Journal of Machine Learning Research 24 (2023) 1-38 Submitted 10/21; Revised 11/22; Published 6/23

PAC-learning for Strategic Classification

Ravi Sundaram R.SUNDARAM@NORTHEASTERN.EDU
Khoury College of Computer Science
Northeastern University
Boston, MA 02115, USA

Anil Vullikanti VSAKUMAR@VIRGINIA.EDU
Department of Computer Science, Biocomplexity Institute
University of Virginia
Charlottesville, VA 22904, USA

Haifeng Xu[*] HAIFENGXU@UCHICAGO.EDU
Department of Computer Science
University of Chicago
Chicago, IL 60637, USA

Fan Yao[†] FY4BC@VIRGINIA.EDU
Department of Computer Science
University of Virginia
Charlottesville, VA 22904, USA

Editor: Vahab Mirrokni

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 by considering strategic agents with heterogenous preferences, 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). Finally, we briefly investigate the power of randomization in our strategic classification setup. We show that randomization may strictly increase the accuracy in general, but will *not* help in the special case of *adversarial* classification with zero-manipulation-cost.

Keywords: Strategic classification, PAC-learning, strategic VC-dimension, linear classifier, strategic empirical risk minimization

∗. Corresponding author; Work done while this author is at the University of Virginia.
†. Corresponding author

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 and Crossroads; Williams and 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 and Haire). Policy makers sometimes prefer to make classification rules confidential (Citron and 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 and 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 and 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 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 setting 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 encompasses the classification under *evasion* attacks (Biggio et al., 2013; Goodfellow et al., 2015; Li and 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 com-

putationally, 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 behavior will *not* change its true label. Such behavior is referred to as *strategic gaming*, which crucially differs from *strategic improvement* studied recently (Kleinberg and 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* $\mathrm{SVC}(\mathcal{H}, R, c)$ 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 $d = \mathrm{SVC}(\mathcal{H}, R, c)$. 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 $\mathrm{SVC}(\mathcal{H}, R, c)$ for $R = \{-1, 1\}$ when data points are allowed to move within region $\{\boldsymbol{z}; c(\boldsymbol{z}; \boldsymbol{x}) \leq 1\}$ in the 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 $\mathrm{SVC}(\mathcal{H}, R, c) \leq 2$ 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 (Hardt et al., 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-invariant* 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 $R = \{-1, 1\}$ 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* preferences — 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.

Randomization in Strategic Classification. We examine the power and limits of *randomization* in our strategic classification setting. It is well-known that randomization does not improve classification accuracy in the non-strategic case (see, e.g., (Braverman and Garg, 2020)). We observe that this ceases to be true in strategic classification — there are examples where randomized linear classifiers can achieve strictly better accuracy than any deterministic linear classifier.[1] Interestingly, however, we prove that randomization does *not* improve accuracy for classification with zero-manipulation-cost adversaries, by leveraging the special properties of the cost function in adversarial classification.

1.2 Related Works

(Brückner and Scheffer, 2011) is one of the first to consider the Stackelberg game formulation of strategic classification, motivated by spam filtering; however they do not study generalization bounds. Zhang and Conitzer (2021) provide the sample complexity result for

1. Similar result has been observed by Braverman and Garg (2020), but their work has focused on on one-dimensional feature space

strategic PAC-learning under the homogeneous preference setting and in particular study the case under the incentive-compatibility constraints, i.e., subject to no data points will misreport features. Both works assume the positive labels are always and equally preferred. There has also been work on understanding the social implications of strategically robust classification (Akyol et al., 2016; Milli et al., 2019; Hu et al., 2019b); these works show that improving the learner's performance may lead to increased social burden and unfairness. Dong et al. (2018); Chen et al. (2020); Ahmadi et al. (2021) extend strategic linear classification to an online setting where the data points come in an online manner. These online learning setups do not have the notion of "training" and "testing" sets. Instead, all their data points are contaminated. Our setting however is in the more canonical PAC-learning setup but with more general agent preferences and thus crucially differs from them in multiple aspects: (1) we assume access to uncontaminated training data whereas testing data are contaminated (like classic strategic classification setups as in (Hardt et al., 2016; Hu et al., 2019b; Zhang and Conitzer, 2021)); (2) data points in our setups have arbitrary preferences and manipulation cost functions whereas these online learning setups all assume homogeneous agent preferences and typically special class of cost functions like l_2 distances (Ahmadi et al., 2021) or positive homogeneous functions (Dong et al., 2018). Finally, all these online setups have so far examined only linear classification whereas our strategic PAC learning framework is general albeit with linear classification as an important case study. To our knowledge, recent work by Tsirtsis et al. (2019) is the only work that considers heterogeneous data point preferences. However, their work focuses purely on the computational problem of computing the optimal classification policy. Moreover, their study a very different classification model with exponentially large discrete feature spaces, and thus their complexity results are not comparable to ours. All these aforementioned works, including the present work, consider *gaming* behaviors. A relevant but quite different line of recent works study *strategic improvements* where the manipulation does really change the inherent quality and labels (Kleinberg and Raghavan, 2019; Miller et al., 2019; Ustun et al., 2019; Bechavod et al., 2020; Shavit et al., 2020). The question there is mainly to design incentive mechanisms to encourage agents' efforts or improvements. Most relevant to ours is perhaps the strategic classification model studied by Hardt et al. (2016) and Zhang and Conitzer (2021), where Hardt et al. (2016) formally formulated the strategic classification problem as a repeated Stackelberg game and Zhang and 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 and Conitzer, 2021) by allowing agents' *heterogeneous* preferences over classification outcomes. Besides the modeling differences, the research questions we study are also quite different from (Hardt et al., 2016). Their positive results are derived under the assumption of *separable cost* functions or its variants, which appear too restrictive and somewhat unrealistic. For example, one consequence of separable cost functions is that for *any* two features $\boldsymbol{x}, \boldsymbol{z}$, the manipulation cost from either $\boldsymbol{x}$ to $\boldsymbol{z}$ or from $\boldsymbol{z}$ to $\boldsymbol{x}$ must be 0.[2]

