Fair Inputs and Fair Outputs: The Incompatibility of Fairness in Privacy and Accuracy

Bashir Rastegarpanah
Boston University
bashir@bu.edu

Mark Crovella
Boston University
crovella@bu.edu

Krishna P. Gummadi
MPI-SWS
gummadi@mpi-sws.org

ABSTRACT

Fairness concerns about algorithmic decision-making systems have been mainly focused on the outputs (e.g., the accuracy of a classifier across individuals or groups). However, one may additionally be concerned with fairness in the inputs. In this paper, we propose and formulate two properties regarding the inputs of (features used by) a classifier. In particular, we claim that fair privacy (whether individuals are all asked to reveal the same information) and need-to-know (whether users are only asked for the minimal information required for the task at hand) are desirable properties of a decision system. We explore the interaction between these properties and fairness in the outputs (fair prediction accuracy). We show that for an *optimal* classifier these three properties are in general incompatible, and we explain what common properties of data make them incompatible. Finally we provide an algorithm to verify if the trade-off between the three properties exists in a given dataset, and use the algorithm to show that this trade-off is common in real data.

ACM Reference Format:
Bashir Rastegarpanah, Mark Crovella, and Krishna P. Gummadi. 2020. Fair Inputs and Fair Outputs: The Incompatibility of Fairness in Privacy and Accuracy. In *Adjunct Proceedings of the 28th ACM Conference on User Modeling, Adaptation and Personalization (UMAP '20 Adjunct), July 14–17, 2020, Genoa, Italy*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3386392.3399568

1 INTRODUCTION

As data-driven decision making systems are increasingly used in modern society in ways that affect individual lives, concerns have been raised about their ethical implications. In particular, recent years have witnessed a fast-growing number of studies on fairness in the decisions made by such systems, including works on developing notions to define, measures to quantify, and mechanism to ensure *fair outputs* (i.e., whether a decision system provides an equitable service to all of its users or groups of users). Despite the natural dependence of decision outcomes on data inputs, fairness concerns that incorporate the *inputs* of decision system are however less studied.

Traditionally, ethical concerns about the *inputs to* (i.e., data used by) decision systems have been the focus of "privacy" studies, while ethical concerns about the *outputs from* decision systems have been the focus of "fairness" studies. However, we observe that privacy and fairness originate from fundamentally different epistemic arguments. At a high-level, privacy concerns are rooted in a desire to protect individuals by limiting or enabling control over the information they reveal to the world. Fairness concerns, on the other hand, are rooted in a desire for equitable treatment of individuals (or groups of individuals). As such, privacy and fairness concerns can independently arise for both the inputs used and the outputs generated by decision systems.

As a motivating example, consider a decision problem where the goal is to decide whether an applicant should be offered a loan. The decision for each applicant is made based on answers that are collected to a number of demographic and financial status questions. In settings similar to this example, we recognize certain social concerns regarding the information that is gathered from each applicant. In particular, we raise two questions motivated by previous proposals and legal regulations:

First, considering each applicant individually, we ask *"what information is necessary for (i.e., what questions are relevant to) solving the decision problem at hand?"* In the loan eligibility problem for example, it seems unnecessary to ask about an applicant's height. Moreover, although it may often be necessary to ask about education level, an applicant who has an excellent credit score and a secure job may find a question about his education level to be unnecessary. Notice that as revealing each piece of information to the decision system is associated with a potential loss in privacy, this concern is related to protecting individuals' privacy.

The above consideration is reflected in the EU General Data Protection Regulation (GDPR) [37] as a principle called *data minimization*, which is defined as: *"Personal data shall be adequate, relevant and limited to what is necessary in relation to the purposes for which they are processed."*

The second ethical question arises when comparing the information used (i.e., set of questions asked) from different applicants. In particular, we ask *"how can using different pieces of information from different applicants amount to discrimination?"* For instance, a loan applicant may find it unfair that she is asked to answer a different set of questions comparing to another applicant.

In order to study these questions in a concrete setting, we consider a classifier and a set of input variables (features). We assume that the classifier is trained using all the features, and we study properties of the classifier when it is applied to a test set. Furthermore, we assume that the classifier is able to classify a given data point using any subset of the input features. In other words, for a given classification instance at the test time, the values of only a

UMAP '20 Adjunct, July 14–17, 2020, Genoa, Italy

ACM ISBN 978-1-4503-7950-2/20/07…$15.00
https://doi.org/10.1145/3386392.3399568

subset of input features may be revealed to the classifier, and the remaining feature values are set to *unknown*[1].

We observe that in such scenarios, one may ask *"What properties can make the set of data inputs used for each classification instance more socially desirable?"* Our first contribution is proposing two properties of classifiers regarding their inputs to address the above question; namely the *need-to-know principle* and the *fair privacy principle*. In the following we introduce each principle and their formal definitions will be presented in section 4.

Notice that we are not concerned with *how* a set of feature values are selected to be used for classifying each instance, but we are rather interested in checking whether an arbitrary set of features meets these properties[2].

The need-to-know principle. This property presents one way of formalizing *"data minimization"* [37] in a classification setting. We propose a formulation that is based on classification accuracy. Intuitively, the need-to-know principle requires that the decision system use only *the minimal amount of information* that is necessary for classifying a data point with a certain accuracy. This may for example result in restricting the use of irrelevant or proxy features.

The justification for this principle is rooted in respecting the privacy rights of individuals to not divulge information about themselves that is not needed for the task at hand. Such a consideration is an important argument for emerging privacy regulations in different countries that require data aggregators to justify the need to collect information about individuals [31, 37, 38].

The fair privacy principle. Intuitively, the fair privacy principle requires that the decision system use the *same information (i.e., data inputs)* about all individuals when making decisions. Put differently, the fair privacy principle prohibits a decision system from using more or less or different pieces of information about different individuals. The justification for the fair privacy principle is two-fold:

First, we observe that in many scenarios it is preferable to use the same data inputs for all individuals since it equalizes the opportunity to get beneficial outcomes. In the loan eligibility problem for example, if a decision system uses different input features for two different individuals say, Alice and Bob, Alice might wonder if she might have been offered a loan had she been asked to provide the same inputs as Bob, and vice versa.

We do not expect this argument to be desirable in every situation. For example, in the case of predicting recidivism rates, it may seem reasonable to ask for more information from one individual comparing to the others in order to achieve an accurate prediction. However, in other domains such as recruiting, it is often considered best practice for all candidates to be asked the same questions, i.e., provide the same data inputs. In fact such considerations have been the main inspiration for *structured interviews*.

Second, note that one approach to achieve "equitable treatment of individuals"— as the basic idea behind fairness— is equitable protection of individuals against disclosure of their private data. The authors in [16] suggest that a desirable property for a privacy protection mechanism is to provide its protections equitably to all its subjects. From this perspective, In decision scenarios where individuals would prefer to not divulge their private data and there is cost to revealing such data, it is preferable that all individuals bear equal cost, and our fair privacy principle guarantees such an equatable share of privacy costs.

Our running assumption in this paper is that the decision system (classifier) itself is a privacy adversary. This assumption is consistent with scenarios such as our loan application example. Thus we do not consider privacy notions that assume an adversary who is different from the party that collects and processes personal data (e.g., differential privacy [14]).

