PAC-Learning for Strategic Classification

Ravi Sundaram [*1] **Anil Vullikanti** [*2 3] **Haifeng Xu** [*2] **Fan Yao** [*2]

Abstract

The study of *strategic* or *adversarial* manipulation of testing data to fool a classifier has attracted much recent attention. Most previous works have focused on two extreme situations where any testing data point either is completely adversarial or always equally prefers the positive label. In this paper, we generalize both of these through a unified framework for strategic classification, and introduce the notion of *strategic VC-dimension* (SVC) to capture the PAC-learnability in our general strategic setup. SVC provably generalizes the recent concept of adversarial VC-dimension (AVC) introduced by Cullina et al. (2018). We instantiate our framework for the fundamental *strategic linear classification* problem. We fully characterize: (1) the *statistical learnability* of linear classifiers by pinning down its SVC; (2) its *computational tractability* by pinning down the complexity of the empirical risk minimization problem. Interestingly, the SVC of linear classifiers is always upper bounded by its standard VC-dimension. This characterization also strictly generalizes the AVC bound for linear classifiers in (Cullina et al., 2018).

1. Introduction

In today's increasingly connected world, it is rare that an algorithm will act alone. When a machine learning algorithm is used to make predictions or decisions about others who have their own preferences over the learning outcomes, it is well known (e.g., *Goodhart's law*) that *gaming behaviors* may arise—these have been observed in a variety of domains such as finance (Tearsheet), online retailing (Hannak et al., 2014), education (Hardt et al., 2016) as well as during the ongoing COVID-19 pandemic (Bryan & Crossroads; Williams & Haire). In the early months of the pandemic, simple decision rules were designed for COVID-19 testing (COVID) by the CDC. However, people had different preferences for getting tested. Those with work-from-home jobs and leave benefits preferred to get tested in order to know their true health status whereas some of the people with lower income, and without leave benefits preferred not to get tested (with fears of losing their income) (Williams & Haire). Policy makers sometimes prefer to make classification rules confidential (Citron & Pasquale, 2014) to mitigate such gaming. However, this is not fool-proof in general since the methods may be reverse engineered in some cases, and transparency of ML methods is sometimes mandated by law, e.g., (Goodman & Flaxman, 2016). Such concerns have led to a lot of interest in designing learning algorithms that are robust to strategic gaming behaviors of the data sources (Perote & Perote-Peña, 2004; Dekel et al., 2010; Hardt et al., 2016; Chen et al., 2018; Dong et al., 2018; Cullina et al., 2018; Awasthi et al., 2019); the present work subscribes to this literature.

This paper focuses on the ubiquitous binary classification problem, and we look to design classification algorithms that are robust to gaming behaviors *during the test phase*. We study a strict generalization of the canonical classification setup that naturally incorporates data points' *preferences* over classification outcomes (which leads to strategic behaviors as we will describe later). In particular, each data point is denoted as a tuple $(\boldsymbol{x}, y, r)$ where $\boldsymbol{x} \in \mathcal{X}$ and $y \in \{-1, +1\}$ are the *feature* and *label*, respectively (as in classic classification problems), and additionally, $r \in \mathbb{R}$ is a real number that describes how much this data point prefers label $+1$ over -1. Importantly, we allow r to be negative, meaning that the data point may prefer label -1. For instance, in the decision rules for COVID-19 testing, individuals who prefer to get tested have $r > 0$, while those who prefer not to be tested have $r < 0$. The magnitude $|r|$ captures how strong their preferences are. For example, in the school choice matching market, students have heterogeneous preferences over universities (Pathak, 2017; Roth, 2008) and may manipulate their application materials during the admission process. Let set $R \subseteq \mathbb{R}$ denote the set of all possible values that the preference value r may take. Obviously, the trivial singleton set $R = \{0\}$ corresponds to

[*] Equal contribution [1] Khoury College of Computer Science, Northeastern University, Boston, MA 02115 [2] Department of Computer Science, University of Virginia, Charlottesville, VA 22904 [3] Biocomplexity Institute and Initiative, University of Virginia, Charlottesville, VA 22904. Correspondence to: Haifeng Xu <hx4ad@virginia.edu>, Fan Yao <fy4bc@virginia.edu>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

the classic classification setup without any preferences. Another special case of $R = \{1\}$ corresponds to the situation where all data points prefer label $+1$ equally. This is the strategic classification settings studied in several previous works (Hardt et al., 2016; Hu et al., 2019b; Miller et al., 2019). A third special case is $R = \{-1, 1\}$. This corresponds to classification under *evasion* attacks (Biggio et al., 2013; Goodfellow et al., 2015; Li & Vorobeychik, 2014; Cullina et al., 2018; Awasthi et al., 2019), where any test data point $(\boldsymbol{x}, y)$ prefers the *opposite* of its true label y, i.e., the "adversarial" assumption.

Our model considers any *general preference* set R. As we will show, this much richer set of preferences may sometimes make learning more difficult, both statistically and computationally, but not always. Like (Hardt et al., 2016; Dong et al., 2018; Goodfellow et al., 2015; Cullina et al., 2018), our model assumes that manipulation is only possible to the data *features* and happens only during the *test* phase. Specifically, the true feature of the test data may be altered by the strategic data point. The cost of masking a true feature $\boldsymbol{x}$ to appear as a different feature $\boldsymbol{z}$ is captured by a *cost function* $c(\boldsymbol{z}; \boldsymbol{x})$. Therefore, the test data point's decision needs to balance the *cost* of altering feature and the *reward* of inducing its preferred label captured by r. As is standard in game-theoretic analysis, the test data point is assumed a rational decision maker and will choose to alter to the feature $\boldsymbol{z}$ that maximizes its quasi-linear utility $[r \cdot \mathbb{I}(h(\boldsymbol{z}) = 1) - c(\boldsymbol{z}; \boldsymbol{x})]$. This naturally gives rise to a *Stackelberg game* (Von Stackelberg, 2010). We aim to learn, from i.i.d. drawn (unaltered) training data, the optimal classifier h^* that minimizes the 0-1 classification loss, assuming any randomly drawn test data point (from the same distribution as testing data) will respond to h^* strategically. Notably, the data point's strategic behaviors will *not* change its true label. Such behavior is referred to as *strategic gaming*, which crucially differs from *strategic improvement* studied recently (Kleinberg & Raghavan, 2019; Miller et al., 2019).

1.1. Overview of Our Results

The Strategic VC-Dimension. We introduce the novel notion of *strategic VC-dimension* SVC$(\mathcal{H}, R, c)$ which captures the learnability of any hypothesis class $\mathcal{H}$ when test data points' strategic behaviors are induced by cost function c and preference values from any set $R \subseteq \mathbb{R}$.

- We prove that any strategic classification problem is agnostic PAC learnable by the empirical risk minimization paradigm with $O\big(\epsilon^{-2}[d + \log(\frac{1}{\delta})]\big)$ samples, where $d =$SVC$(\mathcal{H}, R, c)$. Conceptually, this result illustrates that SVC correctly characterizes the learnability of the hypothesis class $\mathcal{H}$ in our strategic setup.

