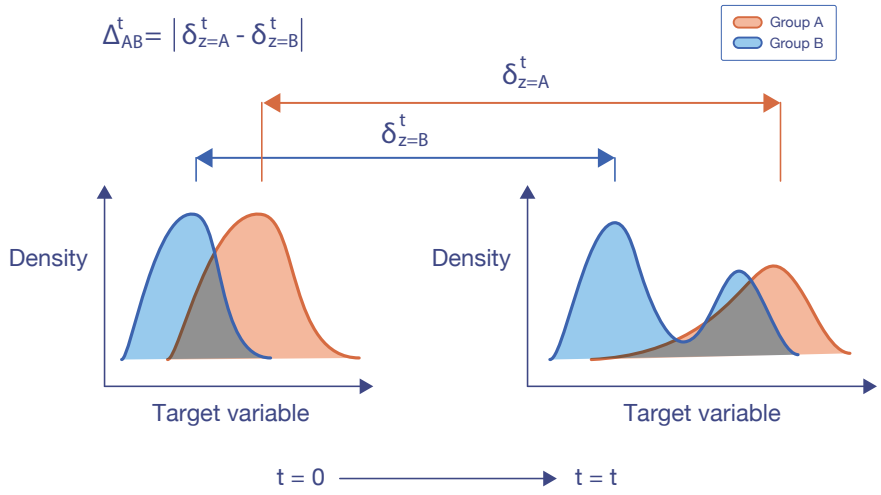


Figure 4-8: An illustration of the backfire effects of a policy. $\delta_{z=A}$ and $\delta_{z=B}$ measures the impact of decisions on the orange group (group A) and blue group (group B) respectively. Here group B is the disadvantaged group since its target variable distribution lies on the lower spectrum. Compared to the initial distribution at time $t = 0$, decisions lead to backfire effects in terms of WGI for group B (the center is decreased and spread is increased. Decisions also lead to backfire effects in terms of BGI where group A and group B's distributions are further apart.

- **Between-group Impact**(BGI): Between-group impact measures the absolute difference between two groups' within-group impact. Between-group disparity captures whether the within-group impact is different among different groups.

Definition 4.4.1 (Within-group impact (g-WGI))**.** *Let Y^t_z be the group z's target variable at time t and $P_{Y^t_z}$ be the distribution of Y^t_z. The within-group impact is defined as the change in the distribution as characterized by a function of the distribution with respect to $t = 0$ for group z, i.e,*

$$\delta^t_z = g(P_{Y^t_z}, P_{Y^0_z}) \tag{4.2}$$

where $g(\cdot)$ is some distance metric.

The choice of the distance metric Previous works studying the long-term impact of fairness decisions [D'Amour et al., 2020b][Mouzannar et al., 2019][Zhang et al., 2020] have focused on how decisions change the outcome in the average sense, where g is the absolute difference of the mean. Here we allow g to stand for general functions that measure

the shift from distribution Y^0 to distribution Y^t to capture the distribution in a more fine-grained way. Here we categorize the possible functions into three categories: (1) functions that measure the shift of the **center** of a distribution; (2) functions that measures the shift of the **spread** of a distribution; (3) functions that measure the shift of the **shape** of a distribution.

- Center shift: Center shift measures the change of the target variable for a typical individual in the distribution.
  - Difference in mean of target variable distribution (Mean-WGI):

$$g(P_{Y_z^t}, P_{Y_z^0}) = \mathbb{E}[Y_z^t] - \mathbb{E}[Y_z^0]$$

  - Difference in quantiles (Quantile-WGI):

$$g(P_{Y_z^t}, P_{Y_z^0}) = Q_k(Y_z^t) - Q_k(Y_z^0)$$

    where Q_k is the k-quantile function. When $k = 2$, this is equivalent to the median of the distribution, which quantifies the change for a median individual in the distribution.

- Spread shift: Spread shift measures the change of variability of a distribution.
  - Difference in variance of target variable distribution (var-WGI):

$$g(P_{Y_z^t}, P_{Y_z^0}) = var[Y_z^t] - var[Y_z^0]$$

  - Application inspired metric: One interesting choice with real-world implications is the Gini coefficient, which measures income inequality within a population group.

- Shape shift: Shape shift measures the distributional change of cumulative density functions.

– Wasserstein-1 distance (W1-WGI):

$$g(P_{Y_z^t}, P_{Y_z^0}) = \int_0^1 |F_{Y_z}^t(y) - F_{Y_z}^0(y)| d_y$$

where $F_{Y_z}^t$ is the CDF function of distribution $P_{Y_z^t}$.

Next, we define between-group impact, which captures how decisions shift distributions of two population groups **differently**.

Definition 4.4.2 (Between-group impact (BGI))**.** *We define the between-group impact at time step t as*

$$\Delta_{AB}^t = |\delta_{z=A}^t - \delta_{z=B}^t| \tag{4.3}$$

The backfire effect appears when algorithmic decisions shape the group-wise distributions in different ways that further increase the disparity between them.

Definition 4.4.3 (Backfire effect)**.** *We say that a policy has a backfire effect if:*

- *g-WGI<0 if g measures center or shape of distribution; or g-WGI<0 if g measures spread of a distribution.*
- *BGI is increased compared to the initial distribution, i.e., $\Delta^T \geq \Delta^0$.*

We use Figure 4-8 to illustrate the backfire effect in terms of within-group impact and between-group impact.

4.4.3 Disparity in a broader context

While disparity could be defined statistically, it is important to understand the implications of disparity and segregation in a broader context. We draw insights from closely related concepts in sociology and economics regarding inequality and discuss how these concepts can be adapted to quantify the inequality introduced by algorithmic decisions.

Social Segregation Racial segregation is a well-studied phenomenon in sociology, where population groups are separated geographically. In the context of machine learning, we can

measure the geometric segregation in the feature space of population groups after applying algorithmic decisions [Heidari et al., 2019].

Economic inequality In economics, the Gini coefficient [Yitzhaki, 1979] is the statistical dispersion metric that measures the inequality of income within a social group. We use x_j to indicate the the income for an individual j and $\bar{x}$ to indicate the average income, and $r_j = \frac{x_j}{\bar{x}}$ to indicate the inequality ratio for individual j. Perfect equality is achieved when the inequality ratio $r_j = \frac{x_j}{\bar{x}}$ equals 1 for everyone, which happens when everyone's income is equal to the average income. A more general measure of inequality can be defined as:

$$Inequality = \sum_j p_j h(r_j)$$

where p_j is the weight of the population, and $h(r_j)$ is a function of the deviation of each individual's r_j from the point of equality. The Gini coefficient is a useful metric for measuring within-group impact as a result of algorithmic decisions.

4.5 Case Studies

Next, we use two case studies to empirically illustrate the impacts of threshold policies. In both cases, the group distribution is time-invariant and sampled from $Z \sim Bernoulli(p_A)$ where p_A is the probability that an individual comes from group A.

Loan Application Example The loan application example was first proposed by Liu et al. [2018] to study the one-step feedback effect of fairness constraints. We first frame the loan application example in the format of a dynamic SCM. The variables in the SCM are as follows: $Z \in \{0, 1\}$ is the binary sensitive attribute, $X \in [c_{min}, c_{max}]$ is the credit score , $D \in \{0, 1\}$ is the binary loan approval/rejection decision, and $Y \in [0, 1]$ is the probability of repaying. The initial feature distribution f_{X^0} and the repay probability f_Y are estimated from the dataset.

Since the data is not sequential in nature, we use a synthetic structural equation for the feature update function f_X, where we experiment different feature update parameters X_{tp}

and X_{fp}.

$$
\begin{aligned}
Z &\sim Bernoulli(p_A) \\
X^0 &= f_{X^0}(Z) \\
Y^t &= f_Y(X^t) \\
O^t &\sim Bernoulli(Y^t) \\
X^{t+1} &= \begin{cases} \min\{X^t + X_{tp}, c_{max}\} & \text{if } O^t = 1, D^t = 1 \\ \max\{c_{min}, X^t - X_{fp}\} & \text{if } O^t = 0, D^t = 1 \end{cases}
\end{aligned}
$$

Synthetic Gaussian In the second example, we extend a previous loan application where the feature variable has only 1-dim. The initial feature distribution X^0 is sampled from a group-specific 2-dim Gaussian distribution. The target variable is the sigmoid of a linear transformation of the feature vectors with weight vector M. The i-th feature positively contributes to the target variable ($[\nabla f_Y(X)]_i > 0$) if the i-th component in M is positive.

$$
\begin{aligned}
Z &\sim Bernoulli(p_A) \\
X^0 &\sim \mathcal{N}_d(\mu_z, \Sigma_z) \\
Y^t &= \frac{1}{1 + e^{-X^t \cdot M}} \\
X^{t+1} &= \begin{cases} X^t + X_{tp} & \text{if } O^t = 1, D^t = 1 \\ X^t - X_{fp} & \text{if } O^t = 0, D^t = 1 \end{cases}
\end{aligned}
$$

4.5.1 Simulation Environment

4.5.1.1 Simulation environment setup

In this part, we briefly outline how the simulation environment using the causal Markov decision process is set up. The state consists of a state vector (X, Y, Z) and a set of struc-

tural equations f_v governing the dependencies between variables. The structural equations implicitly specify the directed edges in the DAG. When we update a state, we first do a topological sort of all the nodes within the graph, and then update with respect to the topological orders. In this case, parent nodes always get updated before child nodes, and the causal structure is preserved.

Algorithm 2 Simulation using causal Markov decision process

Given DAG $\mathcal{G}$ and structural equations f_v
Given initial state $s^0 = (x^0, y^0, z^0)$
Given fairness constraint $\mathcal{C}$
for $t = 1, ..., T$ **do**
  $d^t \leftarrow get\_decision(s^t, \mathcal{C})$ ▷ Find optimal decisions by solving Eq-4.1 through PGD
  $r^t \leftarrow get\_reward(s^t, d^t)$ ▷ Get reward
  for $v \in$ topological_sort($\mathcal{G}$) **do** ▷ Update according to the causal mechanism
    $s^{t+1}[v] = f_v(s^t, d^t)$
  end for
end for

Solving the constrained optimization The sequential decision making process modeled by graphical model provides a natural computation graph. Instead of resolving Eq-4.1 at each time step, we use take projected gradient descent steps at each time step based on previous step's optimal solution. Specifically,

$$\tau^{t+1} = \text{Proj}_{\mathcal{C}}(\tau^t - \alpha^t \nabla \mathcal{U}(\tau^t))$$

where $\text{Proj}_{\mathcal{C}}$ is the projection onto constraint set $\mathcal{C}$ specified by a fairness constraint, and α^t is the learning rate.

4.5.1.2 Evaluation Metrics

Throughout the experiment section, we used different metrics to measure the impact of threshold policies. Each evaluation metric is averaged over 10 simulation runs. In each simulation run, we sample 50000 individuals from the distribution and run 200 steps. The evaluations metrics are:

- Utility: Utility is averaged over a decision maker's utility u_i for making a decision on an individual i. The higher the utility, the better.
- Within-group impact (g-WGI): Within-group impact is measured using equation-4.2 with respect to different g function. For all the WGI metrics on center or shape, a higher number indicates a better WGI. For all the WGI metrics on spread, a higher number indicates a worse WGI.
- Between-group impact (g-BGI): Between-group impact is measured using equation-4.3. For between group impact, the lower the better.

We listed the evaluation metrics and their meanings in the following table.