The trade-off. Our second contribution lies in exposing the trade-offs in simultaneously achieving the proposed fairness and privacy considerations for inputs as well as previously proposed fairness considerations for outputs. Specifically, after formalizing our proposed principles of need-to-know and fair privacy, we show that in general, an *optimal* classifier cannot simultaneously satisfy both principles and achieve fairness in outputs (defined as equal prediction accuracy for all individuals). We then provide a formal specification of all datasets in which this trade-off exists, and a practically efficient algorithm to verify whether a given dataset presents the trade-off.

While each of need-to-know and fair privacy is a desirable property by itself and it is natural to seek a classifier that satisfies both, we further explain why achieving these two properties simultaneously is particularly interesting. Assume a classifier is applied to solve our loan eligibility example. One may decide to achieve fair privacy by asking all applicants to provide answers to all the input features. While this trivial approach will satisfy fair privacy, we observe that for all applicants whose prediction would not change if using a subset of feature values, the need-to-know principle is violated[3].

Therefore, imposing need-to-know constraint can be seen as a way to eliminate trivial solutions for achieving fair privacy. On the other hand, using the same subset of input features for all the applicants such that need-to-know is respected will affect the prediction accuracy of those applicants for whom more data inputs are required. Our incompatibility result in fact formalizes this intuitive argument.

Finally, note that although optimal classifiers are rarely used in practice, our results pose a new challenge to the design of classifiers that aim at optimality: "how much one needs to compromise on optimality in order to simultaneously achieve fairness in the inputs and outputs of a classifier?"

2 RELATED WORK

While some recent work has focused on both privacy and fairness considerations for outputs [3, 11, 17, 22, 29], relatively little work (e.g., [18]) has examined fairness considerations for inputs. In this paper we introduce new notions that simultaneously capture both

[1] While we do not make additional assumptions about the classification algorithm, practical examples of classification with partially known inputs are models that can naturally handle different sets of input features (e.g. the naive Bayes), and a prediction model in which unknown feature are estimated using some imputation procedure.

[2] In practice, one needs to specify how the features are selected (e.g., by using methods that are suggested in [28, 41, 46, 48]). However, by studying these properties and their interaction regardless of the feature selection procedure, we show inherent trade-offs that cannot be avoided using any feature selection procedure.

[3] Another trivial solution is using a baseline classifier that does not use any feature values from all applicant. Notice that this trivial classifier violates the adequacy requirement of input data in the "data minimization" principle.

privacy and fairness properties of inputs in algorithmic decision systems, and explore their interaction with fairness properties of outputs. In the following we review related work on different societal aspects of decision-making systems including privacy and fairness.

Fairness in algorithmic decision-making. In recent years several empirical studies have shown how algorithmic decision systems are prone to unfair treatment of their users in different areas (e.g., online advertisement [44] and criminal justice [26]). For more examples we point the interested readers to survey papers [4, 39]. These findings have raised awareness about the importance of fair decision-making systems by regulatory authorities as well [32, 37].

Research on fair classification can be divided into two parts: formulating fairness nations and measures, and developing techniques to improve the fairness of algorithmic systems. Fairness notions can be categorized as those measuring group unfairness and those measuring unfairness at the level of individuals [13, 42]. Fairness-enhancing techniques in general fall into three categories based on the stage of the classification pipeline that they are employed: (i) *pre-processing* [6, 23], (ii) *in-processing* [1, 24, 50], and (iii) *post-processing* [10, 21]. In this paper we take the accuracy equality [47] approach to fairness in section 4.1, and we apply it at the individual level as it has been done previously in [36, 42].

Privacy. Privacy in information systems is generally understood using two concepts: limitation theory and control theory [45]. Using those theories, several methods have been proposed to protect the privacy of the users in practice such as differential privacy [14] k-anonymity [2] and cryptography [43]. The definitions of fairness in privacy that we put forward are concerned with which information about a user is used by the classifier, and so relate most closely to limitation theory.

Need-to-know as a privacy notion. Achieving *complete privacy* has been the goal of cryptography approaches such as secure multi-party computation (SMC). However, due to practical constraints such as computational efficiency and auditing purposes, the alternative goal of acquiring minimum neccessary data has become important as stated by regulations in different countries [31, 37, 38]. Our proposed need-to-know property follows the similar idea: the system should use the minimum amount of information from users to provide a certain level of quality of service. In a concurrent work Biega et al. [5] define the need to know principle for computational applications and tie it to concepts in data protection laws.

Cost-sensitive learning and privacy as cost. Our definitions of fairness in privacy can be expressed in terms of a cost associated with each feature. This follows a line of research in machine learning that focuses on settings in which acquiring feature values is associated with some cost. The goal then is to make the best possible prediction with minimum cost users incurred at the test time. Some examples are decision trees with minimal cost [27], test-cost sensitive Naive Bayes classification [8], and using a Markov decision process to sequentially acquire feature values [28, 41, 46].

A number of previous papers have associated privacy more explicitly with a cost [15, 33]. Note however that while these works consider privacy as feature costs, the general goal is that the privacy loss of each individual is minimized; there is no consideration of fairness of privacy.

Privacy and fairness. Recently both privacy and fairness researchers have recognized the importance of understanding the interaction between privacy and fairness in algorithmic decision systems [3, 11, 19, 22, 40]. However, as eloquently argued by Ekstrand et. al. [16], much work remains to be done in "characterizing under what circumstances and definitions privacy and fairness are simultaneously achievable?". Our results in this paper can be seen as an effort to answer this question by specifying some of the circumstances in which the interaction between privacy and fairness can be formally studied.

The authors in [16] also interpret fair privacy as whether a privacy scheme protects all individuals equally, and they raise questions about the implications of this property on other fairness notions; however their discussion remains at a high level.

From a practical viewpoint, some studies have proposed techniques to improve fairness and privacy at the same time [19, 20, 40]. Furthermore, the authors in [34, 35] provide a framework for empirical assessment of privacy risks associated with different individuals when different subsets (dataviews) of a dataset are used. This framework allows studying the trade-off between privacy risk and data utility (which in turn is linked to accuracy). However, they do not consider an explicit notion of fairness in privacy or in accuracy.

Differential Privacy and fairness. Another recent line of work considers a common pivacy notion, differential privacy [14], and studies its interaction with existing fairness notions. In particular authors in [11] prove that differential privacy is incompatible with satisfying equal false negative rates among groups, and they provide a differentially private classification algorithm that approximately satisfies group fairness guarantees with high probability. Furthermore, it has been shown that applying differential privacy implies unequal accuracy costs over different subgrups which results in decreasing fairness [3].

In this paper, instead of differential privacy, we use a privacy notion that is based on the set of revealed features of users, which allows us to compare the privacy loss of different users.

Incompatibility results. There are incompatibility results in the area of fairness in machine learning [9, 25]; however, they all consider different fairness measures defined for the output of a learning system, e.g., the trade-off between calibration, equal false positive rates, and equal false negative rates. In contrast, this paper introduces a trade-off between fairness properties related to the outputs and inputs of a classifier.

