Topics in Sequential Decision Making and Algorithmic Fairness

by

Laura K. Niss

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in the University of Michigan
2022

Doctoral Committee:

Assistant Professor Yuekai Sun, Co-Chair
Professor Ambuj Tewari, Co-Chair
Professor Martin Strauss
Assistant Professor Joseph Jay Williams

Laura K. Niss
lniss@umich.edu
ORCID iD: 0000-0001-7467-4153

ACKNOWLEDGMENTS

To start, I must acknowledge the random luck that has led to the many opportunities I have had over the years. It has been my privilege to take advantage of these opportunities, accept support from family–both given and chosen–and be positively influenced by a number of superb individuals. I attribute my success as the culmination of these many encounters.

I thank my advisor, Ambuj Tewari, for encouraging me to explore my interests. It was this freedom that allowed me to delve into diverse research fields, leading to many of my most valued experiences and relationships during this period. Without his influence, support, and patience, I would not have completed this dissertation. I also thank my co-advisor, Yuekai Sun. His enthusiasm and knowledge in shared research interests fostered my identity as a researcher, aiding my growth in technical expertise as well as a sense of belonging in an academic community. To my committee members Martin and Joseph, I also thank you for aiding in my development. Martin had a significant influence at the beginning of my graduate career, and I am thankful for the path our collaboration led me down. Since our first meeting, Joseph has been supportive and encouraging. Thank you for the opportunities and connections you have provided me, and for the insightful and fruitful research conversations over the years. This dissertation is better for them. I also acknowledge the several years of support from the NSF via grant DMS1646108, and thank those in the department who worked so hard to receive this funding.

To my friends old and new, thank you for filling these six years with thoughtful conversations, commiserations, adventures, and laughter. It is the many dinner parties, scotch nights, and porch strawberries that define my best memories of graduate school. I thank my collaborators, whose conversations and questions both sparked and fanned my interests. In particular, I thank both Amanda and Alex. Your friendship, insight, and abilities were a necessary component in my life and my work. I thank all those involved in the Student Council and JEDI, particularly team pedagogy, for being a light of hope and connection when we were plunged into the isolation of 2020. To my oldest friends, thank you Becky and Madalyn. I always appreciated your useful commentary on graduate school and academia, and our many discussions on values, intention, and living a meaningful life. I am so grateful to have you. To Zoe and Charlotte, thank you for being the amazing humans that you are. Thank you Charlotte for your support, insightful knowledge, and all the effort you put into our fun evenings and adventures. Thank you Zoe, for literally everything.

We walked this journey together from the first day, and I could not have done it without you. Thank you for being my companion through every difficult class, success, failure, and milestone. Thank you for being brave and pursuing activities we are both really bad at. Thank you for pushing me to do things I would never have done on my own. Your hard work and dedication both in research and community causes was a constant inspiration to me, lifting me up during times of my own lost motivation.

Without my family, I would not have succeeded in this endeavor. No matter what I set my sights on, they have given me the support, both materially and emotionally, that allowed me to prevail. In particular, I thank my parents for letting me forge my own path, however winding it may seem. It is your endless encouragement and acceptance that led me to this accomplishment. To my brother Tom, I am unceasingly impressed by your ability to bring people together and tie our family ever closer. To my sister Kim, you inspired me with your hard work as well as commiserated and encouraged me when I needed it most. I thank you both for making me a better person. To my partner Collin, I cannot express how grateful I am for you. I could create a list umpteen lines long describing the ways you helped me, though I will refrain and keep it short. Thank you for your infinite patience in tutoring me during my post-bacc, it was your help and understanding that created my opportunity of studying at Michigan. Thank you for both the many flights you endured to visit and for then uplifting your life to live together. Thank you for believing in me when I did not believe in myself. You challenge and inspire me every day, and for that I will always love you. More than anything else, it has been your unwavering support and certainty in my abilities that has led to this culmination of six years' work. Finally, I thank all my family, near and far, for perpetually reminding me what is truly important in life.

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

ABSTRACT

The ability to collect and process data has greatly expanded the areas of application for data driven inference, predictions, and decisions. How to collect and modify data is dependent upon the ultimate goal. Two areas of research with focus on these questions are sequential decision making and algorithmic fairness. Sequential decision making is the process of a learner choosing an action, observing the outcome, and using this and previous information to determine the next action to take. Algorithmic fairness is the overarching term used to describe concern over algorithmic decisions being seen as unfair to certain groups or individuals. Biases present in training data may rise from historical inequities or improper representation. This dissertation addresses four problems in these two areas: policies for contaminated stochastic multi-armed bandits, fair representation through convex hull feasibility sampling, data debiasing, and implications of a sequential pipeline of fair/biased decisions.

We start in Chapter 2 by considering the stochastic multi-armed bandit problem, with the added assumption that rewards can be contaminated some fixed proportion of the time. This reflects the scenario of when the reward is from a human response. Here there is no guarantee the observed reward is from the true reward distribution of the action. To account for the contamination, we propose an Upper Confidence Bound (UCB) policy that relies on robust mean estimators. We derive inequality bounds on these estimators in the contaminated setting and give upper bounds on the regret, showing they are comparable to UCB policies in the standard stochastic setting. Through simulations, we show the effectiveness or our policies under different types of contamination.

Bias in training data is often split into two categories, representation bias and historical bias. Representation bias refers to data with no or limited samples from groups within the target population. Representation bias can result in unfair outcomes for the underrepresented groups. Historical bias refers to unwanted correlations between protected attributes and other features caused by societal inequities. It is an inherent property of the data and cannot be attenuated by more data.

Addressing representational bias, Chapter 3 introduces the convex hull feasibility sampling problem. Here we develop a framework for sequentially testing whether a known point lies within the convex hull of a set of points with unknown distributions. This represents the problem of whether or not it is possible to sample an equally representative data set among labeled groups when the distribution of the sampling sources is unknown. We provide theoretical results in the 2D

setting and simulations of our policy in 2 and 3 dimensions.

In contrast, Chapter 4 addresses historical bias by proposing a data debiasing method based on a factor model. The goal is to remove variation caused by protected attributes that are undesirable during training. We compute the correlation between the debiased data and the original protected attributes and show that in ideal cases there is no correlation. We show empirical results with a case study.

Chapter 5 explores how bias across multiple decisions—what we call a pipeline—impacts the final outcome. We show how fair decisions at each decision point can perpetuate a fair outcome, and also how a biased decision can prevent fair outcomes further down the pipeline. This highlights the importance of representative data at each training and decision period.

CHAPTER 1

Introduction

Machine learning has become a ubiquitous way to optimize and personalize decisions and experiences. As we harness the power of data, we see both capitalistic and humanitarian applications: methods that can optimize website layouts to increase revenue can be applied to online courses to optimize interaction and learning; methods for targeted advertising can also be applied for targeted medical interventions. As we simplify equipment and software (thus lowering the threshold of who is able to harness machine learning) we continue to discover a plethora of benefits and challenges. One sees benefits within smaller scale applications, such as apps that allow small businesses to optimize emails for responses or optimizing interventions to retain learners in an online course. A significant challenge of more recent interest is that of ensuring fair outcomes. As algorithms have impacted more areas of our lives, it has become increasingly clear that human biases are not attenuated by the machine. Motivated by these benefits and challenges, and with a particular focus on sequential decision making methods, this dissertation presents four contributions: policies for contaminated stochastic multi-armed bandits, fair representation through convex hull feasibility sampling, data debiasing, and implications of a sequential pipeline of fair/biased decisions.

Sequential decision making (SDM) is the process of iteratively collecting information while updating decision parameters based on the cumulative information available. Examples of applications include A/B testing web layout to optimize interaction, computer adaptive testing, and fair data collection. Motivated by SDM applications in education, we present our work on contaminated stochastic multi-armed bandits in Chapter 2. When considering applying SDM where feedback comes from students, experience tells us that the feedback may not always be effort-full or intentional. Therefore, we approach this problem as a contaminated stochastic bandit and adapt robust mean estimators for upper confidence bound algorithms in a stochastic setting for when a fixed proportion of rewards are uncorrelated to the action.

Algorithmic fairness is a field of research concerned with the impact of data driven predictions and decisions that can give an unfair advantage or cause harm to a group or an individual. We consider a sequential method for checking feasibility of collecting representationally fair data in Chapter 3, and an offline method for debiasing data with inherent historical bias in Chapter 4.

In Chapter 5, we discuss the implications of fairness within a pipeline of multiple algorithmic decisions.

In the following sections we provide an overview of the stochastic multi-armed bandit problem, whose framework is utilized in Chapters 2 and 3, and an introduction to algorithmic fairness. We also provide the context and motivation along with our research contributions for each of the chapters.

1.1 Stochastic Multi-armed Bandits

The stochastic multi-armed bandit (MAB) problem was first described in Thompson [1933]. The standard problem is often presented as a gambling analogy, which also describes the etymology of the word bandit in multi-armed bandit. A one-armed bandit is a colloquial term for a slot machine, a gambling device where you put in money, pull a lever (hence the term arm), and receive some reward. Those who have gambled understand that you do not, on average, earn more money than you spend (hence the term bandit). Legally though, there must be some chance of winning. We can think of this chance of winning and the amount won as coming from the reward distribution of our slot machine. In the multi-armed bandit problem, a learner is faced with many slot machines, and can play only one at a time. The challenge for the learner is to decide how much time to spend exploring each machine to estimate its average reward, and how much time playing (exploiting) the estimated best machine to gain the most reward.

Bandit problems are sequential decision making problems where a learner only observes the feedback from the chosen action, and gains no information about the unplayed actions. Stochastic multi-armed bandits refers to a class of bandit problems where the rewards for each action are assumed to come from fixed distributions. Notationally, we have $a \in [K]$ possible actions, such that the reward for action a at time t, $r_a(t)$, is a random sample from the fixed distribution D_a. At each time step, the learner only sees the reward for the action chosen, so the difficultly lies in balancing exploring all actions and exploiting the optimal action. The method of determining when to explore and when to exploit is called a policy, typically denoted with π.

There are typically two goals in MABs that determine a policy's strategy, either to identify the optimal arm in relation to a stopping rule, or to minimize regret during play. This first instance is called best arm identification, sometime referred to as pure exploration. Here there are two scenarios. In the first, there is a fixed budget, with the goal being to identify the optimal arm(s) with the highest confidence possible within budget. In the second setting there is a fixed confidence and the goal is to identify the optimal arm(s) with that confidence using the minimum number of samples. We adapt the best arm identification problem with fixed confidence to the convex hull feasibility problem in Chapter 3.

The other goal, minimizing regret, is the same as maximizing reward. To minimize regret is to minimize the difference of your cumulative reward to the optimal reward achieved when sampling only from the action with the highest expected reward. Here we define expected regret for a policy π after T rounds as

$$\bar{R}_T(\pi) = \sum_{t=1}^{T} E[r^* - r_{A_t}],$$

where r^* is the reward of the optimal action and r_{A_t} is the reward for the action chosen at time t.

Adversarial bandits drop the assumption that the rewards are drawn from a fixed distribution. Instead, if the adversary is oblivious, it is allowed to observe the learner's policy before play and then chooses the rewards for each action a and for each time step t. We only consider an oblivious adversary in the following discussions, not an adaptive adversary which can pick rewards during play based on the learner's history.

It might seem questionable where such a strong adversary exists when implementing an adversarial bandit algorithm, but the idea is to account for instances where the rewards are not stochastic or stationary. By proving effectiveness in the extreme case, one has also shown effectiveness in less powerful and more realistic cases. It is intuitive to see that for adversaries that are not malicious but instead close to stochastic, this adversarial assumption is quite strong, and an algorithm that takes advantage of the near stochasticity might perform better than one that doesn't. This is indeed the setting we consider for contaminated stochastic bandits as presented in chapter 2.

1.1.1 Contaminated Stochastic Bandits

One variation of MABs that has been minimally explored is when the stochastic reward assumption is mostly retained, but some rewards are unrelated to the action they are sampled from, which we call contaminated stochastic bandits. This framework and policies under this setting are the purview of Chapter 2. This setting may be considered a mix of stochastic and adversarial bandits, and is motivated by obtaining feedback from people, in particular students. A simple example is the wording of a question. Here, a course instructor wishes to try out several different wordings of a question presented to students in an online setting. The instructor wishes to pick the wording that results in the most correct answers, as that suggests it has the clearest wording. If all students closely read the question and answer to the best of their ability, then the data collected can lead to valid inferences of the best wording. However, this is unlikely to be true. Some students will skim the question and answer on their assumptions of the content, some may randomly answer simply to move on, or some may guess without learning the relevant content first. These are all cases where the information collected is unrelated the instructors question, and may skew the inference.

Formalizing this scenario, in the contaminated stochastic bandits setting we allow the adversary

to corrupt any reward as long as at any time t there is no more than an ϵ-fraction of contaminated points. We allow the adversary to give unbounded contaminated rewards that can be chosen with full knowledge of the learner's history as well as current and future rewards. This setting allows the adversary to act differently across actions. In this setting, we limit calculating the expected regret with respect to the true distributions of the actions, and not the observed, possibly contaminated rewards.

A ubiquitous class of policies in the bandit literature are those based on upper confidence bounds (UCB). The class of UCB algorithms is based on the optimism principle, that in the face of uncertainty, the "learner should act as if the environment is as nice as plausibly possible" [Lattimore and Szepesvári, 2020]. This optimism principle along with the use of confidence bounds first appeared in Lai and Robbins [1985]. UCB algorithms follow the pattern of first choosing each action once, then for each time step t, estimating an upper confidence bound of the mean reward for each action, $UCB_a(t)$, and choosing the next action as the one with the highest estimate, $A_t = \max_{a \in [K]} UCB_a(t)$. Theoretical results for the performance of UCB policies with respect the expected regret depend on the concentration bounds of the mean rewards.

Our contribution is to adapt two known robust statistics for the mean, the trimmed mean and the short mean, to UCB under the contaminated bandits setting. By proving concentration bounds for these robust mean estimators in this bandit setting, we are able to present two UCB policies. We provide theoretical guarantees for the upper bound of regret for the two policies and empirical evidence of their superior performance under several contamination settings.

1.2 Algorithmic Fairness

Biased outcomes in machine learning has seen increasing scrutiny in recent years. Originally, algorithmic decisions had been seen by some as a fix for biased human-made decisions. It is increasingly clear, though, that it is nearly impossible to collect training data that is itself free from bias in some form. These biases in the training data cannot be removed by an impassioned algorithm, which will inevitably pick up on some undesired patterns and may perpetuate or even magnify them.

In general, we can separate data biases into two categories. Representation bias is bias in which data does not accurately capture variation of the target population. For example, this could be data which has no or few examples from minority groups, making it hard to accurately learn relationships between features and outcomes for these groups. Or it could be that the variation of features within one group is significantly less that another, and does not reflect the variation in the true population. In either case, an algorithm may perform well based on collected data in training and poorly in production. The other bias which is typically harder to handle is historical bias.

This is the bias seen in data that is the result of societal inequities. For example, over-policing in certain neighborhoods will result in more documented crimes in those neighborhoods, even if the true relative crime rate is the same as in another less-policed neighborhood. With historical bias, it is sometimes clear what societal mechanisms lead to the bias, but this is not always the case.

In the discourse of algorithmic fairness, a major difficulty is: what is fairness in both an intuitive and mathematical sense? Depending on one's world view, fairness could mean equity in results, or equality in opportunity, among other beliefs. Additionally, there is the question of which groupings to consider fairness against. Is there only one attribute, such as gender identity, that outcomes should be fair for, or a cluster of so called protected attributes? Another proposal is to not consider these groupings and instead say that similar people should be treated similarly, but then the question arises of how to measure similarity.

There are a vast amount of considerations when one wants to achieve a fair algorithm. Significant contributions have been made in the field, each generally pertaining to a specific part in the algorithmic pipeline. We can roughly define this pipeline as data collection, preprocessing, algorithmic training, and post-processing. Methods at each stage have their own pros and cons, and each depends on the result of the process before it. In this dissertation, we consider methods for both data collection and preprocessing to address representation bias and historical bias, respectively.

1.2.1 Representative Data Feasibility Sampling

More and more, data collection is being seen as a primary defense against bias in algorithmic decisions. Indeed, it is often a first choice of action to collect more data if an algorithm results in biased outcomes [Holstein et al., 2019]. While this is possible in settings where group membership and (when applicable) outcome labels are known, there are circumstances where data collection comes from sources with unknown distributions of attributes. In this case, it is not possible to know how exactly to collect representative data or target collecting more data from a specific group. This is the scenario we address is Chapter 3.

To the best of our knowledge, the first work to address data collection as a part of bias mitigation is Abernethy et al. [2020]. Here the goal is to optimize over both a loss function for accuracy and a loss function for fairness. They assume an infinite availability of group labeled data, and at every iteration of sampling they choose the sample which will either minimize the accuracy loss or minimize the fairness loss. Along this vein of work is Shekhar et al. [2021]. Their goal is to identify a minimax optimal classifier across the sampling proportion of protected attributes and the loss of the worst performing group. These methods assume a fixed definition of fairness throughout a project. In reality, the desired fairness at the beginning of a project may not be fixed, and just

like parameters in a training algorithm, they may be adjusted over the life of a project. We explore these implications with regards to these two algorithms in Appendix C.

The ability to sample from particular groups is not always possible. Samples may be drawn from a source where the distribution over desired attributes is unknown. For example, a particular sample's label may be determined only after collection. This is the motivation for introducing the convex hull feasibility sampling problem, which aims to determine if a desired distribution over protected groups is feasible given a fixed number of sources with unknown distributions over the protected groups. In the Bernoulli setting, we give a lower bound on the expected sample size in the infeasible case and an oracle lower bound of the expected sample size in the feasible case. We define the direction of greatest uncertainty and present three policies based on this direction, along with a naive Uniform policy. Using high-probability upper bounds, we prove that one policy, Lower Upper Confidence Bound Mean is superior to Uniform. We define the multinomial version of the problem along with adjusted algorithms, and using simulations show the performance of our three policies outperform Uniform under the Bernoulli setting and the multinomial setting with three dimensions.

1.2.2 Data Debiasing

When bias is the result of systemic historical causes, more data will not reduce the bias present in the data. In Chapter 4, we introduce a factor model prevalent in genetics applications to model the contributions of the protected and permissible attributes to the representation. We treat the variation that is present in the data due to protected attributes (e.g. race) as unwanted, and we propose a method to remove this unwanted variation (and thus debias the data). We further compute the correlation between the debiased data and the original protected attributes. In ideal cases, we show that there is no correlation, and therefore our debiased data satisfies a relaxed version of conditional parity [Ritov et al., 2017]. Using the COMPAS dataset, we show that debiasing reduces disparity in the false positive and false negative rates between the majority and minority class.

1.2.3 Fair Pipelines

Most of the discourse on fair algorithms has focused on the impact of a single point of decision. In Chapter 5, we discuss fairness within a decision pipeline. That is, when multiple, potentially disjoint, decisions are made before a final outcome is measured as fair. We use the example of a two stage hiring process throughout. The first stage represents filtering of applicants for interviews, and the second stage represents the filtering of interviewees for hiring. Using a relaxed definition of equal opportunity, we highlight the fact that biased decisions early in the pipeline can prohibit a fair outcome regardless of the fairness of subsequent decisions, and show the compounding effects

of fair decisions.

1.3 Publications and Contributions

The work in this dissertation is the combination of a first author publication, collaborative papers, and a preprint. Included in the appendix are the results of an exploratory project created to introduce undergraduates to research. When the work is collaborative, we specify the contribution of the author.

- Chapter 2 is based off of Niss and Tewari [2020], which was published in the electronic proceedings of the Uncertainty in Artificial Intelligence conference. This is work in collaboration with Ambuj Tewari.
- Chapter 3 is, at the time of this dissertation submission, a preprint based on collaborative work with Ambuj Tewari and Yuekai Sun [Niss et al., 2022].
- Chapter 4 is based off of Bower et al. [2018], which was presented at the FAT-ML workshop at ICML. All authors contributed equally to the publication. We note that the author discovered the initial proofs, and Amanda Bower performed most of the experimental work. This work is also included in Alexander Vargo's Dissertation as part of their degree requirements [Vargo, 2020].
- Chapter 5 is based off of Bower et al. [2017], which was published as part of the FAT-ML workshop at KDD. The author contributed the initial results and analysis in Section 5.2, with all authors contributing equally to the original publication.
- Appendix C is the result of a project designed to introduce undergraduates to research on algorithmic fairness. The work presented within this dissertation is contributed solely by the author.

CHAPTER 2

What You See May Not Be What You Get: UCB Bandit Algorithms Robust to ϵ-Contamination

2.1 Introduction

We first review the problem of stochastic multi-armed bandits (sMAB) with contaminated rewards, or contaminated stochastic bandits (CSB). This scenario assumes that rewards associated with an action are sampled i.i.d. from a fixed distribution and that the learner observes the reward after an adversary has the opportunity to contaminate it. The observed reward can be unrelated to the reward distribution and can be maliciously chosen to fool the learner. An outline for this setup is presented in Section 2.2.

We are primarily motivated by the use of bandit algorithms in education, where the rewards often come directly from human opinion. Whether responses come from undergraduate students, a community sample, or paid participants on platforms like MTurk, there is always reason to believe some responses are careless or inattentive to the question or could be assisted by bots [Necka et al., 2016, Curran, 2016].

An example in education is a recent paper testing bandit Thompson sampling to identify high quality student generated solution explanations to math problems using MTurk participants [Williams et al., 2016]. Using a rating between 1-10 from 150 participants, the results showed that Thompson sampling identified participant generated explanations that when viewed by other participants significantly improved their chance of solving future problems compared to no explanation or "bad" explanations identified by the algorithm. While the proportion of contaminated responses will always depend on the population, recent work suggests even when screening out fraudulent participants, between $2 - 30\%$ of MTurk participants give low-quality samples [Ahler et al., 2019, Ryan, 2018, Necka et al., 2016]. This is consistent with measurements of careless and inattentive responses seen in survey data, which reports $1 - 30\%$ with an estimated mode of $8 - 12\%$, with the conclusion that these responses are generally not a random sample [Curran, 2016]. Accounting for these low quality responses is especially relevant in educational setting

where the number of iterations an algorithm can run is often significantly smaller than those used by big tech (e.g. advertising).

Recent work in CSB has various assumptions on the adversary, the contamination, and the reward distributions. Many papers require the rewards and contamination to be bounded [Kapoor et al., 2018, Gupta et al., 2019, Lykouris et al., 2018]. Others do not require boundedness, but do assume that the adversary contaminates uniformly across rewards [Altschuler et al., 2019]. All works make some assumption on the number of rewards for an action an adversary can contaminate. We discuss previous work more thoroughly in Section 2.3.

Our work expands on these papers by allowing for a full knowledge adaptive adversary that can give unbounded contamination in any manner. However, there is a trade off when compared to work assuming bounded rewards and contamination: we require an estimate of the upper bound on the reward variance. This can often allow for simpler implementation than some algorithms that require boundedness, as we will discuss in Section 2.4. Our constraint on the adversary is that for some fixed ε, no more than ε proportion of rewards for an action are contaminated. We provide a ε-contamination robust UCB algorithm by first proving concentration inequalities for two robust mean estimators in the ε-contamination context. We are able to show that the regret of our algorithm analyzed on the true reward distributions is $\mathcal{O}(\sqrt{KT \log T})$ provided that the contamination proportion is small enough. Through simulations, we show that with a Bernoulli adversary, our algorithm outperforms algorithms designed for stochastic (UCB1) and adversarial (EXP3) bandits as well as those that have "best of both worlds" guarantees (EXP3++ and TsallisInf) even when our constraint on the adversary is broken.

Though we are motivated by of bandit algorithms applications in education and use this context to determine appropriate parameters in the simulations, we point out opportunities for CSB modeling to arise in other contexts as well.

Human feedback: There is always a chance that human feedback is careless or inattentive, and therefore is not representative of the underlying truth related to an action. This may appear in online surveys that are used for A/B testing, or as is the case above in the explanation generation example. Adaptive surveys, such as choosing question ordering to minimize dropout rates, are also an example where the sample sizes can be small compared to other bandit deployments.

Click fraud: Internet users who wish to preserve privacy can intentionally click on ads to obfuscate their true interests either manually or through browser apps. Similarly, malware can click on ads from one company to falsely indicate high interest, which can cause higher rankings in searches or more frequent use of the ad than it would otherwise merit [Pearce et al., 2014, Crussell et al., 2014].

Measurement errors: If rewards are gathered through some process that may occasionally fail or be inaccurate, then the rewards may be contaminated. For example, in health apps that use activity monitors, vigorous movement of the arms may be perceived as running in place [Feehan et al., 2018, Bai et al., 2018].

2.2 Problem Setting

Here we specify our notation and present the ε-contaminated stochastic bandit problem. We then argue for a specific notion of regret for CSB. We compare our setting to others current in the field in Section 2.3.

Notation We use $[K]$ to represent $\{1, ..., K\}$ for $K \in \mathbb{R}$ to represent the number of actions and the indicator function $\mathbb{I}\{\cdot\}$ to be 1 if true and 0 otherwise. Let $N_a(t)$ be the number of times action a has been chosen at time t and $\mathbf{x}_a(t) = \{x_a(1), ..., x_a(N_a(t))\}$ to be the vector of all observed rewards for action a at time t. The suboptimality gap for action a is Δ_a and we define $\Delta_{\min} = \min_{a \in [K]} \Delta_a$.

2.2.1 ε-Contaminated Stochastic Bandits

A basic parameter in our framework is ε, the fraction of rewards for an action that the adversary is allowed to contaminate. Before play, the environment picks a true reward $r_a(t) \sim D_a$ from fixed distribution D_a for all $a \in [K]$ and $t \in [T]$. The adversary observes these rewards and then play begins. At time $t = 1, 2, ..., T$ the learner chooses an action $A_t \in [K]$. The adversary sees A_t then chooses an observed reward $x_{A_t}(t)$ and then the learner observes only $x_{A_t}(t)$.

We present the contaminated stochastic bandits game in algorithm 1.

Algorithm 1: Contaminated Stochastic Bandits

input: Number of actions K, time horizon T.
fix **:** $r_a(t)\ \forall a \in [K],\ t \in [T]$.
Adversary observes fixed rewards.
for $t = 1, ..., T$ **do**
  Learner picks action $A_t \in [K]$.
  Adversary observes A_t and chooses $x_{A_t}(t)$.
  Learner observes $x_{A_t}(t)$.
end

We allow the adversary to corrupt in any fashion as long as for every time t there is no more than an ε-fraction of contaminated rewards for any action. That is, we constrain the adversary such

that,

$$\forall a \in [K],\ \forall t \in [T],\ \sum_{i=1}^{N_a(t)} \mathbb{I}\{r_a(i) \neq x_a(i)\} \leq \varepsilon \cdot N_a(t).$$

We allow the adversary to give unbounded contamination that can be chosen with full knowledge of the learner's history as well as current and future rewards. This setting allows the adversary to act differently across actions and places no constraints on the contamination itself, but rather the rate of contamination.

2.2.2 Notion of Regret

A traditional goal in bandit learning is to minimize the observed cumulative regret gained over the total number of plays T. Because the adversary in this model can affect the observed cumulative regret, we argue to instead use a notion of regret that considers only the underlying true rewards. We call this uncontaminated regret and give the definition below for any time T and policy π in terms of the true rewards r,

$$\bar{R}_T(\pi) = \max_{a \in [K]} \mathbb{E}\left[\sum_{t=1}^{T} r_a(t) - \sum_{t=1}^{T} r_{A_t}(t)\right]. \tag{2.2.1}$$

This definition equation (2.2.1) is first mentioned in Kapoor et al. [2018] along with another notion of regret that compares the sum of the observed (possibly contaminated) rewards to the sum of optimal, uncontaminated rewards,

$$\bar{R}_T(\pi) = \max_{a \in [K]} \mathbb{E}\left[\sum_{t=1}^{T} r_a(t) - \sum_{t=1}^{T} x_{A_t}(t)\right]. \tag{2.2.2}$$

We argue that equation (2.2.2) gives little information about the performance of an algorithm. This notion of regret can be negative, and with no bounds on the contamination it can be arbitrarily small and potentially meaningless. We believe that any regret that compares a true component to an observed (possibly contaminated) component is not a useful measure of performance in CSB as it is unclear what regret an optimal strategy should produce.

2.3 Reltated Work

We start by briefly addressing why adversarial and "best of both world" algorithms are not optimized for CSB. We then cover relevant work in robust statistics, followed by current work in robust bandits and how our model differs and relates.

2.3.1 Adversarial Bandits

Adversarial bandits with an oblivious environment allows the adversary to first look at the learners policy and then choose all rewards before the game begins. If the learner chooses a deterministic policy, the adversary can choose rewards such that the learner cannot achieve sublinear worst-case regret [Lattimore and Szepesvári, 2020]. Algorithms such as EXP3 [Auer et al., 2002] are thus randomized, but their regret is analysed with respect to the best fixed action where "best" is defined using the *observed* rewards. There are no theoretical guarantees with respect to the uncontaminated regret, so it is not immediately clear how they will perform in a CSB problem. We remark that adversarial analysis assumes uniformly bounded observed rewards whereas we allow observed rewards to be unbounded. Additionally, the general adversarial framework does not take advantage of the structure present in CSB, namely that the adversary can only corrupt a small fraction of rewards, so it is likely that performance improvements can be made.

2.3.2 Best of Both Worlds

A developing line of work is algorithms that enjoy "best of both worlds" guarantees. That is, they perform well in both stochastic and adversarial environments without knowing a priori which environment they will face. Early work in this area [Auer and Chiang, 2016, Bubeck and Slivkins, 2012] started by assuming a stochastic environment and implementing some method to detect a failure of the i.i.d. assumption on rewards, at which point the algorithm switches to an algorithm for the adversarial environment for the remainder of iterations. Further work implements algorithms that can handle an environment that is some mixture of stochastic and adversarial, as in EXP3++ and TsallisInf [Seldin and Slivkins, 2014, Zimmert and Seldin, 2019].

While these algorithms are aimed well for a stochastic environment with some adversarial rewards, they differ from contamination robust algorithms in that all observed rewards are thought to be informative. Their uncontaminated regret has not been analysed and therefore there are no guarantees in the CSB setting.