2. A cost function $c(\boldsymbol{z}; \boldsymbol{x})$ is separable if there exists two functions $c_1, c_2 : \mathcal{X} \to \mathbb{R}$ such that $c(\boldsymbol{z}; \boldsymbol{x}) = \max\{c_2(\boldsymbol{z}) - c_1(\boldsymbol{x}), 0\}$. Since $c(\boldsymbol{x}; \boldsymbol{x}) = 0$, we have $c_2(\boldsymbol{x}) \leq c_1(\boldsymbol{x})$ for any $\boldsymbol{x}$. Therefore, $c_2(\boldsymbol{x}) + c_2(\boldsymbol{z}) - c_1(\boldsymbol{x}) - c_1(\boldsymbol{z}) \leq 0$. Consequently, either $c_2(\boldsymbol{x}) - c_1(\boldsymbol{z}) \leq 0$ or $c_2(\boldsymbol{z}) - c_1(\boldsymbol{x}) \leq 0$, yielding either $c(\boldsymbol{z}; \boldsymbol{x}) = 0$ or $c(\boldsymbol{x}; \boldsymbol{z}) = 0$.

This appears unrealistic in reality. For example, a high-school student with true average math grade 80 and true average literature grade 95 is likely to incur cost if she/he wants to appear as 95 for math and 80 for literature, and vice versa. This is because different students are good at different aspects. Our model imposes less assumptions on the cost functions by allowing any cost functions induced by seminorms. Our characterization of SVC equaling at most 2 under separable cost functions implies the PAC-learnability result of (Hardt et al., 2016), which serves more as our case study. One of our main contribution 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 seminorms. 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 and Vorobeychik, 2014; Carlini and 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 seminorms*.[3]

Strategic classification has been studied in other different settings or domains or for different purposes, including spam filtering (Brückner and Scheffer, 2011), 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 and 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 and 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 and Munoz, 2015; Vanunts and Drutsa, 2019). These works are similar in spirit to ours, but study a completely different set of problems using different techniques. Their results are not comparable to ours.

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 $h : \mathcal{X} \to \{+1, -1\}$. Our model is essentially

3. (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.

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.

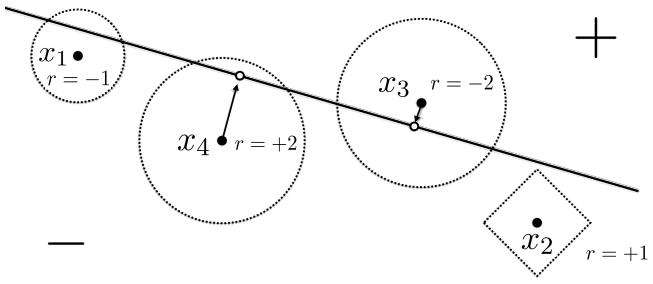


Figure 1: Example illustration of our setup. The line is a linear classifier. Points $\boldsymbol{x}_3, \boldsymbol{x}_4$ have incentive to cross the boundary whereas $\boldsymbol{x}_1, \boldsymbol{x}_2$ do not. The dotted cycles contain all manipulated features which have moving cost exactly 1 and they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

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 $c : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_{\geq 0}$. 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 *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:** With *instance-wise* cost functions, each data point is allowed to possess its own loss function. To capture this situation, we need to augment each data point's representation from $(\boldsymbol{x}, y, r)$ to $(\boldsymbol{x}, y, r, l)$, where l is a function depending on $\boldsymbol{x}$ such that $c(\boldsymbol{z}; \boldsymbol{x}) = l(\boldsymbol{z} - \boldsymbol{x})$ for some semi-norm l. For notational convenience, we often refer to function l associated with this data point as $l_{\boldsymbol{x}}$ though we emphasize that this is a slight abuse of notation because $l_{\boldsymbol{x}}$ is determined not just by the data point's feature $\boldsymbol{x}$ but rather by the data point itself.[4]

4. The only places where this notational discrepancy needs to be recognized are the proof of Theorem 10 and the second situation of Theorem 14, during which we will point out this to avoid confusion.

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:

$$\text{Data Point Best Response:} \qquad \Delta_c(\boldsymbol{x}, r; h) = \arg\max_{\boldsymbol{z}\in\mathcal{X}} \big[\mathbb{I}(h(\boldsymbol{z}) = 1) \cdot r - c(\boldsymbol{z}; \boldsymbol{x})\big]. \tag{1}$$

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 differ from strategic *improvements* (see (Miller et al., 2019) for more discussions on their differences). We also point out that the preference r is assumed to be known for any training data (though no need to be known for testing data). While this may appear a strong assumption at the first glance, we believe this parameter can be estimated in real applications (e.g., how much it matters to someone for being admitted to a university, for receiving a loan or for being tested COVID positive). An interesting future direction may be to understand how robust the strategic classifier is to the estimation error of r in the training data.