Metrics	Expression	Range
Utility	$\frac{1}{n}\sum_{i=1}^{n} u_i$	$[0, 1]$
Mean-WGI	$\frac{1}{n}\sum_{i=1}^{n} y_i^t - \frac{1}{n}\sum_{i=1}^{n} y_i^0$	$[-1, 1]$
Med-WGI	$\frac{1}{n}\sum_{i=1}^{n} y_i^t - \frac{1}{n}\sum_{i=1}^{n} y_i^0$	$[-1, 1]$
var-WGI	$\frac{1}{n}\sum_{i=1}^{n}(y_i^t - \bar{y}^t)^2 - \frac{1}{n}\sum_{i=1}^{n}(y_i^0 - \bar{y}^0)^2$	$[-1, 1]$
Gini-WGI	$G^t - G^0$, $G^t = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}\lvert y_i^t - y_j^t\rvert}{2n^2\bar{y}^t}$	$[-1, 1]$
g-BGI	$\lvert\delta_A^t - \delta_B^0\rvert$	$[0, 1]$

Table 4.8: Evaluation Metrics.

4.5.2 Simulation result: Loan application

4.5.2.1 Simulation setup

The initial group distributions Z^0, initial credit score distributions X^0, and initial repay probability Y^0 are estimated from the FICO score dataset. In this experiment, we set $u_{tp} = u_{fp} = 1$ and $c_{min} = 300, c_{max} = 850$. We also set the feature value change and utility change when not issuing a loan as 0 ($u_{tn} = u_{fn} = X_{tn} = X_{fn} = 0$). The initial distribution of Z^0, X^0, Y^0 is shown below. The groups are White (Group A) and Black (Group B), and we refer to group A (White) as the disadvantaged group.

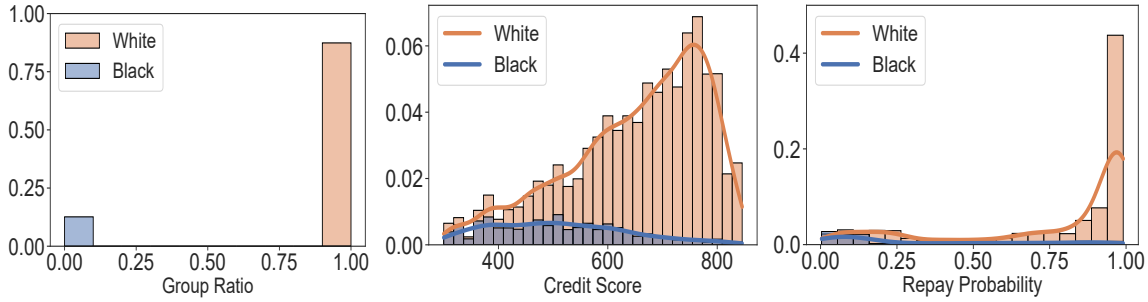


Figure 4-9: The initial distribution for the FICO score dataset. Left: The initial distribution for the group ratio. Middle: The initial distribution for the features (credit score). Right: The initial distribution for the target variable (repay probability).

We categorize the simulation settings into three regimes based on the relative value of the X_{tp} and X_{fp}. In this case $X_{tn} = X_{fn} = 0$, and the cost ratio $q = \frac{X_{fp}}{X_{tp}+X_{fp}}$. Cost ratio specifies the feature change value for a false positive relative to the full feature value range. If X_{fp} is greater, the cost ratio will also be greater. The cost ratio can be interpreted as the relative impact of a false positive on features.

- Forgiving setting : $X_{tp} = 150, X_{fp} = 75$;
- Neutral setting: $X_{tp} = X_{fp} = 75$;
- Harsh setting: $X_{tp} = 75, X_{fp} = 150$.

4.5.2.2 Fixed threshold policies

In this section, we discuss the impact of repeatedly employing a fixed threshold policy on the **center** and **spread** of the target variable distribution. Center and spread are two important summary statistics to describe a distribution. Center describes a typical value of the distribution, and spread describes the variation of the data. We use mean-WGI to measure the impact on the center of the distribution, and var-WGI to measure the impact on the spread of the distribution. We show how the utility and impact change under different fixed thresholds across different simulation settings (forgiving, neutral, and harsh). We show the experiments for one-step simulation as well as multiple-step simulation. In the multi-step simulation, we run the simulation for 200 steps. The multi-step simulation captures the long-term dynamics between the thresholds and the target variable.

Center In figure-4-10, we plot the mean-WGI for one-step simulation. The red dashed line indicates the threshold for `MaxUtil`, which maximizes utility without any constraint and is achieved at $\tau_{MaxUtil} = \frac{u_{fp}}{u_{tp}+u_{fp}}$.

The mean-WGI metric measures how the center of the target variable distribution changes compared to the initial distribution for each group respectively. A positive mean-WGI indicates the policy exerts a positive impact on the center of the distribution. Across different cost ratio settings, the sign of mean-WGI changes as we switch from the forgiving setting to the harsh setting. Specifically, mean-WGI is positive for the forgiving setting and negative for the harsh setting. On the other hand, as the threshold value increases, the magnitude of mean-WGI decreases. This implies that the sign of the mean-WGI depends on the cost ratio, yet lower thresholds increase the magnitude of the mean-WGI. When the cost ratio is lower (as in the forgiving setting), a lower threshold amplifies the positive impact. On the other hand, when the cost ratio is higher (as in the harsh setting), using a lower threshold will amplify the negative impact.

Mean-BGI measures how a policy shifts the center of the two distributions differently and is represented by the gap between the two group-wise mean-WGI lines. As shown in the figure, as the threshold increases, the mean-BGI decreases. This suggests that for a lower cost ratio setting, there is a trade-off: a higher threshold leads to lower positive mean-WGI respectively, but also lower mean-BGI. On the other hand, for a higher cost ratio setting, a higher threshold is always more desirable (lower negative mean-WGI, lower mean-BGI).

In figure-4-11, we plot the mean-WGI for one-step simulation. Multi-step simulation generally shows the same trend for mean-WGI as in one-step simulation, but with an amplification effect on the magnitude for mean-WGI. However, the threshold for `MaxUtil`, which maximizes average utility, appears at the lower spectrum of threshold values at around $\tau = 0.15$. This implies that for long-term simulation, there is a trade-off between utility and mean-BGI. Lower thresholds lead to higher average utility but also a higher disparity between the two groups as measured by the mean of the target variable.

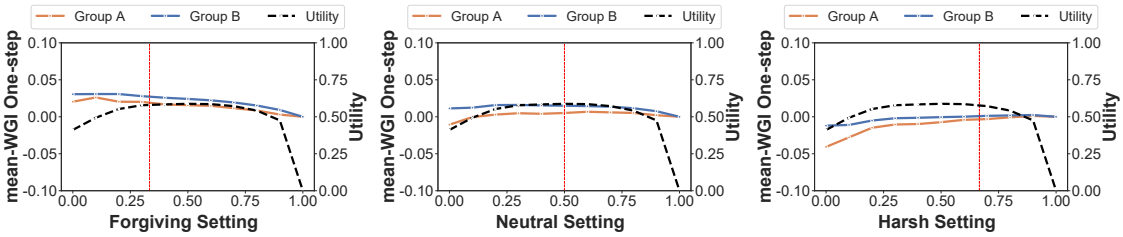


Figure 4-10: One-step simulation on mean-WGI under different fixed threshold values for the forgiving (left), neutral (middle), and harsh settings (right) respectively. The red dashed line indicates optimal threshold for `MaxUtil`.

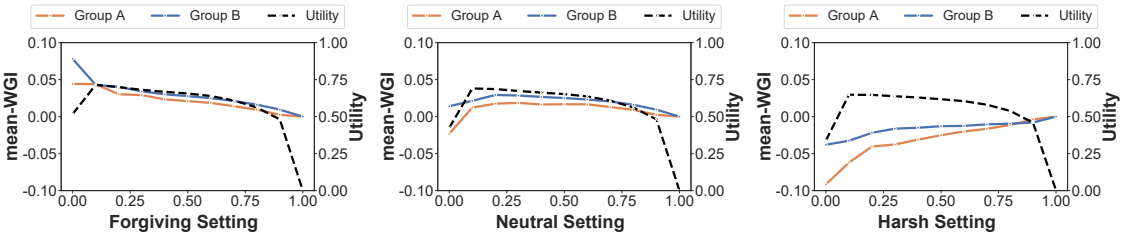


Figure 4-11: Multi-step simulation on mean-WGI under different fixed threshold values for the forgiving (left), neutral (middle), and harsh settings (right) respectively. Here the utility is the average utility over the simulation steps.

Spread In figure-4-12, we plot the var-WGI for one-step simulation. Var-WGI measures how the threshold policies change the variance of the target variable distribution changes. A positive var-WGI indicates the policy increases the spread of the target distribution. In the loan application example, this indicates the inequality of the repaying probability distribution within a group is increased. Contrary to mean-WGI, the relationship between cost ratio setting and var-WGI is not linear. This is reflected as the neutral setting leads to lower var-WGI than either the forgiving setting or the harsh setting. For a fixed cost ratio, a higher threshold is always more desirable as it leads to lower var-WGI. As shown in figure 4-12, all threshold policies lead to a non-negative var-WGI.

Var-BGI measures how the threshold policies shift the variance of the two distributions differently. For each setting, as the threshold increases, var-BGI decreases. This suggests that for var-BGI, a lower threshold is always more desirable. Across different simulation settings, as the cost ratio increases, the gap between var-WGI (var-WBI) decreases.

In figure-4-13, we plot the var-WGI for multi-step simulation. Multi-step simulation generally shows the same trend for var-WGI as in one-step simulation, and also with an amplification effect on the magnitude of var-WGI. The same trade-off for mean-WGI ap-

plies here as well: lower thresholds lead to higher average utility but also a higher disparity between the two groups as measured by the variance of the target variable.

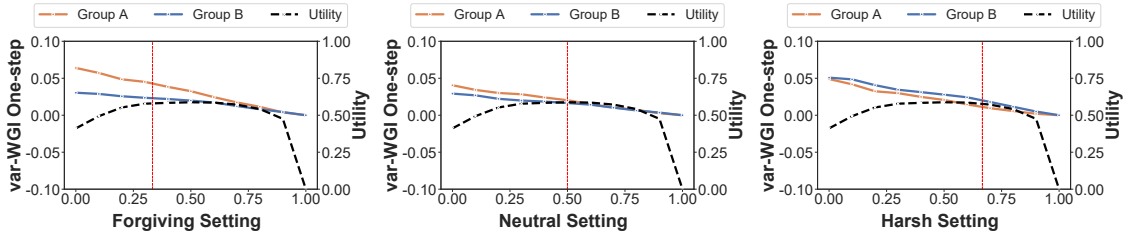


Figure 4-12: One-step simulation on var-WGI under different fixed threshold values for the forgiving (left), neutral (middle), and harsh settings (right) respectively. The red dashed line indicates $\tau_{MaxUtil}$ for each setting.Here the utility is the average utility over the simulation steps.

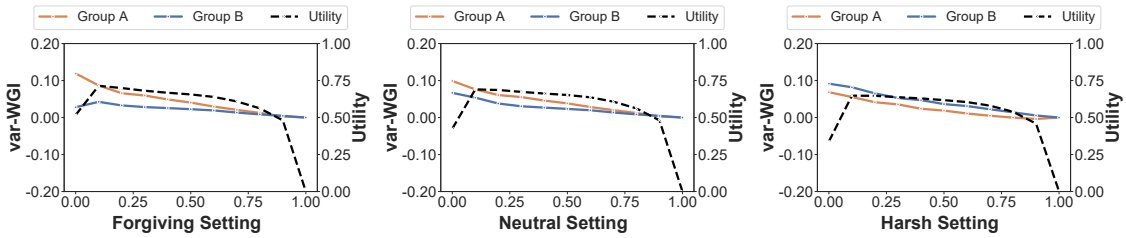


Figure 4-13: Multi-step simulation on var-WGI under different fixed threshold values for the forgiving (left), neutral (middle), and harsh settings (right) respectively.