2.3.3 Contamination Robust Statistics

The ε-contamination model we consider is closely related to the one introduced by Huber in 1964 [Huber, 1964]. Their goal was to estimate the mean of a Gaussian mixture model where ε fraction of the sample was not sampled from the main Gaussian component. There has been a recent increase of work using this model, especially in extensions to the high-dimensional case (Diakonikolas et al. [2019], Kothari et al. [2018], Lai et al. [2016], Liu et al. [2019]). These works often keep the assumption of a Gaussian mixture component, though there has been expanding work with

non-Gaussian models as well.

2.3.4 Contamination Robust Bandits

Some of the first work in CSB started by assuming both rewards and contamination were bounded [Lykouris et al., 2018, Gupta et al., 2019]. These works assume an adversary that can contaminate at any time step, but that is constrained in the cumulative contamination. They bound the cumulative max (over actions) absolute difference of the contaminated reward, x, to the true reward, r, $\sum_t \max_a |r_a(t) - x_a(t)| \leq C$. Lykouris et al. [2018] provides a layered UCB-type active arm elimination algorithm. Gupta et al. [2019] expands on this work to provide an algorithm similar to active arm elimination in spirit, but which never completely eliminates an action, and which has better regret guarantees.

Recent work in implementing a robust UCB replaces the empirical mean with the empirical median, and gives guarantees for the uncontaminated regret with Gaussian rewards [Kapoor et al., 2018]. They consider an adaptive adversary but require the contamination to be bounded, though the bound need not be known. They cite work that can expand their robust UCB to distributions with bounded fourth moments by using the agnostic mean [Lai et al., 2016], though give no uncontaminated regret guarantees. In one dimension, the agnostic mean takes the mean of the smallest interval containing $(1 - \alpha)$ fraction of points. This estimator is also known as the α-shorth mean. Our work expands on this model by allowing for unbounded contamination and analysing the uncontaminated regret for sub-Gaussian rewards when implementing a UCB algorithm with the α-shorth mean.

CSB has also been analysed in the best arm identification problem [Altschuler et al., 2019]. Using a Bernoulli adversary that contaminates any reward with probability ε, Altschuler et al. [2019] consider three adversaries of increasing power, from the oblivious adversary, which does not know the player's history nor the current action or reward, to a malicious adversary, which can contaminate knowing the player's history and the current action and reward. They give analysis of the probability of best arm selection and sample complexity of an active arm elimination algorithm. While their performance measure is different than ours, we generalize their context to allow an adversary to contaminate in any fashion.

There is also work that explores the impact of an adaptive adversarial contamination on ε-greedy and UCB algorithms [Jun et al., 2018]. They give a thorough analysis with both theoretical guarantees and simulations of the effects an adversary can have on these two algorithms when the adversary does not know the optimal action but is otherwise fully adaptive. They show these standard algorithms are susceptible to contamination. Similar work looks at contamination in contextual bandits with a non-adaptive adversary [Ma et al., 2019].

2.4 Main Results

We present concentration bounds for both the α-shorth and α-trimmed mean estimators in the ε-contamination context for sub-Gaussian random variables.

Our contribution to the CSB problem is in providing a contamination robust UCB algorithm that is simple to implement and has theoretical regret guarantees close to those of UCB algorithms in the uncontaminated setting.

2.4.1 Contamination Robust Mean Estimators

The estimators we analyse have been in use for many decades as robust statistics. Our contribution is to analyze them within our ε-contamination model with sub-Gaussian samples and provide simple *finite-sample concentration inequalities* for ease of use in UCB-type algorithms.

2.4.1.1 Trimmed Mean

Our first estimator suggested for use in the contaminated model is the α-trimmed mean [Liu et al., 2019].

α-trimmed mean Trim the smallest and largest α-fraction of points from the sample and calculate the mean of the remaining points. This estimator uses $1 - 2\alpha$ fraction of sample points.

Algorithm 2: α-Trimmed Mean

input : $X_n = (x_1, ..., x_n)$, α
output: α-trimmed mean
$(x_{(1)}, ..., x_{(n)})$ = sorted X_n $s.t.$ $x_{(i)} \leq x_{(i+1)}$
cut = $\lceil \alpha * n \rceil$
return *mean*$(x_{(cut)}, ..., x_{(n\text{-}cut)})$

The intuition being if the contamination is large, then it will be removed from the sample. If it is small, it should have little affect on the mean estimate. Next we provide the concentration inequality for the α-trimmed mean. [Trimmed mean concentration] Let G be the set of points $x_1, ...x_n \in \mathbb{R}$ that are drawn from a σ-sub-Gaussian distribution with mean μ. Let S_n be a sample where an ε-fraction of these points are contaminated by an adversary. For $\varepsilon \leq \alpha < 1/2$, $t \geq n$ we have,

$$|\text{trMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{(1-2\alpha)}\left(\sqrt{\frac{4}{n}\log(t)} + 4\alpha\sqrt{6\log(t)}\right)$$

with probability at least $1 - \frac{4}{t^2}$.

Proof follows from Liu et al. [2019] and can be found in the appendix.

Let G_n be the set of points $x_i, ...x_n \in \mathbb{R}$ that are drawn from a σ-sub-Gaussian distribution. Without loss of generality assume $\mu = 0$. Let S_n be a sample where an ε-fraction of these points are contaminated by an adversary.

Let $\tilde{G} \subset G_n$ represent the points which are not contaminated and $C \subset G_n$ represent the contaminated points. Then our sample can be represented by the union $S_n = \tilde{G} \cup C$. Let R represent the points that remain after trimming α fraction of the largest and smallest points, and T be the set of points that were trimmed. Then we have,

$$
\begin{aligned}
&|\text{trMean}_\alpha(S_n)| = \left| \frac{1}{(1-2\alpha)n} \sum_{x \in R} x \right| \\
&\leq \frac{1}{(1-2\alpha)n} \Bigg(\Bigg| \underbrace{\sum_{x \in \tilde{G}} x}_{A_1} \Bigg| + \Bigg| \underbrace{\sum_{x \in \tilde{G} \cap T} x}_{A_2} \Bigg| + \Bigg| \underbrace{\sum_{x \in C \cap R} x}_{A_3} \Bigg| \Bigg)
\end{aligned}
$$

with

$$
\begin{aligned}
A_1 &\leq \left| \sum_{x \in G_n} x \right| + \left| \sum_{x \in G_n \setminus \tilde{G}} x \right| \\
&\leq n|\bar{x}_{G_n}| + \varepsilon n |\max_{i \in [n]} x_i| && \text{w.p. at least } 1 - \delta_1 - \delta_2, \\
A_2 &\leq 2\alpha n \max_{i \in [n]} |x_i| && \text{w.p. at least } 1 - \delta_2, \\
A_3 &\leq \varepsilon n \max_{i \in [n]} |x_i| && \text{w.p. at least } 1 - \delta_2.
\end{aligned}
$$

Combining we get,

$$
\begin{aligned}
&|\text{trMean}_\alpha(S_n) - \mu| \\
&\qquad \leq \frac{1}{(1-2\alpha)} \left(|\bar{x}_{G_n}| + \max_{i \in [n]} |x_i| (2\varepsilon + 2\alpha) \right) \\
&\qquad \leq \frac{1}{(1-2\alpha)} \left(|\bar{x}_{G_n}| + \max_{i \in [n]} |x_i| (4\alpha) \right) \\
&\qquad \leq \frac{\sigma}{(1-2\alpha)} \left(\sqrt{\frac{2}{n} \log \frac{2}{\delta_1}} + 4\alpha \sqrt{2 \log \frac{2t}{\delta_2}} \right)
\end{aligned}
$$

with probability at least $1 - \delta_1 - \delta_2$. Letting $\delta_1 = \frac{2}{t^2}$ and $\delta_2 = \frac{2}{t^2}$, and assuming $\alpha \geq \varepsilon$, we have,

$$|\text{trMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{(1-2\alpha)}\left(\sqrt{\frac{4}{n}\log(t)} + 4\alpha\sqrt{6\log(t)}\right)$$

with probability at least $1 - \frac{4}{t^2}$.

A more detailed proof can be found in the appendix.

2.4.1.2 Shorth Mean

The agnostic mean from Lai et al. [2016], which we use the more common term α-shorth mean for, can be considered a variation of the trimmed mean.

α-shorth mean Take the mean of the shortest interval that removes the smallest δ_1 and largest δ_2 fraction of points such that $\delta_1 + \delta_2 = \alpha$, where $\delta_1,\ \delta_2$ are chosen to minimize the interval length of remaining points. Uses $1 - \alpha$ fraction of sample points.

The α-shorth mean is less computationally efficient than the trimmed mean, but may be a better mean estimator when the contaminated points are not large outliers and are skewed in one direction. Intuitively this is because the α-shorth mean can trim off contamination that would require removing most of the sample with the trimmed mean. Next we provide the concentration inequality for the α-shorth mean.

Algorithm 3: α-Shorth Mean

input : $X_n = (x_1, ..., x_n)$, α
output: A mean estimate for the distribution of X
$(x_{(1)}, ..., x_{(n)})$ = sorted X_n *s.t.* $x_{(i)} \leq x_{(i+1)}$
$n_\alpha = \lfloor (1-\alpha) * n \rfloor$
$\mathcal{I} \in \text{argmin}_k\{x_{(k+n_\alpha)} - x_{(k)}\}$
Choose uniformly at random from set $\mathcal{I}$ if there is more than one starting index with the smallest interval length
return *sMean*$(X) \leftarrow$ *mean*$(x_{(\mathcal{I})}, ..., x_{(\mathcal{I}+n_\alpha)})$

[α-shorth mean concentration] Let G_n be the set of points $x_1, ...x_n \in \mathbb{R}$ that are drawn from a σ-sub-Gaussian distribution with mean μ. Let S_n be a sample where an ε-fraction of these points

are contaminated by an adversary. For $\varepsilon \leq \alpha < 1/3$, $t \geq n$, we have,

$$|\text{sMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{1-2\alpha}\sqrt{\frac{4}{n}\log t} + \frac{(6\alpha - 8\alpha^2)\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{6\log t}$$

with probability at least $1 - \frac{4}{t^2}$.

Proof sketch. Without loss of generality assume $\mu = 0$ for the underlying true distribution. Let $\tilde{G} \subset G_n$ represent the points which are not contaminated and $C \subset G_n$ represent the contaminated points. Then our sample can be represented by the union $S_n = \tilde{G} \cup C$

Let J be the interval that contains the shortest $1 - \alpha$ fraction of S_n, I be the interval that contains $\tilde{G}$ (i.e. the remaining good points after contamination), and T be the interval that contains the points of S_n after trimming the α largest and smallest fraction of points. Use $|I|$ to denote the length of interval I. It must be that $I \cap J \neq \emptyset$ because otherwise the points in $I \cup J$ would contain $2 - 2\alpha > 1$ fraction of S_n. Let c be a point in $I \cap J$ and x be a point in J. Recall that $\text{trMean}_\alpha(S_n)$ is the trimmed mean of the contaminated sample S_n. Then we have,

$$\begin{aligned}
|x| &\leq |x - c| + |c - \text{trMean}_\alpha(S_n)| + |\text{trMean}_\alpha(S_n)| \\
&\leq |J| + |I| + |\text{trMean}_\alpha(S_n)| \\
&\leq 2|I| + |\text{trMean}_\alpha(S_n)|
\end{aligned}$$

The second step comes from x and c both being in J and because $I \supseteq T$. The third step comes from $|J| \leq |I|$.

To bound the length of I we have,

$$|I| \leq 2 \max_{x \in G_n} |x| \text{ w.p. at least } 1 - \delta_2.$$

Finally, since

$$|\text{trMean}_\alpha(S_n)| \leq \frac{1}{(1-2\alpha)}(|\bar{x}_{G_n}| + 4\alpha \max_{x \in G_n} |x|)$$

with probability at least $1 - \delta_1 - \delta_2$, we get that for $x \in J$,

$$\begin{aligned}
|x| &\leq 4 \max_{i \in [n]} |x_i| + \frac{1}{(1-2\alpha)}(|\bar{x}_{G_n}| + 4\alpha \max_{x \in G_n} |x|) \\
&= \frac{|\bar{x}_{G_n}|}{1-2\alpha} + \left(4 + \frac{4\alpha}{1-2\alpha}\right) \max_{x \in G_n} |x|.
\end{aligned}$$

Now that we have a bound on the contaminated points in J, our analysis follows similarly as the

trimmed mean by bounding A_1, A_2, A_3 as defined below.

$$
\begin{aligned}
&|\text{sMean}_\alpha(S_n)| \\
&\leq \frac{1}{(1-\alpha)n}\left(\left|\underbrace{\sum_{x\in\tilde{G}} x}_{A_1}\right| + \left|\underbrace{\sum_{x\in\tilde{G}\cap\neg J} x}_{A_2}\right| + \left|\underbrace{\sum_{x\in C\cap J} x}_{A_3}\right|\right)
\end{aligned}
$$

□

The full proof is contained in the appendix and follows a similar approach as for the trimmed mean.

Our methods ensured that the first term in each concentration bound is the same, giving them similar regret guarantees when implemented in a UCB algorithm. We emphasize that the α-shorth mean uses $1-\alpha$ fraction of a sample while the α-trimmed mean uses $1-2\alpha$ fraction of a sample. We remark that if there is no contamination and $\alpha = 0$ then our inequalities reduce to the standard concentration inequality for the empirical mean of samples drawn from a sub-Gaussian distribution.

2.4.2 Contamination Robust UCB

We present the contamination robust-UCB (crUCB) algorithm for ε-CSB with sub-Gaussian rewards.

Algorithm 4: crUCB

input: number of actions K, time horizon T, upper bound on fraction contamination α, upper bound on sub-Gaussian constant σ_0, mean estimate function (α trimmed or shorth mean) f.

for $t \leq K$ **do**
  Pick action a when $t = a$.
end
for $t > K$ **do**
  for $a \in [K]$ *compute* **do**
    $f(\mathbf{x}_a(t)) \leftarrow$ mean estimate of rewards.
    $N_a(t) \leftarrow$ number of times action has been played.
  end
  Pick action $A_t = \text{argmax}_{a\in[K]} f(\mathbf{x}_a(t)) + \frac{\sigma_0}{(1-2\alpha)}\left(\sqrt{4\frac{\log(t)}{N_a(t)}}\right)$.
  Observe reward $x_{A_t}(t)$.
end

We provide uncontaminated regret guarantees for crUCB below for both the α-trimmed and the α-shorth mean.

[α-trimmed mean crUCB uncontaminated regret] Let $K > 1$ and $T \geq K - 1$. Then with algorithm 4 with the α-trimmed mean, σ-sub-Gaussian reward distributions with $\sigma_a \leq \sigma_0$, and contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{min}}{4(\Delta_{min}+4\sigma_0\sqrt{6\log T})}$, we have the uncontaminated regret bound,

$$\bar{R}(UCB) \leq 8\sigma_0\sqrt{KT\log T} + \sum 15\Delta_a.$$

[α-trimmed mean crUCB uncontaminated regret bounded rewards] If the rewards are bounded by b, and have contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{\min}}{4(\Delta_{\min}+4b)}$, then

$$\bar{R}_T \leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a.$$

[α-shorth mean crUCB uncontaminated regret] Let $K > 1$ and $T \geq K-1$. Then with algorithm 4 with the α-shorth mean, sub-Gaussian reward distributions with $\sigma_a \leq \sigma_0$, and contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{min}}{4(\Delta_{min}+9\sigma_0\sqrt{6\log T})}$, we have the uncontaminated regret bound,

$$\bar{R}(UCB) \leq 8\sigma_0\sqrt{KT\log T} + \sum 15\Delta_a.$$

[α-shorth mean crUCB uncontaminated regret bounded rewards] If the rewards are bounded by b, and have contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{\min}}{4(\Delta_{\min}+9b)}$, then

$$\bar{R}_T \leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a.$$

Proofs for section 2.4.2 and 2.4.2 and their corollaries follow standard analysis and are provided in the appendix.

From section 2.4.2 and 2.4.2 we get that crUCB has the same order of regret in the CSB setting as UCB1 has in the standard sMAB setting. The constraint on the magnitude of ε is quite strong, but we show in Section 2.5 that they can be broken and still obtain good empirical performance.

Remark Our bounds above do not allow ε to be too big relative to the minimum suboptimality gap $\Delta_{\min}$. This is natural: if $\varepsilon > \Delta_{\min}$ then no algorithm can get sublinear regret since distinguishing between the top two actions is statistically impossible even with infinite samples. We

give a simple example in the Appendix. Furthermore, it is possible to derive a regret bound[1] of $\tilde{O}(\sigma_0\sqrt{KT} + \frac{\alpha\sigma_0}{1-4\alpha}T)$ for any choice of α such that $\varepsilon \leq \alpha < 1/4$. The linear term in regret (which is unavoidable for large ε) may be acceptable if the corruption proportion is not very large.

2.5 Simulations

We compare our crUCB algorithms using the trimmed mean (tUCB) and shorth mean (sUCB) against a standard stochastic algorithm (UCB1, Auer and Cesa-Bianchi [2002]), a standard adversarial algorithm (EXP3, Auer et al. [2002]), two "best of both worlds" algorithms (EXP3++, Seldin and Lugosi [2017], 0.5-TsallisInf, Zimmert and Seldin [2019]), and another contamination robust algorithm (RUCB-MAB, Kapoor et al. [2018]). Each trial has five actions ($K = 5$), is run for 1000 iterations ($T = 1000$), for $\varepsilon \in \{0.05, 0.1\}$. For sUCB and tUCB, we set $\alpha = \varepsilon$ and $\sigma_0 = \sigma$. The plots are average results over 10 trials with error bars showing the standard deviation.

Our choice of T comes from our motivation to apply contaminated bandits in education, where the sample sizes are often much smaller than for example in advertising. While $T = 1000$ would be considered a large university class, it still allows one to visually see regret for smaller iterations and see how performance stabilizes. We similarly chose number K of arms and proportion contamination ε to be in a realistic range for the application we have in mind. All algorithms use recommended parameter settings given within their respective papers.

Rewards and gaps We chose the reward distribution to be binomial(n=10) to simulate likert scale and because this distribution has bounded rewards and is not symmetric for large p. For the optimal action, $p = .9$ and for suboptimal actions $p = .8$, thus the suboptimality gap is $\Delta = 1$. All non-optimal actions have the same true distribution.

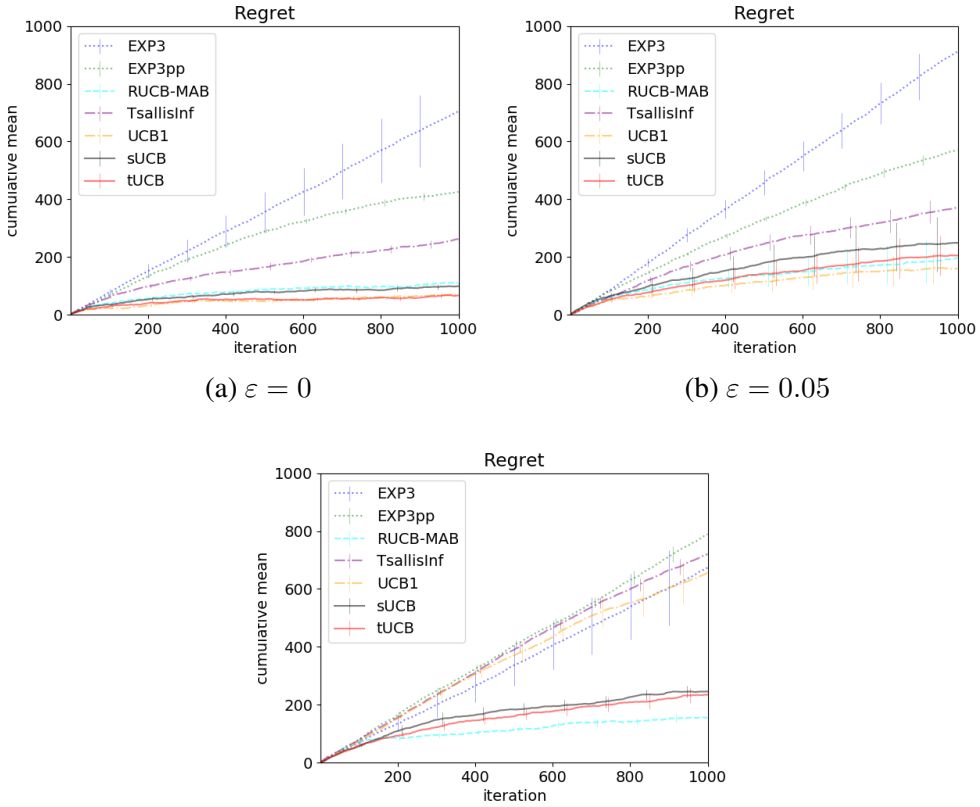
Adversaries We focus on a Bernoulli adversary which gives a contaminated reward at every time step with probability ε. We also implement a cluster adversary which contaminates at the beginning of play to show the weakness of algorithms to this type of attack.

Contamination We use a random malicious contamination scheme which chooses a contaminated reward uniformly from ranges that increase suboptimal action means and decrease the optimal action's mean.

Performance measurement We plot the average regret over 10 trials for 1000 iterations.

We recommend to view the plots on a color screen.

[1]The $\tilde{O}(\cdot)$ notation hides constants and logarithmic terms. See Appendix for details.



(a) $\varepsilon = 0$ (b) $\varepsilon = 0.05$

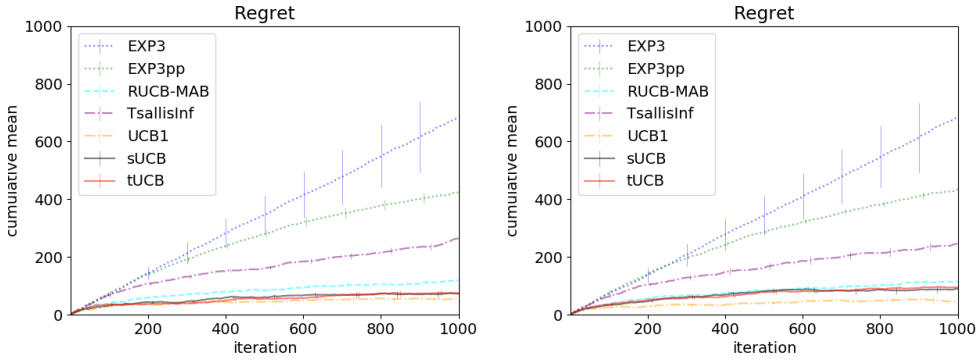
(c) $\varepsilon = 0.1$

Figure 2.1: Binomial Rewards With Varying Proportion Of Contamination

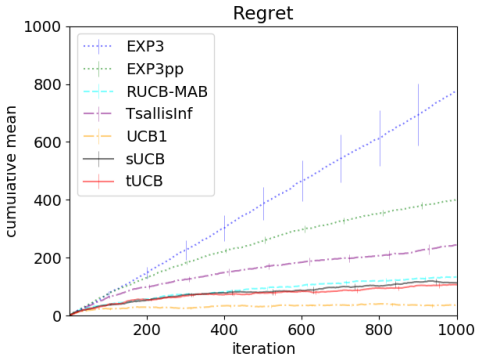
In Figure 2.1a we see that the adversarial and best of both worlds algorithms, EXP3, EXP3++, and TsallisInf, perform poorly in the purely stochastic setting compared to the UCB type algorithms. In Figure 2.1, we see the best of these, TsallisInf, starts to degrade as the proportion of contamination increases while the robust UCB algorithms are only slightly affected. These simulations show a clear performance benefit to using algorithms that specifically account for contaminated rewards.

Figure 2.3 and Figure 2.4 shows that for both sUCB and tUCB, the choice of α is much less sensitive than choice of σ. Over estimating or slightly underestimating α does not degrade performance significantly. Underestimating σ can give a significant boost to performance while over estimating can degrade it. This is consistent with the performance of UCB algorithms in practice, which often scale the exploration term to improve empirical performance [Liu et al., 2014].

To look at the impact of using a contamination robust algorithm when there is no contamination, we plotted various α values when $\varepsilon = 0$, shown in Figure 2.2. Assuming small amounts of contamination when there is none only has a small impact on performance, suggesting it is permissible



(a) $\alpha = 0$ (b) $\alpha = 0.05$



(c) $\alpha = 0.1$

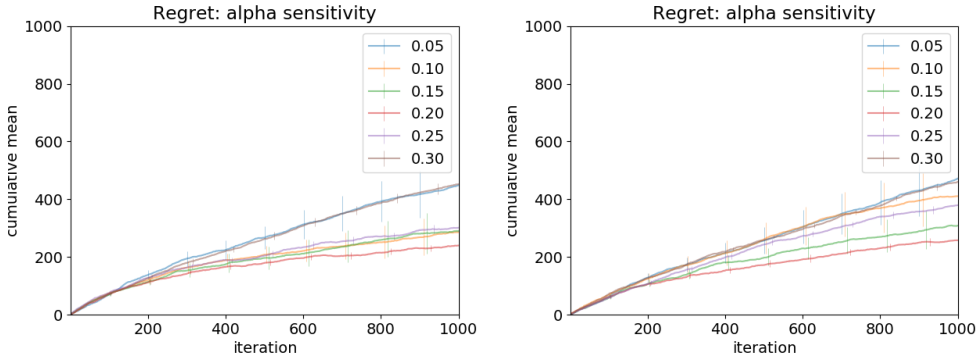
Figure 2.2: Misspecified α For $\varepsilon = 0$.

to use contamination robust methods when there is uncertainty of contamination. Similarly, small K and large Δ can render bounded contamination impotent and would not require algorithms that account for it.

We have included RUCB-MAB in our simulations because it is simple to implement and can perform similarly well to our algorithms. We note it currently has guarantees only for Gaussian rewards [Kapoor et al., 2018].

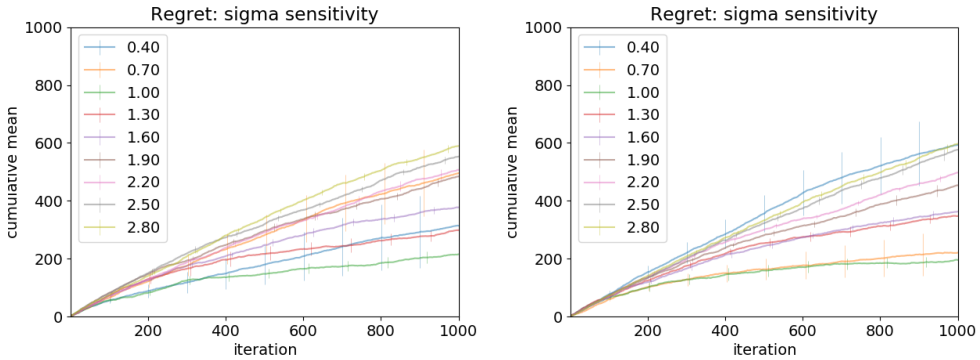
Figure 2.5 shows the poor performance of all algorithms when the first ε rewards are contaminated. TsallisInf and EXP3++ show some recovery, but it is clear this type of adversary is harmful. This remains an open problem for scenarios with small T.

We also considered including the BARBAR algorithm [Gupta et al., 2019] whose epoch scheme is the only algorithm we know that accounts for the front cluster attack. We chose against this as for our setting of $T = 1000$ the BARBAR algorithm only has one epoch, and thus does not make any updates to the estimated gaps, resulting in pure random exploration.



(a) sUCB (b) tUCB

Figure 2.3: Regret Sensitivity For Various α.



(a) sUCB (b) tUCB

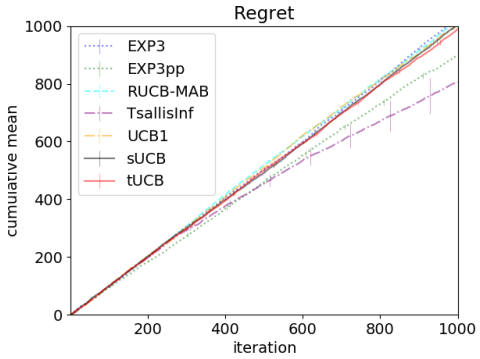
Figure 2.4: Regret Sensitivity For Various σ.

2.6 Discussion

We have presented two variants of an ε-contamination robust UCB algorithm to handle uninformative or malicious rewards in the stochastic bandit setting. As the main contribution, we proved concentration inequalities for the α-trimmed and α-shorth mean in the ε-contamination setting with sub-Gaussian samples and guarantees on the uncontaminated regret of the crUCB algorithms. The regret guarantees are similar to those in the uncontaminated sMAB setting.

We have shown through simulation that these algorithms can outperform "best of both worlds" algorithms and those for stochastic or adversarial environments when using a small number of iterations and ε chosen to be reasonable when implementing bandits in education.

We highlight that our algorithms are simple to implement. In practice, it is often easy to find upper bounds on the parameters which are robust to underestimation. Our algorithms are numerically



(a) $\varepsilon = .1$

Figure 2.5: Front Cluster Attack

stable and have clear intuition to their actions.

A weak point of these algorithms is they require knowledge of α before hand. Choices of α may come from domain knowledge, but could also require a separate study.

In this work we assumed a fully adaptive adversarial contamination, constrained only by the total fraction of contamination at any time step. By making more assumptions about the adversary, it is likely possible to improve uncontaminated regret bounds.

Limitations The adversary used in the simulation is quite simple and does not take full advantage of the power we allow in our model. We designed it as a first test of our algorithms and associated theory. In the future, we would like to design simulated adversaries that are modeled on real world contamination. It will also be important to deploy contamination robust algorithms in the real world. This will require thought on how to select various tuning parameters ahead of the deployment.

There remain many open questions in this area. In particular, we think this work could be improved along the following directions.

Randomized algorithms UCB-type algorithms are often outperformed in applications by the randomized Thompson sampling algorithm. Creating a randomized algorithm that accounts for the contamination model would increase the practicality of this line of work.

Contamination correlated with true rewards One possibility is that the contaminated rewards contain information of the true rewards. For example if contamination can be missing data, we know dropout can be correlated with the treatment condition.

CHAPTER 3

Achieving Representative Data via Convex Hull Feasibility Sampling Algorithms

3.1 Introduction

Implementing algorithmic fairness in practice is a difficult task because most data science pipelines consists of many steps (e.g. data collection, data cleaning, training and post-processing), and any of these steps can affect the fairness of the outcome. Thus implementing algorithmic fairness in practice is generally non-trivial. Representation bias is a known issue when training ML models [Hashimoto et al., 2018, Rolf et al., 2021]. This bias represent a lack of or minimal data from a subgroup of the desired population that can negatively impact the algorithmic outcomes. Unlike historical bias which is inherent in the data [Julia Angwin, 2016], representation bias can be alleviated through intentional data collection. When queried about ways individuals have attempted to address fairness, many cited more data collection as a first approach [Holstein et al., 2019]. While this is possible in settings where group membership and (when applicable) outcome labels are known and can be directly sampled, there are circumstances where data is collected from sources with unknown distributions of protected attributes.