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 is 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:

$$\begin{aligned} &\text{Strategic 0-1 Loss of classifier } h: \\ &L_c(h; \mathcal{D}) = \mathop{\mathbf{Pr}}_{(\boldsymbol{x},y,r)\sim\mathcal{D}} \big[h(\Delta_c(\boldsymbol{x}, r; h)) \neq y)\big]. \end{aligned} \tag{2}$$

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 encompasses the adversarial classification problem (Cullina et al., 2018; Awasthi et al., 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 6 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 compared to our setting. 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*

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

We show that the SVC defined above correctly characterizes the learnability of any strategic classification problem $\text{STRAC}\langle\mathcal{H}, R, c\rangle$. 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 $\text{STRAC}\langle\mathcal{H}, R, c\rangle$ is defined as follows:

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

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 Eq. (2)). Unlike the standard (non-strategic) ERM problem and similar in spirit to the "incentive-aware ERM" in (Zhang and 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* $\text{STRAC}\langle\mathcal{H}, R, c\rangle$, *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 4 *Any strategic classification instance* $\text{STRAC}\langle\mathcal{H}, R, c\rangle$ *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 Let $\mathcal{Y} = \{+1, -1\}$. Define another binary hypothesis class $\tilde{H} = \{\kappa_c(h) : h \in H\}$, where $\kappa_c : (\mathcal{X} \to \mathcal{Y}) \to (\mathcal{X} \times R \to \mathcal{Y})$ is a mapping such that $\kappa_c(h)(\boldsymbol{x}, r) = h(\Delta_c(\boldsymbol{x}, r; h)), \forall(\boldsymbol{x}, r) \in \mathcal{X} \times R$. Note that the input of classifier $\kappa_c(h)$ consists of both the feature vector $\boldsymbol{x}$ and the preference r. By the definition of SVC, we have
$\text{VC}(\tilde{H}) = \text{SVC}(\mathcal{H}, R, c) = d$.

Given any distribution $\mathcal{D}$, cost function c, and $h \in \mathcal{H}$, the strategic 0-1 loss of h is $L_c(h, \mathcal{D}) = \mathbb{E}_{(\boldsymbol{x},y,r)\sim\mathcal{D}}\Big[\mathbb{I}\big[\kappa_c(h)(\boldsymbol{x}, r) \neq y\big]\Big] = L(\kappa_c(h), \mathcal{D})$, where $L(\tilde{h}, \mathcal{D})$ is the standard expected risk of the newly defined $\tilde{h} \in \tilde{\mathcal{H}}$ under the distribution $\mathcal{D}$ in the non-strategic setting. Therefore, studying the PAC sample complexity upper bound for $\mathcal{H}$ under the strategic setting $\langle R, c\rangle$ is equivalent to studying the sample complexity for $\tilde{H}$ in the non-strategic setting. The latter problem can be addressed by employing the standard PAC learning analysis. From the Fundamental Theorem of Statistical Learning (Theorem 6.8 in (Shalev-Shwartz and Ben-David, 2014)), we know $\tilde{H}$ is agnostic PAC learnable with sample complexity $O(\epsilon^{-2}(\text{VC}(\tilde{H}) + \log\frac{1}{\delta}))$, meaning that there exists a constant C such that for

any $(\delta, \epsilon) \in (0,1)^2$ and any distribution $\mathcal{D}$ for $(\boldsymbol{x}, y, r)$, as long as $n \geq C \cdot \epsilon^{-2}(\mathrm{VC}(\tilde{H}) + \log \frac{1}{\delta})$, with at least probability $1-\delta$, we have

$$L(\tilde{h}^*, \mathcal{D}) - \inf_{\tilde{h} \in \tilde{H}} L(\tilde{h}, \mathcal{D}) \leq \epsilon,$$

where $\tilde{h}^*$ is the solution of ERM with n i.i.d. samples from $\mathcal{D}$ as input. Let h^* be the solution of the corresponding SERM conditioned on the same n i.i.d. samples from $\mathcal{D}$. By the definition of $\tilde{H}$ and L_c, we have $L_c(h^*, \mathcal{D}) = L(\tilde{h}^*, \mathcal{D})$, and $\inf_{h \in \mathcal{H}} L_c(h, \mathcal{D}) = \inf_{\tilde{h} \in \tilde{H}} L(\tilde{h}, \mathcal{D})$. Therefore, with probability $1-\delta$, we have

$$L_c(h^*, \mathcal{D}) - \inf_{h \in \mathcal{H}} L_c(h, \mathcal{D}) \leq \epsilon,$$

which implies $\textsc{StraC}\langle \mathcal{H}, R, c \rangle$ is agnostic PAC learnable with sample complexity $O(\epsilon^{-2}[d + \log(\frac{1}{\delta})])$ by the SERM. ■

Next, we illustrate how SVC connects to previous literature, particularly the two most relevant works by (Cullina et al., 2018) and (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 6 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 a 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 5 *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 6 *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 $AVC(\mathcal{H}, \mathcal{B}) = SVC(\mathcal{H}, \{+r, -r\}, c)$.*

