Trade-Offs between Fairness, Interpretability, and Privacy in Machine Learning

Abstract

The concerns of fairness, interpretability, and privacy, in machine learning based systems have received a lot of attention in the research community recently, but have primarily been studied in isolation. In this work, we look at cases where we want to satisfy multiple of these properties simultaneously, and find that it is necessary to make trade-offs between them. We have two theoretical results to demonstrate this. The first main result shows that there is a tension between the requirements of fairness and interpretability of classifiers. More specifically, we consider a formal framework to build simple classifiers as a means to attain interpretability, and show that each simple classifier is strictly improvable, in the sense that every simple classifier can be replaced by a more complex classifier that strictly improves both fairness and accuracy simultaneously. The second main result considers the issue of compatibility between fairness and differential privacy of learning algorithms. In particular, we prove an impossibility theorem which shows that even in simple binary classification settings, one cannot design an accurate learning algorithm that is both ϵ-differentially private and fair (even approximately).

1 Introduction

Technology has entered most aspects of our lives, with automated systems being deployed to make consequential decisions, such as predicting recidivism rates in released prisoners, and estimating the probability of an applicant returning a loan. Now, because these systems are making decisions that are potentially life-altering for many people, there have been many ethical questions raised about how they function. We will look at three ethical considerations in this work: fairness, interpretability, and privacy. We would like the system to be *fair*, and not discriminate against an applicant just because of their membership in a minority/protected group (which could be a particular race, gender, etc.). We would also like the system to be *interpretable*, what that intuitively means is that we would like to be able to understand how it works and convincingly explain any decisions it might make. The third concern is *privacy*. Now, because these decision making systems are typically ML models, and are trained on potentially sensitive data, we would not like to inadvertently leak information about people in the training set, and would like to protect their privacy. These concerns have received a lot of attention in the research community in the last few years (Chaudhuri, Monteleoni, and Sarwate 2011; Corbett-Davies and Goel 2018; Doshi-Velez and Kim 2017). However, they have primarily been studied in isolation, that is, people have primarily looked at scenarios in which we would want to satisfy one of these properties at a time. In this work, we look at cases where we want to satisfy multiple of these properties simultaneously, and analyse how these properties interact. Overall, we find that that these properties are at odds with each other, and it is necessary to make trade-offs between them. We show two theoretical results to demonstrate this.

The **first main result** looks at cases where we would like to have accurate classifiers that are also fair and interpretable, and shows how the desiderata of fairness and accuracy are at odds with interpretability. Creating models that are intuitively simple to humans is a natural strategy to increase their interpretablilty. For example, one could avoid using complex models such as deep neural networks, and instead use simple models such as linear classifiers. Another way to build simple classifiers is to reduce the number of features that are involved in the decision making process, by choosing a small number of the most informative features, or deleting unfair features (Grgić-Hlača et al. 2018). We consider a formal framework to model the construction of simple classifiers, which captures some commonly used methods of building interpretable models. We discuss the interaction between the desiderata of simplicity, fairness and accuracy of binary classifiers in this framework. Given a set of features, we have an optimal classifier (i.e., the most accurate classifier that can be built from the given features). One may wish to simplify the optimal classifier to increase interpretability, or to even increase fairness in some cases. Simpler models can be easier to audit, and we can possibly identify sources of unfairness and correct them with more ease in them (Doshi-Velez and Kim 2017). Deleting features that can be potentially viewed as unfair, has also been adopted in practice, for example, in the well known *ban the box* scenario, where the check box in hiring applications that asks if applicants have a criminal record, is removed (Doleac and Hansen 2016). In contrast, this work discusses the negative effects of building simple classifiers on their fairness. More specifically, we show that every simple classifier can be improved; i.e., replaced by a more complex classifier that strictly increases both fairness

and accuracy simultaneously with respect to the simple classifier. It is quite expected that using a simple model would result in a loss in accuracy, because imposing simplicity requirements on a classifier reduces its expressive power. The surprising finding here is that simplification leads to a loss in fairness as well, i.e., we can always find a more complex classifier that is more fair, in fact, we can always find a more complex classifier that is simultaneously more fair and accurate than the simple classifier. Hence, we see that that the properties of fairness and accuracy clash with interpretability (or simplicity).

Our **second main result** talks about the clash between the requirements of differential privacy, accuracy, and fairness in learning algorithms. We prove an impossibility theorem which states that even in a very simple binary classification setting, no learning algorithm that is ϵ-differentially private (for *any* $\epsilon \geq 0$), and fair (even approximately, i.e., the algorithm outputs an *approximately fair classifier*[1]) can have non-trivial accuracy.

2 Related Work

Although the ethical issues concerning algorithms that we discuss in this work have been considered widely in the now ubiquitous literature on differential privacy, model interpretability, and algorithmic fairness, they have mostly been considered in isolation. In particular, the literature on algorithmic fairness discusses how to handle issues such as bias and discrimination (e.g., Dwork et al. (2012); Kleinberg, Mullainathan, and Raghavan (2017); Feldman et al. (2015)), the literature on model interpretability addresses the growing need for transparent models (e.g., Doshi-Velez and Kim (2017); Rudin (2019); Lipton (2018)), and the literature on differential privacy talks about protecting the privacy of individuals (e.g., Dwork et al. (2006); Dwork, Roth et al. (2014)).

Our work fills a notable gap existing in the current research on ethical AI, and formalises the three desiderata of fairness, interpretability, and privacy of ML systems in a common framework. Very little previous work has looked at cases where one would want to satisfy multiple of these properties simultaneously, and analysed the trade-offs involved, with two notable exceptions being Kleinberg and Mullainathan (2019); Cummings et al. (2019). We significantly improve on prior results in these two works, and show that there are hard trade-offs when we want to simultaneously achieve either fairness and interpretability, or fairness and privacy.

Previously, it has been argued that increasing a model's interpretability makes the model easier to analyse, and therefore assists in (a) deciding whether the model is fair and (b) modifying the model to ensure that it is (Doshi-Velez and Kim 2017). In contrast, we present a result that captures the fact that the interpretability of a model is at odds with its fairness. Our first main result improves upon the main result of Kleinberg and Mullainathan (2019). Our setup is more general, and we remove a restrictive assumption they make on the data distribution. In addition, our notion of fairness is more aligned with accuracy, and enjoys multiple advantages. We will compare our work to Kleinberg and Mullainathan (2019) in greater detail later.

The work that is most relevant to our second main result is that of Cummings et al. (2019). Cummings et al. (2019) consider the trade-offs when considering learning algorithms that satisfy differential privacy and one particular notion of fairness (equal opportunity). In particular, they claim that there is no learning algorithm that achieves ϵ-differential privacy, satisfies equal opportunity, and has accuracy better than a constant classifier. However, there is a gap in their argument (see appendix where we describe what it is), and so unfortunately their proof idea does not go through. Our proof technique is different, and we generalise their result, by showing that such an impossibility holds with respect to every common notion of fairness, and more importantly, even approximate/relaxed versions of these notions of fairness. We discuss other related work in the appendix.

3 Trade-Offs between Fairness and Interpretability

Informal Version of Result This section considers a formal framework to build simple classifiers as a means to achieve interpretability, and shows that if we restrict a classifier to be simple within this framework, it can be replaced by a more complex classifier that strictly improves both fairness and accuracy simultaneously. Therefore, we see that simplicity/interpretability clashes with the properties of fairness and accuracy. We also have a couple of useful variations of the result, dependent on different underlying assumptions. We now formalise the framework, before moving on to the results. Some details and proofs are moved to the appendix in the interest of space.