An example of this is given in Holstein et al. [2019]. Here they describe a company that wishes to automate essay scoring whose current iteration has unfair outcomes for a minority group. Their algorithm is scoring these minority students on average more unfavorably than the majority group based on scores by a human specialist. They desire more high scoring essays from minority students to improve their scoring accuracy within that group. Because they do not know the distribution of these students at the schools they are collecting essays from, they do not have an efficient strategy to collect those needed samples, or know if it is possible to collect a data set with their desired distribution.

An approach to this problem is to have a sampling policy to determine if there exists a distribution across schools that would produce a data set with the desired proportion of high scoring essay from the minority group. The goal of this iterative sampling policy would be to make this

determination using a minimum number of samples. Once this feasibility is known, one can either sample accordingly or seek out other sources.

There are a myriad of strategies now published that are methods to improve fairness at the post-data collection stages [Dwork et al., 2012, Friedler et al., 2019]. These training strategies and post-processing strategies will improve fairness outcomes, but there is a limit to improvement before impacting accuracy. It is always preferred in any machine learning application to start with the best data one can access. This highlights another benefit collecting fair data over post-collection strategies. If fairness is truly a concern, it must also be recognized that data collected today will be used for a different purpose tomorrow. By considering how to curate fair data in isolation, this can impact fairness outcomes regardless of the way data is used apart from its original purpose.

For example, consider that, in general, different definitions of fairness cannot be simultaneously satisfied except for in certain (possibly unattainable) scenarios [Kleinberg et al., 2017, Pleiss et al., 2017]. Collecting data to achieve one type of fairness when trained with a particular algorithm gives no guarantee for outcomes of other measurements of fairness. If the measurement for fairness changes over the life of a project, the data is no longer optimal. Aiming for fair representation from the onset will mitigate some of these problems. Additionally, equal representation is one of the scenarios that can produce fair outcomes in relation to calibration and equalized odds, something that lopsided data cannot achieve.

This work aims to provide a sampling method that tests whether a curator can create a fair data set from available sources, where ”fair” is defined in terms of a predefined proportion of group memberships. To the best of our knowledge, similar work in this area of fair sampling assumes a fair data set is achievable. This work focuses on testing that assumption. Considering the cost of collecting data, the goal will be to determine the feasibility of these sources with a minimum number of samples. When collecting data, if one can sample any protected attribute any number of times, it is simple to create training data that is consistent with some notion of fairness, such as equal proportions of protected attributes. In this chapter, we consider the scenario where the sampling sources have unknown distributions of attributes and the curator has defined a ”balanced” set in regards to the desired proportions of the training data. That is, data can be sampled from different sources (such as polling in different cities) but knowledge of the distributions of data from those sources is unknown. This problem setting is described in full in Section 3.2.

Aside from collecting fair data for training, this method could also be used when fair sampling is the desired end outcome. For example, advertising community services with a desired outcome of equal men and women using those services. Different advertising strategies would reach different populations. A practitioner would want to know as quickly as possible whether their selected strategies can achieve their desired distribution, and if so what combination of strategies would do this.

Contributions We introduce the convex hull feasibility sampling problem. In the Bernoulli setting, we give a lower bound on the expected sample size in the infeasible case and an oracle lower bound of the expected sample size in the feasible case. We define the direction of greatest uncertainty and present three policies that use this direction, along with a naive Uniform policy. Using high-probability upper bounds, we prove that one policy, Lower Upper Confidence Bound (LUCB) Mean is superior to Uniform. We define the Multinomial version of the problem along with adjusted algorithms, and using simulations show the performance of our three policies outperform Uniform in the Bernoulli setting and the Multinomial setting with three dimensions.

3.1.1 Related Work

3.1.1.1 Fair Sampling

To the best of our knowledge, the first work to address data collection as a part of bias mitigation is Abernethy et al. [2020]. Here the goal is to optimize over both a loss function for accuracy and a loss function for fairness. They assume an infinite availability of group labeled data, and at every iteration of sampling they choose the sample which will either minimize the accuracy loss or minimize the fairness loss. The choice of which loss to minimize at every time point is determined by a Bernoulli variable with probability p, where p is a parameter chosen beforehand. When a sample is chosen to increase fairness, a sample is drawn from the group which currently has the worst loss performance. Otherwise a sample is chosen randomly. The intuition in both cases is that more training samples will improve performance, either overall performance when sampling at random or a specific group's performance when sampling to improve fairness. A similar framework is presented in Tae and Whang [2021], where the groupings are predefined slices of a current data set, and the goal is to obtain additional samples within a budget so as to maximize average accuracy as well as minimize the average difference between the accuracy of each slice and that of the total data. Their sampling method relies on estimating learning curves and allocating the sampling budget to slices that will have maximum impact on accuracy and fairness.

Along this vein of work is Shekhar et al. [2021]. Their goal identify a minimax optimal classifier across the sampling proportion of protected attributes and the loss of the worst performing group. Given a function class $\mathcal{F}$, loss l, and protected attributes $z \in \mathcal{Z}$, they propose an adaptive sampling policy that identifies the worst performing group z and dedicates a larger proportion of the sampling budget to that group.

We include an exploration of fairness results when optimizing samples for a particular definition in Appendix C. Using two data sets often seen the algorithmic fairness literature, we sample and train using the policies presented in Abernethy et al. [2020], Shekhar et al. [2021], and discuss the outcomes of several fairness measurements.

In Asudeh et al. [2019], they forgo optimization for a particular learning algorithm and focus on the coverage of features within the data. They define the set maximum uncovered patterns (MUP), which aims to identifying feature combinations that fail to meet predefined threshold counts. In addition to providing several algorithms to identify the set of MUP, they provide a greedy algorithm to sample additions data whose feature patterns are MUP until all meet the required sampling threshold.

The work closest to ours is presented in Nargesian et al. [2021], where the goal is to collect a data set of a given size consisting of a desired count from each defined group. Here they assume a priori that the desired counts are feasible, and if minimums are not achieved they propose over-sampling until minority group counts are met and removing excess majority samples. In addition to results for when the sampling distributions are known, they tackle the unknown distribution model with a multi-armed bandit strategy. They propose a reward function that depends on the true distribution of a group (such as from census population data), with the intuition being if a sample is from a group with a high proportion in the population then the reward is low and if from a minority group the reward should be high. Using a UCB type strategy with this reward function presents a sampling strategy that aims to sample from the distribution with the largest proportion of the minority group. Our work differs substantially by focusing on the feasibility of the desired proportions, and frames the problem through use of a convex hull composed of points defined by a confidence region.

There are several other frameworks around obtaining a fair data set. For example, an active learning application is presented in Anahideh et al. [2021], where the goal is to sequentially select which points to label so as to balance model accuracy along with a predetermined notion of fairness. Data augmentation with synthetic points has also been explored Sharma et al. [2020].

3.1.1.2 Bandit Pure Exploration

The feasibility problem is closely related to the pure exploration multi-armed bandit problem. In pure exploration a learner has k actions with unknown means and the goal is to identify the action or subset of actions with the largest mean from the fewest samples. There are two settings in this problem, fixed-confidence and fixed-budget. In the fixed-confidence setting, a policy aims to minimize the sample complexity while guaranteeing the outcome of a policy is correct with some minimum predetermined probability. In the fixed-budget setting, a policy, given a predetermined sample size, aims to provide the largest confidence with which the largest means are correctly identified.

To see the connection to our feasibility problem to the fixed-confidence setting, consider the two class case, which reduces to identifying if there exists $p_i \leq x \leq p_j$. Here $p_1, \ldots, p_k$ are the k unknown means and the desired mean x encodes our definition of a balanced data set. Then

by determining if x is or isn't the maximum or minimum mean with some probability $1-\delta$ we determine whether we correctly identify feasibility with probability $1-\delta$.

The PAC pure-exploration setting was first presented in Even-dar et al. [2002] for identifying the top action with a fixed confidence. Their successive elimination algorithm relies on uniformly sampling actions from a decreasing set, removing actions from the set as they are determined to be lower than the top action with high confidence. Another set of policies uses lower upper confidence bounds on the means of the actions [Gabillon et al., 2012, Kalyanakrishnan et al., 2012, Kaufmann and Kalyanakrishnan, 2013, Jamieson et al., 2014]. A lower bound on the expected sample complexity for Bernoulli rewards is presented in Mannor and Tsitsiklis [2004], where they provide worst case and gap dependent bounds. This is expanded upon by Garivier and Kaufmann [2016], who provide a lower bound on sample complexity for one parameter exponential families and a policy with a asymptotically matching upper bound.

Relation to Sequential Hypothesis Testing: The multi-armed bandit pure exploration problem (also known as best arm identification) and the convex hull feasibility sampling problem we present in this chapter are a both sequential hypothesis testing problems. They rely on sequential sampling until terminated by a predefined stopping rule. When in a fixed-confidence setting, the sequential nature can allow for reaching a conclusion using smaller sample sizes than offline, batch hypothesis testing. In the fixed-budget setting, sequential sampling can result in higher confidence conclusions. The relation of classical sequential testing theory presented in Wald [1945] to best arm identification sample complexity bounds is discussed in Kaufmann and Kalyanakrishnan [2013].

3.1.1.3 Probabilistic Hyperplane Separability

The fields of computational geometry and computer science are not new to the problems of convex hull feasibility and hyperplane separability with probabilistic points. Though the underlying data assumptions are not quite matched to the convex hull feasibility problem we present in this chapter, there are significant similarities that may ultimately be used in future research and we would be remiss not to point them out. The goal of these papers is typically to provide an algorithm identifying separability or the probability of separability that minimizes run time complexity.

We briefly characterize three variations of these problems that are similar to ours. The first is that which considers the probability of linear separability between two sets of points A and B which are drawn from sets $\mathcal{A}$ and $\mathcal{B}$, as in Fink et al. [2017]. The second variation considers n labeled points from sets A and B, each with a known uncertainty region. The question then is to determine separability of sets of uncertainty regions, as seen in Sheikhi et al. [2017]. Finally, there is the problem formulation where there are n points, with the value of each point i having a probability distribution over a discrete set s_i with the goal to find the probability a set O lies within

the probabilistic convex hull [Yan et al., 2015].

3.2 General Problem Definition

The fixed-confidence ϵ-convex hull feasibility problem is defined as follows. Each of k source distributions, which we will hereto refer to as *actions*, are independently belong to some known family $\mathcal{P}$ with unknown means μ_i in dimension d. We are given a known variable $x \in \mathbb{R}^d$ and a relaxation of $\epsilon \geq 0$ and define x_ϵ as the open set $\{y : ||y - x|| < \epsilon\}$, with $x_\epsilon = x$ when $\epsilon = 0$. We define the *feasible* case as when there exists some $y \in x_\epsilon$ that lies in the convex hull of $\{\mu_1, ..., \mu_k\}$ and the *infeasible* case as when the set x_ϵ lies outside of the convex hull of $\{\mu_1, ..., \mu_k\}$. We include the relaxation of x with ϵ because it may not be necessary to achieve exact feasibility.

If the μ_i's are known, then it is possible to determine whether x is in the convex hull of the μ_i's by solving a linear optimization problem. Instead, we consider the setting in which the μ_i's are unknown, but we can (actively) observe noisy versions of the μ_i's. The goal is to give a determination of the feasibility of x_ϵ with a predetermined confidence while minimizing the number of times the actions are sampled.

In the fairness setting, the dimension d represents the number of groups defined by the protected attribute labels that the curator wishes to balance on. For example $d = 2$ could represent the groupings of 'men' and 'women'. The points μ_i's correspond to data sources: the j-th component of μ_i is the fraction of samples from the j-th group in samples from the i-th data source. The components of the query point x correspond to the desired fractions of samples from each group in the data set. The convex hull feasibility problem is thus equivalent to determining whether there is a set of weights w_i such that drawing w_i fraction of samples from the i-th data source will lead to a data set with the desired fractions of samples from each group.

3.2.1 Feasibility and Infeasbility

Given $i \in [k]$, $\mu_i \in \mathbb{R}^d$, $x \in \mathbb{R}^d$ and $\epsilon \geq 0$, we first state the feasible and infeasible cases more formally.

Definition 3.2.1 (Infeasible Case)**.** The problem is (x, ϵ)-infeasible if there exists some separating hyperplane between x_ϵ and the μ_i.

$$\exists a \in \mathbb{R}^d \text{ such that } \forall i \in [k], y \in x_\epsilon \ (\mu_i - y)^T a < 0.$$

Definition 3.2.2 (Feasible Case)**.** The problem is (x, ϵ)-feasible if there exists a convex combina-

tion that expresses some $y \in x_\epsilon$ in terms of the μ_i's:

$$\exists \lambda \in \Delta^{k-1} \text{such that } y = \sum_{i=1}^{k} \lambda_i \mu_i.$$

where Δ^{k-1} is the $(k-1)$-dimensional probability simplex in $\mathbb{R}^k$.

Because the μ_i are unknown, we must rely on confidence regions to inform a decision of whether the underlying case is *feasible* or *infeasible*. If each confidence region R_i contains μ_i with probability at least $1 - \frac{\delta}{k}$ then we can make a high-confidence decision on the underlying case.

Definition 3.2.3 ($1-\delta$ Confident Infeasible)**.** There exists a separating hyperplane between the set x_ϵ and the confidence regions for all actions.

$$\exists a \in \mathbb{R}^d \text{ such that } \forall i \in [k],\ y \in x_\epsilon,\ v_i \in R_i, \text{ we have that } (v_i - y)^T a < 0.$$

Definition 3.2.4 ($1-\delta$ Confident Feasible)**.** For all sets consisting of a point from each confidence region, there exists a point in x_ϵ within their convex hull.

$$\forall v_i \in R_i, i \in [k],\ \exists \lambda \in \Delta^{k-1},\ y \in x_\epsilon \text{ such that } y = \sum_{i=1}^{k} \lambda_i v_i.$$

3.2.2 Formalization Assumptions and Practical Implementation

This formalization allows us to approach the problem under reasonable assumptions but does not capture the many variations expected in a practical implementation. We highlight some of the assumptions and variations as directions of future work as well as to make them readily apparent to practitioners.

This work assumes each action is of sufficient size that the sampling policy will not sample the entire population. Additionally, we are assuming that the distribution of features uncorrelated to those defining the labels used for balancing the data are similar across actions, so that the sampling policy is not biasing these other features. Finally, we assume that the cost of each action is equal.

An example that holds each of these assumptions could be gathering survey data balanced on political affiliation where each action represents a large city. It is unknown beforehand the distribution of people likely to respond to the survey, and the reward incentive (cost) for each sample would be the same across actions. Depending on the measured features, it can be reasonable to assume a sampling policy would not induce bias.

Conversely, an example that does not hold all these assumptions would be the example given in the introduction: advertising for community services. Different advertising strategies are likely to have different costs, and the population reached by certain strategies may be small and stagnant enough so that additional advertising provides no benefit, making additional sampling moot.

We consider the assumptions in this work reasonable in an initial formulation, and highlight that they are fertile ground for further work in consideration of the convex hull feasibility problem.

3.2.3 Sampling Policy

A *sampling policy* π is a mapping of the history of all samples drawn up to the current time to the choice of which action to sample next and the termination of the algorithm. When a policy terminates, it outputs a result of either *feasible* or *infeasible*. Let τ represent the stopping time of a policy, and $I(\pi, \delta) \in \{$*feasible*, *infeasible*$\}$ be the indicator function of the output for policy π given confidence $1 - \delta$.

Definition 3.2.5 (Sound Policy)**.** Given some δ, We call a policy $(1 - \delta)$-sound if the expected value of the stopping time is finite and if with probability at least $1 - \delta$ the policy selects the correct underlying case,

$$E[\tau] < \infty$$

$$P(I(\pi, \delta) = feasible | feasible) \geq 1 - \delta, \quad P(I(\pi, \delta) = infeasible | infeasible) \geq 1 - \delta$$

3.3 Bernoulli Feasibility Sampling

We focus on the case where there are two protected categories ($d = 2$). In this case the μ_i lie in the 2-dimensional simplex and convex hull feasibility simplifies into testing in 1-dimension with Bernoulli means. This setting maps onto the scenario with two groups labels, $\{0, 1\}$, with $x \in [0, 1]$ representing the desired proportion of samples from group 1 and the probability of sampling group 1 from action i is p_i. For our theoretical analysis, we assume without loss of generality that $p_1 \geq p_2 \geq ... \geq p_k$.

3.3.1 Sample Complexity Lower Bounds

We will take inspiration from the pure exploration bandit literature and give a lower bound on the expected value of the stopping time τ as a measure of sample complexity in the Bernoulli setting.

The multi-armed bandit best arm identification problem and the Bernoulli convex hull feasibility problem share certain similarities pointing towards similar techniques, but significant differences prevent direct application. In the best arm identification problem, to determine the best action with

high confidence, all sub-optimal actions must be sampled to some extent to rule them sub-optimal. This remains true in our problem when the problem instance is infeasible, as all actions must be sampled sufficiently to determine them separable from our set of interest x_ϵ. If the instance is feasible, the relation of the "sub-optimal" actions to each other or x_ϵ becomes irrelevant. For example, if two actions are sampled such they are determined with high confidence to be above and below x_ϵ respectively, sampling from the other actions provides no additional information about the feasibility or infeasibility of the problem. Additionally, there may be multiple sets of actions whose convex hull is feasible.

The possibility of multiple optimal subsets of actions presents a difficulty in determining a lower bound for feasible instances since for any $(1-\delta)$-sound policy, it may not have sampled all actions and there may be multiple sets of actions that would trigger termination with the correct outcome. Therefore, for a specific feasible instance, it becomes difficult to give an expected lower bound for each action, except for the case when the playable actions comprise a unique feasible set.

Considering this, we give a looser oracle lower bound for the feasible case. Here, the oracle knows the optimal subset(s) of actions but does not know their means. The oracle lower bound then is the minimum expected sample complexity when only actions in an optimal subset are played. Note that the oracle lower bound is still a valid lower bound since we are only giving the learner more information about the problem. However, the true lower bound might be much higher than our oracle lower bound.

Notation: For any feasible problem instance, let

$$\Omega = \{\mathcal{J} \subseteq [k] | \{p_i\}_{i\in\mathcal{J}} \text{ is } (x,\epsilon) \text{ feasible}\}$$

be the set of all subsets of actions whose means are (x,ϵ)-feasible. Then we define the optimal subset of actions, $\mathcal{J}^*$, as the subset(s) that is the 'easiest' to distinguish from any infeasible instance. That is, we want to define the set of actions who collectively require the fewest samples to determine with high confidence is not any infeasible instance. In the Bernoulli setting we define this as

$$\mathcal{J}^* = \underset{\mathcal{J}\in\Omega}{\operatorname{argmin}} \begin{cases} \left(\min_{u\in\{-1,1\}} D(\mu_i, x+u\epsilon)\right)^{-1} & |\mathcal{J}| = 1 \\ \sum_{i\in\mathcal{J}} \left(\max_{u\in\{-1,1\}} D(\mu_i, x+u\epsilon)\right)^{-1} & otherwise \end{cases}$$

where D is the Kullback–Leibler divergence. Here we want to maximize the distance between action means and a boundary point of x_ϵ while minimizing the cumulative distance across all actions in the set. There are two feasible cases, either only one action is feasible or two actions are a feasible set, so $|\mathcal{J}^*| \in \{1,2\}$. When analysis differs for these cases and $|\mathcal{J}^*| = 1$ then we write

$\mathcal{J}^* = \{l^*\}$, else we write $\mathcal{J}^* = \{1, k\}$, as in this case the optimal subset consists of the actions with the largest and smallest mean, p_1 and p_k.

Theorem 1 (Oracle Feasible case)**.** *For a problem instance ν that is (x, ϵ)-feasible, for any $(1-\delta)$-sound deterministic policy with $d = 2$, $\delta < 1/2$,*

$$E_\nu[\tau] \geq \begin{cases} \max\left\{D(p_{l^*}|x-\epsilon)^{-1}, D(p_{l^*}|x+\epsilon)^{-1}\right\} \frac{1}{2}\log\left(\frac{1}{4\delta}\right) & \mathcal{J}^* = \{l^*\} \\ \left[\frac{1}{D(p_1|x-\epsilon)} + \frac{1}{D(p_k|x+\epsilon)}\right] \frac{1}{2}\log\left(\frac{1}{4\delta}\right) & \mathcal{J}^* = \{1, k\}, \end{cases}$$

Theorem 2 (Infeasible case)**.** *For a problem instance ν' that is (x, ϵ)-infeasible, for any $(1-\delta)$-sound deterministic policy with $d = 2$, $\delta < 1/2$,*

$$E_{\nu'}[\tau] \geq \sum_{i=1}^{k} \max\left\{D(p_i|x-\epsilon)^{-1}, D(p_i|x+\epsilon)^{-1}\right\} \frac{1}{2}\log(\frac{1}{4\delta})$$

Lower bound proofs can be found in appendix A.1.1.

3.3.2 Sampling Policies

We present four sampling policies, a naive Uniform policy as a baseline along with Lower Upper Confidence Bound (LUCB) Mean, LUCB Ratio and Beta Thompson Sampling. We give high probability upper bounds for Uniform and LUCB Mean, and empirical evidence that LUCB Mean, LUCB Ratio, and Beta TS significantly outperform Uniform.

Notation: Let $B(n, \delta)$ be a confidence margin dependent upon sample size n and confidence parameter δ such that

$$\sum_{n=1}^{\infty} P\left(|p_i - \hat{p}_i(n)| > B\left(n, \delta\right)\right) < \frac{\delta}{k}. \tag{3.3.1}$$

We write $B_i(t)$ to represent the confidence margin for action i given its sample size at time t when δ is implied. Let $\hat{p}_i(t)$ be the estimated mean of action i at time t, and $R_i(t) = \{y : \hat{p}_i(t) - B_i(t) \leq y \leq \hat{p}_i(t) + B_i(t)\}$ be the confidence region of action i at time t. We use a_t to specify the action chosen at time t and $n_i(t)$ the number of times action i has been chosen at time t. Each policy follows the same stopping rules for termination.

We next define the direction of greatest uncertainty, which is used to determine termination and in our sampling policies for action selection. This measure aims to capture which direction away from x, we are least certain an action mean lies on.

Definition 3.3.1 (Direction of greatest uncertainty)**.** Given a confidence margin B_i and mean esti-

mate $\hat{p}_i$, the direction of greatest uncertainty $u \in \{1, -1\}$ is defined as,

$$u = \underset{u \in \{1,-1\}}{\operatorname{argmin}} \max_{i \in [k]} u(\hat{p}_i - x) - B_i.$$

The intuition behind this definition is that it identifies the direction from x we are furthest from determining a mean exists in that direction. For example, if $x = .5$, and there are two confidence regions $(.48, .9)$ and $(.49, .8)$, then the closest lower bound in direction $u = -1$ is .8, and the closest lower bound in direction $u = 1$ is .48. The decision boundary that implies a mean lies below x is further from x than a decision boundary that implies a mean lies above it, so our direction of greatest uncertainty is $u = -1$ and we should sample actions that we have a higher belief are below x.

All the policies presented follow the same stopping rules.

Stopping Rules: If one of the following criteria are met, the policy terminates,

1. **Feasible:** x_ϵ is not separable from any subset consisting of a point from each of the confidence regions.
   $\min_{u \in \{-1,1\}} max_{i \in [k]} (\hat{p}_i - x)u - B_i(t) > -\epsilon$

2. **Infeasible:** x_ϵ is separable from all confidence regions.
   $\min_{u \in \{-1,1\}} max_{i \in [k]} (\hat{p}_i - x)u + B_i(t) < -\epsilon$

Where stopping rule 1 states there is a mean whose confidence interval lies above $x - \epsilon$ and one whose confidence intervals lies below $x + \epsilon$. The same confidence interval may satisfy both of these conditions. Intuitively, stopping rule 1 says that if the true means lie in their respective confidence intervals, then no matter their value, a point in x_ϵ lies in their convex hull.

3.3.2.1 Uniform

This simple policy samples from the active actions and chooses the action with the least samples, leading to uniform sample sizes across active actions. Active actions at time t are those whose confidence regions at time $(t-1)$ contain a boundary point of x_ϵ. The policy is given in algorithm 5.

Algorithm 5: Uniform Bernoulli

input: Number of actions k, confidence $1 - \delta$, x, ϵ.
Sample from each action once.
while *Stop* = *False* **do**
  Update active actions $A_t = \{i : \exists y \in \partial x_\epsilon, y \in R_i(t)\}$.
  $a_{t+1} = \operatorname{argmin}_{i \in A_t} n_i(t)$
end

3.3.2.2 LUCB Mean

This policy is based on the idea of sampling the active action with the confidence boundary furthest from x in the direction of greatest uncertainty, as given in definition 3.3.1. Given this direction, we exploit the action whose confidence bound is furthest from x. The policy is given in algorithm 6.

Algorithm 6: LUCB Mean Bernoulli

input: Number of actions k, confidence $1-\delta$, x, ϵ.
Sample from each action once.
while *Stop* = *False* **do**
  $u_t = \text{argmin}_{u\in\{1,-1\}} \max_{i\in[k]} u(\hat{p}_i(t) - x) - B_i(t)$
  $a_{t+1} = \text{argmax}_{i\in[k]}\, u_t(\hat{p}_i(t) - x) + B_i(t)$
end

3.3.2.3 LUCB Ratio

Using definition 3.3.1 to define the direction of greatest uncertainty, the intuition of this policy is to sample from the active action whose confidence region has the largest proportion of area on the side of x in this direction. It is possible that two actions have the same confidence ratio, at which point exploring the less sampled action provides more information. To account for this, we scale the confidence ratio by $\frac{1}{\sqrt{n_i}}$. The policy is given in algorithm 7.

Algorithm 7: LUCB Ratio Bernoulli

input: Number of actions k, confidence $1-\delta$, x, ϵ.
Sample from each action once.
while *Stop* = *False* **do**
  $u_t = \text{argmin}_{u\in\{1,-1\}} \max_{i\in[k]} u(\hat{p}_i(t) - x) - B_i(t)$
  $a_{t+1} = \text{argmax}_{i\in[k]}\, \frac{1}{\sqrt{n_i}} \frac{u_t(\hat{p}_i(t)-x)+B_i(t)}{u_t(x-\hat{p}_i(t))+B_i(t)}$
end

3.3.2.4 Thompson Sampling

This probabilistic algorithm is a standard choice in the bandit literature. With few changes we adjust it to the convex hull feasibility problem. Again we use the direction of greatest uncertainty, sample a mean from the posterior of each action, and play the action with the mean furthest from x in the given direction. The policy is given in line 8 where $r_i(t)$ are the number of success drawn from action i at time t.

Algorithm 8: Beta Thompson Sampling

input: Number of actions k, confidence $1-\delta$, x, ϵ
Sample from each action once.
while *Stop* = *False* **do**
  Update posteriors $\pi_i(t) = Beta(1 + r_i(t), 1 + (n_i(t) - r_i(t)))$
  $u_t = \text{argmin}_{u \in \{1,-1\}} \max_{i \in [k]} u(\hat{p}_i(t) - x) - B_i(t)$
  Sample $\tilde{p}_i(t)$ from posterior $\pi_i(t)$ for all $i \in [k]$
  $a_{t+1} = \text{argmax}_{i \in [k]}\, u_t(\tilde{p}_i(t) - x)$
end

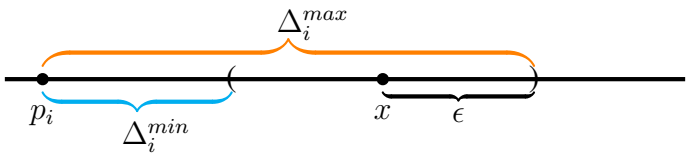


Figure 3.1: Visualization of $\Delta_i^{max}, \Delta_i^{min}$ for some p_i given x, ϵ.

3.3.3 Sample Complexity Upper Bounds

For both Uniform and LUCB Mean policies we give high probability upper bounds on the sample complexity. These policies sample all actions in relation to the optimal feasible subset, and thus allow a simple bounding on the complexity of each action. In Section 3.5, we show that Beta TS, LUCB Ratio, and LUCB Mean outperform Uniform empirically.

Notation: We define Δ_i^{max} to be the maximum distance from p_i to a boundary of x_ϵ and define Δ_i^{min} to be the minimum distance from p_i to a boundary of x_ϵ. Let s_i^{max} be the minimum integer solution to $\Delta_i^{max} > 2B(s_i^{max}, \delta)$ and similarly for s_i^{min}. Therefore we have that with probability at least $1 - \delta/k$, when action i is sampled $s_i^{max}(s_i^{min})$ times, $\Delta_i^{max}(\Delta_i^{min})$ will not be contained in its confidence region. A visualization of the gap relationship is show in figure 3.1. We additionally define $\Delta_{i,j} = |p_i - p_j|$ and $s_{i,j}$ and the smallest integer such that $\Delta_{i,j} > 2B(s_{i,j})$.

For the feasible Bernoulli setting, the optimal subset $\mathcal{J}^*$ will consist of one action, $\mathcal{J}^* = \{l^*\}$, or two actions, $\mathcal{J}^* = \{1, k\}$. In the following theorems we include the general case for any $B(n, \delta)$ that satisfies equation (3.3.1) and where s depends on choice of $B(n, \delta)$, as well as for the case with $B(n, \delta) = \sqrt{\frac{1}{2n} \log(n^2 \frac{5k}{3\delta})}$, which shows the gap dependencies clearly.