3 FORMULATION AND SETTING

We start by establishing notation and a number of definitions. We consider a set of features $F = \{f_1, \ldots, f_d\}$ in which each feature f_i takes values from the domain $\mathcal{F}_i$. A dataset $\mathcal{D}$ is a set of data points (feature vectors) $\mathbf{x}_i \in \mathcal{X}$ where $\mathcal{X} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_d$ together with the corresponding labels $y_i \in \mathcal{Y}$, i.e., $\mathcal{D} \subset \{(\mathbf{x}_i, y_i) | \mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}\}$. For notational convenience, we use $\mathcal{D}_\mathcal{X}$ to denote the set of feature vectors in $\mathcal{D}$, i.e., $\mathcal{D}_\mathcal{X} = \{\mathbf{x}_i | \exists y \in \mathcal{Y} \; s.t. \; (\mathbf{x}_i, y) \in \mathcal{D}\}$. For any $S \subseteq F$ and $\mathbf{x}_i$, $\Omega_S(\mathbf{x}_i)$ denotes feature vector $\mathbf{x}_i$ in which only values of the features in S are revealed.

Let X be a multivariate random variable that takes on values $\mathbf{x} \in \mathcal{D}_\mathcal{X}$, and $Y(X)$ be a random variable that denotes the true label of X in $\mathcal{D}$. If no information about X is known, the probability that

the label of X is $c \in \mathcal{Y}$ equals to[4]

$$Pr[Y(X) = c] = \frac{|\{(\mathbf{x}, y) \in \mathcal{D} | y = c\}|}{|\mathcal{D}|}$$

This probability changes if some features values in X are revealed. In particular, given that $\Omega_S(X) = \Omega_S(\mathbf{x}_i)$ we have:

$$Pr[Y(X) = c \mid \Omega_S(X) = \Omega_S(\mathbf{x}_i)] = \frac{|\{(\mathbf{x}, y) \in \mathcal{D} | \Omega_S(\mathbf{x}) = \Omega_S(\mathbf{x}_i) \wedge y = c\}|}{|\{(\mathbf{x}, y) \in \mathcal{D} | \Omega_S(\mathbf{x}) = \Omega_S(\mathbf{x}_i)\}|}$$

A classifier $\hat{Y}$ is a function that predicts the label of a given feature vector. We assume that $\hat{Y}$ is trained on all the features in F, and at the test time it is applied to data points in a dataset $\mathcal{D}$. Furthermore, We assume that $\hat{Y}$ can make a prediction using any subset of the feature values (see footnote 1). In particular, $\hat{Y}(\Omega_S(\mathbf{x}_i))$ denotes the predicted label for $\mathbf{x}_i$ by $\hat{Y}$ using feature set S. We do not make any assumption about $\hat{Y}$ being a deterministic or a probabilistic function.

$\hat{Y}(X)$ is a random variable that denotes the label predicted for X by $\hat{Y}$; similarly, $\hat{Y}(\Omega_S(X))$ is a random variable that denotes the label predicted for X by the classifier $\hat{Y}$ based on the features in S.

3.1 Predictive Power of a Feature Set

For a given dataset $\mathcal{D}$, we define the *predictive power* $\Phi_S(\mathbf{x}_i)$ of a feature set $S \subseteq F$ for a data point $\mathbf{x}_i \in \mathcal{D}_\mathcal{X}$, as the probability of the most probable label for X given that the values of the features in S are revealed by $\mathbf{x}_i$, i.e., $\Omega_S(X) = \Omega_S(\mathbf{x}_i)$. In other words,

$$\Phi_S(\mathbf{x}_i) = \max_{c \in \mathcal{Y}} Pr[Y(X) = c | \Omega_S(X) = \Omega_S(\mathbf{x}_i)]$$

If $\Phi_S(\mathbf{x}_i) = 1$, we say that $\mathbf{x}_i$ is distinguishable in $\mathcal{D}$ using feature set S.

3.2 Optimal Classifier

We first define the accuracy of a classifier for a data point using a subset of features.

Prediction Accuracy. The *accuracy* of the prediction $\hat{Y}(\Omega_S(\mathbf{x}_i))$ is the probability that the label predicted for X using the features in S is equal to the true label of X, given the feature values revealed by $\Omega_S(\mathbf{x}_i)$. In other words,

$$acc(\hat{Y}(\Omega_S(\mathbf{x}_i))) = Pr[\hat{Y}(\Omega_S(X)) = Y(X) | \Omega_S(X) = \Omega_S(\mathbf{x}_i)]. \quad (1)$$

An optimal classifier is then defined as follows.

Optimal Classifier. Given a dataset $\mathcal{D}$, an optimal classifier $\hat{Y}_{opt}$ is a classifier that for all data points in $\mathcal{D}$ and using any subset of features $S \subseteq F$, has the highest prediction accuracy. In other words, $\hat{Y}_{opt}$ satisfies the following[5]

$$\forall \mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \forall S \subseteq F, \forall \hat{Y};\ acc(\hat{Y}(\Omega_S(\mathbf{x}_i))) \leq acc(\hat{Y}_{opt}(\Omega_S(\mathbf{x}_i)))$$

The following lemma provides a convenient way for computing the accuracy of the predictions made by an optimal classifier. In particular, it states that for any data point in a given dataset, the accuracy of an optimal classifier using a set of features can be computed by finding the predictive power of that feature set for the corresponding data point. We later use this result to measure the performance of an optimal classifier by studying the characteristics of the dataset to which the classifier is applied.

Lemma 1. *A classifier is optimal for a given a dataset $\mathcal{D}$, if and only if for any $\Omega_S(\mathbf{x}_i)$ it returns the most probable label for X given that $\Omega_S(X) = \Omega_S(\mathbf{x}_i)$.*

The proof of lemma 1 is presented in appendix A.1.

Corollary 1. *The prediction accuracy of an optimal classifier for $\Omega_S(\mathbf{x}_i)$ is equal to the predictive power of set S for $\mathbf{x}_i$, i.e., $\Phi_S(\mathbf{x}_i)$.*

4 DESIRED PROPERTIES

We now present formalizations of three properties that involve privacy and fairness of classifiers. The properties that we define in this section depend on both the input features used, and the predictions made by a classifier. For a dataset $\mathcal{D}$, we use $S_i \subset F$ to denote the set of features used by a particular classifier to predict the label of data point $\mathbf{x}_i \in \mathcal{D}_\mathcal{X}$. Note that the following properties can be validated for any arbitrary choice of S_i for each data point.

We emphasize that here we are not concerned about how S_i is selected for a given data point and a particular classifier, but we are rather concerned with the social properties of using S_i compared to other feature sets $S'_i \subset F$. (see footnote 2.)

4.1 Output Property: Fair Prediction Accuracy

In order to define a measure for the fairness in the outputs of a classifier, we use the accuracy equality notion [47], and extend it to the individual level as has been suggested in [42].

For a classifier $\hat{Y}$ and a dataset $\mathcal{D}$, let S_i be the set of features used to predict the label of $\mathbf{x}_i$. $\hat{Y}$ satisfies fair prediction accuracy if labels of all data points are predicted with equal accuracy, i.e.,

$$\exists \gamma \in (0, 1]\ s.t.\ \forall \mathbf{x}_i \in \mathcal{D}_\mathcal{X},\ acc(\hat{Y}(\Omega_{S_i}(\mathbf{x}_i))) = \gamma \quad (2)$$

4.2 Input Property: Need to Know

The need-to-know property states that for any data point, using any proper subset of the features used by the classifier will decrease the prediction accuracy (i.e., the feature set S_i is minimal with respect to the prediction accuracy):

$$\forall \mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \forall S' \subset S_i,\ acc(\hat{Y}(\Omega_{S'}(\mathbf{x}_i))) < acc(\hat{Y}(\Omega_{S_i}(\mathbf{x}_i))) \quad (3)$$

Note that the need-to-know property does not imply that the prediction accuracy must improve monotonically as the number of features that are used by the classifier increases. Furthermore, although we consider accuracy as the criterion for which the use of data is minimized, other measures (e.g., false negative rate) may be more appropriate in specific applications. We leave studying the implications of such alternative definitions of need-to-know for future work.