3.1 Formalising the Framework

We denote the domain set by $\mathcal{X}$. There is an underlying distribution $\mathcal{D}$ over $\mathcal{X}$. We assume the existence of a *Ground Truth* function, that assigns a label to each point in the domain set, that is,

$$\mathcal{G} : \mathcal{X} \to \{0, 1\}.$$

For example, in the case where a bank needs to classify loan applicants, a person in the domain set is assigned the label 1 if they would return the loan, and 0 otherwise. In general, we refer to an instance labeled 1 as *good*, and *bad* otherwise.[2]

Features Each instance in $\mathcal{X}$ is represented by the set of features $\mathcal{F} = \{f_1, \ldots, f_k\}$. For example, in the bank loans case, the features could be things like credit score, income,

[1] An approximately fair classifier refers to classifiers that satisfy even relaxed or approximate versions of common notions of fairness (such as error rates, or false positive/negative rates, being approximately equal across different groups). Proving an impossibility for such relaxations makes our result stronger.

[2] Our results and proofs also go through for the case where the ground truth function $\mathcal{G}$ is non-deterministic, that is, instead of being labeled 0 or 1, a particular instance might be labeled 0 with probability 0.6, and 1 with probability 0.4. However, for simplicity, we assume that the ground truth function is deterministic. If we allow $\mathcal{G}$ to be non-deterministic, the underlying distribution $\mathcal{D}$ would be over $\mathcal{X} \times \{0, 1\}$, not $\mathcal{X}$.

and so on. Each instance also belongs to one of two groups - A or D. A stands for the advantaged group, whereas D stands for the disadvantaged group. D can be thought of as the minority group that we wish to protect from discrimination. The group membership feature $f_m : \mathcal{X} \to \{A, D\}$ maps an instance to their group. For simplicity, we assume that each $f_i : \mathcal{X} \to \{0, 1\}$ is a binary feature.[3]

Task Given an unlabeled set of applicants generated by the underlying distribution, we want build a classifier to admit a fixed fraction r (known as admission rate) of them, such that we are as accurate as possible (i.e., admit as many good applicants as possible).

Partitions and Cells We can partition the domain set $\mathcal{X}$ into different parts, and we call each part a *cell*. A natural way to create cells is based on their feature vectors. That is, two instances are part of the same cell if and only if they have the same feature vector representation. Recall, we are given access to a set of features $\mathcal{F} = \{f_1, \ldots, f_k\}$. We also had the group membership feature f_m and if we append that to the feature set $\mathcal{F}$, we denote the resultant feature set by $\mathcal{F}'$. The partition induced by $\mathcal{F}$ is denoted by f, and we denote the cells of f by $\mathcal{C}_1, \ldots, \mathcal{C}_n$, (where $n = 2^k$, because each feature is binary). The partition induced by $\mathcal{F}'$ is denoted by f', and consists of 2^{k+1} cells, as there are $k + 1$ binary features. The cells in f' are obtained by splitting each cell in f into two parts, according to the group membership feature f_m. For e.g., $\mathcal{C}_1$ is split into $\mathcal{C}_1^A$ and $\mathcal{C}_1^D$, which represent the advantaged and disadvantaged people in the cell $\mathcal{C}_1$ respectively.

Score Function We say that the probability of a random instance sampled according to $\mathcal{D}$ being good (given that it lies in some cell $\mathcal{C}$) is the *score* of $\mathcal{C}$. We denoted the score of $\mathcal{C}$ by $\mathcal{S}(\mathcal{C})$, i.e.,

$$\mathcal{S}(C) = \Pr_{x \sim \mathcal{D}}[\mathcal{G}(x) = 1 \mid x \in \mathcal{C}]$$

By score of an instance $x \in \mathcal{X}$, we mean the score of the cell it belongs to in the partition f'. Given the feature set, and the fact that we only have access to the features of any instance, the score of an instance is the most accurate estimate we can have of the probability of the instance being good.

3.2 Classifiers

A classifier assigns every point in the domain set a label from $\{0, 1\}$. Because each point in the domain set is represented by its feature vector, the classifier is essentially a function from the space of all feature vectors to the label set., i.e., from $\{0, 1\}^{k+1} \to \{0, 1\}$. A given partition h of the domain set and admission rate r induces a threshold classifier that we denote by h_r. The classifier h_r sorts the cells of h in descending order of their scores (after merging together cells with the same score). We then admit applicants in this order until we admit the desired fraction r. We explain how classifiers work in more detail in the appendix. We use the terms classifier and partition interchangeably.

[3]However, the results, and pretty much the same proofs also hold for the case when each feature can take finitely many values.

Recall that we had discussed the partition f' above, which is the partition induced by all the features we have. Given the feature set we have, the most accurate classifier we can construct is the one induced by the partition f'.

Modeling Simple Classifiers We use the framework introduced in Kleinberg and Mullainathan (2019) to model the construction of simple classifiers. Two particular approaches to build simple classifiers that this framework captures are (i) shallow decision trees, and (ii) using a small number of informative features (feature selection). Both these approaches follow a common principle: they simplify the underlying model by combining distinguishable instances (applicants with different feature vector representations) together into larger sets and making a common decision at the level of each set. Previously, the instances were in different sets, and were therefore potentially treated differently. What that means in our framework, is that we would simplify f' (or in general, any partition) by combining multiple cells of it together into one larger cell, to result in a simpler partition, with fewer cells. We define a simplification formally below.

Definition (Simplification)**.** *A partition w of a set $\mathcal{X}$ is a simplification of partition v of $\mathcal{X}$ if every cell of v is a subset of some cell of w, and $w \neq v$.*

Definition (Non-trivial simplification)**.** *A simplification w of f' is non-trivial if it contains at least one cell $\mathcal{C}$ such that $\mathcal{C}$ contains at least two cells of f' with different scores. If a simplification of f' only combines together cells with the same scores, the partition-induced classifier remains unchanged, and hence, such a simplification is not very meaningful.*

Definition (Non-trivial cell)**.** *We say that such a cell $\mathcal{C}$ as above is a non-trivial cell.*

Structured Simplifications The approaches to building simple classifiers that the framework captures, which are (i) shallow decision trees, and (ii) feature selection, do not combine cells at random, but they do it in a constrained way. For example, f is the simplification of f' associated with deleting the group membership feature f_m. Deleting a feature (a way to implement feature selection) is a specific form of simplification that halves the number of cells. We define two other forms of structured simplification in this work, namely, group agnostic simplifications and graded simplifications. We defer defining and proving the theorem for graded simplifications, which is the more general version of simplification, to the appendix.