Theorem 3 (Uniform Complexity)**.** *Let* $j^* = \text{argmax}_{i \in \{1,k\}}\, s_i^{max}$. *Assume* $B(n, \delta)$ *satisfies equation* (3.3.1)*. When the underlying case is feasible, the sample complexity of Uniform is bounded above by*

$$\tau \leq \begin{cases} \mathcal{O}\left(\sum_{i=1}^{k} \min\left(s_{j^*}^{max}, s_i^{min}\right)\right) & \mathcal{J}^* = \{1, k\} \\ \mathcal{O}\left(\sum_{i=1}^{k} \min\left(s_{l^*}^{min}, s_i^{min}\right)\right) & \mathcal{J}^* = \{l^*\} \end{cases}$$

and for $B(n,\delta) = \sqrt{\frac{1}{2n}\log(n^2\frac{5k}{3\delta})}$,

$$\tau \leq \begin{cases} \mathcal{O}\left(\sum_{i=1}^{k} \min\left(\frac{1}{(\Delta_{j^*}^{max})^2}, \frac{1}{(\Delta_i^{min})^2}\right)\right) & \mathcal{J}^* = \{1, k\} \\ \mathcal{O}\left(\sum_{i=1}^{k} \min\left(\frac{1}{(\Delta_{l^*}^{min})^2}, \frac{1}{(\Delta_i^{min})^2}\right)\right) & \mathcal{J}^* = \{l^*\} \end{cases}$$

with probability at least $1-\delta$.

When the cases is infeasible, the sample complexity of Uniform is bounded above by

$$\tau \leq \mathcal{O}\left(\sum_{i=1}^{k} s_i^{min}\right) \qquad \tau \leq \mathcal{O}\left(\sum_{i=1}^{k} \frac{1}{(\Delta_i^{min})^2}\right)$$

For $B(n,\delta) = \sqrt{\frac{1}{2n}\log(n^2\frac{5k}{3\delta})}$

With probability at least $1-\delta$.

Theorem 4 (LUCB Mean Complexity)**.** *Let* $j^* = \text{argmax}_{i\in\{1,k\}}\, s_i^{max}$ *and* $i^* = \text{argmin}_{i\in\{1,k\}}\, s_i^{max}$. *Assume* $B(n,\delta)$ *satisfies equation* (3.3.1)*. When the underlying is feasible, the sample complexity of LUCB Mean is bounded above by*

$$\tau \leq \begin{cases} \mathcal{O}\left(\sum_{i:\Delta_{i,j^*}\leq\Delta_{j^*}^{max}} s_{j^*}^{max} + \sum_{i:\Delta_{i,j^*}>\Delta_{j^*}^{max}} \max\left(s_{i,j^*}, s_{i^*}^{max}\right)\right) & \exists i, j,\ p_i < x < p_j \\ \mathcal{O}\left(\sum_{i=1}^{k} \min\left(s_{i,j^*}, s_{j^*}^{min}\right)\right) & \textit{otherwise} \end{cases}$$

and for $B(n,\delta) = \sqrt{\frac{1}{2n}\log(n^2\frac{5k}{3\delta})}$

$$\tau \leq \begin{cases} \mathcal{O}\left(\sum_{i:\Delta_{i,j^*}\leq\Delta_{j^*}^{max}} \frac{1}{(\Delta_{j^*}^{max})^2} + \sum_{i:\Delta_{i,j^*}>\Delta_{j^*}^{max}} \max\left(\frac{1}{\Delta_{i,j^*}^2}, \frac{1}{(\Delta_{i^*}^{max})^2}\right)\right) & \exists i, j,\ p_i < x < p_j \\ \mathcal{O}\left(\sum_{i=1}^{k} \min\left(\frac{1}{\Delta_{i,j^*}^2}, \frac{1}{(\Delta_{j^*}^{min})^2}\right)\right) & \textit{otherwise} \end{cases}$$

with probability at last $1-\delta$.

When the underlying case is infeasible, the sample complexity of LUCB Mean is bounded above by,

$$\tau \leq \mathcal{O}\left(\sum_{i=1}^{k} s_i^{min}\right) \qquad \tau \leq \mathcal{O}\left(\sum_{i=1}^{k} \frac{1}{(\Delta_i^{min})^2}\right)$$

$$\textit{For } B(n,\delta) = \sqrt{\frac{1}{2n}\log(n^2\frac{5k}{3\delta})}$$

With probability at least $1-\delta$.

This shows that for any problem instance, the worst case sample complexity is lower using the LUCB Mean policy compared to the Uniform policy, since $\min(s_{j^*}^{max}, s_l^{min}) \geq \max(s_{l,j^*}, s_{i^*}^{max})$ for all $l \in [k]$. We leave details of this to Appendix A.1.3.

Intuitively, theorem 3 says that all actions are sampled as many times as the most sampled optimal arm or until it's confidence region is disjoint from x_ϵ. For theorem 4, the bounds are more complicated because it depends upon how close the action mean is to p_{j^*} and the instance setting. Generally speaking, the bounds describe a relationship between the relative distance of an action's mean and the most sampled optimal action's mean, an action's mean and a boundary point in x_ϵ, and the worst case behavior of the action's confidence region. These particular's are detailed in the proof.

3.4 Multinomial Feasibility Sampling

By expanding our definition of the direction of greatest uncertainty and our stopping rules, we can modify each policy to work in higher dimensions.

3.4.1 Feasibility and Infeasibility Checks

Recall the definition of $1-\delta$ Confident Feasible (definition 3.2.4). If we assume $\epsilon = 0$, an equivalent definition would be

$$\forall u, ||u|| = 1,\ \exists R_i \text{ such that } (q_i - (x+w))^T u > 0\ \forall q_i \in R_i$$

which states that x is not separable from any subset of points constructed from the confidence regions. Alternatively, we may say that for all unit vectors u, $\max_{i\in[k]} \min_{q\in R_i} (q_i - x)^T u > 0$. If we limit the confidence regions to be balls with radius B, then we can simplify to say x is feasible

if

$$\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - x)^T u - B_i > 0.$$

Now considering $\epsilon > 0$, we would need to show there exists some point $(x+w) \in x_\epsilon$, $||w|| < \epsilon$,

$$\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - (x+w))^T u - B_i > 0$$

We have that for some $\lambda \in (0,1)$,

$$\begin{aligned}
&\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - x)^T u - B_i > -\lambda\epsilon \\
&\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - x)^T u - B_i - w^T u > -\lambda\epsilon - w^T u \\
&\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - (x+w))^T u - B_i - w^T a > 0
\end{aligned}$$

Where in the last line we have that since u is a unit vector and the length of w is bounded by ϵ, we can pick w, $\lambda \in (0,1)$ such that $w^T u = -\lambda\epsilon$. Therefore a feasibility check becomes if, given some $\lambda \in (0,1)$,

$$\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - (x+w))^T u - B_i > 0.$$

In a similar fashion, an infeasibility check would be if there exists a unit vector a such that,

$$\min_{u:||u||=1} \max_{i\in[k]} (\hat{p}_i - x)^T u + B_i < -\epsilon.$$

3.4.2 Sampling Policies

Using the above formulation for checking feasibility lends itself to defining the direction of greatest uncertainty in any dimension.

Definition 3.4.1 (Direction of greatest uncertainty)**.** Given a confidence margin B_i and mean estimate $\hat{p}_i$, the direction of greatest uncertainty u is defined as,

$$u = \operatorname*{argmin}_{u:\ ||u||=1} \max_{i\in[k]} (\hat{p}_i - x)^T u - B_i.$$

Unfortunately, finding the direction of greatest uncertainty for $d \geq 3$, and thus also checking feasibility, is a non-convex problem, so we cannot obtain the optimal solution. One obvious workaround to this is simply doing a grid search over some subset of points on the unit ball. This is the approach we take.

Let G be some subset of the unit ball in dimension d which will be the directions we search

over, and let $\lambda \in (0, 1)$ be a parameter.

Stopping Rules: If one of the following criteria are met, the policy terminates,

1. x_ϵ is not separable from the confidence balls in any direction $u \in G$.
   $\min_{u \in G} \max_{i \in [k]} (\hat{p}_i(t) - x)^T u - B_i(t) > -\lambda\epsilon$

2. x_ϵ is separable from all confidence balls.
   $\min_{u \in G} \max_{i \in [k]} (\hat{p}_i(t) - x)^T u + B_i(t) < -\epsilon$

Our sampling algorithms do not change significantly to accommodate higher dimensions. The Uniform policy no longer has an active action set, and the other policies use the updated definition of direction of greatest uncertainty and vector dot products instead of scalar multiplication. The policies are given in algorithm 9 (Uniform), algorithm 10 (LUCB Mean), algorithm 11 (LUCB Ratio), and algorithm 12 (Dirichlet Thompson sampling).

Algorithm 9: Uniform

input: Number of actions k, confidence $1 - \delta$, x, ϵ.
Sample from each action once.
while *Stop* = *False* **do**
  $a_{t+1} = \text{argmin}_{i \in [k]}\, n_i(t)$
end

Algorithm 10: LUCB Sampling

input: Number of actions K, confidence $1 - \delta$, unit vectors G.
fix : $A = [k]$
Sample from each action once.
while *Stop* = *False* **do**
  $u_t = \text{argmin}_{u \in G}\, umax_{i \in [k]} (\hat{p}_i - x)^T u - B_i(t)$
  $a_{t+1} = \text{argmax}_{i \in [k]} (\hat{p}_i(t) - x)^T u_t + B_i(t)$
end

Algorithm 11: Confidence Ratio Sampling

input: Number of actions K, confidence $1 - \delta$, unit vectors G.
Sample from each action once.
while *Stop* = *False* **do**
  $u_t = \text{argmin}_{u \in G} \max_{i \in [k]} (\hat{p}_i - x)^T u - B_i(t)$
  $a_{t+1} = \text{argmax}_{i \in [k]} \frac{1}{\sqrt{n_i}} \frac{(\hat{p}_i(t) - x)^T u_t + B_i(t)}{(x - \hat{p}_i(t))^T u_t + B_i(t)}$
end

Algorithm 12: Dirichlet Thompson Sampling

input: Number of actions K, confidence $1-\delta$, priors π_i, unit vectors G
Sample from each action once.
while *Stop* = *False* **do**
  $u_t = \text{argmin}_{a \in G} \max_{i \in [k]} (\hat{p}_i - x)^T u - B_i(t)$
  Sample $p_i(t)$ from posterior $\pi_i(t)$ for all $i \in [k]$
  $a_{t+1} = \text{argmax}_{i \in [k]} (p_i(t) - x)^T u_t$
end
Solve for t in plug optimization. If $t \geq 0$, out = *feasible*, else if $t < 0$ out = *infeasible*.

3.5 Simulations

We compare the average sample size till termination of our three policies against the naive uniform sampling method.

3.5.1 Setup

We run our policies in the Bernoulli setting (which correlates to both $d = 1$ and $d = 2$) and the Multinomial setting with $d = 3$. Each graph shows the average sample size at termination for the four policies when averaged over 30 trials using $B(n, \delta) = \sqrt{\frac{1}{2n} \log(n^2 \frac{5k}{3\delta})}$. In all trials, we set $\delta = .01$, $k = 10$, $\epsilon = 0.1$, and set $\lambda = .99$ when $d = 3$. In the Multinomial setting, we use a grid search over 300 points on the unit sphere. For the Bernoulli setting we run scenarios for $|\mathcal{J}^*| \in \{1, 2\}$ and for the Multinomial setting $|\mathcal{J}^*| \in \{1, 2, 3\}$. In each of these settings we further consider two cases, when $\mathcal{J}^*$ is unique and when it is not. When the $\mathcal{J}^*$ is unique, it is an element of the set of optimal subsets when $\mathcal{J}^*$ is not unique. Therefore the oracle lower bound is the same for unique and non-unique cases and we can compare the two outcomes when all other parameters are fixed. The setting for each scenario is listed in the caption. The desired values is in the Bernoulli case $x = .5$, and $x = (.33, .33, .33)$ in the Multinomial case. The means used in each setting are listed in tables 3.1 and 3.2, and were chosen as a general representation of several different scenarios.

Table 3.1: Bernoulli Mean Values

Bernoulli	$\lvert\mathcal{J}^*\rvert = 1$	$\lvert J^*\rvert = 2$
Optimal	.5	.3, .7
Non-optimal	.48, .52	.48, .52

Table 3.2: Multinomial Mean Vectors

Multinomial	$\lvert J^* \rvert = 1$	$\lvert J^* \rvert = 2$	$\lvert J^* \rvert = 3$
Optimal	(.33, .33, .33)	(.1, .57, .33) (.57, .1, .33)	(.2, .1, .7) (.7, .2, .1) (.1, .7, .2)
Non-optimal	(0, 0, .1)	(.2, .47, .33) (.47, .2, .33)	(.33, .33, .34) (.33, .34, .33) (.34, .33, .33)

3.5.2 Results

When the average sample size of the Uniform policy is substantially larger than that of the best performing policy, the y-axis has a break point to indicate a change in the scale.

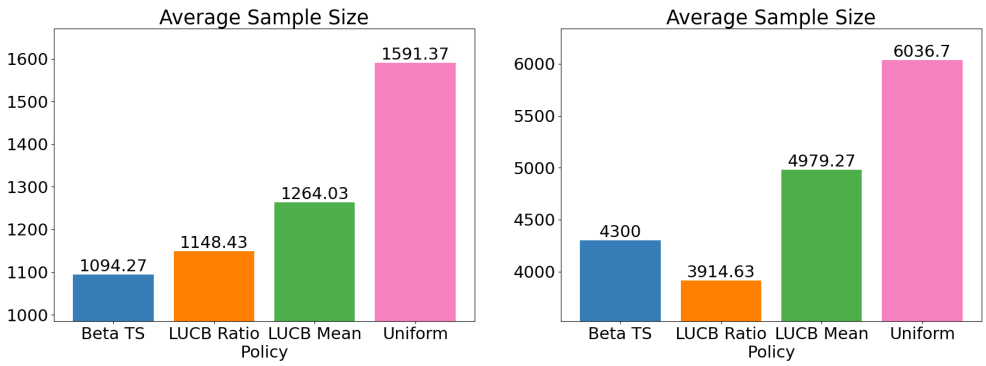
Figures 3.2 and 3.3 show results for Bernoulli sampling and figures 3.4 to 3.6 show resuts for the Multinomial sampling with $d = 3$. It is clear that the Uniform sampling policy performs the worst in all cases, and is improved upon by all other policies presented in this chapter. It is clearly seen, and somewhat surprising, that there is a large relative difference in performance of LUCB Ratio and Thompson sampling between the Bernoulli and Multinomial setting.

In the Multinomial setting Dirichlet Thompson sampling has superior performance, while in the Bernoulli setting LUCB Ratio has the best performance, except when there in one unique optimal action, as seen in figure 3.2a. Here Beta Thompson sampling (Beta TS) outperforms the other policies. We speculate that in this particular Bernoulli setting, Beta TS this may be because this case is most similar to the standard multi-armed bandit problem, which aims to select the action with the highest mean as often as possible. The multi-armed bandit Beta TS policy is one of the simplest and most effective policies in practice [Chapelle and Li, 2011].

Looking at figure 3.2, when $\mathcal{J}^*$ is unique it requires fewer sample sizes on average for each policy than when $\mathcal{J}^*$ is not unique. This relationship reversed in figure 3.3. This example shows that uniqueness of $\mathcal{J}^*$ in the Bernoulli setting does not imply a simpler problem. This is similarly seen in the Multinomial setting. We see in figure 3.4 that Dirichlet TS and LUCB Ratio perform better in the unique optimal subset setting, and there is no difference for LUCB Mean and Uniform. Whereas in figure 3.5, all but LUCB Ratio perform better in the non-unique optimal subset setting.

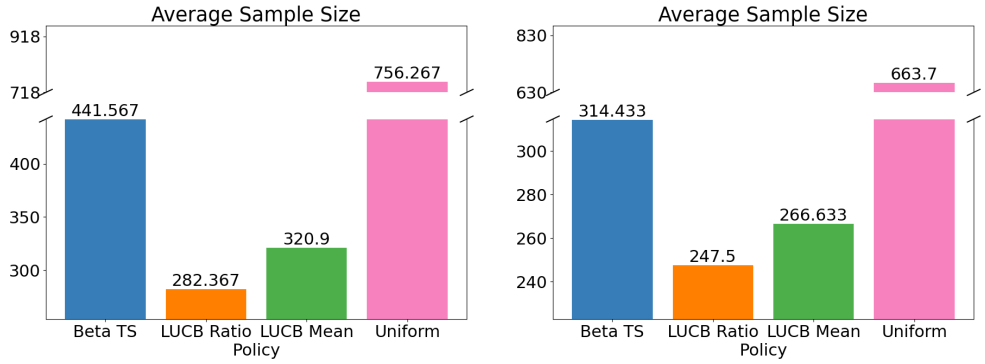
We see in both the Bernoulli and Multinomial setting that the larger the optimal subset, the fewer average samples before termination. This is because when $|\mathcal{J}^*| < d$ the optimal subsets must be sampled until $B(n, \delta) \approx \epsilon$ to ensure there is a mean either on both sides of x or that a confidence region is fully contained in x_ϵ in all directions.

In practice, results will be dependent upon the underlying truth, as can be inferred by the Oracle



(a) $\mathcal{J}^*$ unique. (b) $\mathcal{J}^*$ not unique.

Figure 3.2: Average stopping time in Bernoulli setting, $d = 1,\ |\mathcal{J}^*| = 1, k = 10$.



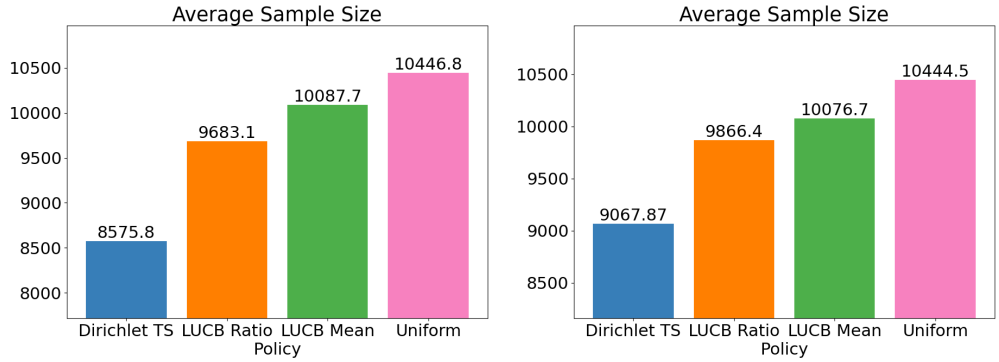
(a) $\mathcal{J}^*$ unique. (b) $\mathcal{J}^*$ not unique.

Figure 3.3: Average stopping time in Bernoulli setting, $d = 1,\ |\mathcal{J}^*| = 2, k = 10$.

average sample complexity lower bound and the high probably sample complexity upper bounds given in this chapter. These simulations give evidence of the magnitude of improvement using an adaptive sampling method over the naive uniform method. Depending on the setting, average sample size can be reasonably small, as seen in figures 3.3a and 3.3b. In the Multinomial setting, average sample sizes are in the thousands. The practicality of this method can be seen to depend upon the true distributions, sampling budget, and parameter values.

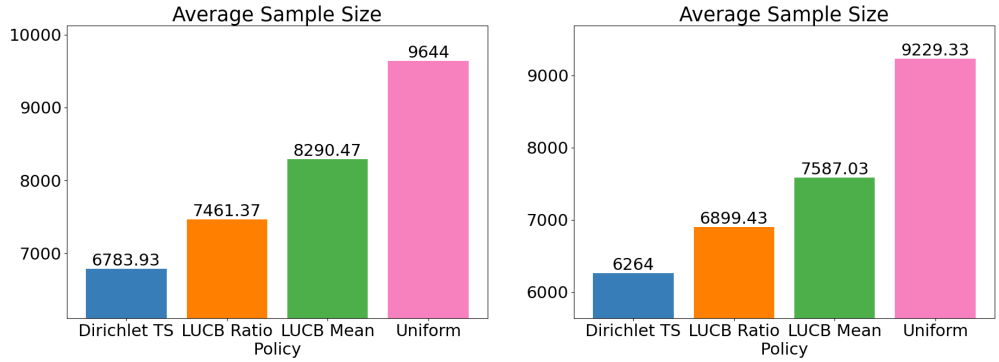
3.6 Summary and Discussion

We introduce the convex hull feasibility problem in the context of fair data collection. In the Bernoulli setting, we give a lower bound on the expected sample complexity in the (x, ϵ)-infeasible instance and an oracle lower bound on the expected sample complexity in the (x, ϵ)-feasible instance. We introduce four sampling policies for the Bernoulli setting, Uniform, LUCB Mean,



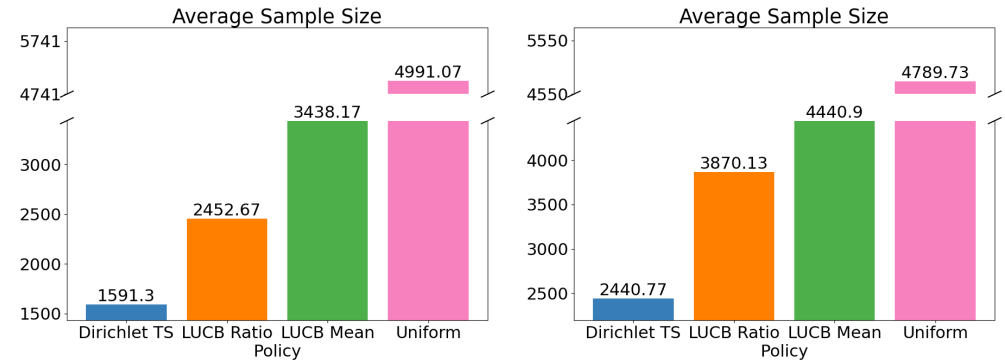
(a) $\mathcal{J}^*$ unique. (b) $\mathcal{J}^*$ not unique.

Figure 3.4: Average stopping time in Multinomial setting, $d = 3,\ |\mathcal{J}^*| = 1, k = 10$.



(a) $\mathcal{J}^*$ unique. (b) $\mathcal{J}^*$ not unique.

Figure 3.5: Average stopping time in Multinomial setting, $d = 3,\ |\mathcal{J}^*| = 2, k = 10$.



(a) $\mathcal{J}^*$ unique. (b) $\mathcal{J}^*$ not unique.

Figure 3.6: Average stopping time in Multinomial setting, $d = 3,\ |\mathcal{J}^*| = 3, k = 10$.

LUCB Ratio, and Beta TS and give high probability upper bounds on sample complexity for the Uniform and LUCB Mean policies. We give the adaptation of the Binomial policies to the Multinomial case. Through simulation, we show LUCB Mean, LUCB Ratio, and the Thompson sampling policies significantly outperform Uniform in the Bernoulli and Multinomial setting. Under our simulation scenarios, we see that LUCB Ratio is typically the best performing policy in our Bernoulli settings, while Dirichlet TS is the best performing policy in our Multinomial settings. We discuss that the practicality of implementation is dependent upon the underlying distributions, sampling budget, and chosen parameters. Large sampling budgets would enable this method practical under most settings, whereas with small sampling budgets this method would only be practical if there was a strong prior that the underlying distributions has a small oracle lower bound.

While this work focused on Bernoulli and Multinomial convex hull feasibility sampling, the general problem is applicable when points are drawn from any distribution for which one can construct a confidence region that satisfies equation (3.3.1).

There are some limitations within this work. Notably, we were only able to give an oracle lower bound on the expected sample complexity in the feasible case. A true lower bound would allow for better comparison of a policy's theoretical performance. Additionally, we provide theoretical results in the Bernoulli settings, but not the Multinomial setting.

3.6.1 Future Work

The work in this chapter is somewhat analogous to multi-armed bandit best arm identification with fixed-confidence problem. Another approach seen in the best arm identification literature is the fixed-budget setting, which could also be applied to the convex hull feasibility problem. Given a set of samples, the confidence regions can be such that they do not meet the definition of either $(1-\delta)$-confident feasible or $(1-\delta)$-confident infeasible. In this case we could ask instead what is the probability of feasibility or infeasible given the current sample, or if adaptively sampling, what is the highest probably of a correct decision when sampling with a budget. One application of this could be to check Pareto frontier feasibility, where if given noisy gradients, we ask what is the probability all groups can be improved versus the probability that improving some groups may harm others.

The problem remains that for higher dimensions, $d > 2$, the optimization problem for determining both feasibility and the direction of greatest uncertainty in non-convex. A relaxation of this formulation could result in a better solutions than the current grid-search allows.

Another avenue of interest would be testing on standard data sets used in the fairness literature. Here one could calculate the theoretical lower bounds given the entire data set, then compare to the performance of the policies presented in this work in terms of average sample size. Finally, with

the curated balanced data sets, one could compare fairness outcomes compared to other curation or preprocessing methods.

CHAPTER 4

Debiasing Representations by Removing Unwanted Variation Due to Protected Attributes

4.1 Introduction

It is generally assumed that the use of algorithms removes human bias from decision making. In practice, this is not usually the case, and there are numerous examples of algorithms with outcomes that are unfair to members of different protected classes (see e.g. Angwin et al. [2016], Steel and Angwin [2010]).

In recent years, there has been a flurry of work aimed at correcting this issue. Starting with the seminal paper Dwork et al. [2012], this work has generally fallen into four categories:

1. Mathematical or statistical definitions of fairness (e.g. Friedler et al. [2016], Ritov et al. [2017]).
2. Algorithms (or recommendations for algorithms) that are modeled to ensure fairness (e.g. Joseph et al. [2016]).
3. Methods of preprocessing data in order to remove inherent bias so that algorithms trained on the debiased data will be fair (e.g. Zemel et al. [2013], Feldman et al. [2015]).
4. Methods of debiasing the outcomes of existing algorithms (a postprocessing step; e.g. Hardt et al. [2016]).

This work falls into the third category of preprocessing. We introduce a factor model prevalent in genetics applications to model the contributions of the protected and permissible attributes to the representation. We treat the variation that is present in the data due to protected attributes (e.g. race) as unwanted, and we propose a method to remove this unwanted variation based on the factor model (and thus debias the data). We further compute the correlation between the debiased data and the original protected attributes. In ideal cases, we show that there is no correlation, and therefore our debiased data satisfies a relaxed version of conditional parity Ritov et al. [2017] in these cases.

4.1.1 Motivating Example

We use ProPublica's COMPAS data set and COMPAS risk recidivism scores as an example throughout. More information can be found from Angwin et al. [2016] and Practitioners Guide to COMPAS [1]. Much has been written questioning the fairness of these scores with respect to race, with concerns about the disparate false negative and false positive rates between African-Americans and Caucasians.

Notation: We denote matrices by uppercase greek or Latin characters and vectors by lowercase characters. A (single) subscript on a matrix indexes its rows (unless otherwise stated). A random matrix $X \in \mathbb{R}^{n \times d}$ is distributed according to a *matrix-variate normal* distribution with mean $M \in \mathbb{R}^{n \times d}$, row covariance $\Sigma_r \in \mathbb{R}^{n \times n}$, and column covariance $\Sigma_c \in \mathbb{R}^{d \times d}$, which we denote by $X \sim \mathsf{MN}(\mathsf{M}, \mathsf{\Sigma_r}, \mathsf{\Sigma_c})$.

4.2 Related Work

The factor model of the representation that motivates the proposed approach (Equation 4.3.1) is widely used in genetics applications to model gene expression data. The model was first introduced by Leek and Storey [2008] to represent wanted and unwanted variation in gene expression data. The model was exploited by Gagnon-Bartsch et al. [2013] to develop the removing unwanted variation (RUV) family of methods.

RUV methods rely on knowledge of a set of control genes: genes whose variation in their expression levels are solely attributed to variation in Z, for example, genes unaffected by the treatments. Formally, a set of controls is a set of indices $\mathcal{I} \subset [d]$ such that $B_{\mathcal{I}} = 0$. Thus

$$Y_{\mathcal{I}} = X A_{\mathcal{I}}^T + E_{\mathcal{I}},$$

where $Y_{\mathcal{I}}$ and $E_{\mathcal{I}}$ consist of subsets of the *columns* of Y and E, which suggests estimating $A_{\mathcal{I}}^T$ by linear regression. This is precisely the "transpose" of the method that we advocate.

4.3 Adjusting for Protected Attributes

Consider the following widely adopted model for matrix-variate data:

$$\underset{(n \times d)}{Y} = \underset{(n \times k)}{X} \underset{(k \times d)}{A^T} + \underset{(n \times l)}{Z} \underset{(l \times d)}{B^T} + \underset{(n \times d)}{E}. \tag{4.3.1}$$

[1] http://www.northpointeinc.com/files/technical_documents/FieldGuide2_081412.pdf

The rows of Y are representations, the rows of X (resp. Z) are protected attributes (resp. permissible attributes) of the samples, and the rows of E are error terms that represent idiosyncratic variation in the representations. In this work, we assume $k, l \ll d$.

In practice, Y is usually observed, X is sometimes observed, and Z is usually unobserved. For example, in Bolukbasi et al. [2016], the representations are embeddings of words in the vocabulary, and the protected attribute is the gender bias of (the embeddings of) words. The rows of Z are unobserved factor loadings that represent the "good" variation in the word embeddings. In analogy to the framework proposed by Friedler et al. [2016], the rows of Z are points in the construct space while the rows of Y are points in the observed space. We emphasize that like the construct space, Z is unobserved.

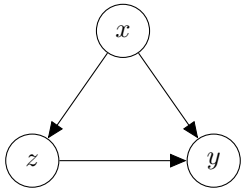


Figure 4.1: The model (4.3.1) and (4.3.2). permissible attributes

We highlight that we permit non-trivial correlation between the protected and permissible attributes. In other words, we allow the protected attribute to *confound* the relationship between the permissible attribute and the representation (see Figure 4.1 for a graphical representation of the dependencies between the rows of Y, X, and Z). This complicates the task of debiasing the representations. To keep things simple, we assume the regression of Z on X is linear:

$$\underset{(n\times l)}{Z} = \underset{(n\times k)(k\times l)}{X}\,\underset{}{\Gamma^T} + \underset{(n\times l)}{W}. \tag{4.3.2}$$

The rows of W are error terms that represent variation in the permissible attributes not attributed to variation in the protected attributes. We specify the distributions of X, E, and W, in Sections 4.3.2 and 4.3.3.

Our goal is to obtain debiased representations Y_{db} such that the debiased representations are uncorrelated with the protected attributes conditioned on the permissible attributes:

$$\mathsf{Cov}\big[[\mathsf{Y_{db}}]_\mathsf{i}, \mathsf{x_i} \mid \mathsf{z_i}\big] = 0. \tag{4.3.3}$$

This is implied by *conditional parity*: $[Y_{\mathsf{db}}]_i \perp x_i \mid z_i$, and we consider (4.3.3) as a first-order approximation of conditional parity. An ideal debiased representation is the variation in the representation attributed to the permissible attributes ZB^T, but this is typically unobservable in practice.

Another way to state our goal is estimating ZB^T.