4.3 Input Property: Fair Privacy

Fair privacy is determined by the input features used by classifier for each data point. We assume each feature is associated with a non-negative cost that denotes the privacy cost of revealing that

[4] In this paper we assume a finite sample model using the given dataset. Thus our setting is an instance of transductive learning as opposed to inductive learning in which the dataset is a sample from some distribution.

[5] This is an extension of the *Bayes optimal classifier* to the settings where any subset of features can be used to make a prediction.

Table 1: An illustrative dataset.

	features		
data point	f1	f2	label
$\mathbf{x}_1$	0	0	-
$\mathbf{x}_2$	0	1	-
$\mathbf{x}_3$	0	3	+
$\mathbf{x}_4$	2	3	-

Table 2: Predictive power of each feature set.

	feature sets			
data point	$\Phi_\emptyset$	$\Phi_{\{f1\}}$	$\Phi_{\{f2\}}$	$\Phi_{\{f1,f2\}}$
$\mathbf{x}_1$	3/4	2/3	1	1
$\mathbf{x}_2$	3/4	2/3	1	1
$\mathbf{x}_3$	3/4	2/3	1/2	1
$\mathbf{x}_4$	3/4	1	1/2	1

feature, and that the privacy cost of each feature is the same across all users. Let vector $\mathbf{c} \in \mathbb{R}^d_{\geq 0}$ denote the privacy costs of the features.

Fair privacy states that the total privacy costs of the used features are equal for all data points, i.e.,

$$\exists \ell \in \mathbb{R}\ s.t.\ \forall \mathbf{x}_i \in \mathcal{D}_X,\ \sum_{f_k \in S_i} \mathbf{c}(k) = \ell \tag{4}$$

There are at least two natural cases for the cost vector $\mathbf{c}$:

Feature Count. One may choose to treat the privacy costs of all features as equal. Setting $\mathbf{c} = c.\mathbf{1}_d$ implies that privacy fairness holds when the number of used features is the same for all data points, i.e.,

$$\exists k \in \mathbb{N}\ s.t.\ \forall \mathbf{x}_i \in \mathcal{D}_X,\ |S_i| = k \tag{5}$$

Feature Match. Another natural approach is to treat two feature sets as equal-privacy-cost if and only if they contain the same features. This can be formalized by making the total cost of every subset of the features distinct (e.g. $\mathbf{c} = \{2^n | 0 \leq n \leq d-1\}$). In this case, privacy fairness means that the exact same set of features are used to make a prediction for all data points:

$$\exists S \subseteq F\ s.t.\ \forall \mathbf{x}_i \in \mathcal{D}_X,\ S_i = S \tag{6}$$

5 THE TRADE-OFF

In this section we study how the different socially important properties of a classifier defined in Section 4 interact. In particular, since all the three properties have important social values, it is natural to ask whether they can be satisfied simultaneously. In other words, we ask whether it is possible for a classifier to use a particular set of input features for each test instance, and satisfy all three properties while maximizing prediction accuracy.

First, we show that there are situations (i.e., datasets) in which an optimal classifier cannot simultaneously satisfy fair privacy, fair accuracy, and need-to-know. Then we present a theorem that precisely characterizes all the datasets in which such a trade-off exists under our definitions. This implies that in general, achieving fairness in the inputs and the outputs of an optimal classifier are incompatible goals.

5.1 Presenting the Incompatibility

We show that the following proposition is true:

Proposition 1. *When applied to an arbitrary dataset, an optimal classifier cannot be guaranteed to simultaneously satisfy fair privacy, fair accuracy, and need-to-know, unless it is the trivial classifier that does not use any feature values for all data points.*

Proof. We provide an example of a dataset for which any optimal classifier can satisfy at most two of fair privacy, fair accuracy, and need-to-know. Table 1 presents our example dataset. The dataset contains two features (f_1 and f_2), and four data points with class labels $y \in \{+, -\}$. Figure 1 shows the data points in a 2D plane.

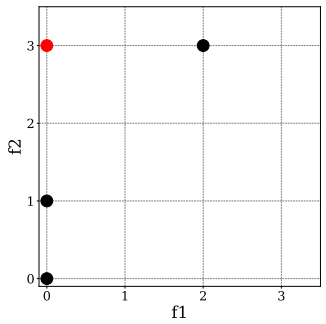

Figure 1

We present our arguments using two different feature cost vectors; each corresponds to one of *feature count* and *feature match* cases introduced in section 4.3.

Feature Count. Assume that privacy costs of all features are 1, i.e., $\mathbf{c} = \mathbf{1}_2$. Using corollary (1), we know that the prediction accuracy of an optimal classifier for each data point is equal to the predictive power of the selected feature set for that data point. Table 2 shows the predictive power of each subset of features for each data point in the dataset.

First, assume an optimal classifier that satisfies fair privacy. Considering the given feature cost vector, the privacy cost for each data point can be either 1 (the classifier uses either f_1 or f_2) or 2 (the classifier uses both f_1 and f_2). (Notice that the case of using no feature values for all data points is excluded from proposition1.) Therefore, in order to satisfy fair privacy, the privacy cost of all data points should be equal, and is either 1 or 2.

If the privacy cost is 1 for all data points (using either f_1 or f_2), from Table 2 we observe that it is not possible to have equal prediction accuracy for all data points. In particular, the prediction accuracy for $\mathbf{x}_3$ is $\frac{2}{3}$ using f_1 and $\frac{1}{2}$ using f_2. However, there is no way to have prediction accuracy of $\frac{2}{3}$ for $\mathbf{x}_4$, or $\frac{1}{2}$ for $\mathbf{x}_1$ and $\mathbf{x}_2$ using either f_1 or f_2. This violates fair prediction accuracy.

If the privacy cost is 2, all the labels can be predicted with accuracy 1.0. However, this violates the need to know property for data points $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_4$ because the same prediction accuracy could be reached using only f_2 for $\mathbf{x}_1$ and $\mathbf{x}_2$, or using f_1 for $\mathbf{x}_4$.

Therefore, any optimal classifier that satisfies fair privacy when applied on this dataset violates either the fair prediction accuracy or the need to know property. □

Feature Match. We could get the same result by assuming feature costs such that the total cost of every feature subset is distinct. In that case, fair privacy reduces to using the same set of features for every data point. From Table 2, we observe that the only feature set that satisfies fair accuracy, i.e., the only column with equal predictive power for all data points, is $\{f_1, f_2\}$; and using this set violates need-to-know for $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_4$.

5.2 Formal Specification

The previous section presents a dataset for which at most two of the properties from Section 4 can be satisfied. However, it remains to formalize when precisely a given dataset exhibits the trade-off, which we do in this section. We do this for an optimal classifier under the Feature Match definition of fair privacy (eq.6) and we leave generalizing to other definitions of fair privacy for future work. In the common case where privacy costs of the features are unknown, the Feature Match definition— i.e., using the same set of features for all individuals— is a reasonable choice.

Similar to the previous section, in the following we exclude the trivial classifier that does not use any feature values for all data points.