Definition (Group Agnostic Simplification)**.** *A simplification of f' such that instances differing only in the group membership feature are mapped to the same cell. This basically means that as a simplification step, the classifier is constrained at the very least to completely ignore/delete the group membership feature. There may or may not be further simplification steps on top of this.*

Evaluating Classifiers

Fairness We propose a notion more in line with *affirmative action*, that actively supports the disadvantaged group. We penalise only the decisions that unfairly hurt the disadvantaged group. Our fairness objective function penalises FP_A

(False Positives for group A) and FN_D (False Negatives for group D), and aims to minimise a weighted sum of the two. We justify and elaborate on this notion in the appendix.

$$FP_A(h_r) = \mathbb{E}(\text{Fraction of bad instances in } A \text{ that } h_r \text{ accepts})$$
$$FN_D(h_r) = \mathbb{E}(\text{Fraction of good instances in } D \text{ that } h_r \text{ rejects})$$

For some $0 < \gamma < 1$,

$$\text{Fairness}(h_r) = \mathbb{E}[-(\gamma(FN_D(h_r)) + (1-\gamma)FP_A(h_r))]$$

Accuracy

$$\text{Accuracy}(h_r) = \frac{\mathbb{E}(\text{Fraction of good instances } h_r \text{ accepts})}{\text{Total fraction of instances } h_r \text{ accepts (i.e., } r)}$$

Comparing Two Classifiers Consider two partitions of $\mathcal{X}$, say h and g. We say that a partition h *improves* on partition g in criteria Q (e.g., accuracy) if for every $r \in [0,1]$, $Q(h_r)$ is at least $Q(g_r)$. We say that a partition h *strictly improves* on partition g in criteria Q (e.g., accuracy) if for every $r \in [0,1]$, $Q(h_r)$ is at least $Q(g_r)$, and there exists an $r' \in [0,1]$ such that $Q(h_{r'})$ is strictly more than $Q(g_{r'})$.

3.3 Group Agnostic Case

We first consider the case where we restrict simplifications to group agnostic ones. We informally explain the result of this section. Recall that the classifier resulting from partition f' is the most accurate classifier we can build with the features we have. If we choose to use a simpler classifier than f', say w, it might lead to an increase in interpretability, or fairness, but we lose accuracy. That might have been a good trade-off, but we show that the simple classifier w is not optimal if we ignore the requirement of interpretability, as there exists a partition h (achievable by the features we have) that is simultaneously more fair, and accurate, than w. Therefore, we would strictly prefer h over w, if we ignore interpretability requirements, and therefore we see that interpretability clashes with the desiderata of fairness, and accuracy.

Theorem 1. *For every non-trivial group-agnostic simplification of f', say w, there exists a classifier h that simultaneously strictly improves both accuracy and fairness over w.*

Assumptions on the Data Distribution Before moving on to the proof, we list the assumptions we use (also used by Kleinberg and Mullainathan (2019)).

1. *Equality assumption:* For every cell $\mathcal{C}_i \in f$, if we split it by group membership, both resultant cells $\mathcal{C}_i^A$ and $\mathcal{C}_i^D$ have the same score. This intuitively means that if we have enough informative features about a person, their membership in a protected group does not affect their performance.
2. *Denseness assumption:* We denote the cells of f' by $\mathcal{C}'_1, \ldots, \mathcal{C}'_{2n}$. $\mathcal{C}^A$ denotes the instances of cell $\mathcal{C}$ that are advantaged. Similarly, $\mathcal{C}^D$ denotes the disadvantaged instances of cell $\mathcal{C}$. For every cell $\mathcal{C}_i \in f$, if we split it by group membership, both resultant cells $\mathcal{C}_i^A$ and $\mathcal{C}_i^D$ have positive measure (The measure of a cell $\mathcal{C}$, denoted by $\mu(C)$, is the mass of the probability distribution $\mathcal{D}$ in cell $\mathcal{C}$). This intuitively means that there exist people in both groups A and D exhibiting every feature vector.
3. *Genericity assumption:* Let $R, T \subseteq f'$ be two distinct sets of cells such that if $R = \mathcal{C}_i^A$ then $T \neq \mathcal{C}_i^D$. We then assume that $\mathcal{S}(R) \neq \mathcal{S}(T)$ (For a set of cells $R \subseteq f'$, use $\mathcal{S}(R)$ to denote the weighted average value of S in the cells of R).

Remark. *This in particular implies that the cells of f can be arranged in strictly descending order of scores. Without loss of generality, we assume that $\mathcal{S}(\mathcal{C}_1) > \mathcal{S}(\mathcal{C}_2) > \cdots > \mathcal{S}(\mathcal{C}_n)$.*

Proof of Theorem 1. Consider non-trivial group-agnostic simplification w of f'. It partitions $\mathcal{X}$ into the cells

$$\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \cdots, , \mathcal{C}_j^\wedge, \mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_t^\wedge, \ldots, \mathcal{C}_d^\wedge$$

with descending order of scores. Take a non trivial cell of w, say $\mathcal{C}_t^\wedge$. The non trivial cell $\mathcal{C}_t^\wedge$ consists of two or more cells of f with different scores. Say $\mathcal{C}_t^\wedge$ is the union of $\mathcal{C}_a, \mathcal{C}_b, \ldots, \mathcal{C}_z \in f$. Let the cell of f in $\mathcal{C}_t^\wedge$ with the highest score be $\mathcal{C}_b$.

Construct h as follows: Remove $\epsilon > 0$ measure of $\mathcal{X}$ from $\mathcal{C}_b^D$ to create a separate cell $\mathcal{C}'$. This is the new partition h. Denote the remainder of $\mathcal{C}_t^\wedge$ by $\mathcal{C}''$. Observe that $\mathcal{S}(\mathcal{C}') > \mathcal{S}(\mathcal{C}_t^\wedge) > \mathcal{S}(\mathcal{C}'')$. Take ϵ small enough to not change order of $\mathcal{C}''$ in the partition w. It should be in the same position as $\mathcal{C}_t^\wedge$ was before. (we can do this because of the genericity assumption) The only change in the order is that $\mathcal{C}'$ jumps to some position ahead of $\mathcal{C}''$. The new partition h is

$$\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \cdots, \mathcal{C}_j^\wedge, \mathcal{C}', \mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_{t-1}^\wedge, \mathcal{C}'', \mathcal{C}_{t+1}^\wedge \cdots, \mathcal{C}_d^\wedge$$

with descending order of scores.

Remark. *Removing $\epsilon > 0$ measure of a cell to create a separate new cell can be viewed as randomising over instances in that cell. Each instance goes to the new cell with probability ϵ, and stays in the old cell with probability $1 - \epsilon$.*

We can show that for all rates r, the fairness, and accuracy of h is at least as good as w, and for at least one value of r, strictly better in both criteria. Let r_j be the fraction of the first j cells of a partition in the order they are represented.

Case 1 $r \geq r_t$ or $r \leq r_j$:

We note that in h, the measure of all cells uptil $\mathcal{C}''$ is r_t. The classifiers resulting from w and h with admission rate r as above classify all cells the same way. Therefore, h_r has the same accuracy, and fairness as w_r.

Case 2 $r_j + \mu(\mathcal{C}') \geq r > r_j$:

Both h_r and w_r classify all instances of $\mathcal{C}_1^\wedge, \ldots \mathcal{C}_j^\wedge$ as 1. The admission rule h_r classifies instances of $\mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_{t-1}^\wedge$ as 0 and some mass $\mu = r - r_j$ of $\mathcal{C}'$ as 1, while the admission rule w_r classifies some mass μ of $\mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_t^\wedge$ as 1, and the remaining as 0 (we start by classifying instances from from $\mathcal{C}_{j+1}^\wedge$ as 1, if $\mu(\mathcal{C}_{j+1}^\wedge) < \mu$, then we move on to $\mathcal{C}_{j+2}$, and so on). Since the score of $\mathcal{C}'$ is greater than the score of each cell $\mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_t^\wedge$, the mass μ of $\mathcal{C}'$ that h_r classifies as 1 has a higher measure of expected true 1's than the mass μ of $\mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_t^\wedge$ that w_r classifies as 1. Therefore, h_r is in expectation more accurate than w_r.