4.3.0.1 COMPAS example

Under this model, each row of Y corresponds to each person's data allowed for recidivism prediction. The data we use in our experiments in Section 4.4 includes age, juvenile and adult felony and misdemeanor counts, and whether the offense was a misdemeanor or felony. Each row of X corresponds to a person's race, where we limit our experiments to Caucasian and African-American. Z is unknown.

4.3.1 Homogeneous Subgroups

The proposed approach relies crucially on knowledge of homogeneous subgroups: groups of samples in which the variation in their representations is mostly attributed to variation in their protected attributes. Formally, we presume knowledge of sets of indices $\mathcal{I}_1, \ldots, \mathcal{I}_G \subset [n]$ such that $H_g Z_{\mathcal{I}_g} \approx 0$, where $H_g = I_{|\mathcal{I}_g|} - \frac{1}{|\mathcal{I}_g|}\mathbf{1}_{|\mathcal{I}_g|}\mathbf{1}_{|\mathcal{I}_g|}^T$ is the centering matrix, for any $g \in [G]$. In other words,

$$H_g Y_{\mathcal{I}_g} \approx H_g X_{\mathcal{I}_g} A^T + H_g E.$$

Ideally, $H_g Z_{\mathcal{I}_g}$ exactly vanishes. This ideal situation arises when the samples in the g-th group share permissible attributes: $Z_{\mathcal{I}_g} = \mathbf{1}_{|\mathcal{I}_g|} z_g^T$ for some $z_g \in \mathbb{R}^l$.

Intuitively, homogeneous subgroups are groups of samples in which we expect a machine learning algorithm that only discriminates by the permissible attributes to treat similarly. For example, in Bolukbasi et al. [2016], the homogeneous subgroups are pairs of words that differ only in their gender bias: (*waiter*, *waitress*), (*king*, *queen*).

4.3.1.1 COMPAS example

In our experiments in Section 4.4, homogeneous groups consist of people who either did not recidivate within two years or people who did recidivate within two years and were charged with the same degree of felony or misdemeanor. Although Z is unknown, we expect subjects who go on to commit similar crimes or those who do not recidivate to have similar z_i regardless of race. We emphasize that the homogeneous subgroups are not defined by having similar attributes in Y.

4.3.2 Adjustment When the Protected Attribute is Unobserved

In this section, we show that the approach proposed by Bolukbasi et al. [2016] produces debiased representations that satisfy (4.3.3).

When the protected attributes are not observed, it is generally not possible to attribute variation in the representations to variation in the protected and permissible attributes. Thus, Bolukbasi et al. [2016] settle on removing the variation in the representations in the subspace spanned by the protected attributes. In other words, we debias the representations by projecting them onto the orthocomplement of $\mathcal{R}(A)$.

Formally, let $Q_g \in \mathbb{R}^{|\mathcal{I}_g| \times (|\mathcal{I}_g|-1)}$ be a subunitary matrix such that $\mathcal{R}(Q_g)$ coincides with $\mathcal{R}(H_g)$. Under (4.3.4), we have

$$Q_g^T Y_{\mathcal{I}_g} \approx Q_g^T X_{\mathcal{I}_g} + Q_g^T E,$$

which implies $\mathsf{Cov}\left[\mathsf{Q_g^T Y_{\mathcal{I}_g}}\right] \approx \mathsf{\Sigma_E + AA^T}$. This is a factor model, which allows us to consistently estimate A by factor analysis under mild conditions. We impose classical sufficient conditions for identifiability of A Anderson and Rubin [1956]:

1. Let A_{-i} be the $(d-1) \times k$ submatrix of A consisting of all but the i-th row of A. For any $i \in [n]$, there are two disjoint submatrices of A_{-i} of rank k.
2. $A^T \Sigma_E^{-1} A$ is diagonal, and the diagonal entries are distinct, positive, and arranged in decreasing order.

We remark that the additional assumptions we imposed in this section are a tad stronger than necessary: the assumptions actually imply identifiability of A, but we only wish to estimate $\mathcal{R}(A)$.

In light of the preceding development, here is a natural approach to adjustment when the protected attribute is unobserved:

1. estimate A by factor analysis:
   $\operatorname{argmin}\{\frac{1}{2}\sum_{g=1}^{G} \|H_g Y_{\mathcal{I}_g} - XA\|_F^2\}$;
2. debias Y by projection onto $\mathcal{R}(A)^\perp$:
   $Y_{\mathsf{db}} = Y(I - P_{\mathcal{R}(A)})$.

Which gives:

$$\mathsf{Cov}[[\mathsf{Y_{db}}]_\mathsf{i}, \mathsf{x_i}|\mathsf{z_i}] = \mathsf{Cov}[\mathsf{P}_{\mathcal{R}(\mathsf{A})^\perp}(\mathsf{Bz_i} + \mathsf{e_i})|\mathsf{z_i}] = 0$$

It is important to note that when $B \subset \mathcal{R}(A)$ then the debiased representations will be non-informative because they only contain noise.

4.3.3 Adjustment if the Protected Attribute is Observed

If the protected attribute is observed, it is straightfoward to debias the representations. The main challenge here is estimating A. Once we have a good estimator $\widehat{A}^T$, we debias the representations by subtracting $X\widehat{A}^T$. We summarize the approach in Algorithm 13.

Algorithm 13: Adjustment if the protected attr. is observed

input: representations $Y \in \mathbb{R}^{n \times d}$, protected attributes $X \in \mathbb{R}^{n \times k}$ and groups $\mathcal{I}_1, \ldots, \mathcal{I}_G \subset [n]$

Estimate A **by regression:**

$$\widehat{A}^T \in \operatorname{argmin}\{\tfrac{1}{2}\sum_{g=1}^{G} \|Y_g - X_g A^T\|_F^2\},$$

where $Y_g = Y_{\mathcal{I}_g} - \mathbf{1}_{|\mathcal{I}_g|}(\frac{1}{|\mathcal{I}_g|}\mathbf{1}_{|\mathcal{I}_g|}^T Y_{\mathcal{I}_g})$ and X_g is defined similarly.

Debias Y**:** subtract the variation in Y attributed to X from Y: $Y_{\mathsf{db}} = Y - X\widehat{A}^T$.

To study the properties of Algorithm 13, we impose the following assumptions on the distributions of X, E, and W:

$$\begin{aligned} X &\sim \mathsf{MN}(0, \mathsf{I_n}, \Sigma_{\mathsf{x}}), \\ E \mid (X, Z) &\sim \mathsf{MN}(0, \mathsf{I_n}, \Sigma_{\epsilon}). \end{aligned} \tag{4.3.4}$$

Proposition 1. *Let Z_g and E_g be defined similarly as Y_g and X_g. Under conditions* (4.3.1)*,* (4.3.2)*, and* (4.3.4)*,*

$$\widehat{A}^T - A^T \mid (X, Z) \sim \mathsf{MN}(\mathsf{T}\sum_{\mathsf{g}=1}^{\mathsf{G}} \mathsf{X_g^T Z_g B^T}, \mathsf{T}, \Sigma_{\epsilon})$$

where $T = (\sum_{g=1}^{G} X_g^T X_g)^{\dagger}$.

We see that the (conditional) bias in the OLS estimator of A depends on the similarity of the permissible attributes in homogeneous subgroups. If $Z_g = 0$ for all $g \in G$, then $\widehat{A}$ is a (conditionally) unbiased estimator of A.

Proposition 2. *Under conditions* (4.3.1)*,* (4.3.2)*, and* (4.3.4)*, we have*

$$\mathsf{Cov}[\mathsf{y_i} - \widehat{\mathsf{A}}\mathsf{x_i}, \mathsf{x_i} \mid \mathsf{z_i}] = -\mathsf{B}\mathsf{Cov}[\tilde{\mathsf{Z}}\tilde{\mathsf{X}}(\tilde{\mathsf{X}}^{\mathsf{T}}\tilde{\mathsf{X}})^{\dagger}\mathsf{x_i}, \mathsf{x_i}|\mathsf{z_i}]$$

We see that if $\hat{A} = A$ or $\tilde{Z} = 0$ then the debiased y_i is uncorrelated with the protected attributes x_i.

4.4 Experiments: Debiased Representations for Recidivism Risk Scores

We empirically demonstrate the efficacy of Algorithm 13 for reducing racial bias in recidivism risk scores based on data that ProPublica [2] used in their investigation of COMPAS scores. Because we do not have access to Northpointe's COMPAS algorithm, we fit our own models to the raw

[2] https://github.com/propublica/compas-analysis/blob/master/compas-scores-two-years.csv

and debiased data. Although simple, the scores output by our model perform comparably to the proprietary COMPAS scores (see Figure 4.2 and Table 4.3).

In particular, we show that logistic regression (LR) trained on debiased data obtained from Algorithm 13 reduces the magnitude of the difference in the false positive rates (FPR) and false negative rates (FNR) between Caucasians and African-Americans (AA) compared to LR trained on raw, potentially biased data. This "fairer" outcome is achieved with a relatively small impact on the percentage of correct predictions. The variables in our LR model are discussed in Section 4.3.0.1.

We split our data into three pieces: a training set used just for debiasing in order to estimate A in Equation (4.3.1), a training set used to fit LR, and a test set to evaluate the performance of the learned model. The same training set is used to fit LR to both the raw data and the debiased data. Figure 4.3 shows the distribution of the probabilities of recidivism for African-Americans according to a logistic model based on the raw and debiased representations. The distribution of the probabilities from the raw representation is skewed to the right. In particular, the right tail of the distribution of probabilities from the raw representation is noticeably heavier than that from the debiased representation.

The ROC curve of the LR model trained on raw data is similar to the ROC curve of COMPAS scores validating the choice of LR as a proxy for COMPAS scores. See Figure 4.2. Over 30 splits

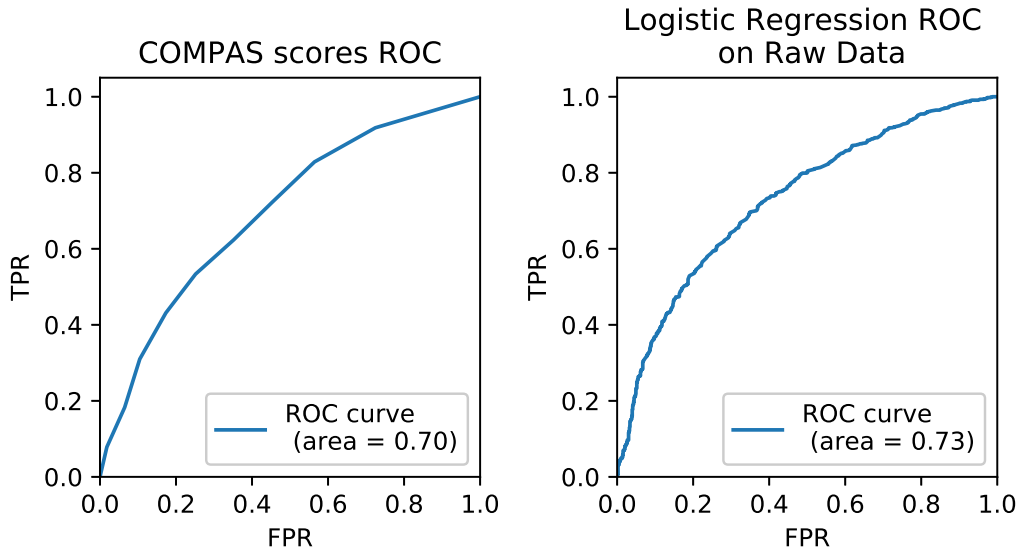


Figure 4.2: ROC curves based on COMPAS scores compared to ROC curves from the Raw COMPAS data using logistic regression.

of the data into train and test sets, the average accuracy (the percentage of correct predictions) is

65% for COMPAS. The accuracy for LR trained on raw data and debiased data is comparable, again validating our proxy and justifying the slight loss in accuracy after debiasing in pursuit of fairer outcomes. See Table 4.3.

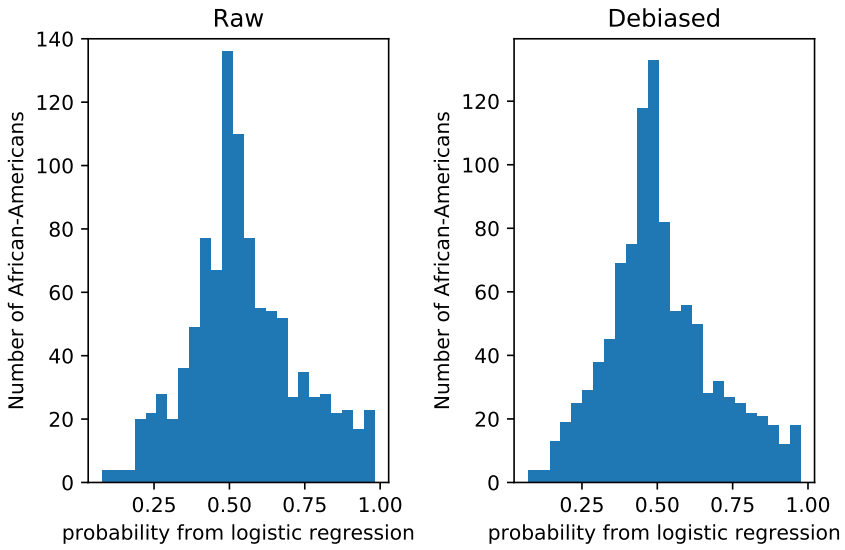


Figure 4.3: Distribution of recidivism probabilities from raw and debiased representations for African-Americans.

For the remaining discussion, we average all results over 30 splits of the data into train and test sets. Table 4.1 and Table 4.2 show the average FPR and FNR for the LR model before and after debiasing. The two tables differ only in the threshold used to declare someone at risk for recidivism based on his or her logistic score; we choose to examine the 50th and 80th quantiles of LR scores since Northpointe specifies that COMPAS scores above the of 50th (respectively 80th) quantile are said to indicate a "Medium" (respectively "High") risk of recidivism. In Table 4.1, we see that there is no difference in FPR after debiasing. The difference in FNR between the races goes from nearly 20% before debiasing to 4% after debiasing. In Table 4.2, we see FNR are nearly equalized, whereas the magnitude of the difference of FPR between both race groups is improved. However, now Caucasians suffer from disparate impact of FPR instead of African-Americans.

4.5 Summary and discussion

We study a factor model of representations that explicitly models the contributions of the protected and permissible attributes. Based on the model, we propose an approach to debias the representations. We show that under certain conditions, we can guarantee relaxed conditional parity for the debiased representations.

	LR raw		LR debiased	
	FPR (SE)	FNR (SE)	FPR (SE)	FNR (SE)
Population	0.8 (0.01)	0.68 (0.01)	0.9 (0.01)	0.69 (0.01)
Caucasian	0.05 (0.01)	0.81 (0.02)	0.9 (0.02)	0.72 (0.03)
AA	0.11 (0.01)	0.62 (0.01)	0.9 (0.01)	0.68 (0.02)

Table 4.1: Average percent FPR and FNR with standard errors (SE) based on the 80th quantile of LR scores.

	LR raw		LR debiased	
	FPR (SE)	FNR (SE)	FPR (SE)	FNR (SE)
Population	0.32 (0.01)	0.32 (0.01)	0.4 (0.01)	0.34 (0.01)
Caucasian	0.22 (0.02)	0.5 (0.02)	0.42 (0.03)	0.31 (0.02)
AA	0.4 (0.02)	0.23 (0.01)	0.27 (0.02)	0.35 (0.02)

Table 4.2: Average percent FPR and FNR with standard errors (SE) based on the 50th quantile of LR scores.

Accuracy	LR Raw (SE)	LR Debiased (SE)	COMPAS (SE)
50 quantile	0.67 (.011)	0.65 (.01)	0.65 (.008)
80 quantile	0.61 (.01)	0.60 (.01)	0.61 (.01)

Table 4.3: Percentage of correct predictions (with standard errors) by logistic regression and thresholding COMPAS scores

CHAPTER 5

Fair Pipelines

Automated-decision making saves time and is implicitly assumed to prevent human bias. However, such automated decisions may unfortunately lead to unfair outcomes. Until recently, the use of automated-decision making has been largely unchecked. In pursuit of this goal, the first question that needs to be addressed is what "fairness" itself actually means and how to quantify it. A meta-analysis of the current literature indicates there are a multitude of inequivalent applications of the term, and consequently metrics. See, for example, Friedler et al. [2016] (which is based off of definitions in the seminal paper, Dwork et al. [2012]) for a general framework that encompasses notions of individual fairness and group fairness ("non-discrimination").

In this work, we are particularly interested in how effects of bias compound in decision-making pipelines. While prior work in algorithmic fairness has focused on fairness of one decision, it is not immediately clear how fairness propagates throughout a compound decision making process. Complicated decisions usually require more than one decision. For example, a hiring process may include two decisions: from an applicant pool, one first decides who gets an interview, and the final hiring decision is made from the pool of interviewees. Although this is a relatively simple two-decision example, one can imagine a recursive-like compound decision-process where the outcome of one decision affects another and vice versa. For instance, perhaps to be brought in for an interview at company A, working for company B helps greatly, but working for company A also helps an applicant greatly to get an interview at company B.

We ask the following questions:

1. Can a decision at point j in a decision pipeline correct for unfairness at point $i < j$?
2. How much fairness from point i is preserved in later points in the pipeline?
3. More specifically, how does the fairness from each stage contribute to the fairness of the final decision?

Our contribution to the algorithmic fairness field is to highlight the need to study compound decision making processes by studying how composability and fairness interact. We emphasize

that pipelines are useful to study because they decouple the intermediate decisions since there may be completely different parties with varying goals and mechanisms responsible for each decision. Perhaps suprisingly, even in the most basic example of a two-stage pipeline, we show under a $(1+\varepsilon)$-equal opportunity definition of fair, the two stages cannot necessarily be combined as expected. Finally, pipelines set the stage for a number of interesting questions detailed in Section 5.3.1.

5.1 Framework

5.1.1 Pipelines

Definition 5.1.1 (Straight Pipeline)**.** An n-stage (straight) pipeline $P(f,g)$ on a set O is an ordered set of decision functions

$$\mathcal{F} = \{f_1 : O \to D_1,\ f_2 : O \times \widehat{D_1} \to D_2,\ \ldots,\ f_T : O \times \widehat{D_{T-1}} \to D_T\}$$

where $\widehat{D_t} = D_{k_t} \times D_{k_t+1} \times \cdots \times D_{t-1} \times D_t$ for $1 \le k_t \le t$ for $t = 1, \ldots, T$ and rule functions

$$\{g_1 : D_1 \to \{0,1\}, \ldots, g_{T-1} : D_{T-1} \to \{0,1\}\}$$

where the final decision for $x \in O$ is given by

$$P(f,g)(x) := \begin{cases} \hat{f}_T(x) & g_t(\hat{f}_t(x)) = 1\ \forall t \in [T-1] \\ \text{FAIL} & \text{otherwise} \end{cases}$$

where for $x \in O$, $\widehat{f}_1(x) = f_1(x)$ and for $t = 2, \ldots, T$, $\hat{f}_t(x) = f_t(x, \widehat{f_{k_t}}(x), \widehat{f_{k_t+1}}(x), \ldots, \widehat{f}_t(x))$. We will say decision function f_t takes place at *stage* t of the pipeline for $t = 1, \ldots, T$.

To understand the above definition, note that a straight pipeline $P(f,g)$ on a set O (for instance, applicants to a job) takes input $x \in O$ and applies a decision function on x. If the first decision ($f_1(x)$), on x is satisfactory (where satisfactory is determined by g_1), x is passed onto the next decision function and so on and so forth until there are no more decisions to be made (reaching f_T) or an intermediate stage declares an unsatisfactory decision (determined by some g_t) at which point no further decisions shall be made. Each subsequent decision function after the first may see some part of the past decisions prior in the pipeline (how many decisions back decision function f_t can see back is determined by k_t).

We expect the following variations and restrictions to be common with illustrative examples below:

1. **Filtering pipeline.** When each decision function f_t is binary, take $g_t(0) = 0$ and $g_t(1) = 1$, i.e., only the positive-decision subset of previous stage gets passed on. In this case, we may use the notation $P(f, g)$ in place of P_f for concision.

2. **Cumulative decisions pipeline.** Take $k_t \leq i$, i.e., the decision at each stage agglomerates some score onto previous stages' decision scores, so that each stage's decision can depend on some of the previous decisions.

3. **Informed pipeline.** Each stage of the pipeline has summary statistics about the outcome of previous stages. In this chapter, those statistics are implicit in the definition of f_t. In particular, our notation above, while sufficient for this work, is insufficient to study linked decisions in which each decision reacts to statistics about the other.

The below diagram illustrates a two-stage pipeline, where the second decision can see what happened in the first decision on a given input:

$$x \xrightarrow{f_1} f_1(x) \xrightarrow{g_1(f_1(x))=1} (x, f_1(x)) \xrightarrow{f_2} f_2(x, f_1(x))$$
$$\Big\downarrow {\scriptstyle g_1(f_1(x))=0} \qquad\qquad\qquad\qquad \Big\downarrow$$
$$P(f,g)(x) = FAIL \qquad\qquad P(f,g)(x) = \hat{f}_2(x)$$

Example 5.1.1 (Hiring decisions as a two-stage filtering pipeline)**.** Let O be the applicant pool, $f_1 : O \to D_1 = \{0, 1\}$ the decision as to who receives an interview, and $f_2 : O \times D_1 \to D_2 = \{0, 1\}$ the hiring decision. Then $P_f(x) = \hat{f}_2(x) = f_2(x, f_1(x))$ for all applicants x such that $f_1(x) = 1$ (i.e. for all applicants who receive an interview), and 0 (or FAIL) otherwise.

5.1.2 Fairness

As we have stated, there is no consensus in the literature on the definition of fairness. However, there have been many recent proposed definitions. For simplicity, in order to illustrate how fairness propagates through a filtering pipeline, we will build on the definition of equal opportunity found in Hardt et al. [2016].

Equal Opportunity

We consider the case of making a binary decision $\hat{Y} \in \{0, 1\}$ and measure fairness with respect to a protected attribute $A \in \{0, 1\}$ (such as age, gender, or race) and the true target outcome $Y \in \{0, 1\}$, which captures if an individual is qualified or not.

Definition 5.1.2. As defined in Hardt et al. [2016], a binary predictor $\hat{Y}$ satisfies equal opportunity with respect to A and Y if

$$\Pr\{\hat{Y} = 1|A = 0, Y = 1\} = \Pr\{\hat{Y} = 1|A = 1, Y = 1\}.$$

In other words, we make sure that the true positive rates are the same across a protected attribute. We usually think of $A = 0$ as a majority class.

$(1 + \varepsilon)$-Equal Opportunity

Equal opportunity may be far too restrictive since it requires exact equality of two probabilities. In addition, because our goal is to measure how fairness propagates through a pipeline, we propose to quantify fairness relative to a majority class with an ε factor. In fact, we propose a framework that consists of boosting the minority class in order to correct for existing bias. Therefore, we introduce the notion of $(1+\varepsilon)$-equal opportunity, which allows for compensation of inherent biases in training data.

Definition 5.1.3 ($(1 + \varepsilon)$-equal opportunity)**.** A binary predictor $\hat{Y}$ satisfies $(1 + \varepsilon)$-equal opportunity with respect to A, Y, and majority class $A = 0$ if

$$(1 + \varepsilon) \Pr\{\hat{Y} = 1|A = 0, Y = 1\} \leq \Pr\{\hat{Y} = 1|A = 1, Y = 1\},$$

where $\varepsilon \in [0, 1)$ can be any real number such that

$$(1 + \varepsilon) \Pr\{\hat{Y} = 1|A = 0, Y = 1\} \in [0, 1].$$

This definition generalizes to more than one protected class in a natural way: if $A = \{a_1, \ldots, a_m\}$ and a_m represents the majority class, then $\hat{Y}$ satisfies $((1 + \varepsilon_1), \ldots, (1 + \varepsilon_m))$-equal opportunity with respect to A, Y, and a_m, if

$$(1 + \varepsilon_t) \Pr\{\hat{Y} = 1|A = a_m, Y = 1\} \leq \Pr\{\hat{Y} = 1|A = a_t, Y = 1\}$$

for $t = 1, \ldots, m - 1$.

That is, we make sure that the true positive rates in the protected class are $1+\varepsilon$ times the rates in the majority class. The factor of $(1 + \varepsilon)$ could be determined, for example, by a Human-Resources professional or lawyer in order to correct known bias, past or present, whose mechanisms may not be fully understood. (The problem of choosing ε properly is a much more difficult control problem, possibly involving feedback, and is deferred to later work.)

5.1.3 Why pipelines?

Examples

We will now illustrate the utility of studying pipelines with a few examples. Many more examples have been identified in O'Neil [2016].

1. **Hiring.** Hiring for a job is at least a two-stage pipeline: (1) determine who to interview out of an applicant pool and (2) determine who to hire out of an interview pool. Hiring is also an example of a *filtering pipeline* since only those who have successfully got an interview are passed onto the interview stage. This pipeline can also be an example of a *informed pipeline* if the final stage gets information about the racial, gender, and age make-up of the applicant pool for instance.

2. **Criminal Justice.** Getting parole can be thought of as a three-stage pipeline: (1) compute a defendant's risk assessment score, (2) if the defendant is convicted, determine the criminal's sentencing, and (3) determine whether the criminal gets parole. This pipeline is an example of a *cumulative decisions pipeline* since the parole board has information about the risk assessment score and sentencing.

3. **Mortgages** Getting a mortgage for a home can be thought of as a *looping pipeline*. An applicant's FICO score is used to determine whether they get a mortgage. However, an applicant's FICO score is affected by prior credit and loan decisions, which also use the applicant's FICO score.

In the following, we show that (under specific circumstances) the fairness of a compound process can be guaranteed by making each link in the pipeline fair. This has the desirable implication that "global" fairness can be obtained via "local" fairness under these specific circumstances. In future work, we aim to develop a more general result that would guarantee fairness over the entire pipeline while allowing for each organization making one of the decisions in the pipeline to consider fairness only in its own decision.

5.2 Results

For the rest of the chapter, we will focus on two-stage pipelines whose decision functions are binary decision functions, i.e., $D_1 = D_2 = \{0, 1\}$. We will usually refer to such a decision function as a binary predictor $\hat{Y}$.

5.2.1 Pipeline Fairness

Consider a two-stage pipeline with binary predictors $f_1 = \hat{X}$ and $f_2 = \hat{Y}$ where again we take $A = 0$ to be the majority class and X and Y be the true target outcomes. Using the hiring scenario from above, for example, $\hat{X}$ would represent the decision about whether or not to interview a candidate and $\hat{Y}$ would represent the decision about whether or not to hire a candidate. If $X = 1$, then the candidate is qualified to get an interview; likewise, a candidate with $Y = 1$ is a good fit for the job.

We define the pipeline to be $(1 + \alpha)$-equal opportunity fair if the final decision is $(1 + \alpha)$-equal opportunity fair:

$$(1 + \alpha) \Pr\{\hat{Y} = 1 | Y = 1, A = 0\} \leq \Pr\{\hat{Y} = 1 | Y = 1, A = 1\}.$$

Assuming

1. $(1 + \varepsilon) \Pr\{\hat{X} = 1 | Y = 1, A = 0\} \leq \Pr\{\hat{X} = 1 | Y = 1, A = 1\}$, a nontrivial assumption that looks something like $(1 + \varepsilon)$-equal opportunity for the first stage in the pipeline.

2. $(1 + \delta) \Pr\{\hat{Y} = 1 | \hat{X} = 1, Y = 1, A = 0\} \leq \Pr\{\hat{Y} = 1 | \hat{X} = 1, Y = 1, A = 1\}$, that is, $(1 + \delta)$-equal opportunity for the second stage in the pipeline.

3. $\hat{Y} = 1 \implies \hat{X} = 1$.

See Section 5.1.3 for a discussion of these assumptions. Then, we have that

$$\begin{aligned}
&(1 + \varepsilon)(1 + \delta) \Pr\{\hat{Y} = 1 | Y = 1, A = 0\} \\
&= (1 + \varepsilon)(1 + \delta) \Pr\{\hat{Y} = 1, \hat{X} = 1 | Y = 1, A = 0\} \\
&= (1 + \varepsilon) \Pr\{\hat{X} = 1 | Y = 1, A = 0\} (1 + \delta) \Pr\{\hat{Y} = 1 | \hat{X} = 1, Y = 1, A = 0\} \\
&\leq \Pr\{\hat{X} = 1 | Y = 1, A = 1\} \Pr\{\hat{Y} = 1 | \hat{X} = 1, Y = 1, A = 1\} \\
&= \Pr\{\hat{Y} = 1, \hat{X} = 1 | Y = 1, A = 1\} \\
&= \Pr\{\hat{Y} = 1 | Y = 1, A = 1\}
\end{aligned}$$

Therefore,

$$(1 + \varepsilon)(1 + \delta) \Pr\{\hat{Y} = 1 | Y = 1, A = 0\} \leq \Pr\{\hat{Y} | Y = 1, A = 1\},$$

so the pipeline is $(1+\varepsilon)(1+\delta) = (1+\varepsilon+\delta+o(\varepsilon+\delta))$-equal opportunity fair and hence fairness is multiplicative over each stage under the above assumptions.

A Toy Example

To provide insight into how a $(1+\varepsilon)$-equal opportunity decision at different stages of the pipeline affect outcomes, we give an example using the two-stage hiring model pipeline.

Suppose a company wishes to interview 20 people, and hire 2 of those 20. Assume 100 applicants apply, with 90 from a majority group and 10 from a minority group. Also assume the proportion of applicants qualified for the job are equal for both groups. For this simple example, we make the strong assumption that we have very good algorithms that choose only people qualified for the job, and that there are enough qualified applicants for each scenario.

Define the interview, the first stage, as a $(1+\varepsilon)$-equal opportunity decision, and the hiring, the second stage, as a $(1+\delta)$ decision. Using the definitions from the above section with strict equality, our decisions satisfy:

1. $(1+\varepsilon)\Pr\{\hat{X}=1|Y=1,A=0\} = \Pr\{\hat{X}=1|Y=1,A=1\}$
2. $(1+\delta)\Pr\{\hat{Y}=1|\hat{X}=1,Y=1,A=0\} = \Pr\{\hat{Y}=1|\hat{X}=1,Y=1,A=1\}$

Table 5.1 and figure 5.1 provide a numerical table and visual of four scenarios. Case one and two present a situation where a perceived bias is accounted for at different stages in the pipeline. Case three presents a scenario where an attempt to fix a bias is implemented at stage one, but is counterbalanced at stage two, and case four presents the reverse circumstance.