Theorem 1. *There exists an optimal non-trivial classifier that satisfies fair privacy, fair accuracy, and need-to-know when applied to a dataset $\mathcal{D}$, if and only if $\mathcal{D}$ satisfies the following condition:*

$$\begin{aligned} &\exists \text{ "non-empty } S\text{"} \subseteq F \; s.t., \\ &\qquad \exists \gamma \in (0,1] \; s.t. \; \forall \mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \; \Phi_S(\mathbf{x}_i) = \gamma \\ &\qquad \wedge \\ &\qquad \forall \mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \forall S' \subset S, \; \Phi_{S'}(\mathbf{x}_i) < \Phi_S(\mathbf{x}_i) \end{aligned} \tag{7}$$

The proof of theorem 1 is presented in appendix A.2.

Theorem 1 provides a necessary and sufficient condition (eq.7) to identify datasets for which an optimal classifier can simultaneously satisfy all the three properties. Notice that this condition is an statement about a dataset and can be verified independently of any classifier. The statement can be written as the following description of a dataset:

"There is a non-empty feature set that has equal predictive power for all data points in the dataset. Furthermore, all subsets of that feature set have lower predictive power for all points in the dataset."

Consequently, the negation of (7) provides a necessary and sufficient condition for the case where any optimal classifier can satisfy at most two of fair prediction accuracy, fair privacy, and need to know (i.e., datasets in which there is a trade-off between the above-mentioned properties of any optimal classifier). By negating (7) we find the following characterization of such datasets:

$$\begin{aligned} &\forall \text{ "non-empty } S\text{"} \subseteq F, \\ &\qquad \exists \mathbf{x}_i, \mathbf{x}_j \in \mathcal{D}_\mathcal{X} \; s.t. \; \Phi_S(\mathbf{x}_i) \neq \Phi_S(\mathbf{x}_j) \\ &\qquad \vee \\ &\qquad \exists \mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \exists S' \subset S \; s.t. \; \Phi_{S'}(\mathbf{x}_i) \geq \Phi_S(\mathbf{x}_i) \end{aligned} \tag{8}$$

For an intuitive interpretation of the above statement, assume an optimal classifier that satisfies fair privacy, i.e., set S is used for all data points in the dataset. Therefore, in order to exhibit the trade-off, using S the classifier should either violate fair prediction accuracy (first clause in (8)), or need-to-know (second clause in (8)). Thus, showing that for all non-empty $S \subseteq F$ either fair prediction accuracy or need-to-know are violated implies that no optimal non-trivial classifier can satisfy all three properties.

Corollary 2. *Given a data set $\mathcal{D}$ and a non-trivial classifier $\hat{Y}$, if $\mathcal{D}$ satisfies (8) and $\hat{Y}$ satisfies fair privacy, fair accuracy, and need-to-know when applied to $\mathcal{D}$, then $\hat{Y}$ is not optimal* [6].

6 THE TRADE-OFF IN REAL DATA

Given the results in the previous section, it is worthwhile to ask whether this trade-off is typical – does it occur often in real-world data? We first develop a practical approach to answering this question for a given dataset, and then we apply our approach to various datasets from the standard UCI machine learning repository [12].

Algorithm 1: verify if a given dataset holds the trade-off between fair accuary, fair privacy, and need-to-know properties for an optimal classifier.

Input: Dataset $\mathcal{D}$ with feature set F
Output: Yes/No

```
initialize queue Q
C = [ ]
Q.put({})
for f in F do
    if f has identical value over all data points then
        remove f from F
while Q is not empty do
    S = Q.get()
    if S ≠ ∅ then
        C.append(S)
    compute Φ_S(x_i) for all x_i ∈ D
    if Φ_S(x_i) ≠ 1 for all x_i ∈ D then
        for all features f in F whose index is larger than the largest
          index in S do
            Q.put(S ∪ {f})
for candidate S in C do
    if S satisfies the 1st clause in (8) then
        continue
    if S satisfies the 2nd clause in (8) then
        continue
    else
        return No
return Yes
```

6.1 A Verification Algorithm

For any given dataset, we may apply (8) to test it, since (8) is a predicate that identifies all and only those datasets for which the trade-off is present. A naive approach to evaluating (8) consists of computing $\Phi_{S_k}(\mathbf{x}_i)$ for all subsets $S_k \subseteq F$ and all $\mathbf{x}_i \in \mathcal{D}$. If for each S_k at least one of the two clauses in (8) are satisfied, no optimal classifier can simultaneously satisfy fair accuracy, fair privacy,

[6] An example of such a non-optimal classifier is provided in appendix B.

and need-to-know when applied on $\mathcal{D}$. However, the universal quantifier in (8) implies a search over the exponential number of subsets in the power set of F. Hence, we must consider how to efficiently verify that a given dataset satisfies (8). In this section, we introduce a verification algorithm that exploits several structures in the feature susbsets to prune the search space and is efficient in practice.

The pseudocode of our dataset verification algorithm is provided in algorithm 1. The algorithm first generates feature subsets (candidates) for which an optimal classifier could possibly satisfy both fair accuracy and need-to-know. Then it eliminates each candidate that satisfies at least one of the two clauses in (8). The algorithm uses an incremental method to generate candidates (i.e., larger sets are generated by adding more features to each of the existing candidates.) This allows the algorithm to recognize many of the candidates that will satisfy (8) before actually generating them. This is a key tool for pruning the search space and obtaining a practical algorithm.

The first pruning step is to notice that if a feature f_i has identical values for all data points, removing f_i from a feature subset S does not change the predictive power of that feature subset. That is, $\Phi_S(\mathbf{x}_i) = \Phi_{S\setminus f_i}(\mathbf{x}_i)$ for all $\mathbf{x}_i \in \mathcal{D}$. Therefore, any feature subset that contains f_i violates need-to-know. Consequently, we do not use such features in our candidate generation procedure (lines 4-6).

The second pruning step is to notice that if $\Phi_{S_k}(\mathbf{x}_i) = 1$ for some $S_k \subseteq F$ and some $\mathbf{x}_i$, then any superset S_k^* of S_k violates need to know property (i.e., second clause of (8)). This is because predictive power cannot be larger than 1. Therefore, we can prune from our search space all the supersets of any feature set whose predictive power is 1 for at least one data point. As we generate new subsets, we compute the predictive power of each subset for all data points, and we stop adding more features to that subset once a data point is distinguishable in the dataset using that subset (lines 7-14).

Finally, for each generated feature subset we first verify the first clause of (8); if it is not satisfied we verify the second clause (lines 15-21). Notice that the time complexity of verifying the first clause is linear in the size of the feature subset while the complexity of verifying the second clause is exponential in the size of the feature subset. If all the candidates (generated feature subsets) satisfy at least one of the two clauses in (8), we conclude that the given dataset holds our desired trade-off, i.e., an optimal classifier can satisfy at most two of fair accuracy, fair privacy, and need to know properties for the given dataset.

6.2 Verifying Real Data

Using Algorithm 1, we find that it is possible to test reasonable-sized datasets and determine whether they exhibit the trade-off. We obtain 18 datasets which have discrete feature domains from the UCI machine learning repository [12], and apply our verification algorithm to check if the trade-offs introduced in section 5 exist in each dataset. Table 4 in appendix C summarizes the datasets and the performance of the verification algorithm for each dataset.