4.5.2.3 Fair policies lead to backfire effects

In this section, we compare policies that maximize utility subject to fairness constraints (`DemoPar`, `EqOpp`) with policy that only maximizes utility (`MaxUtil`). Besides `MaxUtil` policy, all other policies use a different threshold at each time step based on the solution from the optimization problem in eq-4.1. This adds complexity in quantitatively characterizing the backfire effects of fair policies. Instead, we use the simulation results to provide some insights on enforcing fair policies. In this section, we investigate the impact of fair policies on mean of variance of the target variable distribution.

Impact of fair policies on mean In Figure 4-14, we plot mean-WGI and mean-BGI as a function of the cost ratio. As shown in the theoretical result, mean-WGI monotonically decreases as the cost ratio increases. This is showcased in figure-4-14: as the cost ratio increases, all policies exhibit negative impacts on both groups.

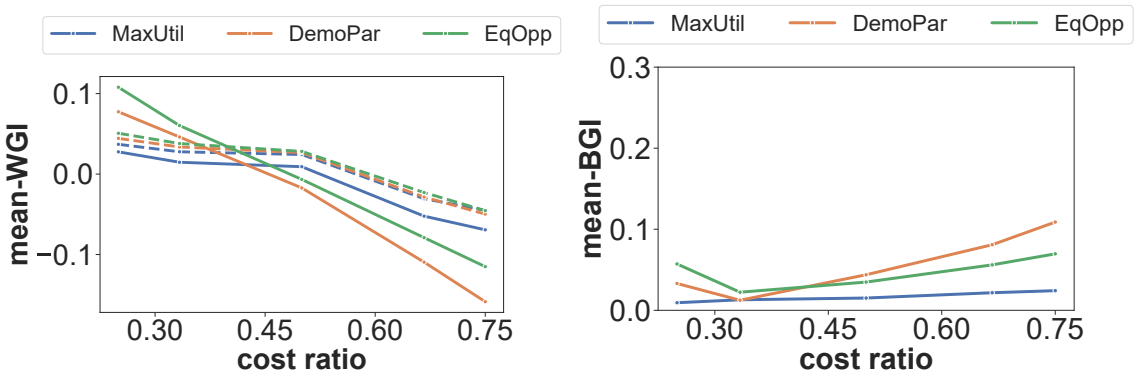


Figure 4-14: Mean-WGI and mean-BGI for different fairness policies.The dashed line indicates the advantaged group, and the solid line indicates the disadvantaged group.

Across different cost ratios, `MaxUtil` always exerts a more positive WGI for the advantaged group compared to the disadvantaged group. On the other hand, the relative WGI for the groups for `DemoPar` and `EqOpp` swaps when shifting from a low-cost ratio to a high-cost ratio regime. Under a low-cost ratio regime, both policies have more positive impacts on the disadvantaged group; and under a high-cost ratio setting, both policies have more positive impacts on the advantaged group. Comparing the three policies, while the WGI is fairly similar for the advantaged group, `MaxUtil` exhibits the least negative impact and `DemoPar` exhibits the most negative impact on the disadvantaged group.

The mean-BGI metric measures how the average change in the target variable differs between two groups. A high mean-BGI indicates a policy increase in the disparity of the target variable between two groups in the average sense. In terms of mean-BGI, `EqOpp` results in the highest mean-BGI when the cost ratio q is lower, and `DemoPar` results in the highest mean-BGI when the cost ratio is higher.

Impact of fair policies on variance In Figure 4-15, we plot mean-WGI and mean-BGI as a function of the cost ratio. As shown with the theoretical result, the direction of var-WGI doesn't monotonically increase as cost ratio increases. This is illustrated empirically in figure-4-15. As cost ratio increases, var-WGI increases for disadvantaged group and decreases for advantaged group.

For var-BGI, `MaxUtil` leads to higher var-BGI for high cost ratio settings, and fair policies (`DemoPar` and `EqOpp`) leads to higher var-BGI for low cost ratio settings.

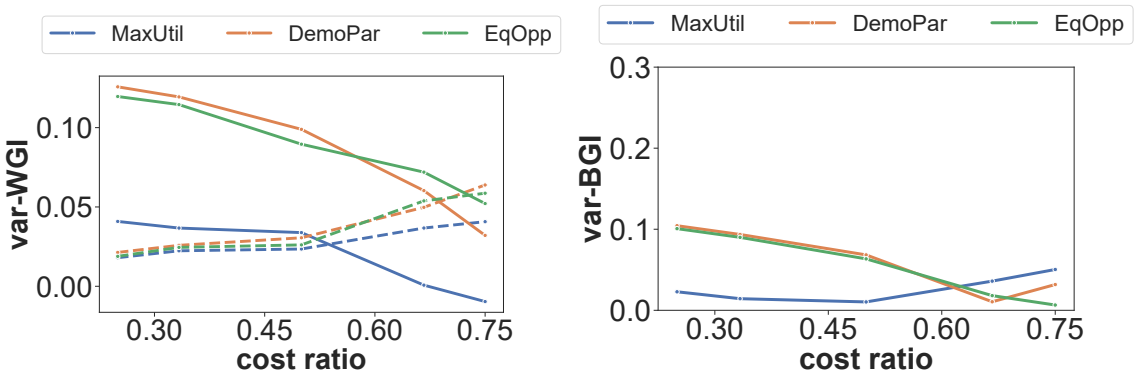


Figure 4-15: Mean-WGI and mean-BGI for different fairness policies. The dashed line indicates the advantaged group, and the solid line indicates the disadvantaged group.

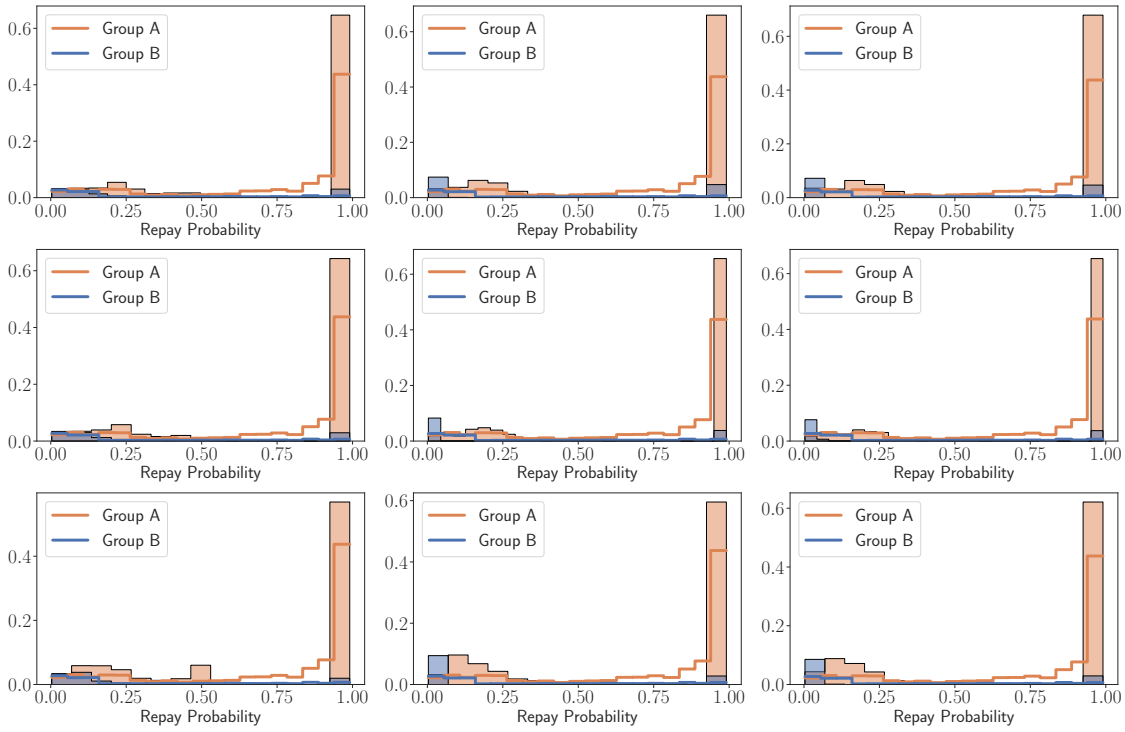
4.5.2.4 The hidden story behind average outcome

In Figure 4-16, we plot the final distributions of the target variable under three different settings. Compared to the initial distributions, repeatedly enforcing a policy changes the shape of the final distributions in a way that cannot not be captured simply by the group mean. In particular, all policies create dichotomies and the Matthew effect [Perc, 2014] on the target variable distribution such that "the rich get richer and the poor get poorer."

This phenomenon is further showcased in Figure 4-17, where we plot the gini-WGI and gini-BGI as a function of the cost ratio q using the Gini coefficient as the g function. A positive gini-WGI indicates that the policy increases the inequality of target variable distribution within a group. On the other hand, a negative gini-WGI indicates the policy decreases the inequality within a group. As shown in the left plot, as the cost ratio increases, the gini-WGI increases for all three policies. The dynamics on the advantaged group are fairly similar among the three policies. For the disadvantaged group, `MaxUtil` is the only policy that doesn't increase the gini-WGI, while gini-WGI decreases drastically with the cost ratio for the other two policies.

As for between-group impact, `DemoPar` leads to the highest gini-BGI consistently. In general, as the cost ratio increases, the gini-BGI also increases for the fair policies.

In general, the comparison of WGI and BGI between different policies highly depends on the simulation setting and metric g. In table 4.9, we evaluate g-BGI using different g function. Depending on the simulation setting parameter and the metric g, we may get dif-



(a) `MaxUtil`. (b) `DemoPar`. (c) `EqOpp`.

Figure 4-16: Histogram for the final distribution for repaying probability after different policies. The unfilled bars indicate the initial distribution and the filled bars indicate the final distribution. Top row: forgiving setting. Middle row: neutral setting. Bottom row: harsh setting.

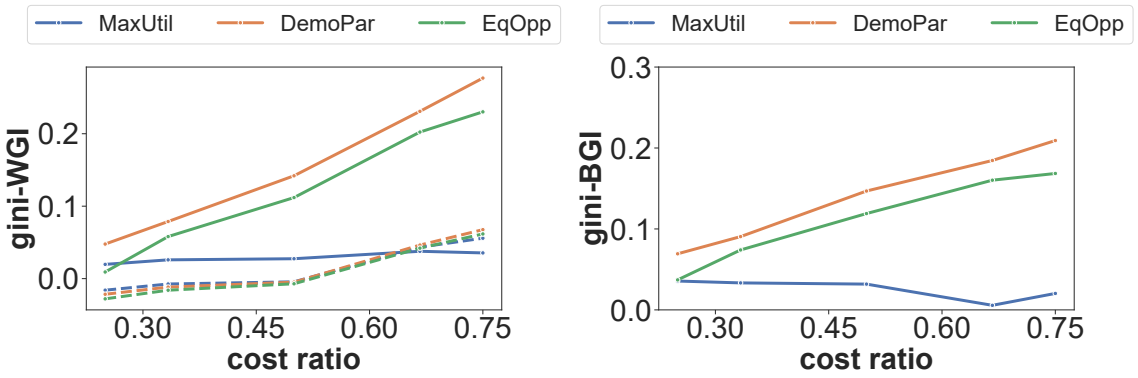


Figure 4-17: Gini-WGI and Gini-BGI for different fairness policies. The dashed line indicates the advantaged group, and the solid line indicates the disadvantaged group.

	MaxUtil	DemoPar	EqOpp
Mean-BGI	0.011	0.005	**0.024**
Median-BGI	0.057	**0.192**	0.159
Var-BGI	0.014	**0.094**	0.090
Gini-BGI	0.031	**0.107**	0.063
W1-BGI	0.002	**0.092**	0.087

Table 4.9: Between-group impact (g-BGI) when measured using different g function (forgiving setting). The bold number indicates the policy that results in the biggest g-BGI. Using different metrics g leads to different conclusions.

ferent conclusions on which policy leads to the biggest backfire effect. This raises the concern that using an average metric (such as groupwise average outcome) to evaluate fairness could lead to an unfair comparison between policies, and comprehensive characterization of the distributional impact of decisions is essential.