For readability, we defer the complete proof for some of the results to the appendix, while only provide their proof sketches in the main paper. The full proof of Proposition 6 is deferred to Appendix A.1. To reveal this relationship, we directly examine the definitions of AVC and SVC and demonstrate that the former captures a special case of the latter. As a corollary of Proposition 6, 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 7 *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 7 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 prefer 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 8 *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 $r > 0$.*

In the proof we construct an instance with a universe set $\mathcal{X} = [n] \cup \mathcal{S}$ where $[n] = \{1, 2, \cdots, n\}$ is the set of n elements and $\mathcal{S}$ be the power set of $[n]$. The hypothesis class $\mathcal{H}$ is the set of all the point classifiers with points from $\mathcal{S}$. We then design an instance-invariant cost function which leads to the desired VC dimension bounds. The detailed proof can be found in Appendix A.3.

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 $c(\boldsymbol{z}; \boldsymbol{x}) = \max\big\{c_2(\boldsymbol{z}) - c_1(\boldsymbol{x}), 0\big\}$.

The following Proposition 9 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.[5] Therefore, separable cost function essentially reduces any classification problem to a problem in lower dimension. Together with Theorem 4, Proposition 9 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 9 *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 $SVC(\mathcal{H}, R, c) \leq 2$.*

The key idea of the proof is to show that the "manipulation regions" A, B, C of three arbitrary points can be ordered so that they must without loss of generality satisfy $A \subseteq B \subseteq C$. Consequently, these points cannot be shattered. We defer the full proof to appendix A.4. The assumption $0 \notin R$ implies that 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 9 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 seminorms.[6] 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 explicitly expressed as:

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

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 $R = \{+1, -1\}$.

5. The model of (Hardt et al., 2016) corresponds to the case $R = \{1\}$ in our model. For that restricted situation, the proof of Proposition 9 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.
6. 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 $\boldsymbol{x} \in \mathcal{X}, \lambda \in \mathbb{R}$.

Theorem 10 *Consider strategic linear classification* $\text{STRAC}\langle\mathcal{H}_d, R, c\rangle$. *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 Let $\mathcal{X} = \mathbb{R}^2$, and consider the linear hypothesis class on $\mathcal{X}$: $\mathcal{H} = \{h = \text{sgn}(\boldsymbol{w} \cdot \boldsymbol{x} + b) : (\boldsymbol{w}, b) \in \mathbb{R}^3, \boldsymbol{x} \in \mathcal{X}\}$. We show that for any $n \in \mathbb{Z}^+$ and $R = \{+1\}$, there exist n points $\{\boldsymbol{x}_i\}_{i=1}^n \in \mathcal{X}^n$ and corresponding cost functions $\{c_i\}_{i=1}^n$, such that the n'th shattering coefficients $\sigma_n(\mathcal{H}, R, \{c_i\}_{i=1}^n) = 2^n$ (see Definition 1 for σ_n). Note that the cost function is instance-wise. Note that our construction here will use different cost functions even though each data point i is at the same location $(0, 0)$.

Let $\boldsymbol{x}_i = (0, 0), \forall i \in [n]$ be the set of data points. The main challenge of the proof is a very careful construction of the cost function for each data point. To do so, we first pick a set of 2^n different points $S = \{s_j\}_{j=1}^{2^n}$ lying on the *unit circle*, i.e., $S \subset \{(x, y) : x^2+y^2 = 1\}$. The number 2^n is *not* arbitrarily chosen — indeed, we will map each point s_j to one of the 2^n subsets of $[n]$ in a *bijective* manner so that each s_j corresponds to a unique subset of $[n]$. What are these 2^n different points will not matter to our construction neither it matters which point is mapped to which subset so long as it is a bijection. Since we have the freedom to pick the locations of elements in S, we will pick any S such that $\bar{S} \cap S = \emptyset$, where $\bar{S} = \{(-x, -y) : (x, y) \in S\}$ is the set that is origin-symmetric to S. We emphasize that $\bar{S}$ is chosen to "symmetrize" our construction in order to obtain a norm and it does not need to have any interpretation. For any $\boldsymbol{x}_i$, we now define its cost function c_i through the following steps :

1. Let $S_i = \{s \subseteq [n] : i \in s\} \subset S$ contain all the 2^{n-1} subsets of $[n]$ that include the element i.

2. Let $\bar{S}_i = \{(-x, -y) : (x, y) \in S_i\} \subseteq \bar{S}$ be the set that is origin-symmetric to S_i.

3. Let G_i be the convex, origin-symmetric polygon with the vertex set being $S_i \cup \bar{S}_i$.

4. The cost function of $\boldsymbol{x}_i$ is defined as $c_i(\boldsymbol{z};\boldsymbol{x}) = \|\boldsymbol{x}-\boldsymbol{z}\|_{G_i}$, where $\|\cdot\|_{G_i} = \inf\{\epsilon \in \mathbb{R}_{\geq 0} : x \in \epsilon G_i\}$ is a norm derived from polygon G_i (note the origin-symmetry of $S_i \cup \bar{S}_i$ and thus G_i).

Next we show that for any label pattern $\mathcal{L} \in \{+1, -1\}^n$, there exists some linear classifier $h \in \mathcal{H}_2$ such that $(h(\Delta_{c_1}(\boldsymbol{x}_1, +1; h)), \cdots, h(\Delta_{c_n}(\boldsymbol{x}_n, +1; h))) = \mathcal{L}$.