- Our SVC notion generalizes the adversarial VC-dimension (AVC) introduced in (Cullina et al., 2018) for adversarial learning with evasion attacks. Formally, we prove that AVC equals precisely SVC$(\mathcal{H}, R, c)$ for $R = \{-1, 1\}$ when data points are allowed to move within region $\{\boldsymbol{z}; c(\boldsymbol{z}; \boldsymbol{x}) \leq 1\}$ in the corresponding adversarial learning setup. However, for general preference set R, SVC can be arbitrarily larger than both AVC and the standard VC dimension. Thus, complex strategic behaviors may indeed make the learning statistically more difficult. Interestingly, to our knowledge, this is the first time that adversarial learning and strategic learning are unified under the same PAC-learning framework.

- We prove SVC$(\mathcal{H}, R, c)\leq 2$ for any $\mathcal{H}$ and R when c is any *separable* cost function (introduced by (Hardt et al., 2016)). Invoking our sample complexity results above, this also recovers a main learnability result of (2016) and, moreover, generalizes their result to arbitrary agent preferences.

Strategic Linear Classification. As a case study, we instantiate our strategic classification framework in perhaps one of the most fundamental classification problems, *linear classification*. Here, features are in $\mathbb{R}^d$ linear space. We assume the cost function $c(\boldsymbol{z}; \boldsymbol{x})$ for any $\boldsymbol{x}$ is induced by *arbitrary* seminorms of the difference $\boldsymbol{z} - \boldsymbol{x}$. We distinguish between two crucial situations: (1) *instance-invariant* cost function which means the cost of altering the feature $\boldsymbol{x}$ to $\boldsymbol{x} + \Delta$ is the same for any $\boldsymbol{x}$; (2) *instance-wise* cost function which allows the cost from $\boldsymbol{x}$ to $\boldsymbol{x} + \Delta$ to be different for different $\boldsymbol{x}$. Our results show that the more general instance-wise costs impose significantly more difficulties in terms of both statistical learnability and computational tractability.

- **Statistical Learnability.** We prove that the SVC of linear classifiers is ∞ for *instance-wise* cost functions even when features are in $\mathbb{R}^2$; in contrast, the SVC is at most $d + 1$ for any *instance-independent* cost functions and any R when features are in $\mathbb{R}^d$. This later result also strictly generalizes the AVC bound for linear classifiers proved in (Cullina et al., 2018), and illustrates an interesting conceptual message: though SVC can be significantly larger than AVC in general, *extending from $R = \{-1, 1\}$ (the adversarial setting) to an arbitrary strategic preference set R does not affect the statistical learnability of strategic linear classification.*

- **Computational Tractability.** We show that the empirical risk minimization problem for linear classifier can be solved in polynomial time only when the strategic classification problem exhibits certain *adversarial* nature. Specifically, an instance is said to have *adversarial* preferences if all negative test points prefer label $+1$ (but possibly to different extents) and all positive test points prefer label -1. A strictly more relaxed situation has *essentially adversarial* references — i.e., any negative test point prefers label $+1$ more than any positive test point. We show that for instance-invariant cost functions, any essentially adversarial instance can be solved in polynomial time whereas for instance-wise cost functions, only adversarial instances can be solved in polynomial time. These positive results are essentially the best one can hope for. Indeed, we prove that the following situations, which goes slightly beyond the tractable cases above, are both NP-hard: (1) instance-invariant cost functions but general preferences; (2) instance-wise cost functions but essentially adversarial preferences.

1.2. Related Work and Comparison

Most relevant to ours is perhaps the strategic classification model studied by (Hardt et al., 2016) and (Zhang & Conitzer, 2021), where Hardt et al. (2016) formally formulated the strategic classification problem as a repeated Stackelberg game and Zhang & Conitzer (2021) studied the PAC-learning problem and tightly characterized the sample complexity via "incentive-aware ERM". However, their model and results all assume homogeneous agent preferences, i.e., all agents *equally* prefer label $+1$. Our model strictly generalizes the model of (Hardt et al., 2016; Zhang & Conitzer, 2021) by allowing agents' *heterogeneous* preferences over classification outcomes. Besides the modeling differences, the research questions we study are also quite different from, and not comparable to, (Hardt et al., 2016). Their positive results are derived under the assumption of *separable cost* functions or its variants. While our characterization of SVC equaling at most 2 under separable cost functions implies a PAC-learnability result of (Hardt et al., 2016), this characterization serves more as our case study and our main contribution here is the study of the novel concept of SVC, which does not appear in previous works. Moreover, we study the efficient learnability of linear classifiers with cost functions induced by semi-norms. This broad and natural class of cost functions is not separable, and thus the results of Hardt et al. (2016) does not apply to this case.

Our model also generalizes the setup of *adversarial classification with evasion attacks*, which has been studied in numerous applications, particularly deep learning models (Biggio et al., 2013; 2012; Li & Vorobeychik, 2014; Carlini & Wagner, 2017; Goodfellow et al., 2015; Jagielski et al., 2018; Moosavi-Dezfooli et al., 2017; Mozaffari-Kermani et al., 2015; Rubinstein et al., 2009); however, most of these works do not yield theoretical guarantees. Our work extends and strictly generalizes the recent work of (Cullina et al., 2018) through our more general concept of SVC and results on computational efficiency. In a different work, (Awasthi et al., 2019) studied *computationally* efficient learning of linear classifiers in adversarial classification with l_∞-norm-induced δ-ball for allowable adversarial moves. Our computational tractability results generalize their results to δ-ball induced by *arbitrary semi-norms*.[1]

Strategic classification has been studied in other different settings or domains or for different purposes, including spam filtering (Brückner & Scheffer, 2011), classification under incentive-compatibility constraints (Zhang & Conitzer, 2021), online learning (Dong et al., 2018; Chen et al., 2020), and understanding the social implications (Akyol et al., 2016; Milli et al., 2019; Hu et al., 2019b). A relevant but quite different line of recent works study *strategic improvements* (Kleinberg & Raghavan, 2019; Miller et al., 2019; Ustun et al., 2019; Bechavod et al., 2020; Shavit et al., 2020). Finally, going beyond classification, strategic behaviors in machine learning has received significant recent attentions, including in regression problems (Perote & Perote-Peña, 2004; Dekel et al., 2010; Chen et al., 2018), distinguishing distributions (Zhang et al., 2019a;b), and learning for pricing (Amin et al., 2013; Mohri & Munoz, 2015; Vanunts & Drutsa, 2019). Due to space limit, we refer the curious reader to Appendix A for detailed discussions.

2. Model

Basic Setup. We consider binary classification, where each data point is characterized by a tuple $(\boldsymbol{x}, y, r)$. Like classic classification setups, $\boldsymbol{x} \in \mathcal{X}$ is the feature vector and $y \in \{+1, -1\}$ is its label. The only difference of our setup from classic classification problems is the additional $r \in R \subseteq \mathbb{R}$, which is the data point's (positive or negative) preference/reward of being labeled as $+1$. The data point's reward for label -1 is, without loss of generality, normalized to be 0. A classifier is a mapping $h : \mathcal{X} \rightarrow \{+1, -1\}$. Our model is essentially the same as that of (Hardt et al., 2016; Miller et al., 2019), except that the r in our model can be any real value from set R whereas the aforementioned works assume $r = 1$ for all data points. Notably, we also allow r to be *negative*, which means some data points prefer to be classified as label -1. This generalization is natural and very useful because it allows much richer agent preferences. For instance, it casts the adversarial/robust classification problem as a special case of our model as well (see discussions later). Intuitively, the set R captures the richness of agents' preferences. As we will prove, how rich it is will affect both the statistical learnability and computational tractability of the learning problem.