4.5.3 Simulation results: synthetic gaussian (2d)

We use the synthetic gaussian dataset to study the effects when the features are multi-dimensional. This allows us to create a more realistic dependency between features and the target variable, where the structural equation f_Y is more complex.

4.5.3.1 Simulation setup

The initial features for the two groups are sampled from $\mathcal{N}(\mu_0, I)$ and $\mathcal{N}(\mu_1, I)$ respectively, where $\mu_0 = [0, 0]^T$ and $\mu_1 = [1, 1]^T$. The initial feature distributions are shown on

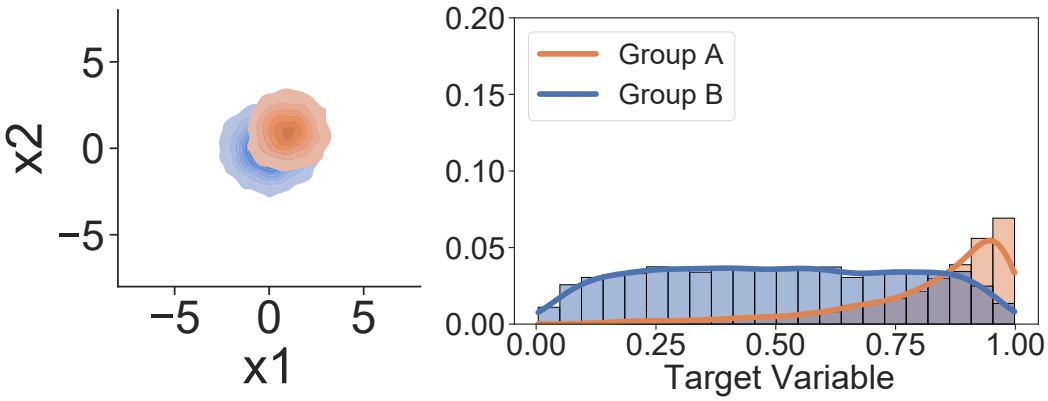


Figure 4-18: Initial distribution for the Gaussian 2d. Left: initial feature distribution. Right: initial target variable distribution.

the right. The feature update is $X_{tp} = [0.02, 0.01]$ and $X_{fp} = [0.01, 0.02]$). We simulate with two feature contribution matrices $M_1 = [1, 1]$ and $M_2 = [1, -1]$, where the first or second feature is a "bad" feature (negatively impacts the target variable) respectively. In Figure 4-18, we plot the initial features and target variable distribution.

Feature segregation In Figure 4-19, we plot the final distribution of the features under different structural equations f_Y. In the top row, both features positively contribute to the target variable ($f_Y(X) > 0$). In the middle row, the first feature negatively contributes to the target variable; and in the third row, the second feature negatively contributes to the target variable. This shapes the feature spaces differently even though the feature transition equation $X^{t+1} = f_X(X^t)$ is the same. Even when the target distribution is close enough, certain features could be segregated more than is desired for the groups. In real-world applications, the structural equation between features and target variable $f_Y(X)$ is rarely known and is in fact what most machine learning models are trying to predict. This interplay between features and target variables adds more complexity to the analysis.

4.6 Conclusion and key takeaways

In this chapter, we model the interactions between decision-makers and individuals using MDP where the transitions could be general structural equations. This allows us to analyze

the distributional impact of algorithmic decisions on the target variable. In particular, we characterize the long-term impact on the center and spread of the target variable distribution as a result of threshold policies. The theoretical results provide useful guidance on choosing the best threshold policies when balancing different considerations.

Better metrics for long-term fairness Fairness constraints ensure that the decisions assigned satisfy some statistical parity in a myopic way that is oblivious to dynamics, yet these decisions could impact features and target variables in an undesired way. In practice, there are often trade-offs between myopic decision fairness and long-term fairness.

In addition, the fairness constraints we discussed are all defined in terms of error metrics that are based on the average outcome and are thus ignorant of the distributions. When risk distributions differ, these error metrics could be poor indicators of inequality. Decisions based on classification parity metrics could lead to dichotomies on the target distribution even when the average outcome remains the same. The within-group and between-group impacts are useful metrics for measuring the dynamic and distributional impacts of algorithmic decisions.

Mitigating the backfire effects Designing fair policies requires careful consideration of interactions between the decisions and the underlying distributions. In this work, we characterize the optimal threshold for maximizing utility, maximizing within-group impact, and minimizing between-group impact.

Our simulation results suggest that there is generally a trade-off between utility and between-group impact. A higher between-group impact indicates the policy leads to a higher disparity between two groups as measured by some metrics. In specific, a lower threshold generally leads to higher utility, but also higher between-group impact as measured both by the mean and variance of the target variable. This trade-off suggests that mitigating backfire effects requires careful consideration and balance between different desiderata. For fairness policies that are computed based on constrained optimization, a different threshold could be used at each round. This further increases the complexity of comparing pros and cons of different fairness policies. In practice, it could be more desir-

able to use some dynamics policies rather than a fixed policy designed for a single fairness constraint.

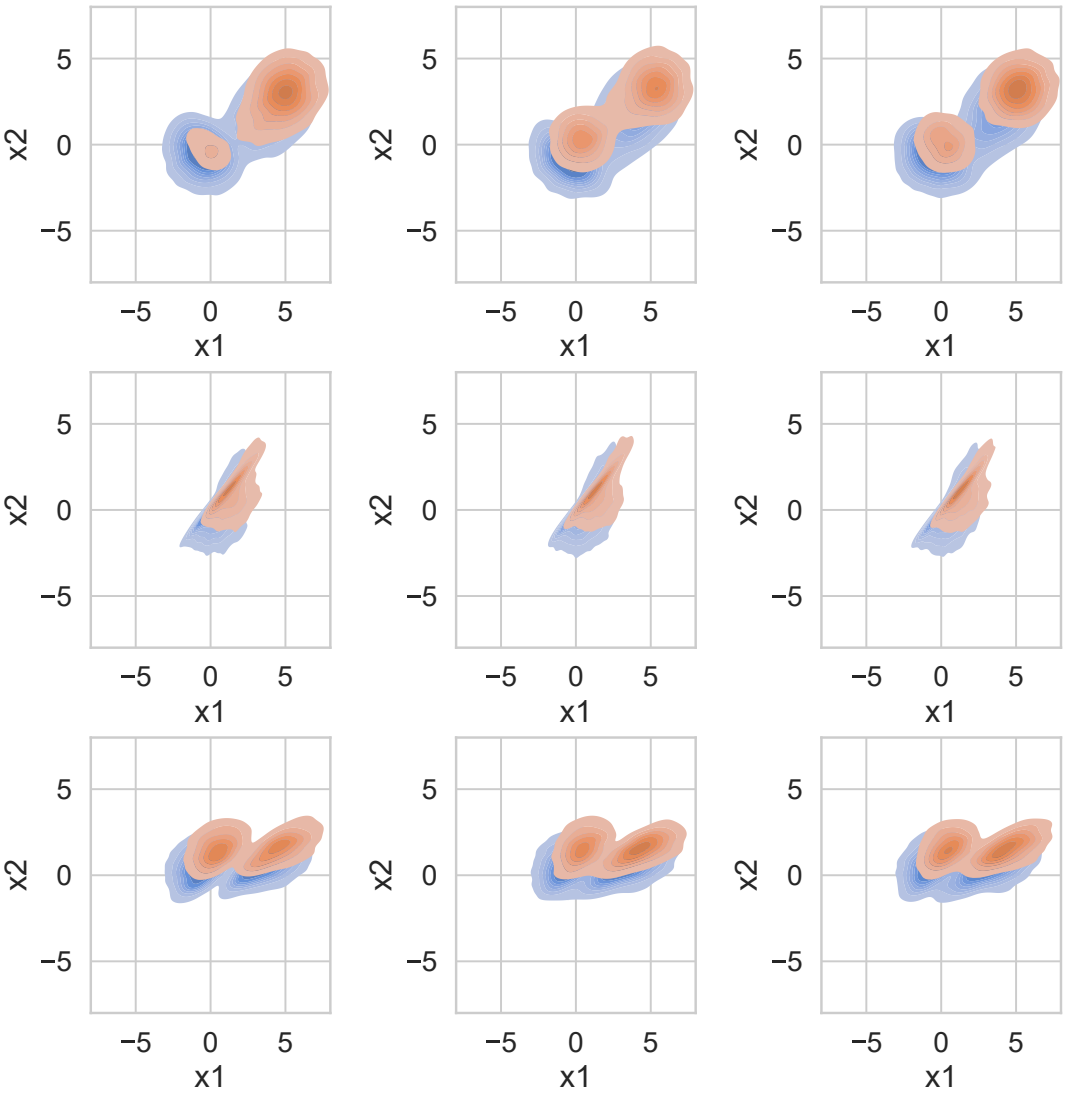


Figure 4-19: Synthetic Gaussian Example. Top: $M_1 = [1, 1]$. Middle: $M_2 = [1, -1]$. Bottom: $M_3 = [-1, 1]$.

Chapter 5

Conclusions and Future Work

5.1 Conclusion

In this thesis, we study the fairness of machine learning algorithms in a sequential decision-making setting. When considering fairness in machine learning, many complexities arise because the data comes from people, and decisions are applied to people. This often creates a feedback loop that involves interactions between the predictions/decisions and the state of the world. While many solutions have been proposed for addressing biases in model predictions, in this thesis we focus on addressing fairness concerns after model predictions. In this thesis, we study two problems: (1) first, how can we translate black-box model predictions into fair decisions? (2) and second, how do fair decisions impact the underlying distributions when there is a feedback loop?

For the first problem, we propose a meta-algorithm that combines black-box predictions in a way that balances different error metrics (FPR and FNR) between groups. For the second problem, we showed that there are often trade-offs between fairness and accuracy in the short term and that fair decisions could also lead to backfire effects in the long term. We argue that applying algorithmic decisions to people requires careful evaluation of the different components that come into play.

In this section, we also discuss how the problems discussed in this thesis can be related to a broad research area beyond fairness. We think that many of these problems and approaches are closely related, opening the door for exciting future work.

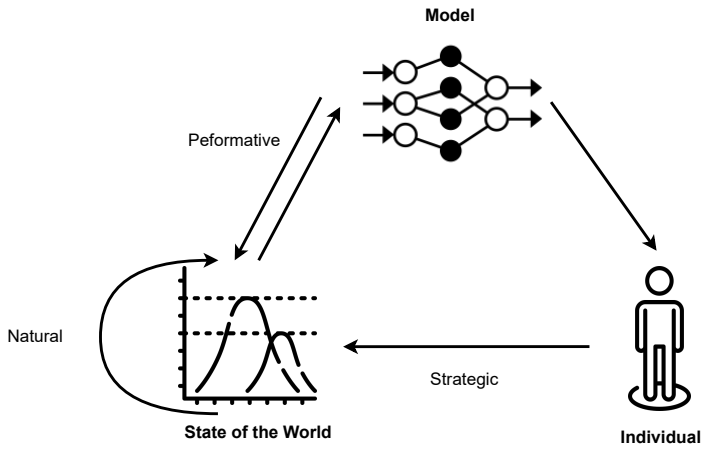


Figure 5-1: Different types of distributional shifts in sequential decision making.

5.2 Connections to machine learning models in a sequential setting

One of the common and essential assumptions in machine learning models is that the training distribution and the distribution on which the model is deployed are the same. Yet much recent work has shown that this assumption is often violated in real-world applications [Perdomo et al., 2020]. Predictions and consequential decisions can lead to changes in distribution, especially when there are humans in the loop. The interplay between predictive models and underlying distributions occurs in many applications. Recommendation systems predict users' preferences and provide suggestions, and these suggestions could shift users' preferences in turn. In traffic prediction, the predicted best route might attract more vehicles, making that very route less desirable. Here we list a taxonomy of scenarios under which the training and testing distribution could be disparate.