The mass μ of $\mathcal{C}'$ that h_r classifies as 1 has a higher measure of disadvantaged instances than the mass μ of $\mathcal{C}^{\wedge}_{j+1}, \cdots \mathcal{C}^{\wedge}_{t}$ that w_r classifies as 1 because $\mathcal{C}'$ only consists of disadvantaged instances, while each cell in $\mathcal{C}^{\wedge}_{j+1}, \cdots \mathcal{C}^{\wedge}_{t}$ consists of both disadvantaged and advantaged instances (because of the denseness assumption). It is easy to see that h_r on expectation has lower FP_A and FN_D values than w_r. Hence, h_r has higher fairness than w_r.

Case 3 $r_t > r \geq r_j + \mu(\mathcal{C}')$:

Both h_r and w_r classify all instances of $\mathcal{C}^{\wedge}_1, \ldots \mathcal{C}^{\wedge}_j$ as 1 and all instances of $\mathcal{C}^{\wedge}_{t+1}, \ldots \mathcal{C}^{\wedge}_d$ as 0. h_r classifies all instances of $\mathcal{C}'$ as 1, while w_r classifies some mass μ of them as 0 and instead classifies some mass μ from $\mathcal{C}^{\wedge}_{j+1}, \ldots \mathcal{C}''$ with score lower than that of $\mathcal{C}'$ as 1. This is where the two classifiers differ. Cells $\mathcal{C}^{\wedge}_{j+1}, \ldots \mathcal{C}''$ have a lower score and lesser proportion of disadvantaged instances than $\mathcal{C}'$. Reasoning similarly as Case 2, we observe that w_r is less fair, and less accurate than h_r. □

3.4 Differences with Respect to Previous Work

As mentioned before, our setup enjoys multiple advantages over Kleinberg and Mullainathan (2019).

1. The following assumption on the data distribution below, which is quite restrictive, is used by Kleinberg and Mullainathan (2019), but we do not use it for our results.

   Disadvantage assumption: Given cells $\mathcal{C}_i, \mathcal{C}_j \in f$ such that $\mathcal{S}(\mathcal{C}_i) < \mathcal{S}(\mathcal{C}_j)$, then

   $$\frac{\mu(\mathcal{C}_i^A)}{\mu(\mathcal{C}_i^D)} < \frac{\mu(\mathcal{C}_j^A)}{\mu(\mathcal{C}_j^D)}.$$

   This condition intuitively means that for every two feature vectors a and b such that instances having feature vector representation a have a higher chance of success than instances having feature vector representation b, instances having feature vector representation a have a higher chance of belonging to the advantaged group than instances having feature vector representation b.

2. They use the notion of equity (defined below) to quantify the fairness of a classifier, which essentially involves maximizing the number of minority group applicants the classifier labels positively.

   $$\text{Equity}(h_r) = \frac{\mathbb{E}(\text{Fraction of instances in } D \text{ that } h_r \text{ accepts})}{\text{Total fraction of instances } h_r \text{ accepts (i.e., } r)}$$

   Our notion of fairness is more aligned with accuracy. We believe that a desirable property of any notion of fairness is that a classifier that is perfectly accurate is also perfectly fair, which is something our notion satisfies but theirs does not. In addition, we also prove similar trade-off results for equity (Theorems 2, 3, 4).

Adding Equity to Theorem 1 If we additionally consider the notion of equity in the scenario of group agnostic simplifications as in Theorem 1, we get the result below.

Theorem 2. *For every non-trivial group-agnostic simplification of f', say w, there exists a classifier h that simultaneously strictly improves accuracy, fairness, and equity over w.*

3.5 General Case

Now we move on from group-agnostic simplifications to a more general notion of simplification, called graded simplification. Note that in group-agnostic simplifications, we constrained the classifier to always ignore/delete the protected group membership feature. Graded simplifications are more general, and do not suffer from this constraint.

Definition (Graded-simplification)**.** *Consider cell partition f' of $\mathcal{X}$: $\mathcal{C}'_1, \mathcal{C}'_2, \ldots, \mathcal{C}'_{2n}$. Consider simplification w of f' that partitions $\mathcal{X}$ into the cells $\mathcal{C}^{\wedge}_1, \mathcal{C}^{\wedge}_2, \ldots, \mathcal{C}^{\wedge}_t, \ldots, \mathcal{C}^{\wedge}_d$ with descending order of scores. Each cell $\mathcal{C}^{\wedge}_i \in w$ can be written as $\mathcal{C}^{\wedge}_i = \cup_{j=1}^{k} \mathcal{C}'_{i_j}$ (i.e., the union of some cells $\mathcal{C}'_{i_1}, \mathcal{C}'_{i_2}, \ldots, \mathcal{C}'_{i_k} \in f'$). We denote the set of indices $\{i_1, i_2 ... i_k\}$ corresponding to $\mathcal{C}^{\wedge}_i$ as $V(\mathcal{C}^{\wedge}_i)$.*

A graded simplification w of f' is one where each cell $\mathcal{C}^{\wedge}_i \in w$ has the property that either $V(\mathcal{C}_i^{\wedge^A}) \subseteq V(\mathcal{C}_i^{\wedge^D})$ or $V(\mathcal{C}_i^{\wedge^D}) \subseteq V(\mathcal{C}_i^{\wedge^A})$.

Result We first informally explain the result of this section. If we use a simpler classifier than f', say w, it might lead to an increase in fairness, interpretability, or equity, but we lose accuracy. We show that the simple classifier w is not optimal if we ignore the requirement of interpretability, as there exists a partition h (achievable by the features we have) that is simultaneously both more fair and accurate than w, while also improving equity. Therefore, we would strictly prefer h over w, if we ignore interpretability requirements, and therefore we see that interpretability clashes with the desiderata of fairness, accuracy, and equity.

Remark. *Unlike Theorem 2, the partition h does not guarantee an increase in equity. This makes sense, as we are now considering a more general notion of simplification.*

Theorem 3. *For every non-trivial graded-simplification, say w, there exists a partition h that simultaneously strictly improves accuracy and fairness, while also improving equity, with respect to w.*

Adding the Disadvantage Condition In Theorem 3, if we make the disadvantage assumption, we can find a partition h that simultaneously guarantees a strict increase in equity as well. That is, we get the following statement below.

Theorem 4. *For every non-trivial graded-simplification, say w, there exists a partition h that simultaneously strictly improves accuracy, fairness, and equity with respect to w.*

4 Trade-Offs between Fairness and Privacy

Informal Version of Result The result in this section essentially shows that even in a simple binary classification setting, there is no learning algorithm that is fair (even approximately, i.e., it is guaranteed to output an approximately fair classifier), and differentially private, while maintaining good accuracy. Hence, we see that, the properties of fairness, differential privacy, and accuracy, are at odds with each other and it is not possible to satisfy the three of them simultaneously.

4.1 Setup

Throughout, we use $\mathcal{X}$ to denote the domain set. There is probability distribution $\mathcal{D}$ over $\mathcal{X}$. The domain set consists of elements of the form $z = (x, a, y)$, where x refers to the element's features (e.g., this could be income, name, etc.), a is a protected (binary) attribute (as before we have an advantaged and a disadvantaged group, and use $a = 0$ to denote the protected class). y is a binary label, that is the thing we want to predict. Additionally, throughout, we assume that $y = 0$ denotes the *bad* label—meaning, for instance, in the context of, say, giving loans, this means that the person will not return the loan.

4.2 Privacy

The notion of privacy we consider is called differential privacy. Differential privacy aims to protect the privacy of each individual in a database. In the case of learning algorithms, the database is the training set.