We observe that the size of the largest generated candidate for most of the datasets is significantly smaller than the number of features in that dataset, which shows that the superset pruning procedure is effective. The verification algorithm terminates in less than a minute for all cases even though a complete search over the power set of the features would be infeasible in most cases. Also notice that except in one case (Nursery dataset), only verifying the first clause of (8) is enough for all data points.

Our algorithm verifies that every dataset we examined exhibits the trade-off between the three properties— hinting that the trade-off is prevalent in real-world data with discrete feature domains.

7 DISCUSSION AND CONCLUDING REMARKS

In this paper we argue that the fairness notions for algorithmic decision-making systems should expand to incorporate the inputs (i.e., features) used by a system, and we formulate two of such input properties: *fair privacy* and *need-to-know*.

We prove that in general an *optimal* classifier cannot satisfy all of fair privacy, need-to-know, and fair accuracy. Furthermore, we characterize all the datasets in which the above trade-off exists using logical predicates. Finally, we provide an algorithm that exploits several computational efficiencies to verify if the trade-off is present in a given dataset.

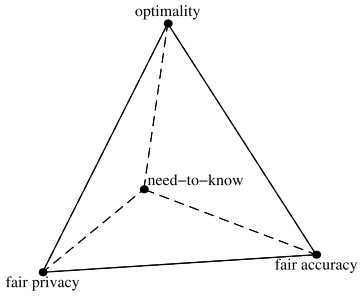

Figure 2

The tetrahedron in Figure 2 can be used to summarize our results. In particular, the properties at any three vertices are satisfiable, but all four can not be satisfied in general. We believe that each vertex offers a potentially interesting direction for future exploration.

First, if one sets aside optimality to achieve the three socially desirable properties, the question arises then how close to optimal can the performance of such a fair-input and and fair-output classifier be on a given dataset.

Second, if one instead sets aside fair privacy, one may seek to achieve the other goals, perhaps following in the general style taken in [30], i.e., using different input features from different individuals.

Third, one may rather choose to set aside need-to-know. For example, the authors in [7] equalize false positive, false negative, false discovery, and false omission rates across the protected groups by deferring on some decisions (i.e., avoid making a decision for some individuals). However, deferred decisions violate our need-to-know principle which requires the system to only use data inputs that are necessary for improving its predictions.

Finally, one may set aside fair accuracy, perhaps in favor of weaker conditions such as fair mistreatment [7, 49]. In that case, the question remains open whether other properties are achievable.

Acknowledgements. This research is based upon work supported by ERC Advanced Grant "Foundations for Fair Social Computing" (No. 789373), and the National Science Foundation under grant numbers IIS-1421759 and CNS-1618207.

REFERENCES

[1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudik, John Langford, and Hanna Wallach. 2018. A Reductions Approach to Fair Classification. In *International Conference on Machine Learning*. 60–69.

[2] Charu C Aggarwal. 2005. On k-anonymity and the curse of dimensionality. In *Proceedings of the 31st international conference on Very large data bases*. VLDB Endowment, 901–909.

[3] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential privacy has disparate impact on model accuracy. In *Advances in Neural Information Processing Systems*. 15453–15462.

[4] Solon Barocas and Andrew D Selbst. 2016. Big data's disparate impact. *Calif. L. Rev.* 104 (2016), 671.

[5] Asia J. Biega, Peter Potash, Hal Daumé, III, Fernando Diaz, and Michèle Finck. 2020. Operationalizing the Legal Principle of Data Minimization for Personalization. In *The 43rd International ACM SIGIR Conference on Research & Development in Information Retrieval*.

[6] Flavio Calmon, Dennis Wei, Bhanukiran Vinzamuri, Karthikeyan Natesan Ramamurthy, and Kush R Varshney. 2017. Optimized pre-processing for discrimination prevention. In *Advances in Neural Information Processing Systems*. 3992–4001.

[7] Ran Canetti, Aloni Cohen, Nishanth Dikkala, Govind Ramnarayan, Sarah Scheffler, and Adam Smith. 2019. From soft classifiers to hard decisions: How fair can we be?. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM, 309–318.

[8] Xiaoyong Chai, Lin Deng, Qiang Yang, and Charles X Ling. 2004. Test-cost sensitive naive bayes classification. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*. IEEE, 51–58.

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

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

[11] Rachel Cummings, Varun Gupta, Dhamma Kimpara, and Jamie Morgenstern. 2019. On the Compatibility of Privacy and Fairness. In *Adjunct Publication of the 27th Conference on User Modeling, Adaptation and Personalization* (Larnaca, Cyprus) *(UMAP'19 Adjunct)*. ACM, New York, NY, USA, 309–315.

[12] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

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

[14] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[15] Kirstin Early, Stephen E Fienberg, and Jennifer Mankoff. 2016. Test time feature ordering with FOCUS: Interactive predictions with minimal user burden. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 992–1003.

[16] Michael D Ekstrand, Rezvan Joshaghani, and Hoda Mehrpouyan. 2018. Privacy for All: ensuring fair and equitable privacy protections. In *Conference on Fairness, Accountability and Transparency*. 35–47.

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

[18] Nina Grgic-Hlaca, Elissa M Redmiles, Krishna P Gummadi, and Adrian Weller. 2018. Human perceptions of fairness in algorithmic decision making: A case study of criminal risk prediction. In *Proceedings of the 2018 World Wide Web Conference*.

[19] Sara Hajian, Josep Domingo-Ferrer, Anna Monreale, Dino Pedreschi, and Fosca Giannotti. 2015. Discrimination-and privacy-aware patterns. *Data Mining and Knowledge Discovery* 29, 6 (2015), 1733–1782.

[20] Sara Hajian, Anna Monreale, Dino Pedreschi, Josep Domingo-Ferrer, and Fosca Giannotti. 2012. Injecting discrimination and privacy awareness into pattern discovery. In *2012 IEEE 12th International Conference on Data Mining Workshops*. IEEE, 360–369.

[21] Moritz Hardt, Eric Price, Nati Srebro, et al. 2016. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*. 3315–3323.

[22] Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. 2019. Differentially Private Fair Learning. In *International Conference on Machine Learning*. 3000–3008.

[23] Faisal Kamiran and Toon Calders. 2012. Data preprocessing techniques for classification without discrimination. *Knowledge and Information Systems* 33, 1 (2012), 1–33.

[24] Toshihiro Kamishima, Shotaro Akaho, and Jun Sakuma. 2011. Fairness-aware learning through regularization approach. In *Data Mining Workshops (ICDMW), 2011 IEEE 11th International Conference on*. IEEE, 643–650.

[25] Jon Kleinberg, Sendhil Mullainathan, and Manish Raghavan. 2016. Inherent trade-offs in the fair determination of risk scores. *arXiv preprint arXiv:1609.05807* (2016).

[26] Jeff Larson, Surya Mattu, Lauren Kirchner, and Julia Angwin. 2016. How we analyzed the COMPAS recidivism algorithm. *ProPublica (5 2016)* 9 (2016).

[27] Charles X Ling, Qiang Yang, Jianning Wang, and Shichao Zhang. 2004. Decision trees with minimal costs. In *Proceedings of the twenty-first international conference on Machine learning*. ACM, 69.

[28] Shlomi Maliah and Guy Shani. 2018. MDP-based cost sensitive classification using decision trees. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

[29] Luca Melis, Congzheng Song, Emiliano De Cristofaro, and Vitaly Shmatikov. 2019. Exploiting unintended feature leakage in collaborative learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 691–706.