Distribution Shift Distribution shift is a general phenomenon in which an underlying distribution changes over time. The cause of this shift could be an exogenous factor or factors unrelated to the decision, such as a change in the weather. It could also be endogenous factors that arise as an artifact of model predictions and consequential decisions.

Performative Prediction The concept of a performative prediction was first proposed by Perdomo et al. [2020], which describes scenarios in which predictive model-based decisions may influence the outcome that the model tries to predict. In this sense, the machine learning model is not only *predictive* of the target but is also *performative* of the target. It's reasonable to assume in this case that the distributional shift is benign or at least predictable. The backfire effects of fairness constraints we discussed previously provide an example of performative feedback.

Strategic Classification When model predictions are converted to decisions made about people and applied downstream, the distributional shift could be strategic or even adversarial. After decisions informed by models are applied to people, individuals could strategically react to the models by nudging their features [Hardt et al., 2016][Milli et al., 2019]. For example, an attacker of a machine learning system can adversarially alter an image [Madry et al., 2018][Kurakin et al., 2017] to intentionally cause the model to predict a wrong label.

5.3 Future Work

Unified framework of studying the interaction between models and humans

- As we discussed above, there are many different ways that an underlying distribution could change over time, and many solutions have been proposed for each of them. Could there be a unified framework for studying interactions between models and humans and their fairness implications?

- Can we design effective interventions that make the interactions between models and humans fairer?

Efficient ways of finding optimal policy When a machine learning model needs to be repeatedly deployed, one common practice is to frequently retrain the model on a new dataset. This is certainly not ideal and leaves room for a lot of interesting future work:

- When we know that the distribution shift is purely caused by the model's decisions (as is assumed to be the case in backfire effects), can we create a more efficient algorithm where frequently retraining and re-balancing different fairness constraints is not necessary?

Chapter 6

Appendix

6.1 Appendix for Chapter 3: Fairness in Sequential Decision Making

6.1.1 Additional Experiment Results

6.1.1.1 Additional Experiment Results on Synthetic Dataset

Pareto Curve on Synthetic Dataset To clearly illustrate the trade-off that can be achieved in the optimization step, we plot the pareto front by varying $\boldsymbol{\lambda}$ defined in the optimization step 3.11. The Pareto curve is in 6-1.

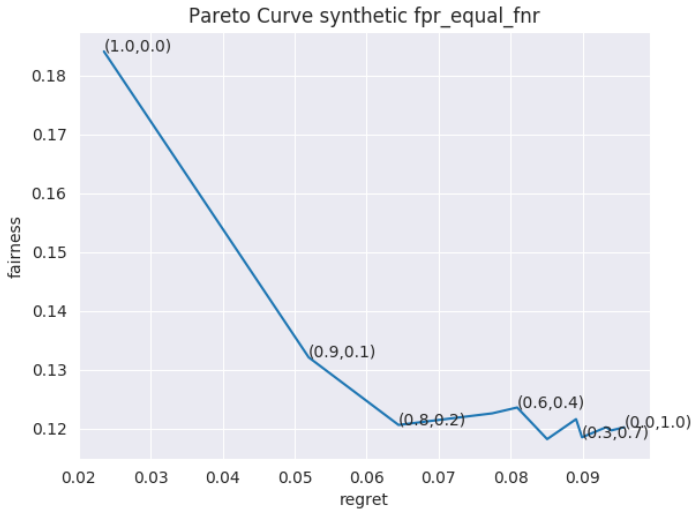


Figure 6-1: Pareto Curve for the synthetic dataset with imbalanced setting. x-axis is the regret and y-axis is the average value of Equalized FPR and Equalized FNR. The pair indicates $(\boldsymbol{\lambda}_{regret}, \boldsymbol{\lambda}_{Fairness})$ where $\boldsymbol{\lambda}_{Fairness} = \boldsymbol{\lambda}_{FPR} = \boldsymbol{\lambda}_{FNR}$.

Different simulation distributions We summarize the experiments for `Synthetic` data in Tables 6.1 and 6.2, where we fix $p_A = 0.9, p_B = 0.1, \mu_{A,+} = 0.7$ and varies $\mu_{B,+}$.

$\mu_{B,+}$	MW	GroupAware	G-FORCE
0.1	0.016 ± 0.013	0.494 ± 0.009	**0.305** ± 0.020
0.3	0.018 ± 0.013	0.487 ± 0.012	**0.182** ± 0.014
0.4	0.026 ± 0.011	0.475 ± 0.019	**0.148** ± 0.032
0.5	0.024 ± 0.018	0.283 ± 0.162	**0.110** ± 0.030
0.6	0.011 ± 0.008	**0.019** ± 0.022	0.032 ± 0.017

Table 6.1: Equalized FPR by fixing $p_A = 0.9, p_B = 0.1, \mu_{A,+} = 0.7$

$\mu_{B,+}$	MW	GroupAware	G-FORCE
0.1	0.473 ± 0.060	0.509 ± 0.055	**0.304**$\pm$ 0.028
0.3	0.490 ± 0.031	0.486 ± 0.022	**0.194**$\pm$ 0.018
0.4	0.508 ± 0.018	0.488 ± 0.019	**0.146**$\pm$ 0.018
0.5	0.488 ± 0.013	0.296 ± 0.162	**0.111** ± 0.030
0.6	0.495 ± 0.020	**0.022** ± 0.019	0.046 ± 0.010

Table 6.2: Equalized FNR by fixing $p_A = 0.9, p_B = 0.1, \mu_{A,+} = 0.7$

6.1.1.2 Additional Experiment Results on Real Dataset

	# of rounds	p_A	$\mu_{A,+}$	$\mu_{B,+}$
`Adult`	24421	0.851	0.26	0.16
`German Credit`	300	0.853	0.73	0.50
`COMPAS`	1584	0.398	0.54	0.39

Summary statistics of real data sets

6.1.2 Proofs for Non-delayed Case

We define the cumulative loss of classifier f on group z as $L_{f,z} = \sum_{t=1}^{T} \ell_{f,z}^t$. The cumulative false positive of f on group z is the cumulative loss made on the negative examples; and its expression is $L_{f,z,-} = \sum_{t=1}^{T} \ell_{f,z}^t \mathbb{1}\{y = -\}$. Similarly, we defined the expected loss on group z as $\mathbb{E}[L_z] = \sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \pi_f^t \ell_{f,z}^t$ and the expected false positive on group z is $\mathbb{E}[L_{z,-}] = \sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \pi_f^t \ell_{f,z}^t \mathbb{1}\{y = -\}$.

6.1.2.1 Proof of Lemma 1

Let $\Phi^t_{z,+} = \sum_{f\in\mathcal{F}} w^t_{f,z,+}$. We start computing the expected loss on group $z,+$:

$$\begin{aligned}
\mathbb{E}[\ell^t_{z,+}] &= \sum_{f\in\mathcal{F}} \pi^t_{f,z}\cdot \ell^t_{f,z}\mathbb{1}\{y=+\} \\
&= \sum_{f\in\mathcal{F}} \left(q^t_{z,+}\cdot \frac{w_{f,z,+}}{\sum_{f\in\mathcal{F}} w_{f,z,+}} + q^t_{z,-}\cdot \frac{w_{f,z,-}}{\sum_{f\in\mathcal{F}} w_{f,z,-}} \right)\cdot \ell^t_{f,z}\mathbb{1}\{y=+\} \\
&= q^t_{z,+}\cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}}\cdot \ell^t_{f,z}\mathbb{1}\{y=+\} + q^t_{z,-}\cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}}\cdot \ell^t_{f,z}\mathbb{1}\{y=+\} \\
&= q^t_{z,+}\cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}}\cdot \ell^t_{f,z,+} + q^t_{z,-}\cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}}\cdot \ell^t_{f,z,+} \qquad (6.1)
\end{aligned}$$

The overall expected loss is composed by two terms: the former, which is the expected loss on group $z,+$ when their associated weights $w_{f,z,+}$ are selected, and the later, when the wrong weights $w_{f,z,-}$ are selected. Both terms are weighted by their respective estimated rates $q^t_{z,+}$ and $q^t_{z,-}$.

Then, we have the following inequality:

$$\begin{aligned}
\Phi^{t+1}_{z,+} &= \sum_{f\in\mathcal{F}} w^{t+1}_{f,z,+} \\
&= \sum_{f\in\mathcal{F}} w^t_{f,z,+}(1-\eta)^{\ell^t_{f,z}\mathbb{1}\{y=+\}} \\
&\leq \sum_{f\in\mathcal{F}} w^t_{f,z,+}(1-\eta\ell^t_{f,z}\mathbb{1}\{y=+\}) \\
&= \sum_{f\in\mathcal{F}} w^t_{f,z,+} - \eta\sum_{f\in\mathcal{F}} w^t_{f,z,+}\ell^t_{f,z,+} \\
&= \Phi^t_{z,+}(1-\eta\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}}\cdot \ell^t_{f,z,+})
\end{aligned}$$

Thus by the recursive function, we have

$$\begin{aligned}\Phi_{z,+}^{T+1} &\leq \Phi_{z,+}^{1} \prod_{t=1}^{T} (1 - \eta \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t}) \\ &= d \prod_{t=1}^{T} (1 - \eta \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t}). \end{aligned} \tag{6.2}$$

Following the updating rule of the MW algorithm, we have

$$\begin{aligned} w_{f,z,+}^{T+1} &= w_{f,z,+}^{1} (1-\eta)^{\sum_{t=1}^{T} \ell_{f,z}^{t} \cdot \mathbb{1}\{y=+\}} \\ &= (1-\eta)^{\sum_{t=1}^{T} \ell_{f,z,+}^{t}} \end{aligned} \tag{6.3}$$

where $w_{f,z,+}^{t} = 1$, as all the weights are initialized.

Using 6.23 and 6.24,

$$w_{f,z,+}^{T+1} = (1-\eta)^{\sum_{t=1}^{T} \ell_{f,z,+}^{t}} \leq \Phi_{z,+}^{T+1} \leq d \prod_{t=1}^{T} (1 - \eta \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t}) \tag{6.4}$$

and taking the logarithm of both sides, we have

$$\ln(1-\eta) L_{f,z,+} \leq \ln d + \sum_{t=1}^{T} \ln(1 - \eta \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t})$$

$$\ln(1-\eta) L_{f,z,+} \leq \ln d - \eta \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t} \tag{6.5}$$

$$\sum_{t=1}^{T} \sum_{f \in \mathcal{F}} \frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}} \cdot \ell_{f,z,+}^{t} \leq (1+\eta) L_{f,z,+} + \frac{\ln d}{\eta} \tag{6.6}$$

Equation 6.26 follows because if $\eta < 1/2$, we can use the inequality $\ln(1-\eta) < -\eta$. This is intuitive as if we always choose weights for positive examples, it reduces to the same bound as in the original MW algorithm.