With slight abuse of notation, let $s_{\mathcal{L}} = \{i \in [n] : \mathcal{L}_i = +1\} \in S$ be the point in S that corresponds to the set of the indexes of $\mathcal{L}$ with $\mathcal{L}_i = 1$. Let $h_{\mathcal{L}}$ be any *linear classifier* whose decision boundary intersects the unit circle centered at $\boldsymbol{x}_i$ and *strictly* separates $s_{\mathcal{L}}$ from all the other elements in $S \cup \bar{S}$. We will use $h_{\mathcal{L}}$ to denote both the linear classifier and its decision boundary (i.e., a line in $\mathbb{R}^2$) interchangeably. Due to the convexity of G_i, such $h_{\mathcal{L}}$ must exist. We further let $h_{\mathcal{L}}$ give prediction result $+1$ for the half plane that contains $s_{\mathcal{L}}$ and -1 for the other half plane. Figure 2 illustrates the geometry of this example.

We now argue that $h_{\mathcal{L}}$ induces the given label pattern $\mathcal{L}$ for instances $\{(\boldsymbol{x}_i, 1, c_i)\}_{i=1}^n$. To see this, we examine $h_{\mathcal{L}}(\Delta_{c_i}(\boldsymbol{x}_i, 1; h))$ for each i:

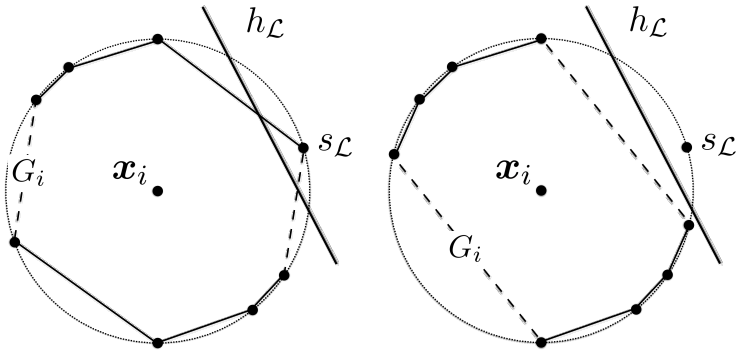


Figure 2: Left: If $i \in s_{\mathcal{L}}$, $h_{\mathcal{L}}$ intersects with G_i, and $\boldsymbol{x}_i$ can manipulate its feature within G_i to cross $h_{\mathcal{L}}$. Right: If $i \notin s_{\mathcal{L}}$, $h_{\mathcal{L}}$ and G_i are disjoint; $\boldsymbol{x}_i$ cannot manipulate its feature within G_i to cross $h_{\mathcal{L}}$. Given any label pattern $\mathcal{L} \in \{+1, -1\}^n$, G_i is the convex, origin-symmetric polygon associated with $\boldsymbol{x}_i$'s cost function. The linear classifier $h_{\mathcal{L}}$ is chosen to separate $s_{\mathcal{L}}$ from all other elements in $\bar{S} \cup S$ and classifies $s_{\mathcal{L}}$ as $+1$. The left/right panel shows the two situations, depending on $i \in s_{\mathcal{L}}$ or $i \notin s_{\mathcal{L}}$.

1. If $i \in s_{\mathcal{L}}$, then $s_{\mathcal{L}} \in S_i$ and $\boldsymbol{x}_i$ can move to $s_{\mathcal{L}}$ with cost $c_i(s_{\mathcal{L}}; \boldsymbol{x}_i) < 1$. This is because G_i is convex and there exists a point $\boldsymbol{x}_i'$ on $h_{\mathcal{L}}$ such that $c_i(\boldsymbol{x}_i'; \boldsymbol{x}_i) < c_i(s_{\mathcal{L}}; x_i) = 1 = r_i$ (e.g., choose $\boldsymbol{x}_i'$ as the intersection point of the segment $[\boldsymbol{x}_i, s_{\mathcal{L}}]$ and $h_{\mathcal{L}}$). Therefore, $h_{\mathcal{L}}$ will classify $\boldsymbol{x}_i$ as positive. This case is shown in the left panel of Figure 2.

2. If $i \notin s_{\mathcal{L}}$, then $s_{\mathcal{L}} \notin S_i$ and G_i does not intersect $h_{\mathcal{L}}$. In this case, $h_{\mathcal{L}}(\boldsymbol{x}) = -1$, and moving across $h_{\mathcal{L}}$ always induces a cost strictly larger than 1. Therefore, the best response for $\boldsymbol{x}_i$ is to stay put and $h_{\mathcal{L}}$ will classify $\boldsymbol{x}_i$ as negative. This case is shown in the right panel of Figure 2.

Now we have shown that the n'th shattering coefficients $\sigma_n(\mathcal{H}, \{+1, -1\}, \{c_i\}_{i=1}^n) = 2^n$. Since n can take any integer, we conclude the strategic VC-dimension in this case is $+\infty$. ■

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 (10) 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 despite allowing richer manipulation preferences of data points.

Theorem 11 *Consider an instance of strategic linear classification* $\text{STRAC}\langle\mathcal{H}_d, R, c\rangle$. *For any* instance-invariant *cost function* $c(\boldsymbol{z};\boldsymbol{x}) = l(\boldsymbol{z}-\boldsymbol{x})$ *where* l *is a seminorm, 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* $\mathcal{B} = \{\boldsymbol{x} : l(\boldsymbol{x}) \le 1\}$.