[30] Alejandro Noriega-Campero, Michiel A Bakker, Bernardo Garcia-Bulle, and Alex'Sandy' Pentland. 2019. Active Fairness in Algorithmic Decision Making. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. ACM, 77–83.

[31] US Department of Health, Human Services, et al. 2003. Health information privacy. The privacy rule.

[32] Executive Office of the President. 2014. Big data: Seizing opportunities, preserving values.

[33] Erman Pattuk, Murat Kantarcioglu, Huseyin Ulusoy, and Bradley Malin. 2015. Privacy-aware dynamic feature selection. In *2015 IEEE 31st international conference on data engineering*. IEEE, 78–88.

[34] Francesca Pratesi, Lorenzo Gabrielli, Paolo Cintia, Anna Monreale, and Fosca Giannotti. 2020. PRIMULE: Privacy risk mitigation for user profiles. *Data & Knowledge Engineering* 125 (2020), 101786.

[35] Francesca Pratesi, Anna Monreale, Roberto Trasarti, Fosca Giannotti, Dino Pedreschi, and Tadashi Yanagihara. 2018. PRUDEnce: a system for assessing privacy risk vs utility in data sharing ecosystems. *Transactions on Data Privacy* 11, 2 (2018), 139–167.

[36] Bashir Rastegarpanah, Krishna P Gummadi, and Mark Crovella. 2019. Fighting Fire with Fire: Using Antidote Data to Improve Polarization and Fairness of Recommender Systems. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. ACM, 231–239.

[37] Protection Regulation. 2016. Regulation (EU) 2016/679 of the European Parliament and of the Council. *REGULATION (EU)* 679 (2016), 2016.

[38] Joel R Reidenberg. 1994. Setting standards for fair information practice in the US private sector. *Iowa L. Rev.* 80 (1994), 497.

[39] Andrea Romei and Salvatore Ruggieri. 2014. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review* 29, 5 (2014), 582–638.

[40] Salvatore Ruggieri, Sara Hajian, Faisal Kamiran, and Xiangliang Zhang. 2014. Anti-discrimination analysis using privacy attack strategies. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 694–710.

[41] Hajin Shim, Sung Ju Hwang, and Eunho Yang. 2018. Joint active feature acquisition and classification with variable-size set encoding. In *Advances in Neural Information Processing Systems*. 1375–1385.

[42] Till Speicher, Hoda Heidari, Nina Grgic-Hlaca, Krishna P Gummadi, Adish Singla, Adrian Weller, and Muhammad Bilal Zafar. 2018. A Unified Approach to Quantifying Algorithmic Unfairness: Measuring Individual & Group Unfairness via Inequality Indices. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2239–2248.

[43] Douglas R Stinson. 2005. *Cryptography: theory and practice*. Chapman and Hall/CRC.

[44] Latanya Sweeney. 2013. Discrimination in online ad delivery. *Queue* 11, 3 (2013), 10–29.

[45] Herman T Tavani. 2007. Philosophical theories of privacy: Implications for an adequate online privacy policy. *Metaphilosophy* 38, 1 (2007), 1–22.

[46] Kirill Trapeznikov and Venkatesh Saligrama. 2013. Supervised sequential classification under budget constraints. In *Artificial Intelligence and Statistics*. 581–589.

[47] Sahil Verma and Julia Rubin. 2018. Fairness definitions explained. In *2018 IEEE/ACM International Workshop on Software Fairness (FairWare)*. IEEE, 1–7.

[48] Jihoon Yang and Vasant Honavar. 1998. Feature subset selection using a genetic algorithm. In *Feature extraction, construction and selection*. Springer, 117–136.

[49] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*. 1171–1180.

[50] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. 2017. Fairness Constraints: Mechanisms for Fair Classification. In *Artificial Intelligence and Statistics*. 962–970.

A PROOFS

A.1 Lemma 1

We write the right hand side of (1) as:

$$\sum_{c\in\mathcal{Y}} Pr[\hat{Y}(\Omega_S(X)) = c, Y(X) = c|\Omega_S(X) = \Omega_S(\mathbf{x}_i)] \tag{9}$$

Since $\hat{Y}(\Omega_S(X))$ only depends on the values of the features in S, $\hat{Y}(\Omega_S(X))$ and $Y(X)$ are conditionally independent given a set of fixed values $\Omega_S(X) = \Omega_S(\mathbf{x}_i)$. Therefore (9) is equal to:

$$\sum_{c\in\mathcal{Y}} Pr[\hat{Y}(\Omega_S(X)) = c|\Omega_S(X) = \Omega_S(\mathbf{x}_i)]Pr[Y(X) = c|\Omega_S(X) = \Omega_S(\mathbf{x}_i)] \tag{10}$$

For any $\Omega_S(\mathbf{x}_i)$, let $p_c = Pr[Y(X) = c|\Omega_S(X) = \Omega_S(\mathbf{x}_i)]$ and $p^* = \max_{c\in\mathcal{Y}} p_c$. Also let $\hat{p}_c = Pr[\hat{Y}(\Omega_S(X)) = c|\Omega_S(X) = \Omega_S(\mathbf{x}_i)]$ for an arbitrary classifier $\hat{Y}$. Using (10) we can write the following for any classifier $\hat{Y}$:

$$acc(\hat{Y}(\Omega_S(\mathbf{x}_i))) = \sum_{c\in\mathcal{Y}} p_c\hat{p}_c \leq \sum_{c\in\mathcal{Y}} p^*\hat{p}_c = p^*$$

Therefore, p^* is an upper bound for the prediction accuracy of any classifier applied on $\Omega_S(\mathbf{x}_i)$. Now let $c^* = \arg\max_{c\in\mathcal{Y}} p_c$, the prediction accuracy of a classifier that deterministically returns c^* for $\Omega_S(\mathbf{x}_i)$ (i.e., $\hat{p}_c = 0$ for all $c \neq c^*$, and $\hat{p}_{c^*} = 1$) is p^*; therefore, such classifier is optimal.

Finally, we show that the prediction accuracy of any classifier with $\hat{p}_{c^*} < 1$ is lower than p^*. Assume a classifier $\hat{Y}'$ for which $\hat{p}_{c^*} = 1 - \epsilon$ and $\hat{p}_{c'} = \epsilon$ for some $\epsilon > 0$ and some $c' \in \mathcal{Y}$ such that $p_{c'} < p^*$. Thus,

$$acc(\hat{Y}'(\Omega_S(\mathbf{x}_i))) = p^*(1-\epsilon) + p_{c'}\epsilon = p^* + \epsilon(p_{c'} - p^*) < p^* \quad \square$$

A.2 Theorem 1

For a classifier $\hat{Y}$ applied to a dataset $\mathcal{D}$, let S_i denote the set of features used from $\mathbf{x}_i$. First we repeat and name the following definitions from section 4,

Fair Accuracy *(p1)*:

$$\exists\gamma \in (0,1]\ s.t.\ \forall\mathbf{x}_i \in \mathcal{D}_\mathcal{X},\ acc(\hat{Y}(\Omega_{S_i}(\mathbf{x}_i))) = \gamma$$

Fair Privacy *(p2)*:

$$\exists S \subseteq F\ s.t.\ \forall\mathbf{x}_i \in \mathcal{D}_\mathcal{X},\ S_i = S$$

Need-To-Know *(p3)*:

$$\forall\mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \forall S' \subset S_i,\ acc(\hat{Y}(\Omega_{S'}(\mathbf{x}_i))) < acc(\hat{Y}(\Omega_{S_i}(\mathbf{x}_i)))$$