We now assume that the expected error on group $z, +$ when wrong weights $w_{f,z,-}$ are

selected is bounded as:

$$\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}} \cdot \ell^t_{f,z,+} \leq \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} + \alpha^t_{z,-} \tag{6.7}$$

where $\alpha^t_{z,-} < 1$ is the difference of loss in expectation made when using the incorrect weights of the MW algorithm on group $z,+$ (Cross-Instance Cost). Then

$$\mathbb{E}[L_{z,+}] = \sum_{t=1}^{T} \left(q^t_{z,+} \cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} + q^t_{z,-} \cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}} \cdot \ell^t_{f,z,+} \right) \tag{6.8}$$

$$\mathbb{E}[L_{z,+}] \leq \sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} + \sum_t q^t_{z,-} \cdot \alpha^t_{z,-} \tag{6.9}$$

where using 6.27, we finally obtain:

$$\mathbb{E}[L_{z,+}] \leq (1+\eta) L_{f,z,+} + \frac{\ln d}{\eta} + \sum_t q^t_{z,-} \cdot \alpha^t_{z,-} \tag{6.10}$$

Similarly,

$$\mathbb{E}[L_{z,-}] \leq (1+\eta) L_{f,z,-} + \frac{\ln d}{\eta} + \sum_t q^t_{z,+} \cdot \alpha^t_{z,+}. \tag{6.11}$$

The expected total errors on group z is, adding the two equations above:

$$\mathbb{E}[L_z] \leq (1+\eta) L_{f,z} + 2\frac{\ln d}{\eta} + (\sum_t q^t_{z,-} \cdot \alpha^t_{z,-} + \sum_t q^t_{z,+} \cdot \alpha^t_{z,+})$$

.

In the same way, the expected total errors (considering $z = A, B$) is:

$$\mathbb{E}[L] \leq (1+\eta) L_f + 4\frac{\ln d}{\eta} + \alpha \tag{6.12}$$

where all the Cross-Instance Costs are condensed in:

$$\alpha = \sum_{z\in\{A,B\}, y\in\{-,+\}} q_{z,y} \sum_t \alpha^t_{z,y}.$$

6.1.2.2 Proof of Lemma 2

Using the same process as for the upper bound, we have:

$$\begin{aligned}
\Phi_{z,+}^{t+1} &= \sum_{f\in\mathcal{F}} w_{f,z,+}^{t+1} \\
&= \sum_{f\in\mathcal{F}} w_{f,z,+}^{t}(1-\eta)^{\ell_{f,z}^{t}\mathbb{1}\{y=+\}} \\
&\geq \sum_{f\in\mathcal{F}} w_{f,z,+}^{t}(1-\eta(1+\eta)\ell_{f,z}^{t}\mathbb{1}\{y=+\} \\
&= \sum_{f\in\mathcal{F}} w_{f,z,+}^{t} - \eta(1+\eta)\sum_{f\in\mathcal{F}} w_{f,z,+}^{t}\ell_{f,z,+}^{t} \\
&= \Phi_{z,+}^{t}(1-\eta(1+\eta)\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\ell_{f,z,+}^{t}).
\end{aligned}$$

Thus, by the recursive function, we have

$$\begin{aligned}
\Phi_{z,+}^{T+1} &\geq \Phi_{z,+}^{1}\prod_{t=1}^{T}(1-\eta(1+\eta)\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\phi_{z,+}^{t}}\cdot\ell_{f,z,+}^{t}) \\
&= d\prod_{t=1}^{T}(1-\eta(1+\eta)\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\ell_{f,z,+}^{t})
\end{aligned}$$

.

Let f^* be the best expert in hindsight in terms of achieving lowest false positives, we

have

$$\begin{aligned}\Phi^t_{z,+} &= \sum_{f\in\mathcal{F}} w^t_{f,z,+} \\ &\leq d \cdot \max_{f\in\mathcal{F}} w^t_{f,z,+} \\ &= d \cdot w^t_{f^*,z,+} \\ &= d \cdot \max_{f\in\mathcal{F}} (1-\eta)^{\sum_{t=1}^T \ell^t_{f,z}\cdot \mathbb{1}\{y=+\}} \\ &= d \cdot (1-\eta)^{\sum_{t=1}^T \ell^t_{f^*,z}\cdot \mathbb{1}\{y=+\}}.\end{aligned}$$

Therefore we have:

$$d \cdot (1-\eta)^{\sum_{t=1}^T \ell^t_{f^*,z}\cdot \mathbb{1}\{y=1\}} \geq \Phi^t_{z,+} \geq d \cdot \prod_{t=1}^T [1-\eta(1+\eta)\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+}]$$

Taking the log of both sides:

$$\ln(1-\eta) L_{f^*,z,+} \geq \sum_{t=1}^T \ln\left(1-\eta(1+\eta)\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+}\right) \tag{6.13}$$

$$\ln(1-\eta) L_{f^*,z,+} \geq \ln\left(1-\eta(1+\eta)\right) \sum_{t=1}^T \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} \tag{6.14}$$

$$\sum_{t=1}^T \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} \geq \gamma(\eta) L_{f^*,z,+} \tag{6.15}$$

where $\gamma(\eta)$ is defined as:

$$\gamma(\eta) = \frac{\ln(1-\eta)}{\ln\left(1-\eta(1+\eta)\right)}$$

using that $ln(1-\eta(1+\eta)x) \geq ln(1-\eta(1+\eta))x$ for all $x \in [0,1]$ and $\eta \in (0, \eta^{max})$, where $\eta^{max} = \frac{-1+\sqrt{5}}{2}$ which does not restrict the range of $\eta \in (0, 0.5)$.

Thus, using 6.29, we have:

$$\mathbb{E}[L_{z,+}] = \sum_{t=1}^{T}(q_{z,+}^{t}\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\ell_{f,z,+}^{t} + q_{z,-}^{t}\sum_{f\in\mathcal{F}}\frac{w_{f,z,-}^{t}}{\Phi_{z,-}^{t}}\cdot\ell_{f,z,+}^{t}$$
$$\geq \gamma(\eta)L_{f^*,z,+} + \sum_{t} q_{z,-}^{t}\cdot\alpha_{z,-}^{t}.$$

and,

$$\mathbb{E}[L_z] \geq \gamma(\eta)L_{f^*,z} + (\sum_{t} q_{z,-}^{t}\cdot\alpha_{z,-}^{t} + \sum_{t} q_{z,+}^{t}\cdot\alpha_{z,+}^{t}).$$

Finally, the total expected error is lower bounded by:

$$\mathbb{E}[L] \geq \gamma(\eta)L_{f^*} + \alpha. \tag{6.16}$$

6.1.2.3 Fairness Bound

Proof We assume group A arrives with probability p, group B arrives with probability 1-p, that is, $\mathbb{P}(z = A) = p$. The expected mean label of group A is defined as $\mu_{A,+} = \mathbb{P}(y = +|z = A)$ and mean label of group B is defined as $\mu_{B,+} = \mathbb{P}(y = +|z = B)$. Each individual classifier is ϵ-fair, thus:

$$|\mathbb{E}_{x,y,z}\left[\frac{L_{f,A,-}}{\sum_{t=1}^{T}\mathbb{1}\{y=-\}\mathbb{1}\{z=A\}}\right] - \mathbb{E}_{x,y,z}\left[\frac{L_{f,B,-}}{\sum_{t=1}^{T}\mathbb{1}\{y=-\}\mathbb{1}\{z=B\}}\right]| \leq \epsilon,\ \forall f \tag{6.17}$$

which represents the cardinality of the selected subset of samples.

The absolute difference of FPR between group A and group B is:

$$|FPR_A - FPR_B| = |\mathbb{E}_{x,y,z}\left[\frac{L_{A,-}}{\sum_{t=1}^{T}\mathbb{1}\{z=A\}\{y=-\}} - \frac{L_{B,-}}{\sum_{t=1}^{T}\mathbb{1}\{z=B\}\{y=-\}}\right]| \tag{6.18}$$

For the sake of notation we define

$$C_{A,-} = \sum_{t=1}^{T} \mathbb{1}\{y = -\}\mathbb{1}\{z = A\} \text{ and } C_{B,-} = \sum_{t=1}^{T} \mathbb{1}\{z = B\}\{y = -\}.$$

Using Lemmas 1 and 2, we have:

$$\begin{aligned}
&|\mathbb{E}_{x,y,z}\left[\frac{L_{A,-}}{C_{A,-}} - \frac{L_{B,-}}{C_{B,-}}\right]| \\
&\leq |\mathbb{E}_{x,y,z}\left[\frac{(1+\eta)L_{f^*(A,-),A,-}}{C_{A,-}} + \frac{\frac{\ln d}{\eta}}{C_{A,-}} + \frac{\sum_t q^t_{A,-} \cdot \alpha^t_{A,-}}{C_{A,-}} - \frac{\gamma(\eta)L_{f^*(B,-),B,-}}{C_{B,-}} - \frac{\sum_t q^t_{B,-} \cdot \alpha^t_{B,-}}{C_{B,-}}\right]| \\
&= |(1+\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(A,-),A,-}}{C_{A,-}}\right] - \gamma(\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \\
&\qquad \left(\sum_t \frac{q^t_{A,-} \cdot \alpha^t_{A,-}}{C_{A,-}} - \frac{\sum_t q^t_{B,-} \cdot \alpha^t_{B,-}}{C_{B,-}}\right) + \mathbb{E}_{x,y,z}\left[\frac{\ln d}{\eta C_{A,-}}\right]| \quad (6.19)
\end{aligned}$$

Using equation 6.37, we have:

$$|\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),A,-}}{C_{A,-}}\right] - \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]| \leq \epsilon$$

Moreover, without loss of generality we assume that f^* makes the smallest average loss on group B. This is,

$$\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(A,-),A,-}}{C_{A,-}}\right] \leq \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),A,-}}{C_{A,-}}\right] \leq \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \epsilon.$$

Thus, equation 6.19 becomes:

$$
\begin{aligned}
&\leq |(1+\eta)\left(\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]+\epsilon\right)-\gamma(\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]+ \\
&\qquad\left(\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{C_{A,-}}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{C_{B,-}}\right)+\mathbb{E}_{x,y,z}\left[\frac{\ln d}{\eta C_{A,-}}\right]| \\
&\leq |\left(1+\eta-\gamma(\eta)\right)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]+\epsilon(1+\eta)+ \\
&\qquad\left(\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{C_{A,-}}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{C_{B,-}}\right)+\mathbb{E}_{x,y,z}\left[\frac{\ln d}{\eta C_{A,-}}\right]| \\
&\leq |\left(1+\eta-\gamma(\eta)\right)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]+\epsilon(1+\eta)+ \\
&\qquad\left(\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)+\frac{\ln d}{\eta p(1-\mu_{A,+})T}| \\
&\leq |\left(1+\eta-\gamma(\eta)\right)FPR_{f^*}+\epsilon(1+\eta)+\left(\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)+\frac{\ln d}{\eta p(1-\mu_{A,+})T}| \\
&\leq |\left(1+\eta-\gamma(\eta)\right)FPR_{f^*}+\epsilon(1+\eta)+\left(\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)|
\end{aligned}
$$

where FPR_{f^*} is the FPR of the best classifier f^* on the best sensitive group z^*. In the fourth line, when $T\to\infty$, the inequality follows from the fact that the last term goes to zero since its numerator is a constant .