The Strategic Manipulation of *Test* Data. We consider strategic behaviors during the *test* phase and assume that the training data is unaltered/uncontaminated. An illustration of the setup can be found in Figure 1. A generic *test* data point is denoted as $(\boldsymbol{x}, y, r)$. The test data point is strategic and may shift its feature to vector $\boldsymbol{z}$ with cost $c(\boldsymbol{z}; \boldsymbol{x})$ where $c : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}_{\geq 0}$. In general, function c can be an arbitrary non-negative cost function. In our study of strategic linear classification, we assume the cost functions are induced by

[1] (Awasthi et al., 2019) also studied computational tractability of learning other classes of classifiers, e.g., degree-2 polynomial threshold classifiers, which we do not consider.

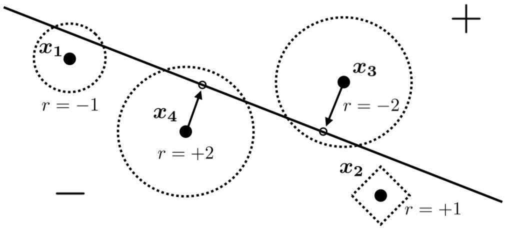

Figure 1. Example illustration of our setup. The line is a linear classifier. Points $\boldsymbol{x}_3, \boldsymbol{x}_4$ have incentive to cross the boundary whereas $\boldsymbol{x}_1, \boldsymbol{x}_2$ do not. The dotted cycles contain all manipulated features which have moving cost exactly 1 and they can be different for different points (i.e., instance-wise costs).

seminorms. We will consider the following two types of cost functions, with increasing generality.

1. **Instance-invariant cost functions:** A cost function c is *instance-invariant* if there is a common function l such that $c(\boldsymbol{z}; \boldsymbol{x}) = l(\boldsymbol{z} - \boldsymbol{x})$ for any point $(\boldsymbol{x}, y, r)$.
2. **Instance-wise cost functions:** A cost function c is *instance-wise* if for each data point $(\boldsymbol{x}, y, r)$, there is a function $l_{\boldsymbol{x}}$ such that $c(\boldsymbol{z}; \boldsymbol{x}) = l_{\boldsymbol{x}}(\boldsymbol{z} - \boldsymbol{x})$. Notably, $l_{\boldsymbol{x}}$ may be different for different data point $(\boldsymbol{x}, y, r)$.

Both cases have been considered in previous works. For instance, the separable cost function studied in (Hardt et al., 2016) is instance-wise, and the cost function induced by a seminorm as assumed by the main theorem of (Cullina et al., 2018) is instance-invariant. We shall prove later that the choice among these two types of cost functions will largely affect the efficient learnability of the problem.

Given a classifier h, the strategic test data point $(\boldsymbol{x}, y, r)$ may shift its feature vector to $\boldsymbol{z}$ and would like to pick the best such $\boldsymbol{z}$ by solving the following optimization problem:

Data Point Best Response:

$$\Delta_c(\boldsymbol{x}, r; h) = \arg\max_{\mathbf{z}\in\mathcal{X}} \left[\mathbb{I}(h(\boldsymbol{z}) = 1)\cdot r - c(\boldsymbol{z}; \boldsymbol{x})\right]. \tag{1}$$

where $\mathbb{I}(S)$ is the indicator function and $[\mathbb{I}(h(\boldsymbol{z}) = 1)\cdot r - c(\boldsymbol{z}; \boldsymbol{x})]$ is the quasi-linear *utility function* of data point $(\boldsymbol{x}, y, r)$. We call $\Delta_c(\boldsymbol{x}, r; h)$ the *manipulated feature*. When there are multiple best responses, we assume the data point may choose any best response and thus will adopt the standard *worst-case* analysis. Note that the test data's strategic behaviors do *not* change its true label. Such strategic *gaming* behaviors differs from strategic *improvements* (see (2019) for more discussions on their differences).

2.1. The Strategic Classification (StraC) Problem

A StraC problem is described by a hypothesis class $\mathcal{H}$, the set of preferences R and a manipulation cost function c. We thus denote it as StraC$\langle\mathcal{H}, R, c\rangle$. Adopting the standard statistical learning framework, the input to our learning task are n *uncontaminated training* data points $(\boldsymbol{x}_1, y_1, r_1), \cdots, (\boldsymbol{x}_n, y_n, r_n)$ drawn independently and identically (i.i.d.) from distribution $\mathcal{D}$. Given these training data, we look to learn a classifier $h \in \mathcal{H}$ which minimizes the basic 0-1 loss, defined as follows:

Strategic 0-1 Loss of classifier h :

$$L_c(h; \mathcal{D}) = \mathop{\mathbf{Pr}}_{(\boldsymbol{x},y,r)\sim\mathcal{D}} \left[h(\Delta_c(\boldsymbol{x}, r; h)) \neq y)\right]. \tag{2}$$

Notably, the classifier h in the above loss definition takes the manipulated feature $\Delta_c(\boldsymbol{x}, r; h))$ as input and, nevertheless, looks to correctly predict the true label y. For notational convenience, we sometimes omit c when it is clear from the context and simply write $\Delta(\boldsymbol{x}, r; h)$ and $L(h; \mathcal{D})$.

2.2. Notable Special Cases

Our strategic classification model generalizes several models studied in previous literature, which we now sketch.

Non-strategic classification. When $R = \{0\}$ and $c(\boldsymbol{z}; \boldsymbol{x}) > 0$ for any $\boldsymbol{x} \neq \boldsymbol{z}$, our model degenerates to the standard non-strategic setting.

Strategic classification with homogeneous preference. When $R = \{1\}$, our model degenerates to the strategic classification model studied in prior work (Hardt et al., 2016; Hu et al., 2019b; Milli et al., 2019)—here all data points have the same incentive of being classified as $+1$.

Adversarial Classification. When $R = \{1, -1\}$ (or $\{\delta, -\delta\}$,$\delta \neq 0$), our model becomes the adversarial classification problem (2018; 2019), where each data point can adversarially move to induce the *opposite* of its true label — within the ball of radius 1 induced by cost function c. Our Proposition 1 provides formal evidence for this connection.

Generalized Adversarial Classification. An interesting generalization of the above adversarial classification setting is that $r < 0$ for all data points with true label $+1$ and $r > 0$ for all data points with true label -1. This captures the situation where each point has different "power" (decided by $|r|$) to play against the classifier. To our knowledge, this generalized setting has not been considered before. Our results yield new efficient statistical learnability and computational tractability for this setting.

3. VC-Dimension for Strategic Classification

In this section, we introduce the notion of *strategic VC-dimension* (SVC) and show that it properly captures the behaviors of a hypothesis class in the strategic setup introduced above. We then show the connection of SVC with previous studies on both strategic and adversarial learning. Before formally introducing SVC, we first define the shattering coefficients in strategic setups.