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 The following lemma is well-known in algebra and will be useful for our analysis.

Lemma 12 *For any seminorm* $l : \mathbb{R}^d \to \mathbb{R}_{\ge 0}$, *and the cost function* $c(\boldsymbol{z};\boldsymbol{x}) = l(\boldsymbol{z}-\boldsymbol{x})$ *induced by* l, *the minimum manipulation cost for* $\boldsymbol{x}$ *to move to the hyperplane* $\boldsymbol{w}\cdot\boldsymbol{x}+b=0$ *is given by the following:*

$$\min_{\boldsymbol{x}'}\{c(\boldsymbol{x}';\boldsymbol{x}) : \boldsymbol{w}\cdot\boldsymbol{x}'+b=0\} = \frac{|\boldsymbol{w}\cdot\boldsymbol{x}+b|}{l^*(\boldsymbol{w})}$$

where $l^*(\boldsymbol{w}) = \sup_{\boldsymbol{z}\in B}\{\boldsymbol{w}\cdot\boldsymbol{z}\} \in \mathbb{R}_{\ge 0}\cup\{+\infty\}$, *and* $\mathcal{B} = \{\boldsymbol{z} : l(\boldsymbol{z}) \le 1\}$ *is the unit ball induced by* l.

The proof has the following two parts. The first part is the more involved one.

Proof of SVC$(\mathcal{H}_d, R, c) \le d + 1 - \dim(V_l)$**:**

It suffices to show that for any $n > d + 1 - \dim(V_l)$ and n data points $(\boldsymbol{x}_i, r_i) \in \mathbb{R}^d \times R, \forall i = 1, \cdots, n$, there exists a label pattern $\mathcal{L} \in \{+1, -1\}^n$, such that for any $h \in \mathcal{H}_d$ it cannot induce $\mathcal{L}$, i.e.,

$$(h(\Delta_c(\boldsymbol{x}_1, r_1; h), \cdots, h(\Delta_c(\boldsymbol{x}_n, r_n; h))) \ne \mathcal{L}.$$

The first step of our proof derives a succinct characterization about the classification outcome for a set of data points. For any seminorm l, it is known the set $\mathcal{B} = \{x : l(x) \le 1\}$ is nonempty, closed, convex, and origin-symmetric. Let $l^*(\boldsymbol{w}) = \sup_{\boldsymbol{z}\in B}\{\boldsymbol{w}\cdot\boldsymbol{z}\}$. We have $l^*(\boldsymbol{w}) > 0$ for all $\boldsymbol{w} \ne \boldsymbol{0}$ since $\boldsymbol{0}$ is an interior point of $\mathcal{B}$. According to Lemma 12, for any $\boldsymbol{x} \in \mathbb{R}^d$ and any linear classifier $h = (\boldsymbol{w}, b) \in \mathcal{H}_d$, the minimum manipulation cost for $\boldsymbol{x}$ to move to the decision boundary of h is $|\boldsymbol{w}\cdot\boldsymbol{x}+b|/l^*(\boldsymbol{w})$. Note that we may w.l.o.g. restrict to $\boldsymbol{w}$'s such that $l^*(\boldsymbol{w}) = 1$ since the sign function $\text{sgn}(\boldsymbol{w}\cdot\boldsymbol{x}+b)$ does not change after re-scaling. For any data point $(\boldsymbol{x}, r) \in \mathcal{X} \times R$ and linear classifier $h \in \mathcal{H}_d$, we define the *signed* manipulation cost to the classification boundary as

$$\delta(h, \boldsymbol{x}) = h(\boldsymbol{x}) \cdot \frac{|\boldsymbol{w}\cdot\boldsymbol{x}+b|}{l^*(\boldsymbol{w})} = \boldsymbol{w}\cdot\boldsymbol{x}+b,$$

using the condition $l^*(\boldsymbol{w}) = 1$. We claim that $h(\Delta_c(\boldsymbol{x}, r; h)) = 2\mathbb{I}(\boldsymbol{w}\cdot\boldsymbol{x}+b \ge -r) - 1$. This follows a case analysis:

1. If $r \le 0$, then $h(\Delta_c(\boldsymbol{x}, r, h)) = 1$ if and only if $h(\boldsymbol{x}) = 1$ and $\boldsymbol{x}$ *cannot* move across the decision boundary of h within cost $|r| = -r$. This implies $h(\Delta_c(\boldsymbol{x}, r; h)) = 2\mathbb{I}(\boldsymbol{w}\cdot\boldsymbol{x}+b \ge -r) - 1$.

2. If $r > 0$, then $h(\Delta_c(\boldsymbol{x}, r, h)) = -1$ if and only if $h(\boldsymbol{x}) = -1$ and $\boldsymbol{x}$ cannot move across the decision boundary of h within cost r. In this case, $h(\Delta_c(\boldsymbol{x}, r; h)) = -(2\mathbb{I}(-(\boldsymbol{w} \cdot \boldsymbol{x} + b) > r) - 1) = 2\mathbb{I}(\boldsymbol{w} \cdot \boldsymbol{x} + b \geq -r) - 1$. Note that the first inequality holds strictly because we assume h always gives $+1$ for those $\boldsymbol{x}$ on the decision boundary.