Table 5.1: Two-stage hiring model expected count outcomes under four different cases for $\delta,\ \epsilon$ values.

Case: ε, δ	majority interviewed	minority interviewed	majority hired	minority hired
1: 2, 0	15	5	1.5	0.5
2: 0, 2	18	2	1.5	0.5
3: 2, -.666	15	5	1.8	0.2
4: -.666, 2	19.28	.71	1.8	0.2

One observation of note is that if one wishes to implement a $(1+\varepsilon)$-decision to fix a perceived bias, the final outcome is independent of the stage at which the $(1+\varepsilon)$-decision is made. Simulation shows that the variance of the final decision is also independent of the stage at which a decision is implemented.

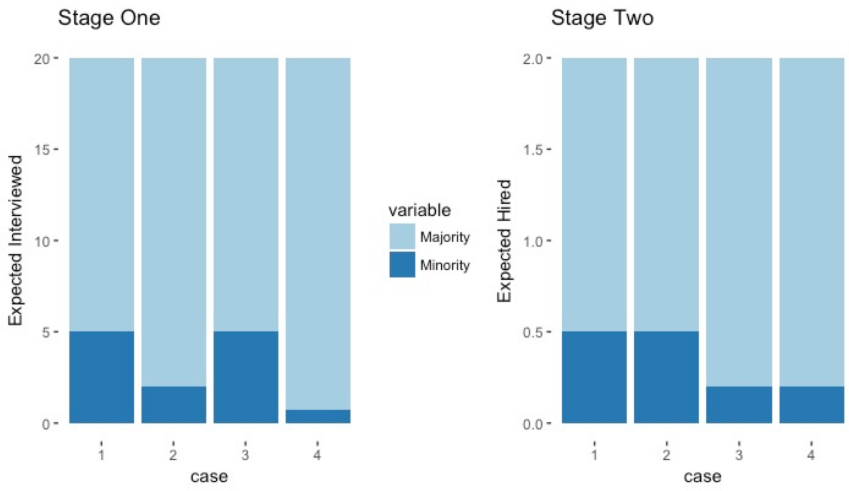


Figure 5.1: Results from two stage example. Number expected interviewed and expected hired.

This may be important for future policy, as giving preference to a minority group during the interview is perhaps more publicly acceptable than giving higher preference to minorities during hiring. Additionally, instead of giving preference to the minority group to receive more interviews from the current unbalanced applicant pool, the same outcome can be achieved by recruiting more minority applicants.

5.2.2 Where Difficulties Lie

Notice that, above we assume that fairness in the first stage of the pipeline to mean $(1+\varepsilon)\Pr\{\hat{X} = 1|Y = 1, A = 0\} \leq \Pr\{\hat{X} = 1|Y = 1, A = 1\}$ and fairness in the second stage to mean $(1+\delta)\Pr\{\hat{Y} = 1|\hat{X} = 1, Y = 1, A = 0\} \leq \Pr\{\hat{Y} = 1|\hat{X} = 1, Y = 1, A = 1\}$. Fairness in stage two fits in the framework of equal opportunity since it's a statement about an applicant getting hired given that the applicant is hiring-qualified and made it successfully through the first stage of the pipeline. Unfortunately, the first stage "fairness" assumption is a bit troublesome because it requires the first stage to make a decision based on the quality measured in the last stage. In a real world application, the first stage may be controlled by different mechanisms or goals than the last stage. In the context of a pipeline process where each portion is controlled by the same organization (or perhaps the portions are controlled by two sub-entities of the one organization), these assumptions make sense. However, there are many scenarios where this assumption will not be met.

For an example, we return to the two-stage pipeline hiring example where the first stage of the pipeline determines who gets an interview and the second stage determines who gets hired. The interview stage may only care about someone's resume to determine if they should be granted an

interview. However, it's not hard to imagine a case where a candidate is hiring qualified ($Y = 1$; they have the skills for the interview and job) but is not interview qualified ($X = 0$; their resume could be bad because they never received guidance on creating a good resume). Therefore, the issue seems to be with the specific ratio for $i = 0, 1$:

$$\frac{\Pr\{\hat{X} = 1 | X = 1, A = i\}}{\Pr\{\hat{X} = 1 | Y = 1, A = i\}}.$$

Ideally, we want these probabilities to be as close as possible so that we can decouple the pipeline. If not, being fair in the interview stage only based on interview qualifications and being fair in the hiring stage may not result in a pipeline that is fair.

5.3 Conclusion

In this chapter, we formalized the notion of a compound decision process called a pipeline, which is ubiquitous in domains like hiring, criminal justice, and finance. A pipeline decouples the final decision into intermediate decisions, which is important since although each decision affects the final outcome, different processes with different goals may be in charge of each intermediate stage. Decoupling allows us to see how fairness in each stage contributes to the fairness in the final decision.

We also modified the definition of equal opportunity to allow boosting of the minority class and showed under what assumptions of fairness on the intermediate stages of the pipeline result in an overall fair pipeline according to this definition. In this case, the fairness from each stage of the pipeline is in some sense independent so that the entire pipeline has fairness factor given by the product. On the other hand, we would like to point out if the first stage is unfair, then the second stage cannot necessarily rectify the situation. For example, if the first stage grants interviews to just one or two from a minority class, then the second stage is limited to hiring both of them, which will not result in overall fairness. Therefore, it is important to get the first stage right.

5.3.1 Future Work

We hope that our work highlights the need and sets the stage to understand compound decision making. We now give many directions for future research.

Stability

In a straight pipeline, will a small amount of unfairness or bias in the beginning turn into a large amount of unfairness at the end?

Bias as pipeline stage

In our main hiring example above, we composed two remedies to implicit bias. Alternatively, bias might be analyzed as a pipeline stage with parameter ϵ of sign opposite to the corresponding ϵ in the remedy stage.

Transparency

How transparent can a pipeline be? One method to test is whether measuring transparency by qualitative input influence Datta et al. [2016] is cumulative, and how it differs by the type of pipeline. Does one need information at each stage, or is it sufficient to have good information of only the final decision?

Variance and small pools

We can ask that the **expected** rate of positive decisions for a subclass be (approximately) proportional to that class's presence in the population. But, as with reliable randomized algorithms, we really want the outcomes to concentrate at (or near) the expectation with high probability. If a protected class is tiny, then small aberrations may cause an outcome far from the mean, with relatively large probability. In the context of pipelines, after asking whether means are preserved through pipelines, we can ask whether concentration is preserved. As in the case with randomized algorithms, it is sometimes appropriate to distort the mean to preserve concentration.

Feedback loops

More generally, some situations involve many interacting decisions, possibly with feedback loops. Under what circumstances can each decision be made autonomously, with some guarantee that the system will converge to meet some overall guarantee of fairness? For example, early in the days of long-distance running, just after women were allowed to enter major marathons without restriction, fewer women than men chose to do so at the elite level. Some race organizers—on the assumption that the sport should attract women and men equally—offered an equal **total** purse (say, for the top ten spots) for each of the men's and women's races and, since fewer women entered, those who did chased a larger expected **individual** payout. Policies like these are designed to lure more women elites the following year. What if the process is decelerated, say, by offering a purse proportional to the square root of the participation rate, e.g., with initial participation rates of $r_0 : r_1$ given by 10:90, the purse is proportional to $\sqrt{r}$, so $\sqrt{.1}$:$\sqrt{.9}$, which is 25:75? What if the process is accelerated, by offering a purse proportional to $1/r$, so $1/.1 : 1/.9$, about 9:1, reversing the participation ratio? Note that, for these types of incentives, Neither class needs to be marked

as protected, which may be desirable, though the goal of equal participation must be assumed. Typically (but depending on the strength of the economic signal to incentivize future runners), $\sqrt{r}$ has an attractive fixed point at 50:50 and $1/r$ has a repulsive fixed point at 50:50. Under appropriate assumptions, we can pose a purely mathematical question: What is the least function $g(r)$ (or give a bound on g) that leads to an attractive fixed point? By analogy with cryptography, can such systems (with or without loops) be tolerant to a bounded number of unfair players? Or bounded "total amount of unfairness," distributed among all the players and not otherwise quantified or understood?

Definitions of fairness

To return to the elite runners example, we may need refined definitions of fairness, say, "interim fairness," that captures increased year-over-year participation of an underrepresented class, and calls the improvement "interim fair" even if the system has not yet converged to a fair state. As for hiring, suppose a University department only hires faculty from a pool of recent PhDs, which is unbalanced. If the hiring process selects underrepresented faculty at a far higher rate than their presence in the PhD pool, that should be deemed "interim fair" for some purposes.

Different notions of fairness

Furthermore, we would like to understand how different definitions of fairness other than $(1+\varepsilon)$-equal opportunity propagate through a pipeline.

CHAPTER 6

Concluding Remarks and Future Work

This dissertation explored topics in sequential decision making and algorithmic fairness. We presented novel problems in established domains and proposed modeling and algorithmic solutions, supporting their performance through theoretical guarantees and empirical results. We conclude with a brief summary of the work presented within and highlight some possible areas of future work.

6.1 Contaminated Stochastic Bandits

We have presented two variants of an ε-contamination robust UCB algorithm to handle uninformative or malicious rewards in the stochastic bandit setting. As the main contribution, we proved concentration inequalities for the α-trimmed and α-shorth mean in the ε-contamination setting with sub-Gaussian samples and guarantees on the uncontaminated regret of the crUCB algorithms. The regret guarantees are similar to those in the uncontaminated stochastic multi-armed bandit setting.

We have shown through simulation that these algorithms can outperform "best of both worlds" algorithms and those for stochastic or adversarial environments when using a small number of iterations and ε chosen to be reasonable when implementing bandits in education.

We highlight that our algorithms are simple to implement. In practice, it is often easy to find upper bounds on the parameters which are robust to underestimation. Our algorithms are numerically stable and have clear intuition to their actions.

A weak point of these algorithms is they require knowledge of α before hand. Choices of α may come from domain knowledge, but could also require a separate study.

In this work we assumed a fully adaptive adversarial contamination, constrained only by the total fraction of contamination at any time step. By making more assumptions about the adversary, it is likely possible to improve uncontaminated regret bounds.

There remain many open questions in this area. In particular, we think this work could be improved along the following directions,

- **Randomized algorithms:** UCB-type algorithms are often outperformed in applications by the randomized Thompson sampling algorithm. Creating a randomized algorithm that accounts for the contamination model would increase the practicality of this line of work.
- **Contamination correlated with true rewards:** One possibility is that the contaminated rewards contain information of the true rewards. For example if contamination is missing data in a clinical trial, where we know dropout can be correlated with the treatment condition.

6.2 Algorithmic Fairness

We explored three problems in algorithmic fairness, all relating to the quality of training data. Our work on convex hull feasibility sampling, in Chapter 3, provides a method of determining whether an equally representative sample is attainable given a fixed set of sources with unknown distributions. Chapter 4 presents a method of removing historical bias present in data. Finally, Chapter 5 on fair pipelines highlights the need for representative data at each training step within a pipeline of decisions.

6.2.1 Convex Hull Feasibility Sampling

We introduce the convex hull feasibility problem in the context of fair data collection. In the Bernoulli setting, we give a lower bound on the expected sample complexity in the (x, ϵ)-infeasible instance and an oracle lower bound on the expected sample complexity in the (x, ϵ)-feasible instance. We introduce four sampling policies for the Bernoulli setting, Uniform, LUCB Mean, LUCB Ratio, and Beta TS and give high probability upper bounds on sample complexity for the Uniform and LUCB Mean policies. We give the adaptation of the Binomial policies for the multinomial case. Through simulation, we show LUCB Mean, LUCB Ratio, and the Thompson sampling policies significantly outperform Uniform in the Bernoulli and multinomial setting. In these simulations, we see that LUCB Ratio is typically the best performing policy in the Bernoulli setting, while Dirichlet TS is the best performing policy in the Multinomial setting.

While this work focused on Bernoulli and multinomial convex hull feasibility sampling, the general problem is applicable when points are drawn from any distribution for which one can construct a confidence region that meets 3.3.1.

The work in this chapter is somewhat analogous to multi-armed bandit best arm identification with fixed-confidence. Another approach seen in the best arm identification literature is the fixed-budget setting, which could also be applied to the convex hull feasibility problem. If given a set of

samples, the confidence regions can be such that they do not meet the definition of either $(1-\delta)$-confident feasible or $(1-\delta)$-confident infeasible. In this case we could ask instead what is the probability of feasibility or infeasible given the current sample, or if adaptively sampling, what is the highest probably of a correct decision when sampling with a budget. One application of this could be to check Pareto frontier feasibility, where if given noisy gradients, we ask what is the probability all groups can be improved versus the probability that improving some groups may harm others.

6.2.2 Debiasing Data

Here we studied a factor model of representations that explicitly models the contributions of the protected and permissible attributes. Based on the model, we propose an approach to debias the representations. We show that under certain conditions, we can guarantee relaxed conditional parity for the debiased representations.

6.2.3 Fair Pipelines

In this chapter, we formalized the notion of a compound decision process called a pipeline, which is ubiquitous in domains like hiring, criminal justice, and finance. A pipeline decouples the final decision into intermediate decisions, which is important since although each decision affects the final outcome, different processes with different goals may be in charge of each intermediate stage. Decoupling allows us to see how fairness in each stage contributes to the fairness in the final decision.

APPENDIX A

Convex Hull Feasibility Sampling Algorithms Appendix

A.1 Proofs

A.1.1 Lower Bounds

Proof of theorem 1. Let ν be the feasible instance. With the optimal set known, to check feasibility we only need to check the relationship between the mean of each action mean and the set x_ϵ. If the set $\mathcal{J}^*$ consists of only one point, it must be sampled enough to determine it lies within x_ϵ. If $\mathcal{J}^*$ = {1,k}, we must sample to determine that one of the means lies above $x - \epsilon$ and one lies below $x + \epsilon$.

We start with the case where $\mathcal{J}^* = \{i^*\} \in \{1, k\}$. Since this is a feasible set, it must be that $|p_i - x| < \epsilon$. The closest infeasible case is the boundary of x_ϵ closest to p_i. The KL divergence from this infeasible case is given by $\min\left(D(p_i|x-\epsilon), D(p_i|x+\epsilon)\right)$

Let $\tilde{t}_i = \max\left\{D(p_i|x-\epsilon)^{-1}, D(p_i|x+\epsilon)^{-1}\right\} \frac{1}{2}\log(\frac{1}{4\delta})$. We let N_i be the random variable representing the number of times action i was sampled when the policy terminates. We will use a proof by contradiction similar to that presented by Mannor and Tsitsiklis [2004] along with a divergence decomposition [Lattimore and Szepesvári, 2020, Lemma 15.1].

Assume $E[N_{i^*}] \leq \tilde{t}_{i^*}$. Let $O \in \{feasible, infeasible\}$ be the output of a policy, and define event $B = \{O = feasible\}$. Then by definition of a $1-\delta$-sound policy, $P_\nu(B) \geq 1 - \delta \geq 1/2$ for $\nu \in \mathcal{E}_f$. Without loss of generality, assume $x - \epsilon < p_{i^*} \leq x$. Then the closest infeasible case would be $p_{i^*} = x - \epsilon$. We will call $H_0 :\ p_{i^*} = p_{i^*},\ H_1 : p_{i^*} = x - \epsilon$. We get that,

$$
\begin{aligned}
P_1(B) &= E_1[1\{B\}] \\
&= E_0\left[\frac{L_1}{L_0}1\{B\}\right] \\
&= E_0\left[\frac{L_1}{L_0}|B\right] P_0(B) \\
&= E_0\left[\exp\left\{-\log\left(\frac{L_0}{L_1}\right)\right\}|B\right] P_0(B) \\
&\geq \exp\left\{-E_0\left[\log\left(\frac{L_0}{L_1}\right)|B\right]\right\} P_0(B) \\
&= \exp\left\{-E_0\left[N_i|B\right] D(p_{i^*}, x-\epsilon)\right\} P_0(B) \\
&\geq \exp\left\{-2\tilde{t}_{i^*} D(p_{i^*}, x-\epsilon)\right\} P_0(B) \\
&= 4\delta P_0(B) \\
&> \delta
\end{aligned}
$$

which contradicts that the policy is $1-\delta$ sound under hypothesis H_1, which means that

$$
E_0[N_{i^*}] \geq \max\left\{D(p_*|x-\epsilon)^{-1}, D(p_{i^*}|x+\epsilon)^{-1}\right\} \frac{1}{2}\log\left(\frac{1}{4\delta}\right).
$$

For $\mathcal{J}^* = \{1, k\}$, we define $H_0:\ p_1 = p_1, p_k = p_k$ and $H_1:\ P_1 = x-\epsilon$ or $p_k = p+\epsilon$. Because $p_1 \geq p_k$ by definition, if either $p_1 = x-\epsilon$ or $p_k = x+\epsilon$ then the problem is infeasible. By setting $\tilde{t}_i = \max\left\{D(p_i|x-\epsilon)^{-1}, D(p_i|x+\epsilon)^{-1}\right\} \frac{1}{2}\log(\frac{1}{4\delta})$ and following the same method as above for actions $i \in \{1, k\}$, get

$$
E_0[T_i] \geq \min\left\{D(p_i|x-\epsilon)^{-1}, D(p_i|x+\epsilon)^{-1}\right\} \frac{1}{2}\log\left(\frac{1}{4\delta}\right).
$$

□

Proof sketch of theorem 2. The proof for the infeasible lower bound follows closely to that of the feasible case, therefore we provide a brief proof outline. Because all means must lie outside x_ϵ, the closest feasible case is the boundary of x_ϵ nearest the means. To determine infeasibility, all actions must be sampled sufficiently to reject this boundary. Without loss of generality, if we assume $p_i < x-\epsilon$, then the closest boundary would be $x-\epsilon$. Setting $H_i: p_i = p_i$, $H_1: p_i = x-\epsilon$ for all i, $\tilde{t}_i = \max\left\{D(p_i|x-\epsilon)^{-1}, D(p_i|x+\epsilon)^{-1}\right\} \frac{1}{2}\log(\frac{1}{4\delta})$ and following the methods from the feasible case, we get the desired result. □

A.1.2 Upper Bounds

Proof of theorem 3. Given some δ and $B(n, \delta)$ that satisfies equation (3.3.1), let event E be the event that all confidences regions contain their mean, $E = \{\forall i \in [k], n \in \mathbb{N},\ \hat{p}_i(n) - B(n, \delta) \leq p_i \leq \hat{p}_i(n) + B(n, \delta)\}$. Under event E, each action i will become inactive at or before being sampled s_i^{min} times.

We start with the feasible cases. When where $\mathcal{J}^* = \{i^*\}$ and under event E, action i^* can be sampled at most $s_{i^*}^{min}$ times before the policy will terminate due to stopping rule 1. Thus the bound on the sample size of each action is the minimum of the sample size it is guaranteed to become inactive under E, which is s_i^{min}, and the sample size of $s_{i^*}^{min}$ when the policy terminates. Since event E happens with probability at least $1 - \delta$, this concludes the proof when the optimal subset is one action.

In the case where $\mathcal{J}^* = \{1, k\}$, under event E the policy will terminate due to stopping rule 1, which will happen when action 1 is sampled s_1^{max} times and action k is sampled s_k^{max} times. Again, under E an action becomes inactive when sampled at most s_i^{min} times, this gives that each action is sampled at most $(s_i^{min}, \max(s_1^{max}, s_k^{max}))$ times. Again, event E happens with probability at least $1 - \delta$.

When the problem is infeasible, each action will be sampled until its confidence region is disjoint form x_ϵ. Under event E, this sample size is bounded above by s_i^{min} for all i. □

Proof of theorem 4. We Start with the feasible case where there exists a mean on both sides of x. Let event E be the event that all confidences regions contain their mean, $E = \{\forall i \in [k], n \in \mathbb{N},\ \hat{p}_i(n) - B(n, \delta) \leq p_i \leq \hat{p}_i(n) + B(n, \delta)\}$. Without loss of generality, let actions i, j be the action that triggers termination at time τ. Define the sample size of action l at time t as $N_l(t)$. Assume without loss of generality that $j^* = k$, and $p_j < x$, thus $i^* = 1$ and $p_i > x$. If $j = k$, then under event E, $N_j(\tau) \leq s_k^{max}$. If $j \neq k$ it must be that,

$$\begin{aligned} \hat{p}_j(N_j(\tau) - 1) - B(N_j(\tau) - 1) &\leq p_k && \text{by } E \\ \hat{p}_j(N_j(\tau) - 1) + B(N_j(\tau) - 1) &\geq x + \epsilon && \text{definition of } \tau \end{aligned}$$

Therefore

$$\begin{aligned}
\hat{p}_j(N_j(\tau)-1) + B(N_j(\tau)-1) &\geq x+\epsilon \\
2B(N_j(\tau)-1) &\geq x+\epsilon-(\hat{p}_j(N_j(\tau)-1) - B(N_j(\tau)-1)) \\
&\geq x+\epsilon-p_k \\
&= \Delta_k^{max} \\
&> 2B(s_k^{max})
\end{aligned}$$

since B is a decreasing function, $N_j(\tau)-1 < s_k^{max} \implies N_j(\tau) \leq s_k^{max}$. Similarly for action i we have that $N_j(\tau) \leq s_1^{max}$

For non-terminating actions we have that under E,

$$\hat{p}_l(N_l(\tau)-1) - B(N_l(\tau)-1) \leq p_k \qquad \hat{p}_l(N_l(\tau)-1) + B(N_l(\tau)-1) \geq \max(x+\epsilon, p_l)$$

which implies that

$$2B(N_l(\tau)-1) \geq \max(|p_k-(x+\epsilon)|, |p_l-p_k|) = \max(\Delta_k^{max}, \Delta_{l,j}) > \max(2B(s_k^{max}), 2B(s_{k,l}))$$

giving $N_l(\tau) \leq \min(s_k^{max}, s_{k,l})$ and similarly $N_l(\tau) \leq \min(s_1^{max}, s_{1,l})$. To meet both these bounds, it must be that $N_l(\tau) \leq \max(\min(s_1^{max}, s_{1,l}), \min(s_k^{max}, s_{k,l})) = \max(\min(s_{i^*}^{max}, s_{l,i^*}), \min(s_{j^*}^{max}, s_{l,j^*}))$.

When $s_{j^*}^{max} \leq s_{l,j^*}$, which is equivalent to $\Delta_{j^*}^{max} \geq \Delta_{l,j^*}$, then $s_{j^*}^{max} \geq s_{i^*}^{max}$ by definition. If $s_{i^*}^{max} \geq s_{l,i^*}$, then we have that

$$\begin{aligned}
\Delta_{l,i^*} &= \Delta_{i^*} + \Delta_l^{min} \\
&\geq \Delta_{j^*}^{max} + \Delta_l^{min} \\
&\geq \Delta_{j^*}^{max}
\end{aligned}$$

Thus $s_{j^*}^{max} \geq s_{l,i^*}$. So when $\Delta_{j^*}^{max} \geq \Delta_{l,j^*}$, the sample size of action l is bounded above by $s_{j^*}^{max}$.

When $s_{j^*}^{max} > s_{l,j^*}$, which is equivalent to $\Delta_{j^*}^{max} < \Delta_{l,j^*}$, it must be that p_{i^*} and p_l are on the same side of x_ϵ, thus $s_{i^*}^{max} < s_{l,i^*}$ and the sample size of action i is bounded above by $\max(s_{l,j^*}, s_{i^*}^{max})$.

When $\mathcal{J}^* = l^*$. there are two scenarios. Either all means lies in x_ϵ, or all means lie in one direction from x_ϵ. In the first case, the outcome is the same as above. In the second case, it must be that the optimal action mean must be sample at most $s_{l^*}^{min}$ times, at which point it would trigger

termination under E. Let j be the action that triggered stopping rule 1. Without loss of generality, assume $p_{l^*} < x$. If $j = l^*$, then under E, $N_j(\tau) \leq s_{l^*}^{min}$. If $j \neq l^*$,

$$\begin{aligned} \hat{p}_j(N_j(\tau)-1) + B(N_j(\tau)-1) &\geq p_{l^*} && \text{by } E \\ \hat{p}_j(N_j(\tau)-1) - B(N_j(\tau)-1) &\leq x - \epsilon && \text{definition of } \tau \end{aligned}$$

and as in the feasible case,

$$\begin{aligned} \hat{p}_j(N_j(\tau)-1) - B(N_j(\tau)-1) &\leq x - \epsilon \\ 2B(N_j(\tau)-1) &\geq \hat{p}_j(N_j(\tau)-1) - B(N_j(\tau)-1) - (x+\epsilon) \\ &\geq p_{l^*} - (x-\epsilon) \\ &= \Delta_{l^*}^{min} \\ &> 2B(s_{l^*}^{min}) \end{aligned}$$

which gives that $N_j(\tau) \leq s_{l^*}^{min}$. For all other actions $l \neq j$,

$$\hat{p}_l(N_l(\tau)-1) + B(N_l(\tau)-1) \geq p_j \qquad \hat{p}_l(N_l(\tau)-1) - B(N_l(\tau)-1) \leq \min(x-\epsilon, p_l)$$

and using the same logic from above gives $N_l(\tau) \leq \min(s_{l^*}^{min}, s_{l,l^*})$.

In the infeasible case, assume without loss of generality that $p_i \leq x_\epsilon$ for all $i \in [k]$. Under E, it must be that $\hat{p}_i(N_i(\tau)-1) - B(N_i(\tau)-1) \leq p_i$ and $\hat{p}_i(N_i(\tau)-1) - B(N_i(\tau)-1) \geq x - \epsilon$ for all $i \in [k]$. Therefore

$$\begin{aligned} 2B(N_i(\tau)-1) &\geq x - \epsilon - p_i \\ &= \Delta_i^{min} \\ &> 2B(s_i^{min}) \end{aligned}$$

and $N_i(\tau) \leq s_i^{min}$. Since E happens with probability at least $1-\delta$, this concludes the proof. $\square$

A.1.3 Upper bound improvement of LUCB Mean over Uniform

We start with the case where $\exists i, j,\ p_i < x < p_j,\ |\mathcal{J}^*| = 2$. If $\Delta_{l,j^*} \leq \Delta_{j^*}^{max}$ then $\min(s_{j^*}^{max}, s_l^{min}) \geq s_{j^*}^{max}$, because

$$\begin{aligned}\Delta_l^{min} &\leq \begin{cases} \Delta_{j^*}^{min} & p_l \notin x_\epsilon \\ \epsilon & p_l \in x_\epsilon \end{cases} \\ &< \Delta_{j^*}^{max}\end{aligned}$$

Therefore $\min(s_{j^*}^{max}, s_l^{min}) \geq s_{j^*}^{max}$ when $\Delta_{l,j^*} \leq \Delta_{j^*}^{max}$.

If $\Delta_{l,j^*} > \Delta_{j^*}^{max}$, then p_l is on the same side of x_ϵ as p_{i^*}. To show $\min(s_{j^*}^{max}, s_l^{min}) \geq \max(s_{i^*}^{max}, s_{l,j^*})$, we have that,

$$\begin{aligned}&\Delta_l^{min} < \Delta_{j^*}^{max} + \Delta_l^{min} = \Delta_{l,j^*} \\ &\Delta_l^{min} < \Delta_{i^*}^{max} && \text{by definition}\end{aligned}$$

and by definition

$$\begin{aligned}\Delta_{j^*}^{max} &< \Delta_{i^*}^{max} \\ \Delta_{j^*}^{max} &< \Delta_{l,j^*}\end{aligned}$$

Since $\min(s_{j^*}^{min}, s_l^{min}) \geq \min(s_{j^*}^{max}, s_l^{min}) \geq \max(s_{i^*}^{max}, s_{l,j^*})$ this covers the $|\mathcal{J}^*| = 1$ case as well.

When all means are on one side of x, then $|\mathcal{J}^*| = 1$ and must show $\min(s_{j^*}^{min}, s_l^{min}) \geq \min(s_{l,j^*}, s_{j^*}^{min})$. If $\Delta_{j^*}^{max} \leq \Delta_l^{min}$ then we have,

$$\Delta_l^{min} < \Delta_{j^*}^{min} + \Delta_{l^*}^{min} = \Delta_{l,j^*}$$

We have therefore shown in all cases, LUCB Mean has a lower high probability upper bound on sample complexity in the (x, ϵ)-feasible Bernoulli setting than Uniform.

APPENDIX B

Contamination Robust Bandits Appendix

B.1 Proofs

B.1.1 Theorem 2.4.1.1

[Trimmed mean concentration] Let G be the set of points $x_1, ...x_n \in \mathbb{R}$ that are drawn from a σ-sub-Gaussian distribution with mean μ. Let S_n be a sample where an ε-fraction of these points are contaminated by an adversary. For $\varepsilon \leq \alpha < 1/2$, $t \geq n$ we have,

$$|\text{trMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{(1-2\alpha)}\left(\sqrt{\frac{4}{n}\log(t)} + 4\alpha\sqrt{6\log(t)}\right)$$

with probability at least $1 - \frac{4}{t^2}$.