Differential Privacy Differential privacy protects the privacy of an individual by ensuring that an algorithm will generate similar outputs on neighboring databases. It roughly protects the privacy of an individual in the database in the following way; changing an individual's entry, or deleting or adding it, will lead to what we call a neighboring database, and because the algorithm will generate similar outputs on neighboring databases, an observer seeing its output essentially cannot tell if a particular individual's information was used in the computation, or what that information is.

Definition ((ϵ, δ)-differential privacy (Dwork et al. 2006))**.** *For any $\epsilon, \delta \geq 0$, a randomized algorithm $\mathcal{A}$ is said to be (ϵ, δ)-differentially private if for all pairs of neighboring databases $\mathcal{D}$, $\mathcal{D}'$ and for all sets $S \in$ Range($\mathcal{A}$) of outputs,*

$$\Pr[\mathcal{A}(\mathcal{D}) \in S] \leq \exp(\epsilon) \cdot \Pr[\mathcal{A}(\mathcal{D}') \in S] + \delta.$$

Remark. *Although we have defined differential privacy in its full generality, note that throughout we will be talking about $(\epsilon, 0)$-differential privacy.*

Database We talk two different notions of a database. The first one, is a finite sample, with entries drawn i.i.d. from the distribution $\mathcal{D}$ over the domain $\mathcal{X}$. The second notion is to consider the whole distribution $\mathcal{D}$ as a database. The first notion is standard in the privacy literature, where databases are viewed as a finite collection of data points from n individuals. The second notion is standard for statistical notions of fairness, where the goal is to ensure fairness over a large population. Notion 2 can simply be considered a generalization of Notion 1. We will be using the second notion (also used by Cummings et al. (2019)), but the same results and proofs also work for the first notion.

Neighbouring Databases Given our definition of a database, it now remains to be defined what we mean by *neighboring databases*. Here we use the notion of σ-closeness, which is also used by Cummings et al. (2019).

Definition (σ-closeness (McGregor et al. 2010))**.** *Distributions (i.e., databases) $\mathcal{D}$ and $\mathcal{D}'$ are said be σ-close if*

$$\frac{1}{2} \sum_{z \in \mathcal{X}} |\mathcal{D}(z) - \mathcal{D}'(z)| \leq \sigma.$$

We calculate the distance between two distributions (databases) by the above expression. If the distance is lesser than σ, for some pre-specified value of σ, then the distributions are said to be neighboring.

4.3 Fairness

What notion of fairness do we use? Our results hold for pretty much all the common notions proposed in popular literature (for example: Demographic Parity, Equal Opportunity, Equalised Odds, etc., see appendix for definitions) (Dwork et al. 2012; Hardt et al. 2016; Verma and Rubin 2018). Essentially, any reasonable notion of fairness, that does not allow one group to be treated much worse than the other. More importantly, our results hold for even relaxed or approximate versions of these notions (This means that, for example, instead if demanding equality in false positive/negative rates for both groups, we require that there should not be a high difference in these rates between the two groups). Proving an impossibility for such relaxations makes our result stronger.

4.4 Result

Our main result is an incompatibility theorem showing how differential privacy and fairness are at odds with each other when we consider a learning algorithm with non-trivial accuracy. In particular, we consider a simple binary classification setting where the learning algorithm is given full access to the underlying distribution, and show that even under this severe restriction,[4] any learning algorithm that is $(\epsilon, 0)$-differentially private (for *any* $\epsilon \geq 0$), and even approximately fair, cannot achieve accuracy better than that of a constant classifier (that outputs the same label for every input).

Theorem 5. *If a learning algorithm $\mathcal{A}$ is $(\epsilon, 0)$-differentially private and is guaranteed to output an approximately fair classifier, then $\mathcal{A}$ is constrained to output a constant classifier, i.e., $\mathcal{A} : \tilde{D} \to \Delta(\mathcal{H})$, where $\tilde{D}$ denotes the set of all distributions, and*

$$\mathcal{H} = \{h : \mathcal{X} \to \{0, 1\} \mid h \text{ is a constant function}\}.$$

We start with an informal overview of the proof. The main idea in the proof is to first observe that, due to differential privacy constraints, if there is a classifier that is output with positive probability by $\mathcal{A}$ on a distribution $\mathcal{D}_1 \in \tilde{D}$, then $\mathcal{A}$ has to output this classifier with positive probability on any other distribution $\mathcal{D}_1' \in \tilde{D}$. Now, what the claim above implies is that, if algorithm $\mathcal{A}$ has to be (approximately) fair as well, and it outputs classifier h on some input distribution, then h is always (approximately) fair, irrespective of the underlying distribution. Now, once we have the observation above, then it just remains to show that such classifiers—i.e., ones that are (approximately) fair with respect to any underlying distribution—belong to a very restricted set, namely $\mathcal{H}$ as defined in the theorem. Below, we present a formal argument by first proving the following claim.

[4] The result also holds for the case where the algorithm has access to a finite training set, and not the underlying distribution. Giving the algorithm access to the underlying distribution is an easier task (equivalent to providing infinite training samples). Proving an impossibility for this case makes our result stronger.

Claim 6. *Let $\mathcal{A}$ be a learning algorithm that is $(\epsilon, 0)$-differentially private. Then, $\forall \mathcal{D}_1, \mathcal{D}'_1 \in \tilde{D}$, and for all classifiers h,*

$$\Pr[\mathcal{A}(\mathcal{D}_1) = h] > 0 \implies \Pr[\mathcal{A}(\mathcal{D}'_1) = h] > 0.$$

Proof. Consider an arbitrary distribution $\mathcal{D}_1 \in \tilde{D}$ and a classifier h such that $\Pr[A(\mathcal{D}_1) = h] > 0$. Next, consider any arbitrary distribution $\mathcal{D}'_1 \in \tilde{D}$. We need to show that $\Pr[\mathcal{A}(\mathcal{D}'_1) = h] > 0$.

To see this, first let us consider, for any $i \in [n]$ and $\eta > 0$, two η-close distributions $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$ (i.e., they are neighboring databases). Since $\mathcal{A}$ is ϵ-differentially private, if $\Pr[\mathcal{A}(\mathcal{D}_i) = h] > 0$, then we have that $\Pr[\mathcal{A}(\mathcal{D}_{i+1}) = h] > 0$, for if otherwise, then we have,

$$0 < \Pr[A(\mathcal{D}_i) = h] \leq \exp(\epsilon) \Pr[A(\mathcal{D}_{i+1}) = h] = 0,$$

which is a contradiction.