Let $q_{A,-}$ and $q_{B,-}$ indicates the converged true value $q^t_{A,-}$ and $q^t_{B,-}$ respectively, where $q_{A,-}=\lim_{t\to\infty}q^t_{A,-}$ and $q_{B,-}=\lim_{t\to\infty}q^t_{B,-}$. Let $\delta^t_{A,-}=q^t_{A,-}-q_{A,-}$ and $\delta^t_{B,-}=q^t_{B,-}-q_{B,-}$ be the estimation errors at round t. By the classical central limit theory, the estimation errors converge at the rate of $O(\frac{1}{\sqrt{t}})$. Therefore,

$$
\begin{aligned}
&\frac{\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T} \\
&=\frac{\sum_t(q^t_{A,-}-q_{A,-})\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t(q^t_{B,-}-q_{B,-})\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}+\underbrace{\frac{q_{A,-}\sum_t\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{q_{B,-}\sum_t\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}}_{Q_{FPR}} \\
&=\frac{\sum_t\delta^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T}-\frac{\sum_t\delta^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}+Q_{FPR} \\
&=\frac{\sum_t O(\frac{1}{\sqrt{t}})}{p_A(1-\mu_{A,+})T}-\frac{\sum_t O(\frac{1}{\sqrt{t}})}{p_B(1-\mu_{B,+})T}+Q_{FPR}
\end{aligned}
$$

where the last inequality follows the fact that $\alpha^t_{A,-} < 1$. Since $\frac{\sum_t O(\frac{1}{\sqrt{t}})}{p_A(1-\mu_{A,+})T} < \frac{O(2\sqrt{T})}{p_A(1-\mu_{A,+})T}$. When $T \to \infty$, the estimation errors are sub-linear and thus go to zero. Therefore,

$$|\mathbb{E}_{x,y,z}\left[\frac{L_{A,-}}{C_{A,-}} - \frac{L_{B,-}}{C_{B,-}}\right]| \leq |\left(1+\eta-\gamma(\eta)\right) FPR_{f^*} + \epsilon(1+\eta) + \underbrace{\left(\frac{q_{A,-}\cdot\sum_t \alpha^t_{A,-}}{p_A(1-\mu_{A,+})T} - \frac{q_{B,-}\cdot\sum_t \alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)}_{Q_{FPR}}| \tag{6.20}$$

Similarly, for the absolute difference of FNR between group A and group B, we have:

$$|\mathbb{E}_{x,y,z}\left[\frac{L_{A,+}}{C_{A,+}} - \frac{L_{B,+}}{C_{B,+}}\right]| \leq |\left(1+\eta-\gamma(\eta)\right) FNR_{f^*} + \epsilon(1+\eta) + \underbrace{\left(\frac{q_{A,+}\cdot\sum_t \alpha^t_{A,+}}{p_A\mu_{A,+}T} - \frac{q_{B,+}\cdot\sum_t \alpha^t_{B,+}}{p_B\mu_{B,+}T}\right)}_{Q_{FNR}}| \tag{6.21}$$

where $FPR_{f^*}(FNR_{f^*})$ is the FPR(FNR) of the best classifier f^* on the best sensitive group z^*.

6.1.3 Proofs for Delayed Case

6.1.3.1 Proof of Lemma 1

Let $\Phi^t_{z,+} = \sum_{f\in\mathcal{F}} w^t_{f,z,+}$. Note that in the delayed feedback setting, the weight updated will incur a delay, but the loss will still be updated every round. Thus the expected loss on group $z,+$ at each round is the same as before:

$$\mathbb{E}[\ell^t_{z,+}] = q^t_{z,+}\cdot\sum_{f\in\mathcal{F}}\frac{w^t_{f,z,+}}{\Phi^t_{z,+}}\cdot\ell^t_{f,z,+} + q^t_{z,-}\cdot\sum_{f\in\mathcal{F}}\frac{w^t_{f,z,-}}{\Phi^t_{z,-}}\cdot\ell^t_{f,z,+} \tag{6.22}$$

Then, the weight transition would be different, where multiple update step could happen in each round. On the other hand, if D_t is a empty set, there will be no update for round t.

$$\begin{aligned}\Phi_{z,+}^{t+1} &= \sum_{f\in\mathcal{F}} w_{f,z,+}^{t+1} \\ &= \sum_{f\in\mathcal{F}} w_{f,z,+}^{t}(1-\eta)^{\sum_{\tau\in D_t}\ell_{f,z}^{\tau}\mathbb{1}\{y=+\}} \\ &= \Phi_{z,+}^{t}(1-\eta\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\sum_{\tau\in D_t}\ell_{f,z,+}^{\tau})\end{aligned}$$

Thus by the recursive function, we have

$$\begin{aligned}\Phi_{z,+}^{T+1} &\leq \Phi_{z,+}^{1}\prod_{t=1}^{T}(1-\eta\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\sum_{\tau\in D_t}\ell_{f,z,+}^{\tau}) \\ &= d\prod_{t=1}^{T}(1-\eta\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\sum_{\tau\in D_t}\ell_{f,z,+}^{\tau}).\end{aligned} \tag{6.23}$$

Although the weight update uses a different schedule, there will still be T updates after T rounds. Thus the same MW update rule applies:

$$\begin{aligned}w_{f,z,+}^{T+1} &= w_{f,z,+}^{1}(1-\eta)^{\sum_{t=1}^{T}\ell_{f,z}^{t}\cdot\mathbb{1}\{y=+\}} \\ &= (1-\eta)^{\sum_{t=1}^{T}\ell_{f,z,+}^{t}}\end{aligned} \tag{6.24}$$

where $w_{f,z,+}^{t}=1$, as all the weights are initialized.

Using 6.23 and 6.24,

$$w_{f,z,+}^{T+1} = (1-\eta)^{\sum_{t=1}^{T}\ell_{f,z,+}^{t}} \leq \Phi_{z,+}^{T+1} \leq d\prod_{t=1}^{T}(1-\eta\sum_{f\in\mathcal{F}}\frac{w_{f,z,+}^{t}}{\Phi_{z,+}^{t}}\cdot\sum_{\tau\in D_t}\ell_{f,z,+}^{\tau}) \tag{6.25}$$

and taking the logarithm of both sides, we have

$$\ln(1-\eta)L_{f,z,+} \leq \ln d + \sum_{t=1}^{T} \ln(1-\eta \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+})$$

$$\ln(1-\eta)L_{f,z,+} \leq \ln d - \eta \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} \tag{6.26}$$

$$\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} \leq (1+\eta)L_{f,z,+} + \frac{\ln d}{\eta} \tag{6.27}$$

Equation 6.26 follows because if $\eta < 1/2$, we can use the inequality $\ln(1-\eta) < -\eta$. This is intuitive as if we always choose weights for positive examples, it reduces to the same bound as in the original MW algorithm.

As before, we assume that the expected error on group $z,+$ when wrong weights $w_{f,z,-}$ are selected is bounded as:

$$\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}} \cdot \ell^t_{f,z,+} \leq \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \ell^t_{f,z,+} + \alpha^t_{z,-} \tag{6.28}$$

Let $D^{max}_{z,+} = \max_t |\mathcal{D}_t|$ be the maximum cardinality of feedback set, we have:

$$\sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} \leq \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} + \sum_{\tau\in D_t} \ell^\tau_{f,z,+}\alpha^t_{z,-} \tag{6.29}$$

where $\alpha^t_{z,-} < 1$ is the difference of loss in expectation made when using the incorrect weights of the MW algorithm on group $z,+$ (Cross-Instance Cost). Then

$$\mathbb{E}[L_{z,+}] = \sum_{t=1}^{T} \left(q^t_{z,+} \cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} + q^t_{z,-} \cdot \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,-}}{\Phi^t_{z,-}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} \right) \tag{6.30}$$

$$\mathbb{E}[L_{z,+}] \leq \sum_{t=1}^{T} \sum_{f\in\mathcal{F}} \frac{w^t_{f,z,+}}{\Phi^t_{z,+}} \cdot \sum_{\tau\in D_t} \ell^\tau_{f,z,+} + \sum_{t=1}^{T} q^t_{z,-} \sum_{\tau\in D_t} \ell^\tau_{f,z,+}\alpha^t_{z,-} \tag{6.31}$$

where using 6.27, we finally obtain:

$$\mathbb{E}[L_{z,+}] \leq (1+\eta)L_{f,z,+} + \frac{\ln d}{\eta} + \sum_{t=1}^{T} q_{z,-}^{t} \sum_{\tau \in D_t} \ell_{f,z,+}^{\tau} \alpha_{z,-}^{t} \tag{6.32}$$

$$\leq (1+\eta)L_{f,z,+} + \frac{\ln d}{\eta} + D_{z,+}^{max} \sum_{t=1}^{T} q_{z,-}^{t} \alpha_{z,-}^{t} \tag{6.33}$$

Similarly,

$$\mathbb{E}[L_{z,-}] \leq (1+\eta)L_{f,z,-} + \frac{\ln d}{\eta} + D_{z,-}^{max} \sum_{t} q_{z,+}^{t} \cdot \alpha_{z,+}^{t}. \tag{6.34}$$

The expected total errors on group z is, adding the two equations above:

$$\mathbb{E}[L_z] \leq (1+\eta)L_{f,z} + 2\frac{\ln d}{\eta} + (D_{z,+}^{max} \sum_{t} q_{z,-}^{t} \cdot \alpha_{z,-}^{t} + D_{z,-}^{max} \sum_{t} q_{z,+}^{t} \cdot \alpha_{z,+}^{t})$$

. Let $D^{max} = \max_{t,z,y} D_{z,y}^{max}$ In the same way, the expected total errors (considering $z = A, B$) is:

$$\mathbb{E}[L] \leq (1+\eta)L_f + 4\frac{\ln d}{\eta} + \alpha D^{max} \tag{6.35}$$

where all the Cross-Instance Costs are condensed in:

$$\alpha = \sum_{z \in \{A,B\}, y \in \{-,+\}} q_{z,y} \sum_{t} \alpha_{z,y}^{t}.$$

6.1.3.2 Proof of Lemma 2

The proof of the lower bound for the loss is largely the same as the non-delayed case, and therefore we only present the final result here:

$$\mathbb{E}[L] \geq \gamma(\eta)L_{f^*} + \alpha D^{max}. \tag{6.36}$$

6.1.3.3 Fairness Bound

Proof We assume group A arrives with probability p, group B arrives with probability 1-p, that is, $\mathbb{P}(z = A) = p$. The expected mean label of group A is defined as $\mu_{A,+} = \mathbb{P}(y = +|z = A)$ and mean label of group B is defined as $\mu_{B,+} = \mathbb{P}(y = +|z = B)$. Each individual classifier is ϵ-fair, thus:

$$|\mathbb{E}_{x,y,z}\left[\frac{L_{f,A,-}}{\sum_{t=1}^{T}\mathbb{1}\{y=-\}\mathbb{1}\{z=A\}}\right] - \mathbb{E}_{x,y,z}\left[\frac{L_{f,B,-}}{\sum_{t=1}^{T}\mathbb{1}\{y=-\}\mathbb{1}\{z=B\}}\right]| \leq \epsilon,\ \forall f \tag{6.37}$$

which represents the cardinality of the selected subset of samples.