Proof of section 2.4.1.1. Without loss of generality assume $\mu = 0$ for the underlying true distribution. For $X \sim \sigma$-sub-Gaussian, by definition, we have:

$$P\Big(|X| \geq \mu + \eta\Big) \leq 2\exp(-\frac{\eta^2}{2\sigma^2})$$
$$P\left(|\bar{x}_n - \mu| \geq \sigma\sqrt{\frac{2}{n}\log\frac{2}{\delta_1}}\right) \leq \delta_1$$

and

$$P\Big(\max_{i\in[n]}|X_i| \geq t\Big) \leq 2n\exp\left(-\frac{t^2}{2\sigma^2}\right)$$
$$P\Big(\max_{i\in[n]}|X_i| \geq \sigma\sqrt{2\log\frac{2n}{\delta_2}}\Big) \leq \delta_2.$$

Let $\tilde{G} \subset G_n$ represent the points which are not contaminated and $C \subset G_n$ represent the contaminated points. Then our sample can be represented by the union $S_n = \tilde{G} \cup C$. Let R represent the points that remain after trimming α fraction of the largest and smallest points, and T be the set of points that were trimmed. Then we have that.

$$\begin{aligned}
|\text{trMean}_\alpha(S_n)| &= \left| \frac{1}{(1-2\alpha)n} \sum_{x \in R} x \right| \\
&= \frac{1}{(1-2\alpha)n} \left| \sum_{x \in \tilde{G} \cap R} x + \sum_{x \in C \cap R} x \right| \\
&\leq \frac{1}{(1-2\alpha)n} \left| \underbrace{\sum_{x \in \tilde{G}} x}_{A_1} - \underbrace{\sum_{x \in \tilde{G} \cap T} x}_{A_2} + \underbrace{\sum_{x \in C \cap R} x}_{A_3} \right| \\
&\leq \frac{1}{(1-2\alpha)n} \left(\left| \underbrace{\sum_{x \in \tilde{G}} x}_{A_1} \right| + \left| \underbrace{\sum_{x \in \tilde{G} \cap T} x}_{A_2} \right| + \left| \underbrace{\sum_{x \in C \cap R} x}_{A_3} \right| \right)
\end{aligned}$$

with

$$A_1 = \left| \sum_{x \in G_n} x - \sum_{x \in G_n \setminus \tilde{G}} x \right| \leq \left| \sum_{x \in G_n} x \right| + \left| \sum_{x \in G_n \setminus \tilde{G}} x \right| \leq n|\bar{x}_{G_n}| + \varepsilon n \max_{x \in G_n} |x| \quad \text{w.p. at least } 1 - \delta_1 - \delta_2,$$

$$A_2 \leq 2\alpha n \max_{x \in G_n} |x| \quad \text{w.p. at least } 1 - \delta_2,$$

$$A_3 \leq \varepsilon n \max_{x \in G_n} |x| \quad \text{w.p. at least } 1 - \delta_2.$$

Combining we get,

$$\begin{aligned}
|\text{trMean}_\alpha(S_n) - \mu| &\leq \frac{1}{(1-2\alpha)} \left(|\bar{x}_{G_n}| + \max_{x \in G_n} |x| (2\varepsilon + 2\alpha) \right) \\
&\leq \frac{1}{(1-2\alpha)} \left(|\bar{x}_{G_n}| + \max_{x \in G_n} |x| (4\alpha) \right) \\
&\leq \frac{\sigma}{(1-2\alpha)} \left(\sqrt{\frac{2}{n} \log \frac{2}{\delta_1}} + 4\alpha \sqrt{2 \log \frac{2t}{\delta_2}} \right)
\end{aligned}$$

with probability at least $1 - \delta_1 - \delta_2$. Letting $\delta_1 = \frac{2}{t^2}$ and $\delta_2 = \frac{2}{t^2}$, and assuming $\alpha \geq \varepsilon$, we have,

$$|\text{trMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{(1-2\alpha)} \left(\sqrt{\frac{4}{n} \log(t)} + 4\alpha \sqrt{6 \log(t)} \right)$$

with probability at least $1 - \frac{4}{t^2}$. □

B.1.2 Theorem 2.4.1.2

[α-shorth mean concentration] Let G_n be the set of points $x_1, ...x_n \in \mathbb{R}$ that are drawn from a σ-sub-Gaussian distribution with mean μ. Let S_n be a sample where an ε-fraction of these points are contaminated by an adversary. For $\varepsilon \leq \alpha < 1/3$, $t \geq n$, we have,

$$|\text{sMean}_\alpha(S_n) - \mu| \leq \\ \frac{\sigma}{1-2\alpha}\sqrt{\frac{4}{n}\log t} + \frac{(6\alpha - 8\alpha^2)\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{6\log t}$$

with probability at least $1 - \frac{4}{t^2}$.

Proof of section 2.4.1.2. Without loss of generality assume $\mu = 0$ for the underlying true distribution. Let $X \sim \sigma$-sub-Gaussian.

We want to bound the impact of the contaminated points in our interval. Once we have this bound, the proof follows just as in the trimmed mean.

Assume $\alpha < 1/3$ and $\varepsilon \leq \alpha$. Let J be the interval that contains the shortest $1 - \alpha$ fraction of S_n, I be the interval that contains $\tilde{G}$ (i.e. the remaining good points after contamination), and T be the interval that contains the points of S_n after trimming the α largest and smallest fraction of points. Use $|I|$ to denote the length of interval I. It must be that $I \cap J \neq \emptyset$ because otherwise the points in $I \cup J$ would contain $2 - 2\alpha > 1$ fraction of S_n. Let c be a point in $I \cap J$ and x be a point in J. Recall that $\text{trMean}_\alpha(S_n)$ is the trimmed mean of the contaminated sample S_n from above. Then we have,

$$\begin{aligned} |x| &\leq |x - c| + |c - \text{trMean}_\alpha(S_n)| + |\text{trMean}_\alpha(S_n)| \\ &\leq |J| + |I| + |\text{trMean}_\alpha(S_n)| \\ &\leq 2|I| + |\text{trMean}_\alpha(S_n)| \end{aligned}$$

The second step comes from x and c both being in J and because $I \supseteq T$. The third step comes from $|J| \leq |I|$.

To bound the length of I we have,

$$|I| \leq 2 \max_{x \in G_n} |x| \text{ w.p. at least } 1 - \delta_2.$$

Finally, since

$$|\text{trMean}_\alpha(S_n)| \leq \frac{1}{(1-2\alpha)}(|\bar{x}_{G_n}| + 4\alpha \max_{x \in G_n} |x|)$$

with probability at least $1-\delta_1-\delta_2$, we get that for $x \in J$,

$$\begin{aligned}|x| &\le 4\max_{x\in G_n}|x| + \frac{1}{(1-2\alpha)}(|\bar{x}_{G_n}| + 4\alpha\max_{x\in G_n}|x|) \qquad \text{w.p. at least } 1-\delta_1-\delta_2,\\ &= \frac{|\bar{x}_{G_n}|}{1-2\alpha} + \Big(4+\frac{4\alpha}{1-2\alpha}\Big)\max_{x\in G_n}|x|.\end{aligned}$$

Now that we have a bound on the contaminated points in J, our analysis follows as before,

$$\begin{aligned}&|\text{sMean}_\alpha(S_n)|\\ &\qquad \le \frac{1}{(1-\alpha)n}\Bigg(\Big|\underbrace{\sum_{x\in\tilde{G}} x}_{A_1}\Big| + \Big|\underbrace{\sum_{x\in\tilde{G}\cap\neg J} x}_{A_2}\Big| + \Big|\underbrace{\sum_{x\in C\cap J} x}_{A_3}\Big|\Bigg)\end{aligned}$$

where

$$\begin{aligned}A_1 &\le n|\bar{x}_{G_n}| + \varepsilon n\max_{x\in G_n}|x| && \text{w.p. at least } 1-\delta_1-\delta_2,\\ A_2 &\le \alpha n\max_{x\in G_n}|x| && \text{w.p. at least } 1-\delta_2,\\ A_3 &\le \varepsilon n\left(\frac{|\bar{x}_{G_n}|}{1-2\alpha} + \Big(4+\frac{4\alpha}{1-2\alpha}\Big)\max_{x\in G_n}|x|\right) && \text{w.p. at least } 1-\delta_1-\delta_2.\end{aligned}$$

Combining we get,

$$\begin{aligned}&|\text{sMean}_\alpha(S_n)-\mu|\\ &\le \frac{1}{(1-\alpha)}\left(|\bar{x}_{G_n}|\big(1+\frac{\varepsilon}{1-2\alpha}\big) + \max_{x\in G_n}|x|\big(5\varepsilon+\alpha+\frac{4\alpha\varepsilon}{1-2\alpha}\big)\right)\\ &\le \frac{1}{(1-\alpha)}\left(|\bar{x}_{G_n}|\big(\frac{1-\alpha}{1-2\alpha}\big) + \max_{x\in G_n}|x|\big(6\alpha+\frac{4\alpha^2}{1-2\alpha}\big)\right)\\ &= \frac{1}{1-\alpha}\left(|\bar{x}_{G_n}|\big(\frac{1-\alpha}{1-2\alpha}\big) + \max_{x\in G_n}|x|\frac{6\alpha-8\alpha^2}{1-2\alpha}\right)\\ &\le \frac{\sigma}{1-2\alpha}\sqrt{\frac{2}{n}\log\frac{2}{\delta_1}} + \frac{(6\alpha-8\alpha^2)\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{2\log\frac{2t}{\delta_2}}\end{aligned}$$

With probability at least $1-\delta_1-\delta_2$. Letting $\delta_1 = \frac{2}{t^2}$ and $\delta_2 = \frac{2}{t^2}$, and assuming $\alpha \ge \varepsilon$, we have,

$$\begin{aligned}&|\text{sMean}_\alpha(S_n)-\mu|\\ &\qquad \le \frac{\sigma}{1-2\alpha}\sqrt{\frac{4}{n}\log t} + \frac{(6\alpha-8\alpha^2)\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{6\log t}\end{aligned}$$

With probability at least $1 - \frac{4}{t^2}$. $\square$

B.1.3 Theorem 2.4.2

[α-trimmed mean crUCB uncontaminated regret] Let $K > 1$ and $T \geq K-1$. Then with algorithm 4 with the α-trimmed mean, σ-sub-Gaussian reward distributions with $\sigma_a \leq \sigma_0$, and contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{min}}{4(\Delta_{min}+4\sigma_0\sqrt{6\log T})}$, we have the uncontaminated regret bound,

$$\bar{R}(UCB) \leq 8\sigma_0\sqrt{KT\log T} + \sum 15\Delta_a.$$

Proof of section 2.4.2. First will show that $\mathbb{E}[N_a(t)] < \infty$ for non-optimal actions. Assume $N_a(t) \geq \frac{64\sigma_0^2\log(T)}{\Delta_a^2}$.

$$\begin{aligned}
&\hat{\mu}_a + \frac{\sigma_0}{(1-2\alpha)}\left(\sqrt{\frac{4}{N_a(t)}\log t} + 4\alpha\sqrt{6\log(t)}\right) \\
&\leq \mu_a + \frac{\sigma_i + \sigma_0}{(1-2\alpha)}\left(\sqrt{\frac{4}{N_a(t)}\log t} + 4\alpha\sqrt{6\log(t)}\right) && \text{w.p. at least } 1 - \frac{4}{t^2} \\
&\leq \mu^* - \Delta_a + \frac{2\sigma_0}{(1-2\alpha)}\left(\sqrt{\frac{4}{N_a(t)}\log t} + 4\alpha\sqrt{6\log(t)}\right) \\
&\leq \mu^* - \Delta_a + \frac{\Delta_a}{2(1-2\alpha)} + \frac{2\sigma_0 4\alpha}{(1-2\alpha)}\sqrt{6\log t} && N_a(t) \geq \frac{64\sigma_0^2\log(T)}{\Delta_a^2} \\
&\leq \mu^* && \alpha \leq \frac{\Delta_a}{4(\Delta_a + 4\sigma_0\sqrt{6\log(t)})} \\
&\leq \hat{\mu}^* + \frac{\sigma_{i^*}}{(1-2\alpha)}\left(\sqrt{\frac{4}{N^*(t)}\log t} + 4\alpha\sqrt{6\log(t)}\right) && \text{w.p. at least } 1 - \frac{4}{t^2} \\
&\leq \hat{\mu}^* + \frac{\sigma_0}{(1-2\alpha)}\left(\sqrt{\frac{4}{N^*(t)}\log t} + 4\alpha\sqrt{6\log(t)}\right).
\end{aligned}$$

Now to find $\mathbb{E}[N_a(T)]$ for non-optimal actions.

$$\begin{aligned}
\mathbb{E}[N_a(T)] &= 1 + \mathbb{E}\Big[\sum_{t=K+1}^{T} \mathbf{1}\{A_t = a\}\Big] \\
&= 1 + \mathbb{E}\Big[\sum_{t=K+1}^{T} \mathbf{1}\Big\{A_t = a, N_a(t) \leq \frac{64\sigma_0^2 \log(T)}{\Delta_a^2}\Big\} + \mathbf{1}\Big\{A_t = a, N_a(t) > \frac{64\sigma_0^2 \log(T)}{\Delta_a^2}\Big\}\Big] \\
&\leq 1 + \frac{64\sigma_0^2 \log(T)}{\Delta_a^2} + \sum_{t=K+1}^{T} \mathbb{P}\Big[A_t = a, N_a(t) > \frac{64\sigma_0^2 \log(T)}{\Delta_a^2}\Big] \\
&= 1 + \frac{64\sigma_0^2 \log(T)}{\Delta_a^2} + \sum_{t=K+1}^{T} \mathbb{P}\Big[A_t = a | N_a(t) > \frac{64\sigma_0^2 \log(T)}{\Delta_a^2}\Big]\mathbb{P}\Big[N_a(t) > \frac{64\sigma_0^2 \log(T)}{\Delta_a^2}\Big] \\
&\leq 1 + \frac{64\sigma_0^2 \log(T)}{\Delta_a^2} + \sum_{t=K+1}^{T} \frac{8}{t^2} \\
&\leq \frac{64\sigma_0^2 \log(T)}{\Delta_a^2} + 15.
\end{aligned}$$

Finally, we can find the regret following the standard analysis,

$$\begin{aligned}
\bar{R} &= \sum_{a=2}^{K} \Delta_a \mathbb{E}[N_a(T)] \\
&= \sum_{\Delta_a < \Delta} \Delta_a \mathbb{E}[N_a(T)] + \sum_{\Delta_a \geq \Delta} \Delta_a \mathbb{E}\Big[N_a(T)\Big] \\
&\leq \Delta T + \sum_{\Delta_a \geq \Delta} \big[\frac{64\sigma_0^2 \log(T)}{\Delta_a} + 15\Delta_a\big] && \mathbb{E}[N_a(t)] \leq \frac{64\sigma_0^2 \log(T)}{\Delta_a} + 15 \\
&\leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a && \Delta = \sqrt{\frac{64K\sigma_0^2 \log(T)}{T}}.
\end{aligned}$$

□

B.1.4 Corollary 2.4.2

[α-trimmed mean crUCB uncontaminated regret bounded rewards] If the rewards are bounded by b, and have contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{\min}}{4(\Delta_{\min}+4b)}$, then

$$\bar{R}_T \leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a.$$

Proof of section 2.4.2. By replacing the part of the concentration bound for the trimmed mean that

is based on the maximum value in the sample with b, we get that,

$$|\text{trMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{(1-2\alpha)}\sqrt{\frac{4}{n}\log(t)} + \frac{4\alpha}{1-2\alpha}b$$

with probability at least $1 - \frac{4}{t^2}$.

First will show that $\mathbb{E}[N_a(t)] < \infty$ for non-optimal actions. Assume $N_a(t) \geq \frac{64\sigma_0^2\log(T)}{\Delta_a^2}$.

$$\begin{aligned}
&\hat{\mu}_a + \frac{\sigma_0}{(1-2\alpha)}\sqrt{\frac{4}{N_a(t)}\log t} + \frac{4\alpha}{1-2\alpha}b \\
&\leq \mu_a + \frac{\sigma_i + \sigma_0}{(1-2\alpha)}\sqrt{\frac{4}{N_a(t)}\log t} + \frac{8\alpha}{1-2\alpha}b && \text{w.p. at least } 1 - \frac{4}{t^2} \\
&\leq \mu^* - \Delta_a + \frac{2\sigma_0}{(1-2\alpha)}\sqrt{\frac{4}{N_a(t)}\log t} + \frac{8\alpha}{1-2\alpha}b \\
&\leq \mu^* - \Delta_a + \frac{\Delta_a}{2(1-2\alpha)} + \frac{8\alpha}{(1-2\alpha)}b && N_a(t) \geq \frac{64\sigma_0^2\log(T)}{\Delta_a^2} \\
&\leq \mu^* && \alpha \leq \frac{\Delta_a}{4(\Delta_a + 4b)} \\
&\leq \hat{\mu}^* + \frac{\sigma_{i^*}}{(1-2\alpha)}\sqrt{\frac{4}{N^*(t)}\log t} + \frac{4\alpha}{1-2\alpha}b && \text{w.p. at least } 1 - \frac{4}{t^2} \\
&\leq \hat{\mu}^* + \frac{\sigma_0}{(1-2\alpha)}\sqrt{\frac{4}{N^*(t)}\log t} + \frac{4\alpha}{1-2\alpha}b.
\end{aligned}$$

Results follow with a similar analysis as above. □

B.1.5 Theorem 2.4.2

[α-shorth mean crUCB uncontaminated regret] Let $K > 1$ and $T \geq K - 1$. Then with algorithm 4 with the α-shorth mean, sub-Gaussian reward distributions with $\sigma_a \leq \sigma_0$, and contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{min}}{4(\Delta_{min} + 9\sigma_0\sqrt{6\log T})}$, we have the uncontaminated regret bound,

$$\bar{R}(UCB) \leq 8\sigma_0\sqrt{KT\log T} + \sum 15\Delta_a.$$

Proof of section 2.4.2. The proof for the contamination robust UCB using the α-shorth mean is similar to that of the trimmed mean.

$$
\begin{aligned}
&\hat{\mu}_a + \frac{\sigma_0}{1-2\alpha}\sqrt{\frac{4}{N_a(t)}\log t} + \frac{(6\alpha - 8\alpha^2)\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{6\log t} \\
&\leq \mu^* - \Delta_a + \frac{2\sigma_0}{1-2\alpha}\sqrt{\frac{4}{N_a(t)}\log t} + 2\frac{(6\alpha - 8\alpha^2)\sigma_0}{(1-2\alpha)(1-\alpha)}\sqrt{\log t} \quad \text{w.p.a.l } 1 - \frac{4}{t^2} \\
&\leq \mu^* - \Delta_a + \frac{\Delta_a}{2(1-2\alpha)} + \frac{18\alpha\sigma_0}{(1-2\alpha)}\sqrt{6\log t} \qquad N_a(t) \geq \frac{64\sigma_0^2\log(t)}{\Delta_a^2},\ \alpha < 1/3 \\
&\leq \mu^* \qquad \alpha \leq \frac{\Delta_a}{4(\Delta_a + 9\sigma_0\sqrt{6\log t})} \\
&\leq \hat{\mu}^* + \frac{\sigma_0}{1-2\alpha}\sqrt{\frac{4}{N^*(t)}\log t} + \frac{6\alpha - 8\alpha^2\sigma}{(1-2\alpha)(1-\alpha)}\sqrt{6\log t}
\end{aligned}
$$

Using the analysis from the trimmed mean regret, we again get,

$$\mathbb{E}[N_a(t)] \leq \frac{64\sigma_0^2\log T}{\Delta_a} + \sum 15\Delta_a$$

Using this value and standard regret analysis yields

$$\bar{R}_T \leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a.$$

□

B.1.6 Corollary 2.4.2

[α-shorth mean crUCB uncontaminated regret bounded rewards] If the rewards are bounded by b, and have contamination rate $\varepsilon \leq \alpha \leq \frac{\Delta_{\min}}{4(\Delta_{\min}+9b)}$, then

$$\bar{R}_T \leq 8\sigma_0\sqrt{KT\log(T)} + \sum 15\Delta_a.$$

Proof of section 2.4.2. By replacing the part of the concentration bound for the trimmed mean that is based on the maximum value in the sample with b, we get that,

$$|\text{sMean}_\alpha(S_n) - \mu| \leq \frac{\sigma}{1-2\alpha}\sqrt{\frac{4}{n}\log t} + \frac{6\alpha - 8\alpha^2}{(1-2\alpha)(1-\alpha)}b$$

With probability at least $1 - \frac{4}{t^2}$.

Follow similar analysis as in appendix B.1.4 but setting constraint to be,

$$\varepsilon \leq \alpha \leq \frac{\Delta_{\min}}{4(\Delta_{\min} + 9b)}$$

□

B.2 Relationship of ε and Δ_{min}

One quick example showing that $\varepsilon > \Delta_{\min}$ can prohibit sublinear regret is to consider the CSB game with two actions and Bernoulli rewards. If $a_1 \sim B(p)$ and $a_2 \sim B(p - \varepsilon)$ then an adversary can choose all the contaminated rewards for a_2 to be 1 making it appear that $a_2 \sim B(p)$. Thus the actions are indistinguishable to the learner.

However, we can still provide a bound for larger values of ε provided one is willing to tolerate a linear term in the regret. We outline the argument only for the trimmed mean case since the argument for the shorth mean is very similar. Note that argument for bounding $\mathbb{E}[N_a(T)]$ in Section 2.4.2 works under the condition

$$\alpha \leq \frac{\Delta_a}{4(\Delta_a + 4\sigma_0\sqrt{6\log(T)})}.$$

Let $\mathcal{S}$ be the set of actions satisfying this condition. The arguments in the proof of Section 2.4.2 show that

$$\sum_{a>1, a\in\mathcal{S}} \Delta_a \mathbb{E}[N_a(T)] \leq 8\sigma_0\sqrt{KT\log(T)} + \sum_{a>1, a\in\mathcal{S}} 15\Delta_a.$$

Therefore the bound of $\tilde{O}(\sigma_0\sqrt{KT})$ holds only for the regret due to actions $a \in \mathcal{S}$. For any action $a \notin \mathcal{S}$, we have

$$\Delta_a < \frac{16\alpha\sigma_0\sqrt{6\log(T)}}{1 - 4\alpha}$$

assuming $\alpha < 0.25$. The total regret contribution for $a \notin \mathcal{S}$ is therefore

$$\begin{aligned}\sum_{a>1, a\notin\mathcal{S}} \Delta_a \mathbb{E}[N_a(T)] &\leq \frac{16\alpha\sigma_0\sqrt{6\log(T)}}{1 - 4\alpha} \sum_{a>1, a\notin\mathcal{S}} \mathbb{E}[N_a(T)] \\ &\leq \frac{16\alpha\sigma_0\sqrt{6\log(T)}}{1 - 4\alpha} T\end{aligned}$$

So the total regret is $\tilde{O}(\sqrt{KT} + \frac{\alpha}{1-4\alpha}T)$.

APPENDIX C

Exploration of Effects on Various Fairness Violations When Optimizing Fair Data Collection

C.1 Introduction

Collecting good data is a high priority when developing any algorithmic decision policy. Poor data leads to poor decisions that when implemented can have serious negative consequences on peoples' lives. The field of fair algorithms has begun to consider sampling strategies to collect data that optimizes upon some desirable fairness definition Abernethy et al. [2020], Shekhar et al. [2021], Asudeh et al. [2019]. These strategies have been shown to be effective for fixed fairness definitions. This exploration considers the impact of fairness outcomes for metrics not used for optimization in the sampling procedure. This represents the situation where the measurement used for the desired fairness changes over the course of a project. The desired fairness at the beginning of a project may not be fixed, and just like parameters in a training algorithm they may be adjusted over the course of a project before implementation.

With this in mind, we explore the fairness violation outcomes defined by multiple fairness measurements when optimized for a single fairness measure. We additionally compare the results to a data set with equal representation among the protected groups.

C.2 Sampling Algorithms

We selected two methods to implement, those given in Abernethy et al. [2020] and Shekhar et al. [2021].

C.2.1 Abernathy 2020

This sampling strategy measures a fairness loss at each time point, and with probability p sample randomly from all protected groups, and with probability $(1-p)$ sample from the protected group

with largest fairness loss. The intuition behind their method is that more samples will decrease loss. This sampling policy can be thought of as an $\epsilon - Greedy$ policy.

Algorithm 14: Abernathy 2020

input : Parameter $p \in [0, 1]$, number of rounds T.
Start with some initial classifier h_0 (e.g., trained on an initial training set S_0 or chosen at random)
output: a classifier $h \in \mathcal{H}$
for $t = 1$ *to* $t = T$ **do**
  Let G_a be the group for which $f_{|a}(h_{t-1})$ is largest, evaluated on a validation set.
  With probability p sample $(x, y, a) \sim Pr$ and with probability $1 - p$ sample $(x, y, a) \sim Pr_{|G_a}$.
  Set $S_t = S_{t-1} \cup \{(x, y, a)\}$; update h_{t-1} to obtain the classifier h_t.
end
return h_t

C.2.2 Shekhar 2021

This policy uses the optimism principle to estimate the protected group with the largest loss. They additionally include forced exploration by ensuring all groups are sampled a minimum number of times relative to the current step. Here $\mathcal{A}$ is the set of protected groups.

The upper bound of loss for protected group a is defined as

$$U_t(a, h_t, c) = \frac{1}{|D_{a,t}|} \sum_{(x,y) \in D_{a,t}} l(h_t, x, y) + e_a(N_{a,t}) + \frac{2c}{\pi_t(a)} \sum_{a' \in A} \pi_{a'} e_{a'}(N_{a',t})$$

where

$$e_a(N) = \frac{2}{\sqrt{N}} \sqrt{6 \log\left(\frac{2}{3} eN\right) + 2 \log\left(\frac{2}{3\delta} N^2 \pi^2 |\mathcal{A}|\right)}.$$

C.3 Analysis

C.3.1 Data

Using two data sets common in the fairness literature, the Adult Data Set and the Default of Credit Card Clients Data Set Dua and Graff [2017], Yeh and Lien [2009]. We dropped all data points with missing values.

Algorithm 15: Shekhar 2021: Optimistic Sampling for Fair Classification ($\mathcal{A}_{opt}$)

input : Parameter $c > 0$, number of rounds T.
output: a classifier $h \in \mathcal{H}$
Draw two independent samples from each $a \in A$, assigning one to the training set $\mathcal{D}$ and one to the group validation set $\mathcal{D}_a$.
for $t = 2d$ *to* $t = T$ **do**
  if $\min_{a \in A} N_{a,t} < \sqrt{t}$ **then**
    $a_t = \text{argmin}_{a \in A} N_{a,t}$
  end
  else
    $a_t = \text{argmax}_{a \in A} U_t(a, h_{t-1}, c)$
  end
  $(X_t^{(i)}, Y_t^{(i)})_{i=1,2} \sim P_{|G_a}$
  Update $(e_a(N_{a,t}))_{a \in \mathcal{A}}$
  $\mathcal{D}_t = \mathcal{D}_{t-1} \cup (X_t^{(1)}, Y_t^{(1)})$, $\mathcal{D}_{a_t,t} = \mathcal{D}_{a_t,t-1} \cup (X_t^{(2)}, Y_t^{(2)})$
  Update h_t
end
return h_t

For the Adult data set, we used '¡=50K' and '¿50K' as the class labels, and defined the protected groups as 'Male' and 'Female'. Quantitative data was normalized, and categorical data was split into multiple features using one-hot encoding. We kept the given split between test and train, using the train data set to sample from and as the validation set specified in the sampling algorithms.

For the Default of Credit Card Clients data set, we again used gender as the protected attribute. We similarly normalized quantitative data and transformed categorical data using one-hot encoding.

C.3.2 Fairness measurements

We sampling from the two data sets using the samplings algorithms with three different definitions of fairness, overall accuracy equality, false negative rate equality (also known as equal opportunity), and predictive parity.

Overall Accuracy Equality: To satisfy this fairness definition, all groups must have equal prediction accuracy.

False Negative Rate Equality (Equal Opportunity): To satisfy this fairness definition, all groups must have equal false negative rates.

Predictive Parity: To satisfy this fairness definition, all groups must have equal true positive rates.

Considering the different fairness definitions and how to decide loss, it is not always clear how

to consider the loss. For example, if one group have higher false positive rates than another, which one has a bigger loss? Is is the group that has the higher false positive rate because more training may lower it, or the group that has lower false positive rate, because the term 'loss' implies they are worse of opportunity wise. Additionally, some fairness definitions consider multiple metrics, such as equalized odds, which defines equality as equal true positive rates (TPR) and equal false positive rates (FPR). From an accuracy standpoint, we want high TPR and low FPR. How then to combine them to determine a loss? Should that loss be minimized or maximized?

C.3.3 Results

We ran the sampling algorithms using each fairness definition to 1000 samples, with a batch size of 10, using 30 values in [0,1] for the parameters p and C. At sizes 500 and 1000, we measure the fairness violation using each fairness definition. We also randomly sampled equally representative data as a baseline measure. Results are an average over 10 trials.

The Pareto frontier represents the best in error and fairness violations for each sampling algorithm. To compare outcomes, we first determined which parameters defined the Pareto frontier of the fairness violations measured from the fairness definition used for sampling. The Pareto frontier for all other fairness violations is then determined only from the parameters in the original frontier. This choice is to used to represent the best results for one fairness definition, if we are Pareto optimal on a different definition.

We will refer to policy A for results from the sampling algorithm in Abernethy et al. [2020], and policy S for results from Shekhar et al. [2021]

When looking at the outcomes for the Adult data set, seen in figures C.1 and C.2, we see that the policy A outperforms policy S. This is most likely simply because that policy S uses a large portion of the sampled data as a validation set, whereas policy A only used a validation set of size 300.

A more interesting observation in that in policy A, for each fairness measure, the data collected to optimize fairness for that measure generally outperforms data collected to optimize over other fairness measure, with the only exception being in appendix C.3.3. This does not hold for policy S, where there are several cases of the equal representation data or and data collected to optimize a different fairness measure had a better frontier. This can be seen in appendix C.3.3.

This is not the case with the Credit Default data, where both policy A and policy S have that a fairness measure is not always optimized using data collected using that fairness measure, as seen in figures C.3 and C.4. While policy A almost always outperforms the equally representative data, we can see that this is not the case for policy S, which is often outperformed by the equally representative data.

C.4 Discussion

We see that in the adult data set, policy A performs as expected, and performance when optimized over the desired fairness measure it better that when optimized over other fairness measure. It is interesting that is does not hold true for the Credit Default data set, where optimizing for overall fairness accuracy often outperforms when sampling using other fairness measures, even when it is the other fairness measure we desire to be equal. This is particularly interesting because we are deriving Pareto frontiers only from the subset of points optimal for the fairness used in sampling.

This implies that in certain circumstances, at least with these sampling policies, it may be better to sampling with respect to a different fairness measure than the one ultimately desired.

This work is limited by the small sample sizes and consideration of only two data sets. We also used a large batch size, whereas both policies referenced in this paper present their simulations and empirical results using a batch size of 1.