Definition 1 (Strategic Shattering Coefficients)**.** *The n'th shattering coefficient of any strategic classification problem*

$\textsc{StraC}\langle\mathcal{H}, R, c\rangle$ *is defined as*

$$\sigma_n(\mathcal{H}, R, c) = \max_{(\boldsymbol{x},\boldsymbol{r})\in\mathcal{X}^n\times R^n} |\{(h(\Delta_c(\boldsymbol{x}_1, r_1; h)), \cdots, h(\Delta_c(\boldsymbol{x}_n, r_n; h)) : h \in \mathcal{H}\}|,$$

where $\Delta_c(\boldsymbol{x}_i, r_i; h)$ defined in Eq. (1) is a best response of data point $(\boldsymbol{x}_i, y_i, r_i)$ to classifier h under cost function c.

That is, $\sigma_n(\mathcal{H}, R, c)$ captures the maximum number of classification behaviors/outcomes (among all choices of data points) that classifiers in $\mathcal{H}$ can possibly induce by using manipulated features as input. Like classic definition of shattering coefficient, the $\sigma_n(\mathcal{H}, R, c)$ here does not involve the labels of the data points at all. In contrast, in the shattering coefficient definition for adversarial VC-dimension of (Cullina et al., 2018), the "max" is allowed to be over data labels as well. This is an important difference from us. Given the definition of the strategic shattering coefficients, the definition of strategic VC-dimension is standard.

Definition 2. *The Strategic VC-dimension (SVC) for strategic classification problem* $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$ *is defined as*

$$SVC(\mathcal{H}, R, c) = \sup\{n \in \mathbb{N} : \sigma_n(\mathcal{H}, R, c) = 2^n\}. \quad (3)$$

We show that the SVC defined above correctly characterizes the learnability of any strategic classification problem $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$. We consider the standard Empirical risk minimization (ERM) paradigm for strategic classification, but take into account training data's manipulation behaviors. Specifically, given any cost function c, any n uncontaminated training data points $(\mathbf{x}_1, y_1, r_1), \cdots, (\mathbf{x}_n, y_n, r_n)$ drawn independently and identically (i.i.d.) from the same distribution $\mathcal{D}$, the *strategic empirical risk minimization* (SERM) problem computes a classifier $h \in \mathcal{H}$ that minimizes the empirical strategic 0-1 loss in Eq. (2). Formally, the SERM for $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$ is defined as follows:

$$\begin{aligned} \text{SERM}: \quad & \text{argmin}_{h\in\mathcal{H}} \; L_c(h, \{(\boldsymbol{x}_i, y_i, r_i)\}_{i=1}^n) \\ &= \sum_{i=1}^n \mathbb{I}\big[h(\Delta_c(\boldsymbol{x}_i, r_i; h)) \neq y_i\big] \end{aligned} \quad (4)$$

where $L_c(h, \{(\boldsymbol{x}_i, y_i, r_i)\}_{i=1}^n)$ is the *empirical loss* (compared to the expected loss $L_c(h, \mathcal{D})$ defined in Equation (2)). Unlike the standard (non-strategic) ERM problem and similar in spirit to the "incentive-aware ERM" in (Zhang & Conitzer, 2021), classifiers in the SERM problem take each data point's strategic response $\Delta_c\,(\boldsymbol{x}_i, r_i; h)$ as input, while not the original feature vector $\boldsymbol{x}_i$.

Given the definition of strategic VC-dimension and the SERM framework, we state the sample complexity result for PAC-learning in our strategic setup:

Definition 3 (PAC-Learnability)**.** *In a strategic classification problem* $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$*, the hypothesis class* $\mathcal{H} \subseteq (\mathcal{X} \to \{+1, -1\})$ *is* Probably Approximately Correctly (PAC) learnable *by an algorithm $\mathcal{A}$ if there is a function $m_{\mathcal{H},R,c} : (0, 1)^2 \to \mathbb{N}$ such that $\forall(\delta, \epsilon) \in (0, 1)^2$, for any $n \geq m_{\mathcal{H},R,c}(\delta, \epsilon)$ and any distribution $\mathcal{D}$ for $(\boldsymbol{x}, y, r)$, with at least probability $1 - \delta$, we have $L_c(h^*, \mathcal{D}) \leq \epsilon$ where h^* is the output of the algorithm $\mathcal{A}$ with n i.i.d. samples from $\mathcal{D}$ as input. The problem is* agnostic PAC learnable *if $L_c(h^*, \mathcal{D}) - \inf_{h\in\mathcal{H}} L_c(h, \mathcal{D}) \leq \epsilon$.*

Theorem 1. *Any strategic classification instance* $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$ *is agnostic PAC learnable with sample complexity $m_{\mathcal{H},R,c}(\delta, \epsilon) \leq C\epsilon^{-2}[d + \log(\frac{1}{\delta})]$ by the SERM in Eq. (4), where $d = SVC(\mathcal{H}, R, c)$ is the strategic VC-dimension and C is an absolute constant.*

Proof Sketch. The key idea of the proof is to convert our new strategic learning setup to a standard PAC learning setup, so that the Rademacher complexity argument can be applied. This is done by viewing the preference $r \in R$ as an additional dimension in the augmented feature vector space. Formally, we consider the new binary hypothesis class $\tilde{H} = \{\kappa_c(h) : h \in H\}$, where classifier κ_c satisfies $\kappa_c(h) : (\boldsymbol{x}, r) \mapsto h(\Delta_c(\boldsymbol{x}, r; h)), \forall(\boldsymbol{x}, r) \in \mathcal{X} \times R$. It turns out that the SVC defined above captures the VC-dimension for this new hypothesis class $\tilde{H}$. Formally proof can be found in Appendix B.1. □

Due to space limit, we defer all formal proofs to the appendix. Proof sketches are provided for some of the results. Next, we illustrate how SVC connects to previous literature, particularly the two most relevant works by Cullina et al. (Cullina et al., 2018) and Hardt et al. (Hardt et al., 2016).

3.1. SVC generalizes Adversarial VC-Dimension (AVC)

We show that SVC generalizes the adversarial VC dimension (AVC) introduced by (Cullina et al., 2018). We give an intuitive description of AVC here, and refer the curious reader to Appendix 1 for its formal definition. At a high level, AVC captures the behaviors of binary classifiers under *adversarial* manipulations. Such adversarial manipulations are described by a binary nearness relation $\mathcal{B} \subseteq \mathcal{X} \times \mathcal{X}$ and $(\boldsymbol{z}; \boldsymbol{x}) \in \mathcal{B}$ if and only if data point with feature $\boldsymbol{x}$ can manipulate its feature to $\boldsymbol{z}$. Note that there is no direct notion of agents' *utilities* or *costs* in adversarial classification since each data point simply tries to ruin the classifier by moving within the allowed manipulation region (usually an δ-ball around the data point). Nevertheless, our next result shows that AVC with binary nearness relation $\mathcal{B}$ always equals to SVC as long as the set of strategic manipulations induced by the data points' incentives is the same as $\mathcal{B}$. To formalize our statement, we need the following consistency definition.