For a set of samples $(\boldsymbol{X}, \boldsymbol{r})$ where $\boldsymbol{X} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n), \boldsymbol{r} = (r_1, \cdots, r_n)$, define the set of all possible vectors (over the choice of linear classifiers $(\boldsymbol{w}, b) \in \mathcal{H}_d$) of signed manipulation costs as

$$\mathcal{D}(\mathcal{H}_d, \boldsymbol{X}) = \{(\boldsymbol{w} \cdot \boldsymbol{x}_1 + b, \cdots, \boldsymbol{w} \cdot \boldsymbol{x}_n + b) : h \in \mathcal{H}_d\}, \tag{6}$$

there is a $h \in \mathcal{H}_d$ that achieves a label pattern $\mathcal{L}$ on $(\boldsymbol{X}, \boldsymbol{r})$ if and only if there exists an element in $D(\mathcal{H}_d, \boldsymbol{x}) + \boldsymbol{r}$ with the corresponding sign pattern $\mathcal{L}$.

Recall that a linear classifier is described by $(\boldsymbol{w}, b) \in \mathbb{R}^{d+1}$. The second step of our proof rules out "trivial" linear classifiers under strategic behaviors, and consequently allows us to work with only linear classifiers in a linear space of smaller dimension. Let $\mathcal{B} = \{\boldsymbol{x} : l(\boldsymbol{x}) \leq 1\}$ and V_l be the largest linear space contained in $\mathcal{B}$. We argue that it suffice to consider only linear classifiers $(\boldsymbol{w}, b)$ with $\boldsymbol{w} \perp V_l$. This is because for any $\boldsymbol{w}$ that is not orthogonal to the subspace V_l, we can find $\bar{\boldsymbol{z}} \in V_l$ such that $c(\bar{\boldsymbol{z}}; \boldsymbol{x}) = 0$ and $\boldsymbol{w} \cdot \bar{\boldsymbol{z}} \to \infty$ since V_l is a linear subspace. This means any data point can induce its preferred label $\text{sgn}(r)$ with 0 cost, by moving to $\bar{\boldsymbol{z}}$ if $\text{sgn}(r) = +$ and $-\bar{\boldsymbol{z}}$ otherwise. Any such linear classifier will result in the same label pattern, simply specified by $\text{sgn}(r)$. As a consequence, we only need to focus on linear classifiers $(\boldsymbol{w}, b)$ with $\boldsymbol{w} \perp V_l$. Let $\tilde{H}_d = \{(\boldsymbol{w}, b) : \boldsymbol{w} \perp V_l\}$ denote all such linear classifiers.

Next, we argue that when restricting to the non-trivial class of linear classifiers $\tilde{\mathcal{H}}_d$, the $\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})$ defined in Equation (6) lies in a linear subspace with dimension at most $d + 1 - \dim(V_l)$. Consider the linear mapping $\mathcal{G}_{\boldsymbol{X}} : \tilde{\mathcal{H}}_d \to \mathbb{R}^n$ determined by the data features $\boldsymbol{X}$, defined as

$$\mathcal{G}_{\boldsymbol{X}}(\boldsymbol{w}, b) = (\boldsymbol{w} \cdot \boldsymbol{x}_1 + b, \cdots, \boldsymbol{w} \cdot \boldsymbol{x}_n + b), \quad \forall (\boldsymbol{w}, b) \in \tilde{\mathcal{H}}_d.$$

Since $\boldsymbol{w} \perp V_l$, $\boldsymbol{w}$ is from a linear subspace of $d - \dim(V_l)$. Linear mapping will not increase the dimension of the image space, therefore $\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})$ lies in a space with dimension at most $d + 1 - \dim(V_l)$.

Finally, we prove that there must exist label patterns that cannot be induced by linear classifiers whenever the number of data points $n > d + 1 - \dim(V_l)$. Let $\text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big)$ denote the smallest linear space that contains $\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})$. Since $\text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big)$ has dimension at most $d + 1 - \dim(V_l) < n$ but $\text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big) \subset \mathbb{R}^n$, there must exist a non-zero vector $\bar{\boldsymbol{u}} \in \mathbb{R}^n$ such that: (1) $\bar{\boldsymbol{u}} \neq \boldsymbol{0}$; (2) $\bar{\boldsymbol{u}} \perp \text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big)$ (i.e., $\bar{\boldsymbol{u}} \cdot \boldsymbol{v} = 0, \forall \boldsymbol{v} \in \text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big)$); and (3) $\bar{\boldsymbol{u}} \cdot \boldsymbol{r} \leq 0$ (if $\bar{\boldsymbol{u}} \cdot \boldsymbol{r} \geq 0$, simply takes its negation). Note that this implies $\bar{\boldsymbol{u}} \cdot \boldsymbol{v} \leq 0, \forall \boldsymbol{v} \in \text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big) + \boldsymbol{r}$.