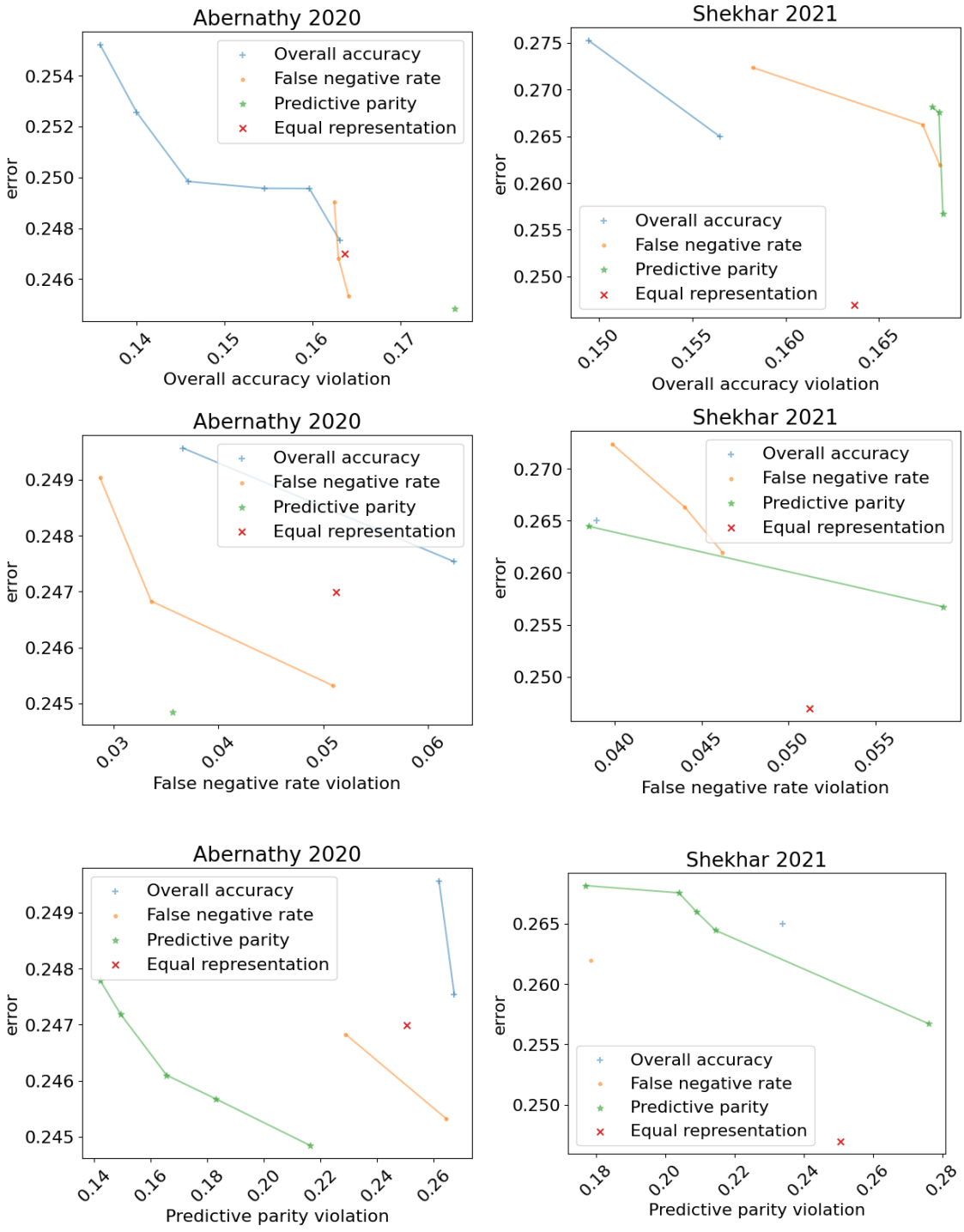


Figure C.1: Pareto frontier of fairness violations from Adult data set, sample size 500. Frontier for fairness measure not used for sampling determined only from subset of parameters that define frontier for fairness measurement used in sampling.

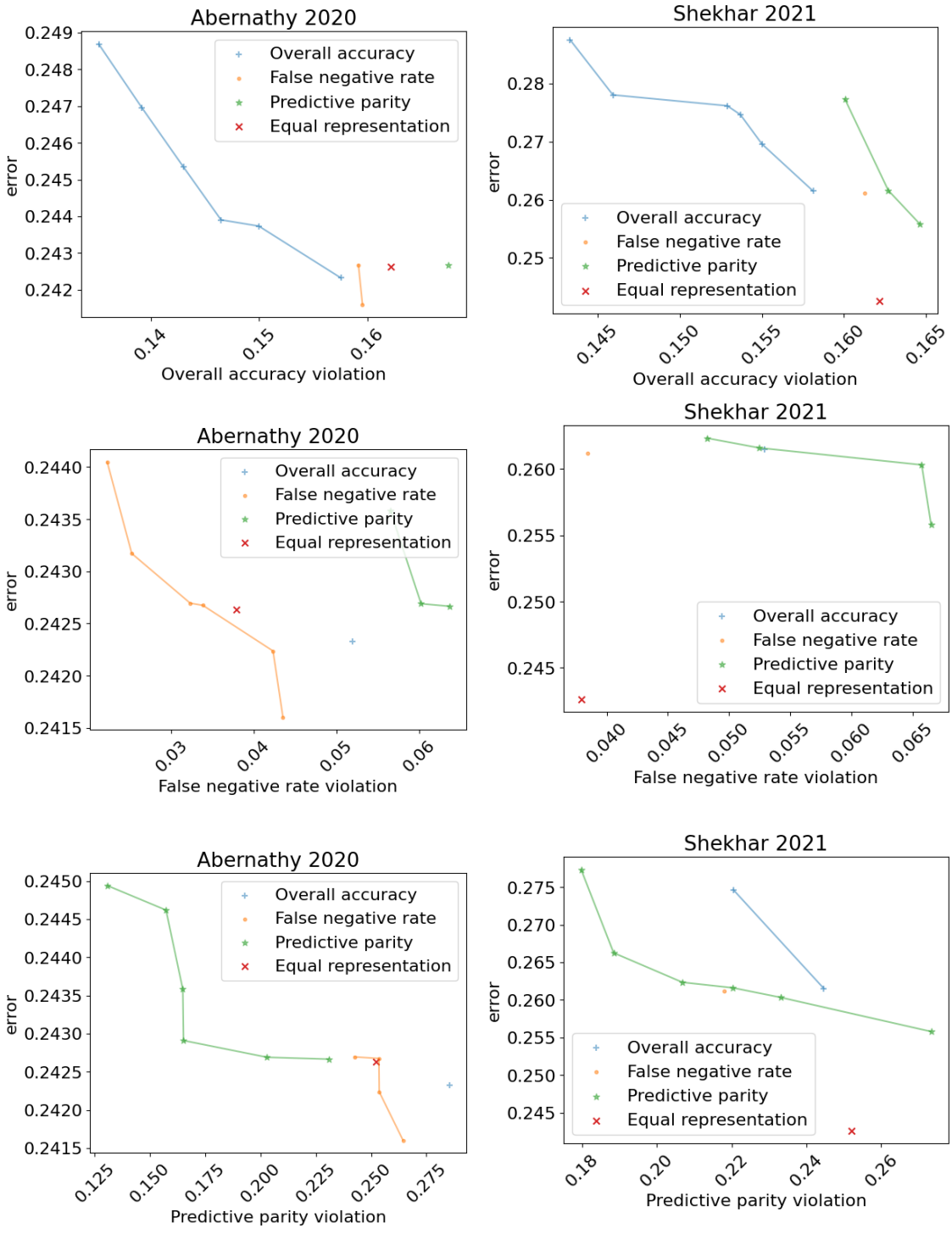


Figure C.2: Fairness violations from Adult data set, sample size 1000. Frontier for fairness measure not used for sampling determined only from subset of parameters that define frontier for fairness measurement used in sampling.

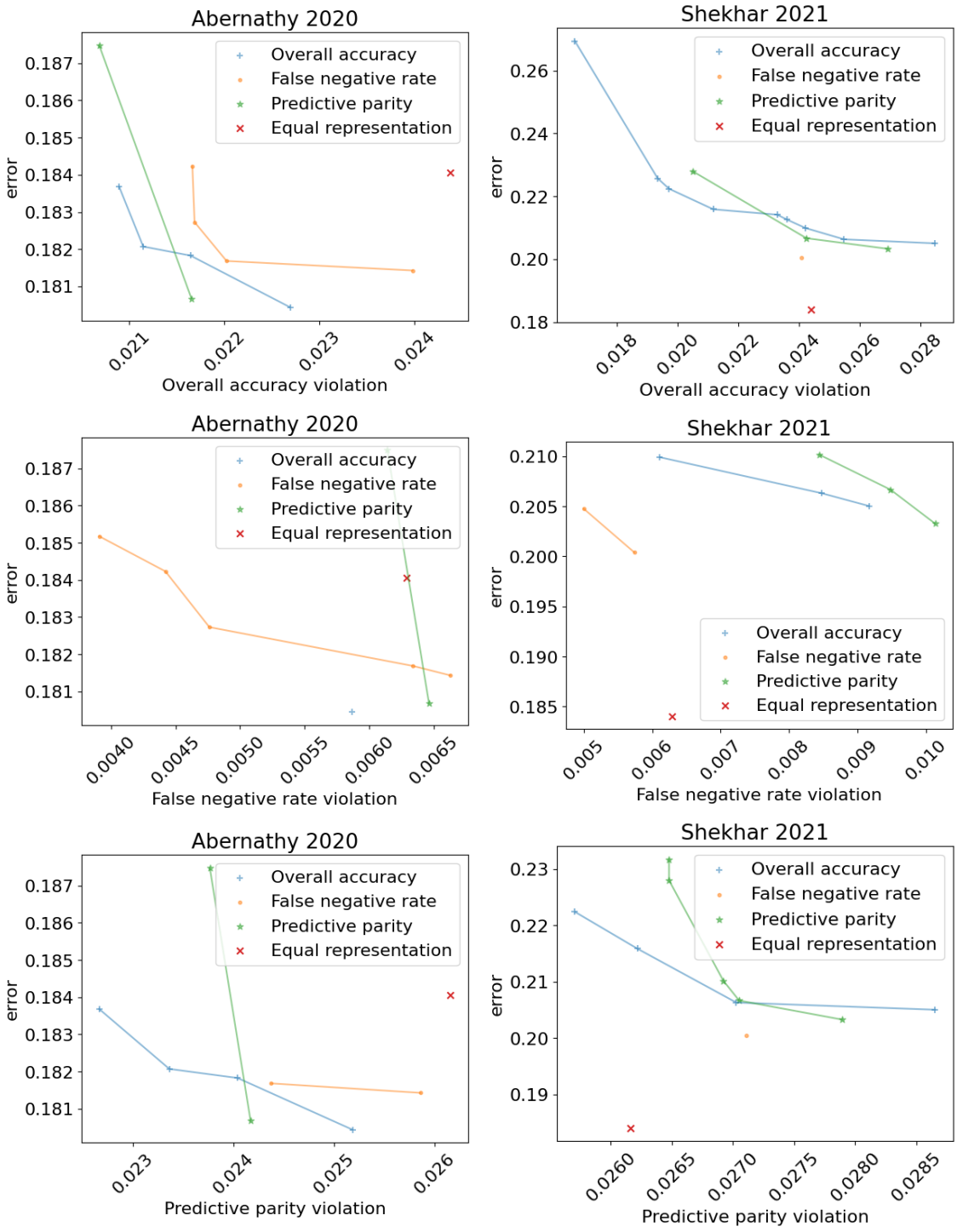


Figure C.3: Fairness violations from Default of Credit Card Clients data set, sample size 500. Frontier for fairness measure not used for sampling determined only from subset of parameters that define frontier for fairness measurement used in sampling.

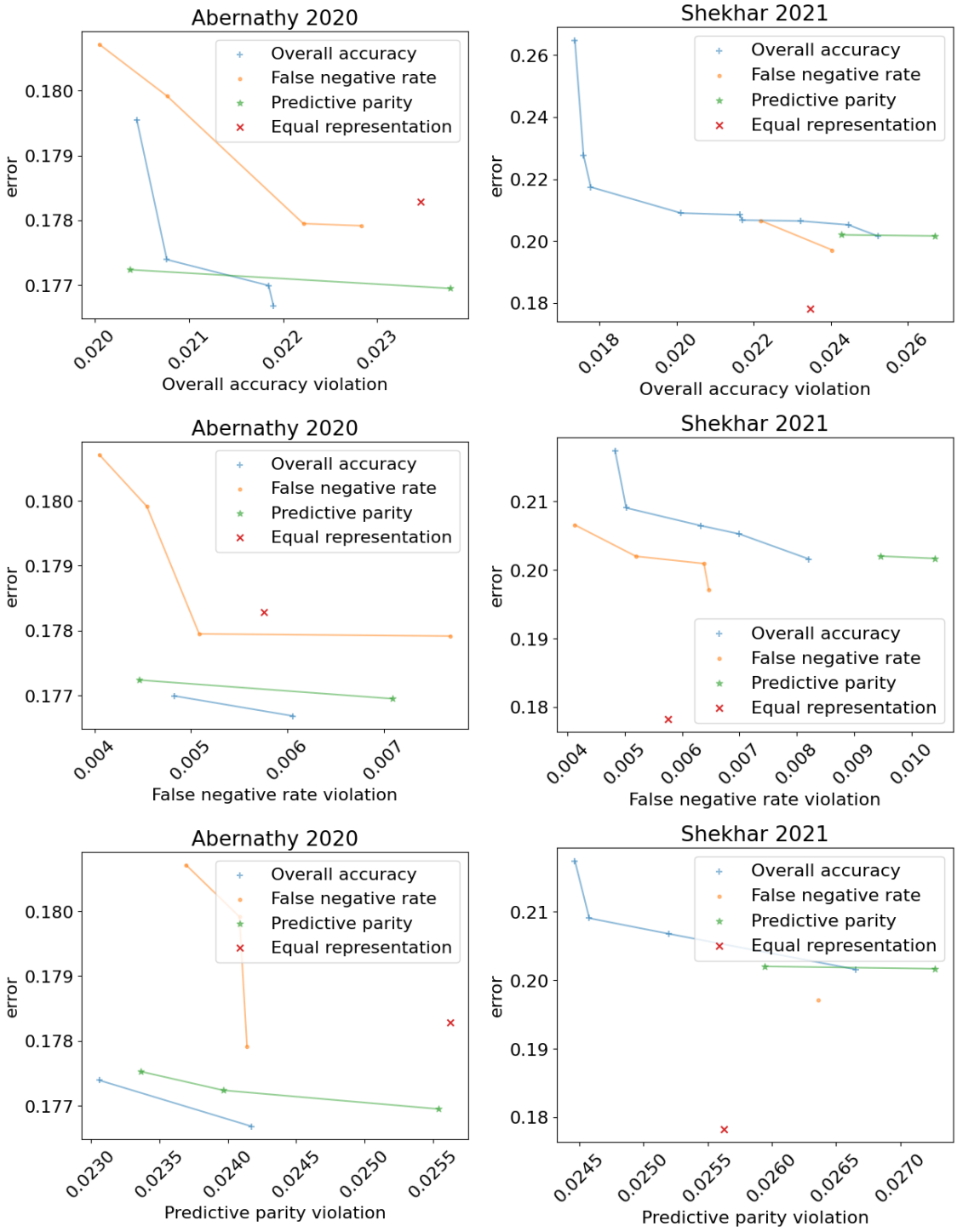


Figure C.4: Fairness violations from Default of Credit Card Clients data set, sample size 1000. Frontier for fairness measure not used for sampling determined only from subset of parameters that define frontier for fairness measurement used in sampling.

BIBLIOGRAPHY

Jacob Abernethy, Pranjal Awasthi, Matthäus Kleindessner, Jamie Morgenstern, and Jie Zhang. Adaptive Sampling to Reduce Disparate Performance. In *arXiv:2006.06879 [Cs, Stat]*, June 2020.

Douglas J Ahler, Carolyn E Roush, and Gaurav Sood. The micro-task market for lemons: Data quality on amazon's mechanical turk. In *Meeting of the Midwest Political Science Association*, 2019.

Jason Altschuler, Victor-Emmanuel Brunel, and Alan Malek. Best Arm Identification for Contaminated Bandits. *Journal of Machine Learning Research*, 20:1–39, 2019.

Hadis Anahideh, Abolfazl Asudeh, and Saravanan Thirumuruganathan. Fair Active Learning. In *arXiv:2001.01796 [Cs, Stat]*, March 2021.

Theodore W Anderson and Herman Rubin. Statistical inference in factor analysis. In *Proceedings of the third Berkeley symposium on mathematical statistics and probability*, volume 5, pages 111–150, 1956.

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

Abolfazl Asudeh, Zhongjun Jin, and H. V. Jagadish. Assessing and Remedying Coverage for a Given Dataset. In *International Conference on Data Engineering*. IEEE, February 2019.

Peter Auer and Nicolo Cesa-Bianchi. Finite-time Analysis of the Multiarmed Bandit Problem. *Machine learning,*, page 22, 2002.

Peter Auer and Chao-Kai Chiang. An algorithm with nearly optimal pseudo-regret for both stochastic and adversarial bandits. *Conference on Learning Theory*, pages 116–120, May 2016.

Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The Nonstochastic Multi-armed Bandit Problem. *SIAM Journal on Computing*, 32(1):48–77, January 2002. ISSN 0097-5397, 1095-7111. doi: 10.1137/S0097539701398375.

Yang Bai, Paul Hibbing, Constantine Mantis, and Gregory J. Welk. Comparative evaluation of heart rate-based monitors: Apple Watch vs Fitbit Charge HR. *Journal of Sports Sciences*, 36 (15):1734–1741, August 2018. ISSN 0264-0414. doi: 10.1080/02640414.2017.1412235.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357, 2016.

Amanda Bower, Sarah N. Kitchen, Laura Niss, Martin J. Strauss, Alexander Vargas, and Suresh Venkatasubramanian. Fair Pipelines. In *4th Workshop on Fairness, Accountability, and Transparency in Machine Learning*, Halifax, NS, Canada, July 2017.

Amanda Bower, Laura Niss, Yuekai Sun, and Alexander Vargo. Debiasing representations by removing unwanted variation due to protected attributes. In *5th Workshop on Fairness, Accountability, and Transparency in Machine Learning*, Stockholm, Sweden, July 2018.

Sebastien Bubeck and Aleksandrs Slivkins. The best of both worlds: Stochastic and adversarial bandits. *Conference on Learning Theory*, February 2012.

Olivier Chapelle and Lihong Li. An Empirical Evaluation of Thompson Sampling. In *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc., 2011.

Jonathan Crussell, Ryan Stevens, and Hao Chen. MAdFraud: Investigating ad fraud in android applications. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services - MobiSys '14*, pages 123–134, Bretton Woods, New Hampshire, USA, 2014. ACM Press. ISBN 978-1-4503-2793-0. doi: 10.1145/2594368.2594391.

Paul G. Curran. Methods for the detection of carelessly invalid responses in survey data. *Journal of Experimental Social Psychology*, 66:4–19, September 2016. ISSN 00221031. doi: 10.1016/j.jesp.2015.07.006. URL https://linkinghub.elsevier.com/retrieve/pii/S0022103115000931.

A. Datta, S. Sen, and Y. Zick. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 598–617, May 2016. doi: 10.1109/SP.2016.42.

Ilias Diakonikolas, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. Robust Estimators in High Dimensions without the Computational Intractability. *SIAM Journal on Computing*, 48(2):742–864, 2019.

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

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

Eyal Even-dar, Shie Mannor, and Yishay Mansour. PAC bounds for multi-armed bandit and Markov decision processes. In *In Fifteenth Annual Conference on Computational Learning Theory (COLT)*, pages 255–270, 2002.

Lynne M Feehan, Jasmina Geldman, Eric C Sayre, Chance Park, Allison M Ezzat, Ju Young Yoo, Clayon B Hamilton, and Linda C Li. Accuracy of Fitbit Devices: Systematic Review and Narrative Syntheses of Quantitative Data. *JMIR mHealth and uHealth*, 6(8):e10527, August 2018. ISSN 2291-5222. doi: 10.2196/10527.

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

Martin Fink, John Hershberger, Nirman Kumar, and Subhash Suri. Hyperplane separability and convexity of probabilistic point sets. *Journal of Computational Geometry (Old Web Site)*, 8(2): 32–57, February 2017. ISSN 1920-180X. doi: 10.20382/jocg.v8i2a3.

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

Sorelle A. Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P. Hamilton, and Derek Roth. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency - FAT* '19*, FAT* '19, Atlanta, GA, USA, 2019. doi: 10.1145/3287560.3287589.

Victor Gabillon, Mohammad Ghavamzadeh, and Alessandro Lazaric. Best Arm Identification: A Unified Approach to Fixed Budget and Fixed Confidence. In *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

Johann A Gagnon-Bartsch, Laurent Jacob, and Terence P Speed. Removing unwanted variation from high dimensional data with negative controls. *Berkeley: Tech Reports from Dep Stat Univ California*, pages 1–112, 2013.

Aurelien Garivier and Emilie Kaufmann. Optimal Best Arm Identification with Fixed Confidence. In *Conference on Learning Theory*, pages 998–1027. PMLR, 2016.

Anupam Gupta, Tomer Koren, and Kunal Talwar. Better algorithms for stochastic bandits with adversarial corruptions. In *Conference on Learning Theory*, pages 1562–1578, 2019.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

Tatsunori B. Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. Fairness Without Demographics in Repeated Loss Minimization. In *International Conference on Machine Learning*. PMLR, July 2018.

Kenneth Holstein, Jennifer Wortman Vaughan, Hal Daumé, Miro Dudik, and Hanna Wallach. Improving Fairness in Machine Learning Systems: What Do Industry Practitioners Need? In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–16, Glasgow Scotland Uk, May 2019. ACM. ISBN 978-1-4503-5970-2. doi: 10.1145/3290605.3300830.

Peter Huber. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1):73–101, March 1964.

Kevin Jamieson, Matthew Malloy, Robert Nowak, and Sebastien Bubeck. Lil' UCB : An Optimal Exploration Algorithm for Multi-Armed Bandits. In *Conference on Learning Theory*, page 17. PMLR, 2014.

Matthew Joseph, Michael J. Kearns, Jamie Morgenstern, and Aaron Roth. Fairness in learning: Classic and contextual bandits. *CoRR*, abs/1605.07139, 2016. URL http://arxiv.org/abs/1605.07139.

Jeff Larson Julia Angwin. Machine Bias. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing, May 2016.

Kwang-Sung Jun, Lihong Li, Yuzhe Ma, and Jerry Zhu. Adversarial attacks on stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 3640–3649, 2018.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC Subset Selection in Stochastic Multi-armed Bandits. In *ICML*, pages 655–662, 2012.

Sayash Kapoor, Kumar Kshitij Patel, and Purushottam Kar. Corruption-tolerant bandit learning. *Machine Learning*, pages 1–29, August 2018. ISSN 0885-6125, 1573-0565. doi: 10.1007/s10994-018-5758-5.

Emilie Kaufmann and Shivaram Kalyanakrishnan. Information Complexity in Bandit Subset Selection. In *Conference on Learning Theory*, page 24. PMLR, 2013.

Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. Inherent Trade-Offs in the Fair Determination of Risk Scores. In *8th Innovations in Theoretical Computer Science Conference*, 2017. ISBN 978-3-95977-029-3.

Pravesh K Kothari, Jacob Steinhardt, and David Steurer. Robust moment estimation and improved clustering via sum of squares. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1035–1046, 2018.

K. A. Lai, A. B. Rao, and S. Vempala. Agnostic Estimation of Mean and Covariance. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 665–674, October 2016. doi: 10.1109/FOCS.2016.76.

T.L Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, March 1985. ISSN 01968858. doi: 10.1016/0196-8858(85)90002-8.

Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, first edition, July 2020. ISBN 978-1-108-57140-1 978-1-108-48682-8. doi: 10.1017/9781108571401.

Jeffrey T Leek and John D Storey. A general framework for multiple testing dependence. *Proceedings of the National Academy of Sciences*, 105(48):18718–18723, 2008.

Liu Liu, Tianyang Li, and Constantine Caramanis. High Dimensional Robust Estimation of Sparse Models via Trimmed Hard Thresholding. *arXiv preprint*, January 2019.

Yun-En Liu, Travis Mandel, Emma Brunskill, and Zoran Popovic. Trading off scientific knowledge and user learning with multi-armed bandits. In *EDM*, pages 161–168, 2014.

Thodoris Lykouris, Vahab Mirrokni, and Renato Paes Leme. Stochastic bandits robust to adversarial corruptions. *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 114–122, March 2018.

Yuzhe Ma, Kwang-Sung Jun, Lihong Li, and Xiaojin Zhu. Data poisoning attacks in contextual bandits. In *International Conference on Decision and Game Theory for Security*, 2019.

Shie Mannor and John N Tsitsiklis. The Sample Complexity of Exploration in the Multi-Armed Bandit Problem. *Journal of Machine Learning Research*, 5:623–648, 2004.

Fatemeh Nargesian, Abolfazl Asudeh, and H. V. Jagadish. Tailoring data source distributions for fairness-aware data integration. *Proceedings of the VLDB Endowment*, 14(11):2519–2532, July 2021. ISSN 2150-8097. doi: 10.14778/3476249.3476299.

Elizabeth A. Necka, Stephanie Cacioppo, Greg J. Norman, and John T. Cacioppo. Measuring the Prevalence of Problematic Respondent Behaviors among MTurk, Campus, and Community Participants. *PLOS ONE*, 11(6):e0157732, June 2016. ISSN 1932-6203. doi: 10.1371/journal.pone.0157732.

Laura Niss and Ambuj Tewari. What You See May Not Be What You Get: UCB Bandit Algorithms Robust to {\epsilon}-Contamination. *36th Conference on Uncertainty in Artificial Intelligence*, 2020.

Laura Niss, Yuekai Sun, and Ambuj Tewari. Achieving representative data via convex hull feasibility sampling algorithms. *arXiv preprint arXiv:2204.06664*, 2022. URL https://arxiv.org/abs/2204.06664.

C. O'Neil. *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown/Archetype, 2016. ISBN 9780553418828. URL https://books.google.com/books?id=NgEwCwAAQBAJ.

Paul Pearce, Vacha Dave, Chris Grier, Kirill Levchenko, Saikat Guha, Damon McCoy, Vern Paxson, Stefan Savage, and Geoffrey M. Voelker. Characterizing Large-Scale Click Fraud in ZeroAccess. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security - CCS '14*, pages 141–152, Scottsdale, Arizona, USA, 2014. ACM Press. ISBN 978-1-4503-2957-6. doi: 10.1145/2660267.2660369.

Geoff Pleiss, Manish Raghavan, Felix Wu, Jon Kleinberg, and Kilian Q Weinberger. On Fairness and Calibration. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Y. Ritov, Y. Sun, and R. Zhao. On conditional parity as a notion of non-discrimination in machine learning. *ArXiv*, June 2017. URL http://arxiv.org/abs/1706.08519.

Esther Rolf, Theodora Worledge, Benjamin Recht, and Michael I. Jordan. Representation Matters: Assessing the Importance of Subgroup Allocations in Training Data. In *arXiv:2103.03399 [Cs, Stat]*, March 2021.

Timothy Ryan. Data contamination on MTurk — Timothy J. Ryan, 2018.

Yevgeny Seldin and Gábor Lugosi. An improved parametrization and analysis of the EXP3++ algorithm for stochastic and adversarial bandits. In *Proceedings of the 2017 Conference on Learning Theory*, volume 65, pages 1743–1759. PMLR, 2017. URL http://proceedings.mlr.press/v65/seldin17a.html.

Yevgeny Seldin and Aleksandrs Slivkins. One practical algorithm for both stochastic and adversarial bandits. In *ICML*, pages 1287–1295, 2014.

Shubham Sharma, Yunfeng Zhang, Jesús M. Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R. Varshney. Data Augmentation for Discrimination Prevention and Bias Disambiguation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, pages 358–364, New York NY USA, February 2020. ACM. ISBN 978-1-4503-7110-0. doi: 10.1145/3375627.3375865.

Farnaz Sheikhi, Ali Mohades, Mark de Berg, and Ali D. Mehrabi. Separability of imprecise points. *Computational Geometry*, 61:24–37, February 2017. ISSN 09257721. doi: 10.1016/j.comgeo.2016.10.001.

Shubhanshu Shekhar, Greg Fields, Mohammad Ghavamzadeh, and Tara Javidi. Adaptive Sampling for Minimax Fair Classification. In *arXiv:2103.00755 [Cs]*, July 2021.

Emily Steel and Juila Angwin. On the web's cutting edge, anonymity in name only. *The Wall Street Journal*, 2010. URL https://www.wsj.com/articles/SB10001424052748703294904575385532109190198.

Ki Hyun Tae and Steven Euijong Whang. Slice Tuner: A Selective Data Acquisition Framework for Accurate and Fair Machine Learning Models. In *Proceedings of the 2021 International Conference on Management of Data*, pages 1771–1783, Virtual Event China, June 2021. ACM. ISBN 978-1-4503-8343-1. doi: 10.1145/3448016.3452792.

William Thompson. ON THE LIKELIHOOD THAT ONE UNKNOWN PROBABILITY EXCEEDS ANOTHER IN VIEW OF THE EVIDENCE OF TWO SAMPLES. *Biometrika*, 25: 285–294, 1933.

Alexander H S Vargo. *Applications of Machine Learning: From Single Cell Biology to Algorithmic Fairness*. PhD thesis, University of Michigan, 2020.

Abraham Wald. Sequential tests of statistical hypotheses. *The annals of mathematical statistics*, 16(2):117–186, 1945.

Joseph Jay Williams, Juho Kim, Anna Rafferty, Samuel Maldonado, Krzysztof Z. Gajos, Walter S. Lasecki, and Neil Heffernan. AXIS: Generating Explanations at Scale with Learnersourcing and Machine Learning. In *Proceedings of the Third (2016) ACM Conference on Learning @ Scale*

- *L@S '16*, pages 379–388, Edinburgh, Scotland, UK, 2016. ACM Press. ISBN 978-1-4503-3726-7. doi: 10.1145/2876034.2876042.

Da Yan, Zhou Zhao, Wilfred Ng, and Steven Liu. Probabilistic Convex Hull Queries over Uncertain Data. *IEEE Transactions on Knowledge and Data Engineering*, 27(3):852–865, March 2015. ISSN 1558-2191. doi: 10.1109/TKDE.2014.2340408.

I. C. Yeh and C. H. Lien. UCI Machine Learning Repository: Default of credit card clients Data Set. *Expert Systems with Applications*, 36(2):2473–2480, 2009.

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

Julian Zimmert and Yevgeny Seldin. An optimal algorithm for stochastic and adversarial bandits. In *Proceedings of Machine Learning Research*, volume 89, pages 467–475, 2019. URL http://proceedings.mlr.press/v89/zimmert19a.html.