The absolute difference of FPR between group A and group B is:

$$|FPR_A - FPR_B| = |\mathbb{E}_{x,y,z}\left[\frac{L_{A,-}}{\sum_{t=1}^{T}\mathbb{1}\{z=A\}\{y=-\}} - \frac{L_{B,-}}{\sum_{t=1}^{T}\mathbb{1}\{z=B\}\{y=-\}}\right]| \tag{6.38}$$

For the sake of notation we define

$$C_{A,-} = \sum_{t=1}^{T}\mathbb{1}\{y=-\}\mathbb{1}\{z=A\} \text{ and } C_{B,-} = \sum_{t=1}^{T}\mathbb{1}\{z=B\}\{y=-\}.$$

Using Lemmas 1 and 2, we have:

$$
\begin{aligned}
&|\mathbb{E}_{x,y,z}\left[\frac{L_{A,-}}{C_{A,-}} - \frac{L_{B,-}}{C_{B,-}}\right]| \\
&\leq |\mathbb{E}_{x,y,z}[\frac{(1+\eta)L_{f^*(A,-),A,-}}{C_{A,-}} + \frac{\frac{\ln d}{\eta}}{C_{A,-}} + \frac{D^{max}_{A,-}\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{C_{A,-}} \\
&\qquad\qquad - \frac{\gamma(\eta)L_{f^*(B,-),B,-}}{C_{B,-}} - \frac{D^{max}_{B,-}\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{C_{B,-}}]| \\
&= |(1+\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(A,-),A,-}}{C_{A,-}}\right] - \gamma(\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \\
&\qquad\qquad \left(\frac{D^{max}_{A,-}\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{C_{A,-}} - \frac{D^{max}_{B,-}\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{C_{B,-}}\right) + \mathbb{E}_{x,y,z}\left[\frac{\ln d}{\eta C_{A,-}}\right]|
\end{aligned}
$$

Using equation 6.37, we have:

$$
\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),A,-}}{C_{A,-}}\right] - \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right]| \leq \epsilon
$$

Moreover, without loss of generality we assume that f^* makes the smallest average loss on group B. This is,

$$
\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(A,-),A,-}}{C_{A,-}}\right] \leq \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),A,-}}{C_{A,-}}\right] \leq \mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \epsilon.
$$

Thus, equation 6.19 becomes:

$$
\begin{aligned}
&\leq |(1+\eta)\left(\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \epsilon\right) - \gamma(\eta)\mathbb{E}_{x,y,z}\left[\frac{L_{f^*(B,-),B,-}}{C_{B,-}}\right] + \\
&\qquad\qquad \left(\frac{D^{max}_{A,-}\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{C_{A,-}} - \frac{D^{max}_{B,-}\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{C_{B,-}}\right) + \mathbb{E}_{x,y,z}\left[\frac{\ln d}{\eta C_{A,-}}\right]| \\
&\leq |\left(1+\eta-\gamma(\eta)\right)FPR_{f^*} + \epsilon(1+\eta) + \left(\frac{D^{max}_{A,-}\sum_t q^t_{A,-}\cdot\alpha^t_{A,-}}{p_A(1-\mu_{A,+})T} - \frac{D^{max}_{B,-}\sum_t q^t_{B,-}\cdot\alpha^t_{B,-}}{p_B(1-\mu_{B,+})T}\right)|
\end{aligned}
$$

6.2 Appendix for Chapter 4: Fairness with Dynamic Feedback

6.2.1 Additional Experiments Results

6.2.1.1 Thresholds for each policy as cost ratio increases

In figure 6-2, we plot the average thresholds of each policy as a function of the cost ratio.

Regardless of the group, `MaxUtil`'s threshold only depends on the parameter for the utility function ($\frac{u_{fp}}{u_{fp}+u_{tp}}$), which is set as 0.5 in the experiment.

For `DemoPar` policy, it consistently overcompensate for the disadvantaged group by assigning a lower average threshold for the disadvantaged group. Remember that demographic parity constraint equalize the rate of possible decisions, and if the disadvantaged group has a lower target variable, the threshold will also be lower.

The case with `EqOpp` is a little bit more complicated. As cost ratio increases, `EqOpp` switches from lower threshold for disadvantaged group to lower threshold for advantaged group. The reason is that equalized opportunity requires both group have equalized false positive rates. In the loan application case, this means the decision maker should issues the same percentage of people loans among those who can repay. Higher cost ratio means the stake of defaulting (false positive rate) is higher for an individual. When the stake of false positive rate is high for an individual, and will lead to the greater target variable decreases. By assigning a lower threshold for the advantaged group, `EqOpp` intentionally increases the false positive rate of the advantaged group in order to match that of the disadvantaged group.

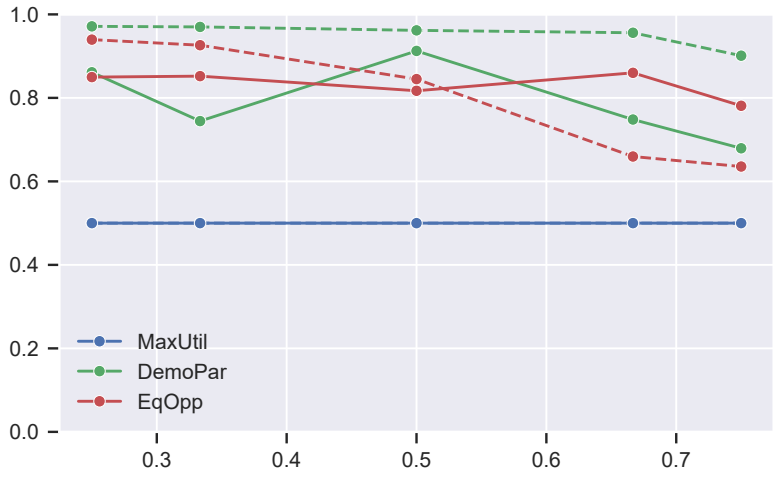


Figure 6-2: Threshold for different fairness policies as a function of cost ratio. The dashed line indicates the threshold for advantaged group, and the solid line indicates the threshold for disadvantaged group.

Bibliography

Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Z. Chen, Nikhil Thorat, Fernanda B. Viégas, Martin Wattenberg, Gregory S. Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, L. Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Vedavyas Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy P. Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529:484–489, 2016.

Ziad Obermeyer, Brian W. Powers, Christine Vogeli, and S. Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366:447 – 453, 2019.

Julia Dressel and Hany Farid. The accuracy, fairness, and limits of predicting recidivism. *Science Advances*, 4, 2018.

Matthew Kay, Cynthia Matuszek, and Sean A Munson. Unequal representation and gender stereotypes in image search results for occupations. *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, 2015.

Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356:183 – 186, 2017.

Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. *ProPublica*, May 23rd, 2016.

S. Verma and J. Rubin. Fairness definitions explained. *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*, pages 1–7, 2018.

White-House. Big data: A report on algorithmic systems, opportunity, and civil rights. 2016.

Kimmo Kärkkäinen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1547–1557, 2021.

Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2434–2442, 2020.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *NIPS*, 2016.

Richard S. Zemel, Ledell Yu Wu, Kevin Swersky, Toniann Pitassi, and Cynthia Dwork. Learning fair representations. In *ICML*, 2013.

Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard S. Zemel. The variational fair autoencoder. *CoRR*, abs/1511.00830, 2016.

Flávio du Pin Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R. Varshney. Optimized pre-processing for discrimination prevention. In *NIPS*, 2017.

Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Kr. Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John T. Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *ArXiv*, abs/1810.01943, 2018.

Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *KDD*, 2017.

Richard A. Berk, Hoda Heidari, Shahin Jabbari, Matthew Joseph, Michael Kearns, Jamie H. Morgenstern, Seth Neel, and Aaron Roth. A convex framework for fair regression. *ArXiv*, abs/1706.02409, 2017.

Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *NeurIPS, 29*, pp. 3315–3323, 2016.

Smitha Milli, John Miller, Anca D. Dragan, and Moritz Hardt. The social cost of strategic classification. *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019.

Ganesh Ghalme, Vineet J. Nair, Itay Eilat, Inbal Talgam-Cohen, and Nir Rosenfeld. Strategic classification in the dark. *ArXiv*, abs/2102.11592, 2021.

Xueru Zhang, Mohammad Mahdi Khalili, Kun Jin, Parinaz Naghizadeh Ardabili, and M. Liu. Fairness interventions as (dis)incentives for strategic manipulation. In *ICML*, 2022.

Vijay Keswani and L. Elisa Celis. Addressing strategic manipulation disparities in fair classification. *ArXiv*, abs/2205.10842, 2022.

Yi Sun, Iván Díaz, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. Towards reducing biases in combining multiple experts online. In *IJCAI*, 2021.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard S. Zemel. Fairness through awareness. *ArXiv*, abs/1104.3913, 2012a.

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

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent trade-offs in the fair determination of risk scores. Innovations in Theoretical Computer Science, 2017.

Matt J. Kusner, Joshua R. Loftus, Chris Russell, and Ricardo Silva. Counterfactual fairness. In *NIPS*, 2017.

Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez-Rodriguez, and Krishna P. Gummadi. Fairness beyond disparate treatment and disparate impact: Learning classification without disparate mistreatment. In *Proc. of the 26th Int. Conf. on World Wide Web*, pages 1171–1180, 2017.

H. Heidari, Vedant Nanda, and K. Gummadi. On the long-term impact of algorithmic decision policies: Effort unfairness and feature segregation through social learning. In *ICML*, 2019.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proc. of the 3rd Innovations in Theoretical Computer Science Confe.*, ITCS '12, pages 214–226, 2012b. doi: 10.1145/2090236.2090255.

Ben Green. The false promise of risk assessments: epistemic reform and the limits of fairness. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020.

Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *ICML*, 2018.

Silvia Chiappa. Path-specific counterfactual fairness. In *AAAI*, 2019.

Elliot Creager, David Madras, T. Pitassi, and R. Zemel. Causal modeling for fairness in dynamical systems. *ArXiv*, abs/1909.09141, 2020.

Amir-Hossein Karimi, Bernhard Schölkopf, and Isabel Valera. Algorithmic recourse: from counterfactual explanations to interventions. *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 2021.

Stephen Gillen, Christopher Jung, Michael Kearns, and Aaron Roth. Online learning with an unknown fairness metric. In *NeurIPS, 31*, pages 2600–2609. 2018.

Matthew Joseph, Michael Kearns, Jamie H. Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. In *NeurIPS*, pages 325–333, 2016.

Yang Liu, Goran Radanovic, Christos Dimitrakakis, Debmalya Mandal, and David C. Parkes. Calibrated fairness in bandits. In *Proceedings of the 4th Workshop on Fairness, Accountability, and Transparency in Machine Learning (Fat/ML 2017)*, 2017.

Yahav Bechavod, Katrina Ligett, Aaron Roth, Bo Waggoner, and Steven Z. Wu. Equal opportunity in online classification with partial feedback. In *NeurIPS, 32*, pages 8972–8982. 2019.

Avrim Blum, Suriya Gunasekar, Thodoris Lykouris, and Nati Srebro. On preserving non-discrimination when combining expert advice. In *NeurIPS, 31*, pages 8376–8387. NeurIPS 2018, 2018.

L. Liu, Sarah Dean, Esther Rolf, Max Simchowitz, and Moritz Hardt. Delayed impact of fair machine learning. *ArXiv*, abs/1803.04383, 2018.

A. D'Amour, Hansa Srinivasan, J. Atwood, P. Baljekar, D. Sculley, and Yoni Halpern. Fairness is not static: deeper understanding of long term fairness via simulation studies. *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 2020a.

Hussein Mouzannar, Mesrob I. Ohannessian, and Nathan Srebro. From fair decision making to social equality. *Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2019.

Min Wen, Osbert Bastani, and Ufuk Topcu. Algorithms for fairness in sequential decision making. In *AISTATS*, 2021.

Jiechuan Jiang and Zongqing Lu. Learning fairness in multi-agent systems. In *NeurIPS*, 2019.

X. Zhang, Ruibo Tu, Y. Liu, M. Liu, Hedvig Kjellstrom, Kun Zhang, and C. Zhang. How do fair decisions fare in long-term qualification? *ArXiv*, abs/2010.11300, 2020.

Sanjeev Arora, Elad Hazan, and Satyen Kale. The multiplicative weights update method: a meta-algorithm and applications. *Theory of Computing*, 8:121–164, 2012.

Pooria Joulani, A. György, and Csaba Szepesvari. Online learning under delayed feedback. In *ICML*, 2013.

Michele Donini, Luca Oneto, Shai Ben-David, John S Shawe-Taylor, and Massimiliano Pontil. Empirical risk minimization under fairness constraints. In *NeurIPS, 31*, pages 2791–2801, 2018.

Alexander D'Amour, Hansa Srinivasan, James Atwood, Pallavi Baljekar, D. Sculley, and Yoni Halpern. Fairness is not static: Deeper understanding of long term fairness via simulation studies. In *Proc. of the Conference on Fairness, Accountability, and Transparency*, page 525–534, 2020b. doi: 10.1145/3351095.3372878.

Shlomo Yitzhaki. Relative deprivation and the gini coefficient. *Quarterly Journal of Economics*, 93:321–324, 1979.

Matja Perc. The matthew effect in empirical data. *Journal of the Royal Society Interface*, 11, 2014.

Juan C. Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. *ArXiv*, abs/2002.06673, 2020.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *ArXiv*, abs/1706.06083, 2018.

Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *ArXiv*, abs/1607.02533, 2017.