Now, given the observation above, observe that, for any $\eta > 0$, one can construct a (finite) series of distributions $\mathcal{D}_2, \cdots, \mathcal{D}_n$ such that $\forall i \in [n]$, $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$ are η-close (i.e., they are neighboring databases) and where $\mathcal{D}_{n+1} = \mathcal{D}'_1$. This in turn implies that we have,

$$\begin{aligned} \Pr[A(\mathcal{D}_1) = h] > 0 &\implies \Pr[A(\mathcal{D}_2) = h] > 0 \\ &\implies \Pr[A(\mathcal{D}_3) = h] > 0 \\ &\vdots \\ &\implies \Pr[A(\mathcal{D}_{n+1}) = h] > 0, \end{aligned}$$

where all the implications above are obtained by using the argument made above that for two neighboring databases $\mathcal{D}_i$ and $\mathcal{D}_{i+1}$, $\Pr[\mathcal{A}(\mathcal{D}_i) = h] > 0 \implies \Pr[\mathcal{A}(\mathcal{D}_{i+1}) = h] > 0$. This in turn proves our claim. □

Proof of Theorem 5. From Claim 6 we know that if a learning algorithm $\mathcal{A}$ is $(\epsilon, 0)$-differentially private and is guaranteed to output a fair classifier, then for all fair classifiers h and $\forall \mathcal{D}_1, \mathcal{D}'_1 \in \tilde{D}$, $\Pr[\mathcal{A}(\mathcal{D}_1) = h] > 0 \implies \Pr[\mathcal{A}(\mathcal{D}'_1) = h] > 0$. In other words, what this implies is that, for a fair learning algorithm $\mathcal{A}$, any fair classifier h that is output by $\mathcal{A}$ is fair with respect to any distribution in $\tilde{D}$. Below, we show how any h satisfying the property mentioned above should belong to $\mathcal{H}$, where $\mathcal{H}$ is as defined in the statement of the theorem.

To do this, consider for the sake of contradiction any $h \notin \mathcal{H}$. This implies that, there exist points $p_1 = (x_1, 0, y_1)$ and $p_2 = (x_2, 1, y_2)$ classified differently by h (Because h is not constant, we can find two points $a, b \in \mathcal{X}$ such that $h(a) \neq h(b)$. If they are in different groups, we are done. If they are in the same group, then choose any $c \in \mathcal{X}$ in the other group. It will hold that either $h(c) \neq h(a)$, or $h(c) \neq h(b)$, and we are done). Then, either of the following two cases holds:

1. $h(p_1) = 0$ and $h(p_2) = 1$, or
2. $h(p_1) = 1$ and $h(p_2) = 0$.

Now, if this is the case, then we will construct a distribution on which h is unfair. We construct a distribution for Case 1. To construct such a distribution, let us first consider the following points.

$$q_1 = (x_1, 0, 1) \qquad q_2 = (x_2, 1, 0)$$

Next, let us define the following distribution $\mathcal{D}'$.

$$\mathcal{D}'(q_1) = \frac{1}{2} \qquad \mathcal{D}'(q_2) = \frac{1}{2}$$

Note that $h(q_1) = 0$ and $h(q_2) = 1$. However, if this is the case, then note that by any reasonable notion of fairness, h is unfair to group 0 as compared to group 1, since group 0 always has true label 1 but is always labeled 0, whereas group 1 always has true label 0 but is always labeled 1.

We omit the construction for Case 2. Essentially the same idea as Case 1 can be used for Case 2. □

5 Conclusion and Future Work

Through this work, we see that in ML based decision systems, the desiderata of fairness, interpretability, and privacy are at odds with each other and it is necessary to make trade-offs between them, if we want to maintain accuracy. We prove two results to demonstrate this.

The first result considers a formal framework to build simple classifiers as a means to achieve interpretability, and shows that if we restrict our classifier to be simple within this framework, it can be replaced by a more complex classifier that strictly improves both fairness and accuracy. Therefore, we see that simplicity/interpretability clashes with the properties of fairness and accuracy. There are many variants of the setup that we could investigate for further work. While this result talks about the tradeoffs between fairness and simplicity, it is important to note that not all forms of building simple classifiers are captured by this framework (for e.g., linear classifiers). It would be interesting to investigate the compatibility between fairness and other notions of simplicity. Also, we deploy a particular objective function to quantify unfairness, and it might be worth looking into the interplay between interpretability and fairness for other fairness objectives.

The second result is an incompatibility theorem showing a setting where differential privacy and fairness are at odds with each other when we want a learning algorithm with non-trivial accuracy. In particular, we consider the task of learning a classifier for a simple binary classification setting and show that any learning algorithm that is $(\epsilon, 0)$-differentially private, and even approximately fair, cannot achieve accuracy better than that of a constant classifier. The current statement allows the the learning algorithm to be faced with any underlying distribution (without any restrictions). But in reality, it's probably more likely that the set of distributions the learning algorithm will encounter follow some niceness properties. So, if we restrict the distributions by these niceness properties, can we prove something similar? Additionally, in the result, we require each output classifier should be fair. An algorithm that generates a fair classifier with high probability could also be considered as fair, and such relaxations could definitely be looked at.

Another interesting direction of work could be to look at situations where one would want to have both interpretability, and privacy, and study the trade-offs these two requirements.

References

Bagdasaryan, E.; Poursaeed, O.; and Shmatikov, V. 2019. Differential privacy has disparate impact on model accuracy. In *Advances in Neural Information Processing Systems*, 15453–15462.

Chaudhuri, K.; Monteleoni, C.; and Sarwate, A. D. 2011. Differentially Private Empirical Risk Minimization. *Journal of Machine Learning Research* 12(29): 1069–1109. URL http://jmlr.org/papers/v12/chaudhuri11a.html.

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

Cummings, R.; Gupta, V.; Kimpara, D.; and Morgenstern, J. 2019. On the Compatibility of Privacy and Fairness. In *Fairness in User Modeling, Adaptation and Personalization (FairUMAP 2019)*.

Doleac, J. L.; and Hansen, B. 2016. Does "ban the box" help or hurt low-skilled workers? Statistical discrimination and employment outcomes when criminal histories are hidden. Technical report, National Bureau of Economic Research.

Doshi-Velez, F.; and Kim, B. 2017. Towards A Rigorous Science of Interpretable Machine Learning.

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

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, 265–284. Springer.

Dwork, C.; and Mulligan, D. K. 2013. It's not privacy, and it's not fair. *Stan. L. Rev. Online* 66: 35.

Dwork, C.; Roth, A.; et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9(3–4): 211–407.

Ekstrand, M. D.; Joshaghani, R.; and Mehrpouyan, H. 2018. Privacy for all: Ensuring fair and equitable privacy protections. In *Conference on Fairness, Accountability and Transparency*, 35–47.

Feldman, M.; Friedler, S. A.; Moeller, J.; Scheidegger, C.; and Venkatasubramanian, S. 2015. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 259–268. ACM.

Grgić-Hlača, N.; Zafar, M. B.; Gummadi, K. P.; and Weller, A. 2018. Beyond distributive fairness in algorithmic decision making: Feature selection for procedurally fair learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Hardt, M.; Price, E.; Srebro, N.; et al. 2016. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, 3315–3323.

Hotchkiss, M.; and Phelan, J. 2017. Uses of census bureau data in federal funds distribution. *US Dept. of Commerce, Econ. and Statistics Administration* .

Jagielski, M.; Kearns, M.; Mao, J.; Oprea, A.; Roth, A.; Sharifi-Malvajerdi, S.; and Ullman, J. 2019. Differentially Private Fair Learning. In *International Conference on Machine Learning*, 3000–3008.

Kleinberg, J.; and Mullainathan, S. 2019. Simplicity creates inequity: implications for fairness, stereotypes, and interpretability. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, 807–808.

Kleinberg, J.; Mullainathan, S.; and Raghavan, M. 2017. Inherent Trade-Offs in the Fair Determination of Risk Scores. In Papadimitriou, C. H., ed., *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, 43:1–43:23. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-95977-029-3. ISSN 1868-8969. doi:10.4230/LIPIcs.ITCS.2017.43. URL http://drops.dagstuhl.de/opus/volltexte/2017/8156.

Lipton, Z. C. 2018. The mythos of model interpretability. *Queue* 16(3): 31–57.