Definition 4. *Given any binary relation $\mathcal{B}$ and any cost function c, we say $\mathcal{B}, c$ are r-consistent if $\mathcal{B} = \{(\boldsymbol{z}; \boldsymbol{x}) :$*

$c(\boldsymbol{z};\boldsymbol{x}) \leq r\}$. *In this case, we also say $\mathcal{B}$ [resp. c] is r-consistent with c [resp. $\mathcal{B}$].*

By definition any cost function c is r-consistent with the natural binary nearness relation it induces $\mathcal{B}_c = \{(\boldsymbol{z};\boldsymbol{x}) : c(\boldsymbol{z};\boldsymbol{x}) \leq r\}$. Conversely, any binary relation $\mathcal{B}$ is r-consistent (for any $r > 0$) with a natural cost function that is simply an indicator function of $\mathcal{B}$ defined as follows

$$c_{\mathcal{B}}(\boldsymbol{z};\boldsymbol{x}) = \begin{cases} \infty, & \text{if } (\boldsymbol{z};\boldsymbol{x}) \in \mathcal{B} \\ 0, & \text{if } (\boldsymbol{z};\boldsymbol{x}) \notin \mathcal{B} \end{cases}. \tag{5}$$

Note that, $\mathcal{B}$ and c may be r-consistent for infinitely many different r, as shown in the above example with $\mathcal{B}$ and $c_{\mathcal{B}}$.

Proposition 1. *For any hypothesis class $\mathcal{H}$ and any binary nearness relation $\mathcal{B}$, let AVC($\mathcal{H}, \mathcal{B}$) denote the adversarial VC-dimension defined in (Cullina et al., 2018). Suppose $\mathcal{B}$ and c are r-consistent for some $r > 0$, then we have AVC($\mathcal{H}, \mathcal{B}$) =SVC($\mathcal{H}, \{+r, -r\}, c$).*

As a corollary of Proposition 1, we know that SVC is in general larger than or at least equal to AVC when the strategic behaviors it induces include $\mathcal{B}$. This is formalized in the following statement.

Corollary 1.1. *Suppose a cost function c is r-consistent with binary nearness relation $\mathcal{B}$ and $\pm r \in R$, then we have*

$$SVC(\mathcal{H}, R, c) \geq AVC(\mathcal{H}, \mathcal{B}).$$

Corollary 1.1 illustrates that for any cost function c, the SVC with a rich preference set R is generally no less than the corresponding AVC under the natural binary nearness relation that c induces. One might wonder how large their gap can be. Our next result shows that for a general R the gap between SVC and AVC can be *arbitrarily large* even in natural setups. The intrinsic reason is that a general preference set R will lead to different extents of preferences (i.e., some data points strongly prefers label 1 whereas some slightly prefers it). Such variety of preferences gives rise to more strategic classification outcomes and renders the SVC larger than AVC, and sometimes significantly larger, as shown in the following proposition.

Proposition 2. *For any integer $n > 0$, there exists a hypothesis class $\mathcal{H}$ with point classifiers, an instance-invariant cost function $c(\boldsymbol{z};\boldsymbol{x}) = l(\boldsymbol{z} - \boldsymbol{x})$ for some* metric c *and preference set R such that $SVC(\mathcal{H}, R, c) = n$ but $VC(\mathcal{H}) = AVC(\mathcal{H}, \mathcal{B}_c(r)) = 1$ for any $r \in R$ where $\mathcal{B}_c(r) = \{(\boldsymbol{x},\boldsymbol{z}) : c(\boldsymbol{z};\boldsymbol{x}) \leq r\}$ is the natural nearness relation induced by c and $r > 0$.*

3.2. SVC under Separable Cost Functions

Not only restricting the set R of preference values can reduce the SVC. This subsection shows that restricting to special classes of cost functions can also lead to a small SVC. One special class of cost functions studied in many previous works is the *separable cost functions* (Hardt et al., 2016; Milli et al., 2019; Hu et al., 2019a). Formally, a cost function $c(\boldsymbol{z};\boldsymbol{x})$ is separable if there exists function $c_1, c_2 : \mathcal{X} \to \mathbb{R}$ such that $c(\boldsymbol{z};\boldsymbol{x}) = \max\{c_2(\boldsymbol{z}) - c_1(\boldsymbol{x}), 0\}$.

The following Proposition 3 shows that when the cost function is separable, SVC is at most 2 for *any* hypothesis class $\mathcal{H}$ and any class of preference set R.[2] Therefore, separable cost function essentially reduces any classification problem to a problem in lower dimension. Together with Theorem 1, Proposition 3 also recovers the PAC-learnability result of (Hardt et al., 2016) in their strategic-robust learning model (specifically, Theorem 1.8 of (2016)) and, moreover, generalizes their learnability from homogeneous agent preferences to the case with *arbitrary* agent preference values.

Proposition 3. *For any hypothesis class $\mathcal{H}$, any preference set R satisfying $0 \notin R$, and any separable cost function $c(\boldsymbol{z};\boldsymbol{x})$, we have SVC($\mathcal{H}, R, c$) ≤ 2.*

The assumption $0 \notin R$ means each agent must strictly prefer either label $+1$ or -1. This assumption is necessary since if $0 \in R$, SVC will be at least the classic VC dimension of $\mathcal{H}$ and thus Proposition 3 cannot hold. We remark that the above SVC upper bound 2 holds for any hypothesis class $\mathcal{H}$. This bound 2 is tight for some classes of hypothesis, e.g., linear classifiers.

4. Strategic Linear Classification

This section instantiates our previous general framework in one of the most fundamental special cases, i.e., linear classification. We will study both the *statistical* and *computational* efficiency in *strategic linear classification*. Naturally, we will restrict $\mathcal{X} \subseteq \mathbb{R}^d$ in this section. Moreover, the cost functions are always assumed to be induced by semi-norms.[3] A linear classifier is defined by a hyperplane $\boldsymbol{w} \cdot \boldsymbol{x} + b = 0$; feature vector $\boldsymbol{x}$ is classified as $+1$ if and only if $\boldsymbol{w} \cdot \boldsymbol{x} + b \geq 0$. With slight abuse of notation, we sometimes also call $(\boldsymbol{w}, b)$ a *hyperplane* or a *linear classifier*. Let $\mathcal{H}_d$ denote the hypothesis set of all d-dimensional linear classifiers. For linear classifier $(\boldsymbol{w}, b)$, the data point's best response can be more explicit expressed as:

$$\Delta_c(x, r; \boldsymbol{w}, b) = \arg\max_{\boldsymbol{z}} \left[\mathbb{I}(\boldsymbol{w} \cdot \boldsymbol{z} + b \geq 0) \cdot r - c(\boldsymbol{z};\boldsymbol{x})\right].$$

[2] The model of (Hardt et al., 2016) corresponds to the case $R = \{1\}$ in our model. For that restricted situation, the proof of Proposition 3 can be simplified to prove SVC = 1 when $R = \{1\}$. It turns out that arbitrary preference set R only increases the SVC by at most 1.