Let $\mathcal{H}_{opt}$ be the set of all optimal non-trivial classifiers for $\mathcal{D}$. Assume there exists an optimal non-trivial classifier that satisfies all of *p1*, *p2*, and *p3* when applied on $\mathcal{D}$, i.e., the following statement is true:

$$\exists\hat{Y} \in \mathcal{H}_{opt}\ s.t.\ p1 \wedge p2 \wedge p3 \tag{11}$$

From *p2* we infer that the same set of features is used by the classifier to predict the label of all the data points. We call this set S and we replace S_i with S in *p1* and *p3* (S is non-empty since $\hat{Y}$ is non-trivial). Moreover, since $\hat{Y}$ is an optimal classifier, by Corollary (1) we can replace the prediction accuracy of $\hat{Y}$ for any data point and any feature set with the predictive power of that feature set for that data point. Thus, from (11) we infer that the following statement is true:

$$\begin{aligned}&\exists \text{ “non-empty } S\text{”} \subseteq F\ s.t.,\\ &\qquad \exists\gamma \in (0,1]\ s.t.\ \forall\mathbf{x}_i \in \mathcal{D}_\mathcal{X},\ \Phi_S(\mathbf{x}_i) = \gamma\\ &\qquad \wedge\\ &\qquad \forall\mathbf{x}_i \in \mathcal{D}_\mathcal{X}, \forall S' \subset S,\ \Phi_{S'}(\mathbf{x}_i) < \Phi_S(\mathbf{x}_i)\end{aligned} \tag{12}$$

On the other hand, assume we are given a dataset for which statement (12) is true. We can then define a classifier $\hat{Y}$ such that it uses the features in S for all $\mathbf{x}_i \in \mathcal{D}$, and it returns $\arg\max_{c\in\mathcal{Y}} Pr[Y(X) = c|\Omega_{S_k}(X) = \Omega_{S_k}(\mathbf{x}_i)]$ for any $S_k \subseteq F$ and $\mathbf{x}_i \in \mathcal{D}$, i.e., $\hat{Y}$ is optimal. Therefore, $acc(\hat{Y}(\Omega_{S_k}(\mathbf{x}_i))) = \Phi_{S_k}(\mathbf{x}_i)$ and from (12) we infer that $\hat{Y}$ satisfies *p1* and *p3*. Furthermore, $\hat{Y}$ satisfies *p2* because it uses S for all data points, and is optimal by definition. Therefore, statement (11) is satisfied for $\mathcal{D}$.

Thus the property defined in (12) is a necessary and sufficient condition to recognize datasets for which there exists an optimal classifier that satisfies all properties *p1*, *p2*, and *p3*. $\square$

B A NON-OPTIMAL CLASSIFIER THAT SATISFIES ALL THE THREE PROPERTIES

As we discussed in Section 7, one may give up optimality in order to achieve a classifier that simultaneously satisfies all the three properties defined in Section 4. In this section, we present an example of such a non-optimal classifier for the dataset in Table 1.

We use a probabilistic classifier for our discussion. Let $\mathbf{x}.f_i$ denote the value of feature f_i in data point $\mathbf{x}$. Now consider the classifier defined by equation (13). This classifier first selects a linear classifier based on the set of known feature values S. Then it returns a binary label using an additional randomization step.

$$\hat{Y}(\Omega_S(\mathbf{x})) = \begin{cases} S = \{f_1, f_2\} & \begin{cases}\mathbf{x}.f_2 - \mathbf{x}.f_1 \geq 2 & + \quad w.p.\ \frac{4}{5}\\ otherwise & - \quad w.p.\ \frac{4}{5}\end{cases}\\ S = \{f_1\} & \begin{cases}\mathbf{x}.f_1 \geq 1 & - \quad w.p.\ \frac{3}{4}\\ otherwise & + \quad w.p.\ \frac{3}{4}\end{cases}\\ S = \{f_2\} & \begin{cases}\mathbf{x}.f_1 \geq 2 & + \quad w.p.\ \frac{3}{4}\\ otherwise & - \quad w.p.\ \frac{3}{4}\end{cases}\\ S = \emptyset & \begin{cases}+ \quad w.p.\ \frac{1}{2}\\ - \quad w.p.\ \frac{1}{2}\end{cases}\end{cases} \tag{13}$$

Table 3 shows the accuracies of the predictions made by this classifier for each data point and each feature set. The values in the table are calculated using equation (1). First observe that by using feature set $\{f_1, f_2\}$ for all the data points, the classifier satisfies fair privacy. Furthermore, the accuracy of the classifier for all data points using this feature set is $\frac{4}{5}$, meaning that fair accuracy is also satisfied. Finally, we observe that using any subset of $\{f_1, f_2\}$ will result in a lower prediction accuracy for all the data points in the dataset, thus the need-to-know principle is also satisfied.

Table 3: Accuracies of the predictions made by the classifier in equation (13).

	$\emptyset$	$\{f1\}$	$\{f2\}$	$\{f1, f2\}$
$\mathbf{x}_1$	1/2	5/12	3/4	4/5
$\mathbf{x}_2$	1/2	5/12	3/4	4/5
$\mathbf{x}_3$	1/2	5/12	1/2	4/5
$\mathbf{x}_4$	1/2	3/4	1/2	4/5

In order to see that the classifier defined by equation (13) is not optimal, notice that our optimality definition requires the classifier to be optimal for all data points and using any subset $S \subset F$. However, using $\{f_1, f_2\}$ we see that the accuracy of the classifier for all the data points is $\frac{4}{5}$ while all the data points are distinguishable using $\{f_1, f_2\}$, i.e., the accuracy of an optimal classifier using $\{f_1, f_2\}$ is 1 for all data points in the dataset.

This example illustrates an interesting direction for future research, which can be stated as the following question: *"how much one needs to compromise on optimality in order to simultaneously achieve all the three socially desirable properties introduced in this paper?"*

C DATASET VERFIFICATION RESUTLS

Table 4: The results of applying our verification algorithm to 18 UCI datasets.

Dataset	size	# features	# labels	largest candidate size	all candidates satisfy the 1st condition	both 1st and 2nd conditions were verified	trade-off
Handwritten Digits	1797	64	10	4	✓	✘	YES
Haberman's Survival	306	3	2	2	✓	✘	YES
Letter Recognition	20000	16	26	2	✓	✘	YES
Somerville Happiness	143	6	2	2	✓	✘	YES
Vehicle Silhouettes	846	18	4	2	✓	✘	YES
Caesarian	80	5	2	3	✓	✘	YES
Musk (Version 1)	476	166	2	1	✓	✘	YES
Musk (Version 2)	6598	166	2	1	✓	✘	YES
Optical Digits	3823	64	10	2	✓	✘	YES
Pen-Based Digits	7494	16	10	2	✓	✘	YES
Mushroom	5644	22	2	3	✓	✘	YES
Nursery	12960	8	5	8	✘	✓	YES
Census Income	32561	14	2	3	✓	✘	YES
Chess	28056	6	18	5	✓	✘	YES
Contraceptive Method	1473	9	3	4	✓	✘	YES
Balance Scale	625	4	3	3	✓	✘	YES
Breast Cancer	277	9	2	4	✓	✘	YES
Car Evaluation	1728	6	4	4	✓	✘	YES