McGregor, A.; Mironov, I.; Pitassi, T.; Reingold, O.; Talwar, K.; and Vadhan, S. 2010. The limits of two-party differential privacy. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, 81–90. IEEE.

Pujol, D.; McKenna, R.; Kuppam, S.; Hay, M.; Machanavajjhala, A.; and Miklau, G. 2020. Fair Decision Making Using Privacy-Protected Data. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, 189–199. New York, NY, USA: Association for Computing Machinery. ISBN 9781450369367. doi:10.1145/3351095.3372872. URL https://doi.org/10.1145/3351095.3372872.

Rudin, C. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1(5): 206–215.

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

Appendix

Organisation We cover some parts in the appendix that were left out due to space constraints. The appendix is organised as follows. Section 6 continues Section 2 and discusses some other literature relevant to our work. Section 7 continues on Section 3, and starts off with some additional remarks about the framework we use. We then give a formal description of how the classifier works. We conclude the section by justifying our proposed notion of fairness, and including the missing proofs. Section 8 continues on Section 4, where we first go over some common fairness notions discussed in literature, and then go over the gap in the proof in Cummings et al. (2019).

6 Related Work (continued)

Although there are many settings where one might only care about one of the issues (either privacy, or fairness), they are not always mutually exclusive, for one can easily think of several scenarios where one might not only need privacy but also need to ensure that the procedure is fair. A typical example of such a setting is allocation of scarce resources—be it research funding, natural resources, loans, etc. Given this, it is imperative that the issues of privacy and fairness be studied together. However, unfortunately, there has been very little work that has looked at these issues simultaneously. Especially in light of the fact that the 2020 U.S. census is going to employ differential privacy and that the annual distribution of at least 675 billion dollars relies on census data (Hotchkiss and Phelan 2017), we believe that having a good understanding of the privacy-fairness trade-offs involved is of prime importance.

For the fairness-privacy trade-off result, apart from the work of Cummings et al. (2019), another important paper that was a motivation for pursuing this line of work was that of Pujol et al. (2020). Pujol et al. (2020) empirically show how there might be privacy-fairness trade-offs involved in certain settings. In particular, they consider three resource allocation settings and use census data to which noise has been added to demonstrate how adding noise so as to achieve differential privacy could disproportionately affect some groups over others in the settings that they consider. Besides the paper mentioned above, there is also work by Dwork and Mulligan (2013), and Ekstrand, Joshaghani, and Mehrpouyan (2018), where they consider the issues of privacy and fairness together. They intuitively discuss about why there may be trade-offs involved in deploying algorithms in real life scenarios, and advocate for the issues of privacy and fairness to be studied together. Bagdasaryan, Poursaeed, and Shmatikov (2019) empirically demonstrate that in neural networks trained using differentially private stochastic gradient descent (DP-SGD), accuracy of DP models drops much more for the underrepresented classes. Our work can be considered as essentially lending some theoretical support to the observations in the above works, as we prove how even in very simple settings it may be impossible to achieve fairness and privacy together, while maintaining accuracy.

There is also work by Jagielski et al. (2019), where they shows two algorithms that satisfy (ϵ, δ)-differential privacy and a particular notion of fairness (equalized odds). However, they consider the relaxed notion of approximate differential privacy ((ϵ, δ)-DP, $\delta > 0$), while we consider pure differential privacy ($(\epsilon, 0)$-DP) throughout.

7 Trade-Offs between Fairness and Interpretability (continued)

7.1 Additional Remarks

1. In practice, to build the classifier, we have access to a labeled sample of points generated by the underlying distribution (i.e., the training set). In our setup, we assume we have full access to the distribution and ground truth function to build the classifier. We do this because we want to analyse the behavior of the classifiers in isolation without considering any added complications due to sampling error.
2. In Theorems 1, 2, 3, and 4, note that we require partition h to be achievable with the features we have. If we do not require partition h to be achievable with the features we have, it is trivial to find an h that strictly improves in fairness and accuracy over any w (where w is a non-trivial simplification of f'). For example, the following partition would work: $h = \mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \mathcal{C}_3^\wedge, \mathcal{C}_4^\wedge$, where $\mathcal{C}_1^\wedge =$ Good instances in D, $\mathcal{C}_2^\wedge =$ Good instances in A, $\mathcal{C}_3^\wedge =$ Bad instances in D, $\mathcal{C}_4^\wedge =$ Bad instances in A. Here we ensure to not merge any cells in h while admitting instances.

7.2 Formal Description of Classifier

A given partition h of the domain set and admission rate r induces a threshold classifier that we denote by h_r. Consider an arbitrary partition h which partitions $\mathcal{X}$ into the cells $\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \ldots \mathcal{C}_t^\wedge, \ldots, \mathcal{C}_d^\wedge$. We sort the cells of h in descending order of their scores. Without loss of generality assume that h partitions $\mathcal{X}$ into cells $\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \ldots, \mathcal{C}_d^\wedge$ with decreasing (not necessarily strict) order of scores. We merge cells with the same scores to form a new partition h^* with cells $\mathcal{C}_1^*, \mathcal{C}_2^*, \ldots, \mathcal{C}_{d'}^*$, in strictly decreasing order of scores. Now, start admitting applicants in order as follows until you admit a fraction r of them. Let r_j be the fraction of the first j cells of h^* in the order they are represented. If $j(r)$ is the unique index j such that $r_{j-1} \leq r < r_j$, then the instances admitted consist of all the applicants in the cells $\mathcal{C}_1^*, \mathcal{C}_2^*, \ldots, \mathcal{C}_{j(r)-1}^*$, together with a subset of $\mathcal{C}_{j(r)}^*$ of fraction $(r - r_{j-1})$. The instances in $\mathcal{C}_{j(r)}^*$ to be admitted will be picked randomly.

7.3 Why this new notion to quantify fairness?

We believe that the notion we use to quantify (un)fairness is one of the contributions of the paper, and could be used widely in the future. Most notions of fairness just tell us when something is unfair, but do not quantify the amount of unfairness. The few notions that have been proposed previously to quantify unfairness have essentially been of the following form: Unfairness = $|R(A) - R(D)|$, which is the absolute value in the difference of some quantity R (such as error rate, or false positive/negative rates), between the advantaged group A and disadvantaged group D. Such notions

support the viewpoint that error rates, say 0 and 1 for A and D respectively is as unfair as error rates 1 and 0 for A and D respectively, which we do not agree with. We believe that the former should be considered more unfair than the latter. We propose a notion more in line with *affirmative action*, that actively supports the disadvantaged group. We penalise only the decisions that unfairly hurt the disadvantaged group (false negatives of the disadvantaged group, and false positives of the advantaged group). Also, our notion of fairness is more aligned with accuracy (than the one in Kleinberg and Mullainathan (2019)). We believe that a desirable property of any notion of fairness is that a classifier that is perfectly accurate is also perfectly fair, which is something our notion satisfies but theirs does not. In addition, we also prove similar results for the notion of fairness in Kleinberg and Mullainathan (2019), namely equity.

7.4 Proofs

Proof of Theorem 2. It is easy to see that the same classifier h as constructed in the proof of Theorem 1 works for this case as well.

□

Proof of Theorem 3. Consider simplification w. It partitions $\mathcal{X}$ into the cells $\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \ldots \mathcal{C}_t^\wedge, \ldots, \mathcal{C}_d^\wedge$ with descending order of scores. Take a non trivial cell of w, say $\mathcal{C}_t^\wedge$. Say $\mathcal{C}_t^\wedge$ is the union of $\mathcal{C}_a, \mathcal{C}_b, \ldots, \mathcal{C}_z \in f'$.