[3] A function $l : \mathcal{X} \to \mathbb{R}_{\geq 0}$ is a *seminorm* if it satisfies: (1) triangle inequality: $l(\boldsymbol{x} + \boldsymbol{z}) \leq l(\boldsymbol{x}) + l(\boldsymbol{z})$ for any $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$; and (2) homogeneity: $l(\lambda \boldsymbol{x}) = |\lambda| l(\boldsymbol{x})$ for any $\boldsymbol{x} \in \mathcal{X}, \lambda \in \mathbb{R}$.

4.1. Strategic VC-Dimension of Linear Classifiers

We first study the statistical learnability by examining the strategic VC-dimension (SVC). Our first result is a negative one, showing that SVC can be unbounded in general even for linear classifiers with features in $\mathbb{R}^2$ (i.e., $\mathcal{X} \subset \mathbb{R}^2$) and with simple preference set $R = \{+1, -1\}$.

Theorem 2. *Consider strategic linear classification* $\textsc{StraC}\langle\mathcal{H}_d, R, c\rangle$*. There is an instance-wise cost function* $c(\boldsymbol{z}; \boldsymbol{x}) = l_{\boldsymbol{x}}(\boldsymbol{z} - \boldsymbol{x})$ *where each* $l_{\boldsymbol{x}}$ *is a norm, such that* $SVC(\mathcal{H}_d, R, c) = \infty$ *even when* $\mathcal{X} \subset \mathbb{R}^2$ *and* $R = \{1\}$.

Proof Sketch. We consider a set of data points $\{\boldsymbol{x}_i\}_{i=1}^n$ in $\mathbb{R}^2$ whose features are close but with cost functions induced by different norms. The cost functions are designed such that each point $\boldsymbol{x}_i$ is allowed to strategically alter its feature within a carefully designed polygon G_i centered at the origin. Specifically, for any label pattern $\mathcal{L} \in \{+1, -1\}^n$, it has a corresponding node $s_{\mathcal{L}}$ on a unit cycle. The polygon G_i for $\boldsymbol{x}_i$ is the convex hull of all $s_{\mathcal{L}}$ whose label pattern $\mathcal{L}$ classifies i as $+1$. Figure 2 illustrates a high-level idea of our construction. Under such cost functions, we prove that $\mathcal{H}$ can induce all 2^n possible label patterns. The formal construction and proof can be found in Appendix C. □

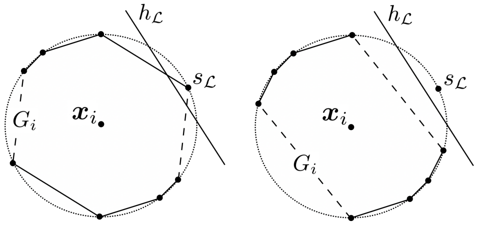

Figure 2. G_i is the polygon associated with $\boldsymbol{x}_i$'s cost function. Given any label pattern $\mathcal{L} \in \{+1, -1\}^n$, the linear classifier $h_{\mathcal{L}}$ is carefully chosen to induce exactly the label pattern $\mathcal{L}$ on $\{\boldsymbol{x}_i\}_{i=1}^n$.

In the study of adversarial VC-dimension (AVC) by (Cullina et al., 2018), the feature manipulation region of each data point is assumed to be *instance-invariant*. As a corollary, Theorem (2) implies that AVC also becomes ∞ for linear classifiers in $\mathbb{R}^2$ if each data point's manipulation region is allowed to be different.

It turns out that the ∞-large SVC above is mainly due to the instance-wise cost functions. Our next result shows that under instance-invariant cost functions, the SVC will behave nicely and, in fact, equal to the AVC for linear classifiers despite the much richer data point manipulation behaviors. This result also strictly generalizes the characterization of AVC by (Cullina et al., 2018) for linear classifiers and shows that linear classifiers will be no harder to learn statistically even allowing richer manipulation preferences of data points.

Theorem 3. *Consider strategic linear classification* $\textsc{StraC}\langle\mathcal{H}_d, R, c\rangle$*. For any* instance-invariant *cost function* $c(\boldsymbol{z}; \boldsymbol{x}) = l(\boldsymbol{z} - \boldsymbol{x})$ *where* l *is a semi-norm, we have* $SVC(\mathcal{H}_d, R, c) = d + 1 - \dim(V_l)$ *for any bounded* R*, where* V_l *is the largest linear space contained in the ball* $\mathcal{B} = \{\boldsymbol{x} : l(\boldsymbol{x}) \leq 1\}$.

In particular, if l *is a norm (i.e.,* $l(\boldsymbol{x}) = 0$ iff $\boldsymbol{x} = 0$*), then* $\dim(V_l) = 0$ *and* $SVC(\mathcal{H}, R, c) = d + 1$.

Proof Sketch. The key idea of the proof relies on a careful construction of a linear mapping which, given any set of samples $\{(\boldsymbol{x}_i, y_i, r_i)\}_{i=1}^n$, maps the class of linear classifiers to the vector space of points' utilities and moreover, the sign of each data point's utility will correspond exactly to the label of the data point after its strategic manipulation. Using the structure of this construction, we can identify the relationship between the dimension of the linear mapping and the strategic VC-dimension. By bounding the dimension of the space of signed agent utilities, we are able to derive the SVC though with some involved algebraic argument to exactly pin down the dimension of the linear mapping due to possible degeneracy of V_l ball. □

4.2. The Complexity of Strategic Linear Classification

In this subsection, we turn our attention to the computational efficiency of learning. The standard ERM problem for linear classification to minimize the 0-1 loss is already known to be NP-hard in the general agnostic learning setting (Feldman et al., 2012). This implies that agnostic PAC learning by SERM is also NP-hard in our strategic setup. Therefore, our computational study will focus on the more interesting PAC-learning case, that is, assuming there exists a strategic linear classifier that perfectly separates all the data points. In the non-strategic case, the ERM problem can be solved easily by a linear feasibility problem.

It turns out that the presence of gaming behaviors does make the resultant SERM problem significantly more challenging. We prove essentially tight computational tractability results in this subsection. Specifically, any strategic linear classification instance can be efficiently PAC-learnable by the SERM when the problem exhibits some "adversarial nature". However, the SERM problem immediately becomes NP-hard even when we go slightly beyond such adversarial situations. We start by defining what we mean by "adversarial nature" of the problem.

Definition 5 (Essentially Adversarial Instances)**.** *For any strategic classification problem* $\textsc{StraC}\langle\mathcal{H}, R, c\rangle$*, let*

$$\begin{aligned} &\textit{min}^- = \min\{r : (\boldsymbol{x}, y, r) \textit{ with } y = -1\} \textit{ and} \\ &\textit{max}^+ = \max\{r : (\boldsymbol{x}, y, r) \textit{ with } y = +1\} \end{aligned} \tag{6}$$

be the minimum reward among all -1 *points and the maximum reward among all* $+1$ *points, respectively. We say* *the instance is* "adversarial" *if* $\min^- \geq 0 \geq \max^+$ *and is* "essentially adversarial" *if* $\min^- \geq \max^+$.

In other words, an instance is "adversarial" if each data point would like to move to the opposite side of its label though with different magnitudes of preferences, and is "essentially adversarial" if any negative data point has a stronger preference to move to the positive side than any positive data point. Many natural settings are essentially adversarial, e.g., all the four examples in Subsection 2.2.

Our first main result of this subsection (Theorem 4) shows that when the strategic classification problem exhibits the above adversarial nature, linear strategic classification can be efficiently PAC-learnable by SERM. The second main result Theorem 5 shows that the SERM problem becomes NP-hard once we go slightly beyond the adversarial setups identified in Theorem 4. These results show that the computational tractability of strategic classification is primarily governed by the preference set R. Interestingly, this is in contrast to the statistical learnability results in Theorem 2 and 3 where the preference set R did not play much role.

Theorem 4. *Any separable strategic linear classification instance* $\text{STRAC}\langle \mathcal{H}_d, R, c\rangle$ *is efficiently PAC-learnable by the SERM in polynomial time in the following two situations:*

1. *The problem is* essentially *adversarial (*$\min^- \geq \max^+$*) and cost function* $c(\boldsymbol{z};\boldsymbol{x}) = l(\boldsymbol{z}-\boldsymbol{x})$ *is instance-invariant and induced by seminorm* l.
2. *The problem is adversarial (*$\min^- \geq 0 \geq \max^+$*) and the instance-wise cost function* $c(\boldsymbol{z};\boldsymbol{x}) = l_{\boldsymbol{x}}(\boldsymbol{z}-\boldsymbol{x})$ *is induced by seminorms* $l_{\boldsymbol{x}}$.

Proof Sketch. For situation 1, we can formulate the SERM problem as the following feasibility problem:

$$\begin{array}{lll} \text{find} & \boldsymbol{w}, b, \epsilon > 0 & \\ \text{s.t.} & \boldsymbol{w}\cdot\boldsymbol{x}_i + b \geq -r_i, & \text{for } y_i = 1. \\ & \boldsymbol{w}\cdot\boldsymbol{x}_i + b \leq -(r_i+\epsilon), & \text{for } y_i = -1. \\ & l^*(\boldsymbol{w}) = 1 & \end{array}$$

where l^* is the dual norm of l.

This unfortunately is not a convex program due to the non-convex constraint $l^*(\boldsymbol{w}) = 1$ — indeed we prove later that this program is NP-hard to solve in general. Therefore, instead, we solve a relaxation of the above program, by relaxing constraint $l^*(\boldsymbol{w}) = 1$ to $l^*(\boldsymbol{w}) \leq 1$ to obtain a convex program. The crucial step of our proof is to show that this relaxation is tight under the *essentially* adversarial assumption. This is proved by converting any optimal solution of the relax convex program to a feasible and optimal solution to the original problem. This is the crucial and also difficult step since the solution to the relaxed convex program may not satisfy $l^*(\boldsymbol{w}) = 1$ — in fact, it will satisfy $l^*(\boldsymbol{w}) < 1$ generally which is why the original program is NP-hard in general. Fortunately, using the essentially adversarial assumption, we are able to employ a carefully crafted construction to generate an optimal solution the the above non-convex program.

For situation 2, we can formulate it as another *non-convex* program with parameter $\boldsymbol{w}, \epsilon$:

$$\begin{array}{lll} \text{find} & \boldsymbol{w}, b, \epsilon > 0 & \\ \text{s.t.} & \boldsymbol{w}\cdot\boldsymbol{x}_i + b \geq (-r_i)\cdot l^*_{\boldsymbol{x}_i}(\boldsymbol{w}), & \text{for } r_i < 0. \\ & -(\boldsymbol{w}\cdot\boldsymbol{x}_i + b) \geq (r_i+\epsilon)\cdot l^*_{\boldsymbol{x}_i}(\boldsymbol{w}), & \text{for } r_i \geq 0. \end{array}$$

Fortunately, for any fixed $\epsilon > 0$, the above program is convex in variable $\boldsymbol{w}$. Moreover, if the system is feasible for $\epsilon_0 > 0$ then it is feasible for any $0 < \epsilon \leq \epsilon_0$. This allows us to determine the feasibility problem above for any ϵ via binary search, which overall is in polynomial time. □

Our next result shows that the positive claim in Theorem (4) are essentially the best one can hope for. Indeed, the SERM immediately becomes NP-hard if one goes slightly beyond the two tractable situations in Theorem (4). Note that our results did not rule out the possibility of other computationally efficient learning algorithms other than the SERM. We leave this as an intriguing open problem for future works.

Theorem 5. *Suppose the strategic classification problem is linearly separable, then the SERM Problem for linear classifiers is NP-hard in the following two situations:*

1. *Preferences are arbitrary and the cost function is instant-invariant and induced by the standard* l_2 *norm, i.e.,* $c(\boldsymbol{z};\boldsymbol{x}) = \|\boldsymbol{x}-\boldsymbol{z}\|_2^2$.
2. *The problem is* essentially *adversarial (*$\min^- \geq \max^+$*) and the cost function is instance-wise and induced by norms.*

Remark 1. *Theorem 3, Theorem 4 and Theorem 5 together imply that for strategic linear classification:*
(1) the problem is efficiently PAC-learnable (both statistically and computationally) when the cost function is instance-invariant and preferences are essentially adversarial;
(2) SERM can be solved efficiently but SVC is infinitely large when the cost function is instance-wise and preferences are adversarial;
(3) the problem is efficiently PAC learnable in statistical sense, but SERM is NP-hard when the cost function is instance-invariant and preferences are arbitrary.

5. Summary

In this work, we propose and study a general strategic classification setting where data points have different preferences over classification outcomes and different manipulation costs. We establish the PAC-learning framework for this strategic learning setting and characterize both the statistical and computational learnability result for linear classifiers. En route, we generalize the recent characterization of adversarial VC-dimension (Cullina et al., 2018) as well as computational tractability for learning linear classifiers by (Awasthi et al., 2019). Our conclusion reveals two important insights. First, the additional intricacy of having different preferences harms the statistical learnability of general hypothesis classes, but *not* for linear classifiers. Second, learning strategic linear classifiers can be done efficiently only when the setup exhibits some adversarial nature and becomes NP-hard in general.

Our learnability result for linear classifiers applies to cost functions induced by semi-norms. A future direction is to generalize the theory to cost function induced by asymmetric semi-norms or even any metrics. We also note that the strategic classification model we consider is under the full-information assumption, i.e., the cost function and the strategic preferences are transparent. This is analogous to the evasion attack in the adversarial machine learning literature, where the training data is supposed to be uncontaminated and the manipulation only happens during testing. What if we cannot observe the strategic preferences during training or do not know the adversaries' cost function? This can be reformulated as online learning through repeated Stackelberg games and has been studied in (Dong et al., 2018), but it does not apply to classifiers with 0-1 loss.

Acknowledgement. We thank anonymous ICML reviewers for helpful suggestions. H. Xu is supported by a GIDI award from the UVA Global Infectious Diseases Institute and a Google Faculty Research Award. A. Vullikanti is supported by NSF grants IIS-1931628, CCF-1918656, NSF IIS-1955797, and NIH grant R01GM109718. R. Sundaram is supported by NSF grants CNS-1718286 and IIS-2039945.

References

Akyol, E., Langbort, C., and Basar, T. Price of transparency in strategic machine learning. *arXiv*, pp. arXiv–1610, 2016.