Case 1: $V(\mathcal{C}_t^{\wedge A}) \subseteq V(\mathcal{C}_t^{\wedge D})$

There exists a cell $\mathcal{C}_a$ such that $\mathcal{C}_a \in f', \mathcal{C}_a \subset \mathcal{C}_t^\wedge$, such that $\mathcal{C}_a$ has the highest score amongst all cells $\mathcal{C}_a, \mathcal{C}_b, \ldots, \mathcal{C}_z \subset \mathcal{C}_t^\wedge$ and only consists of disadvantaged instances.

Construct h as follows: Remove $\epsilon > 0$ mass of $\mathcal{X}$ from $\mathcal{C}_a$ to create a separate cell $\mathcal{C}'$. Denote the remainder of $\mathcal{C}_t^\wedge$ by $\mathcal{C}''$. Observe that $\mathcal{S}(\mathcal{C}') > \mathcal{S}(\mathcal{C}_t^\wedge) > \mathcal{S}(\mathcal{C}'')$. Take ϵ small enough to not change order of $\mathcal{C}''$ in the partition w (we can do this because of the genericity assumption). It should be in the same position as $\mathcal{C}_t^\wedge$ was before. The only change in the order is that $\mathcal{C}'$ jumps to some position ahead of $\mathcal{C}''$.

The new partition h is

$$\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \cdots, \mathcal{C}_j^\wedge, \mathcal{C}', \mathcal{C}_{j+1}^\wedge, \cdots \mathcal{C}_{t-1}^\wedge, \mathcal{C}'', \mathcal{C}_{t+1}^\wedge \cdots, \mathcal{C}_d^\wedge$$

with descending order of scores.

Similar to the proof of Theorem 1, it is easy to check that for all rates r, the fairness and accuracy of h is at least as good as w, and for at least one value of r, strictly better in both criteria. We also see that the equity does not reduce.

Case 2: $V(\mathcal{C}_t^{\wedge D}) \subseteq V(\mathcal{C}_t^{\wedge A})$

There exists a cell $\mathcal{C}_a$ such that $\mathcal{C}_a \in f', \mathcal{C}_a \subset \mathcal{C}_t^\wedge$, such that $\mathcal{C}_a$ has the lowest score amongst all cells $\mathcal{C}_a, \mathcal{C}_b, \ldots, \mathcal{C}_z \subset \mathcal{C}_t^\wedge$ and only consists of advantaged instances.

Construct h as follows: Remove $\epsilon > 0$ mass of $\mathcal{X}$ from $\mathcal{C}_a$ to create a separate cell $\mathcal{C}'$. Denote the remainder of $\mathcal{C}_t^\wedge$ by $\mathcal{C}''$. Observe that $\mathcal{S}(\mathcal{C}') < \mathcal{S}(\mathcal{C}_t^\wedge) < \mathcal{S}(\mathcal{C}'')$. Take $\epsilon > 0$ small enough to not change order of $\mathcal{C}''$ in the partition w. It should be in the same position as $\mathcal{C}_t^\wedge$ was before (We can do this because of the genericity assumption). The only change in the order is that $\mathcal{C}'$ jumps to some position behind $\mathcal{C}''$.

The new partition h is

$$\mathcal{C}_1^\wedge, \mathcal{C}_2^\wedge, \cdots, \mathcal{C}_{t-1}^\wedge, \mathcal{C}'', \mathcal{C}_{t+1}^\wedge \cdots, \mathcal{C}_v^\wedge, \mathcal{C}', \mathcal{C}_{v+1}^\wedge, \cdots \mathcal{C}_d^\wedge$$

with descending order of scores.

Similar to the proof of Theorem 1, it is easy to check that for all rates r, the fairness and accuracy of h is at least as good as w, and for at least one value of r, strictly better in both criteria. We also see that the equity does not reduce.

□

Proof of Theorem 4. We omit the proof because it is easy to see that essentially the same construction as in the main result of Kleinberg and Mullainathan (2019) works for this result as well.

□

8 Trade-Offs between Fairness and Privacy (continued)

8.1 Fairness definitions

Definition (Demographic parity)**.** *A binary classifier h satisfies demographic parity if with respect to random variables A and Y*

$$\Pr_{z\sim D}[h(z) = 1|A = 1] = \Pr_{z\sim D}[h(z) = 1|A = 0].$$

Definition (Equal opportunity (Hardt et al. 2016))**.** *A binary classifier h satisfies equal opportunity if with respect to random variables A and Y*

$$\Pr_{z\sim D}[h(z) = 1|Y = 1, A = 1] = \Pr_{z\sim D}[h(z) = 1|Y = 1, A = 0].$$

In words, h satisfies equal opportunity if it produces equal true positive rates across the two groups.

Definition (Equalized odds (Hardt et al. 2016))**.** *A binary classifier h satisfies equalized odds if*

- *h has equal false positive rates across the two groups, i.e., with respect to random variables A and Y*

$$\Pr_{z\sim D}[h(z) = 1|Y = 0, A = 1] = \Pr_{z\sim D}[h(z) = 1|Y = 0, A = 0]$$

- *h satisfies equal opportunity.*

8.2 Gap in Proof in Previous Work (Cummings et al. 2019, Theorem 1).

As mentioned previously, our result here is a stronger version to one claimed in the paper by Cummings et al. (2019), but their proof has a gap. Below we briefly describe what this gap is (in the proof of Theorem 1 in Section 3).

Error in Theorem 1. On a high level, what their proof tries to do is, given a distribution $\mathcal{D}$ and a classifier h that satisfies equal opportunity (and is output with non-zero probability) for this distribution, to construct a neighboring distribution $\mathcal{D}'$ on which h does not satisfy equal opportunity. Now, h is output with non-zero probability on $\mathcal{D}$, and because of differential privacy constraints, it is output with non-zero probability on $\mathcal{D}'$ as well. This would imply the algorithm is

not fair, because on input distribution $\mathcal{D}'$, it outputs an unfair classifier (h) with non-zero probability. However, there is error in this construction, and h does indeed satisfy fairness (equal opportunity) on the distribution $\mathcal{D}'$, contrary to what is claimed.

The equal opportunity notion of fairness requires that a classifier h satisfies (with respect to group A and label Y on distribution $\mathcal{D}'$)

$$\Pr_{D'}[h(z) = 1|Y = 1, A = 1] = \Pr_{D'}[h(z) = 1|Y = 1, A = 0].$$

It is therefore crucial for their proof to show inequality of group-conditional true positive classification rates for classifier h on the distribution $\mathcal{D}'$, denoted by

$$\gamma_{ya}(h) = \Pr_{D'}[h = 1|Y = y, A = a].$$

I.e., they require that

$$\gamma_{10}^{D'} \neq \gamma_{11}^{D'}.$$

which does not hold.

The claim is that

$$\gamma_{10}^{D'} = \frac{1}{4} - \tau \neq \frac{1}{4} + \tau = \gamma_{11}^{D'}.$$

However, it is easy to see that

$$\gamma_{10}^{D'} = 1 = \gamma_{11}^{D'}.$$

and therefore h does indeed satisfy equal opportunity on the distribution $\mathcal{D}'$, contrary to what is claimed.

The error seems to stem from an incorrect usage of conditional probability arguments, and unfortunately this error does not seem fixable within the same proof idea.

In any case, we do think that the statement is correct, and we prove a stronger claim.

□