Amin, K., Rostamizadeh, A., and Syed, U. Learning prices for repeated auctions with strategic buyers. In *Advances in Neural Information Processing Systems*, pp. 1169–1177, 2013.

Awasthi, P., Dutta, A., and Vijayaraghavan, A. On robustness to adversarial examples and polynomial optimization. In *Advances in Neural Information Processing Systems*, pp. 13737–13747, 2019.

Bechavod, Y., Ligett, K., Wu, Z. S., and Ziani, J. Causal feature discovery through strategic modification. *arXiv preprint arXiv:2002.07024*, 2020.

Biggio, B., Nelson, B., and Laskov, P. Poisoning attacks against support vector machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1467–1474, Madison, WI, USA, 2012. Omnipress. ISBN 9781450312851.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Srndic, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pp. 387–402. Springer, 2013.

Brückner, M. and Scheffer, T. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 547–555, 2011.

Bryan, M. and Crossroads, K. Coronavirus unemployment benefits are high, putting workers and employers at odds. https://whyy.org/articles/coronavirus-unemployment-benefits-are-high-putting-workers-and-employers-at-odds/.

Carlini, N. and Wagner, D. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pp. 39–57, 2017.

Chen, Y., Podimata, C., Procaccia, A. D., and Shah, N. Strategyproof linear regression in high dimensions. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, pp. 9–26, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358293. doi: 10.1145/3219166.3219175. URL https://doi.org/10.1145/3219166.3219175.

Chen, Y., Liu, Y., and Podimata, C. Learning strategy-aware linear classifiers. *Advances in Neural Information Processing Systems*, 33, 2020.

Citron, D. K. and Pasquale, F. A. The scored society: Due process for automated predictions. 2014.

COVID. Covid-19 testing overview. https://www.cdc.gov/coronavirus/2019-ncov/symptoms-testing/testing.html.

Cullina, D., Bhagoji, A. N., and Mittal, P. Pac-learning in the presence of adversaries. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31*, pp. 230–241. Curran Associates, Inc., 2018.

Dekel, O., Fischer, F., and Procaccia, A. D. Incentive compatible regression learning. *Journal of Computer and System Sciences*, 76(8):759–777, 2010.

Dong, J., Roth, A., Schutzman, Z., Waggoner, B., and Wu, Z. S. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, pp. 55–70, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450358293. doi: 10.1145/3219166.3219193. URL https://doi.org/10.1145/3219166.3219193.

Feldman, V., Guruswami, V., Raghavendra, P., and Wu, Y. Agnostic learning of monomials by halfspaces is hard. *SIAM Journal on Computing*, 41(6):1558–1590, 2012.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In *ICLR 2015 : International Conference on Learning Representations 2015*, 2015.

Goodman, B. and Flaxman, S. EU regulations on algorithmic decision-making and a "right to explanation", 2016. URL http://arxiv.org/abs/1606.08813. Presented at 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY.

Hannak, A., Soeller, G., Lazer, D., Mislove, A., and Wilson, C. Measuring price discrimination and steering on e-commerce web sites. In *Proceedings of the 2014 conference on internet measurement conference*, pp. 305–318, 2014.

Hardt, M., Megiddo, N., Papadimitriou, C., and Wootters, M. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, ITCS '16, pp. 111–122, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450340571. doi: 10.1145/2840728.2840730. URL https://doi.org/10.1145/2840728.2840730.

Hu, L., Immorlica, N., and Vaughan, J. W. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 259–268, 2019a.

Hu, L., Immorlica, N., and Vaughan, J. W. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pp. 259–268, New York, NY, USA, 2019b. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287597. URL https://doi.org/10.1145/3287560.3287597.

Jagielski, M., Oprea, A., Biggio, B., Liu, C., Nita-Rotaru, C., and Li, B. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. *2018 IEEE Symposium on Security and Privacy (SP)*, pp. 19–35, 2018.

Kleinberg, J. and Raghavan, M. How do classifiers induce agents to invest effort strategically? In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pp. 825–844, 2019.

Li, B. and Vorobeychik, Y. Feature cross-substitution in adversarial classification. In *Advances in neural information processing systems*, pp. 2087–2095, 2014.

Miller, J., Milli, S., and Hardt, M. Strategic classification is causal modeling in disguise. *arXiv*, pp. arXiv–1910, 2019.

Milli, S., Miller, J., Dragan, A. D., and Hardt, M. The social cost of strategic classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, pp. 230–239, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361255. doi: 10.1145/3287560.3287576. URL https://doi.org/10.1145/3287560.3287576.

Mohri, M. and Munoz, A. Revenue optimization against strategic buyers. In *Advances in Neural Information Processing Systems*, pp. 2530–2538, 2015.

Moosavi-Dezfooli, S., Fawzi, A., Fawzi, O., and Frossard, P. Universal adversarial perturbations. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 86–94, 2017.

Mozaffari-Kermani, M., Sur-Kolay, S., Raghunathan, A., and Jha, N. K. Systematic poisoning attacks on and defenses for machine learning in healthcare. *IEEE Journal of Biomedical and Health Informatics*, 19(6):1893–1905, 2015.

Pathak, P. A. What really matters in designing school choice mechanisms. *Advances in Economics and Econometrics*, 1:176–214, 2017.

Perote, J. and Perote-Peña, J. Strategy-proof estimators for simple regression. *Math. Soc. Sci.*, 47:153–176, 2004.

Roth, A. E. What have we learned from market design? *Innovations: Technology, Governance, Globalization*, 3 (1):119–147, 2008.

Rubinstein, B. I., Nelson, B., Huang, L., Joseph, A. D., Lau, S.-h., Rao, S., Taft, N., and Tygar, J. D. Stealthy poisoning attacks on pca-based anomaly detectors. *SIGMETRICS Perform. Eval. Rev.*, 37(2): 73–74, October 2009. ISSN 0163-5999. doi: 10.1145/1639562.1639592. URL https://doi.org/10.1145/1639562.1639592.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shavit, Y., Edelman, B., and Axelrod, B. Causal strategic linear regression. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8676–8686, 2020.

Tearsheet. Gaming the system: Loan applicants are reverse engineering the online lending algorithms. https://tearsheet.co/data/gaming-the-system-online-loan-applicants-are-reverse-engineering-the-algorithms/.

Ustun, B., Spangher, A., and Liu, Y. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 10–19, 2019.

Vanunts, A. and Drutsa, A. Optimal pricing in repeated posted-price auctions with different patience of the seller and the buyer. In *Advances in Neural Information Processing Systems*, pp. 939–951, 2019.

Von Stackelberg, H. *Market structure and equilibrium*. Springer Science & Business Media, 2010.

Williams, J. and Haire, B. Why some people don't want to take a covid-19 test. https://theconversation.com/why-some-people-dont-want-to-take-a-covid-19-test-141794.

Zhang, H. and Conitzer, V. Incentive-aware pac learning. *AAAI 2021*, 2021.

Zhang, H., Cheng, Y., and Conitzer, V. Distinguishing distributions when samples are strategically transformed. In *Advances in Neural Information Processing Systems*, pp. 3193–3201, 2019a.

Zhang, H., Cheng, Y., and Conitzer, V. When samples are strategically selected. In *International Conference on Machine Learning*, pp. 7345–7353, 2019b.