We argue that the sign pattern of the vector $\bar{\boldsymbol{u}}$, denoted as $\text{sgn}(\bar{\boldsymbol{u}})$, and the sign pattern of all negatives ($\mathcal{L} = (-1, \cdots, -1)$) cannot be achieved simultaneously by $\tilde{\mathcal{H}}_d$. Suppose $\text{sgn}(\bar{\boldsymbol{u}})$ can be achieved by $\tilde{\mathcal{H}}_d$, then there must exist $\boldsymbol{v}^1 \in \text{span}(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})) + \boldsymbol{r}$ such that $\text{sgn}(\bar{\boldsymbol{u}}) = \text{sgn}(\boldsymbol{v}^1)$ and $\bar{\boldsymbol{u}} \cdot \boldsymbol{v}^1 \leq 0$. Since $\text{sgn}(\bar{\boldsymbol{u}}) = \text{sgn}(\boldsymbol{v}^1)$ also implies $\bar{\boldsymbol{u}} \cdot \boldsymbol{v}^1 \geq 0$, we thus have $\bar{\boldsymbol{u}} \cdot \boldsymbol{v}^1 = \sum_{j=1} \bar{u}_j v_j^1 = 0$. We claim that there must exist j such that $\bar{u}_j > 0$. First of all, we cannot have $\bar{u}_j < 0$ for any j since that implies $v_j^1 < 0$ (only strictly less v_j^1's

will be assigned -1 pattern due to our tie breaking rule) and consequently, $\bar{\boldsymbol{u}} \cdot \boldsymbol{v}^1 < 0$, a contradiction. Also note that $\bar{\boldsymbol{u}} \neq \boldsymbol{0}$, so there exists $j \in [n]$ such that $\bar{u}_j > 0$.

Utilizing the above property of $\bar{\boldsymbol{u}}$, we show that the sign pattern $\mathcal{L} = (-1, \cdots, -1)$ cannot be achieved by $\tilde{\mathcal{H}}_d$. Suppose, for the sake of contradiction, that this is not true. Then there exists another $\boldsymbol{v}^2 = (v_1^2, \cdots, v_n^2) \in \text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big) + \boldsymbol{r}$ with all its elements being strictly negative. Now consider $\boldsymbol{v} = \boldsymbol{v}^1 - \boldsymbol{v}^2 \in \text{span}(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X}))$, we have $\bar{u} \cdot \boldsymbol{v} = \bar{u} \cdot \boldsymbol{v}^1 - \bar{u} \cdot \boldsymbol{v}^2 = 0 - \bar{u} \cdot \boldsymbol{v}^2 > 0$. Here the inequality holds because $\bar{u}_j \geq 0, v_j^2 < 0$ for all j and there exists some j such that $\bar{u}_j > 0$. Therefore, we draw a contradiction to the fact that $\bar{\boldsymbol{u}} \cdot \boldsymbol{v} = 0$ for any $\boldsymbol{v} \in \text{span}(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X}))$.

Now we proved that $\text{sgn}(\bar{\boldsymbol{u}})$ and $\mathcal{L} = (-1, \cdots, -1)$ cannot be achieved simultaneously by non-trivial classifiers $\tilde{\mathcal{H}}_d$, and the only achievable sign pattern for trivial classifiers is $\text{sgn}(\boldsymbol{r})$. Note that $\boldsymbol{r} \in \text{span}\big(\mathcal{D}(\tilde{\mathcal{H}}_d, \boldsymbol{X})\big) + \boldsymbol{r}$, $\text{sgn}(\boldsymbol{r})$ is thus also achievable by $\tilde{\mathcal{H}}_d$. Therefore, the trivial classifier has no contributions to the shattering coefficient, and we conclude at least one of $\text{sgn}(\bar{\boldsymbol{u}})$ and $\mathcal{L} = (-1, \cdots, -1)$ cannot be achieved by $\mathcal{H}_d$.

Proof of SVC$(\mathcal{H}_d, R, c) \geq d + 1 - \dim(V_l)$:

The second step of the proof shows $\text{SVC}(\mathcal{H}, R, c) \geq d + 1 - \dim(V_l)$ by giving an explicit construction of $(\boldsymbol{X}, \boldsymbol{r})$ that can be shattered by $\mathcal{H}_d$. Let $\boldsymbol{x}_0 = \boldsymbol{0}$, and $(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_t)$ be a basis of the subspace orthogonal to V_l, $(\boldsymbol{x}_{t+1}, \cdots, \boldsymbol{x}_d)$ be a basis of the subspace V_l, where $t = d - \dim(V_l)$.

We claim that the $t + 1 = d + 1 - \dim(V_l)$ data points in $\{0, 1, \cdots, t\}$ can be shattered by $\mathcal{H}_d$. In particular, for any given subset $S \subseteq \{0, 1, \cdots, t\}$, consider the linear system

$$\begin{cases} \boldsymbol{x}_i \cdot \boldsymbol{w}_S + b_S = 1, & \text{if} \quad i \in S \\ \boldsymbol{x}_i \cdot \boldsymbol{w}_S + b_S = -1, & \text{if} \quad i \leq t, \text{ and } i \notin S \\ \boldsymbol{x}_i \cdot \boldsymbol{w}_S = 0, & t + 1 \leq i \leq d. \end{cases}$$

Because $(\boldsymbol{x}_1, \cdots, \boldsymbol{x}_d)$ has full rank, the solution $(\boldsymbol{w}_S, b_S)$ must exist. Therefore, the half-plane $h = \boldsymbol{w}_S \cdot \boldsymbol{x} + b_S$ separates S and $\{\boldsymbol{x}_0, \cdots, \boldsymbol{x}_d\}/S$. Now consider the case when each $\boldsymbol{x}_i$ has a strategic preference $r_i \in R$. Since $\boldsymbol{w}_S$ is chosen to be orthogonal to V_l, $\boldsymbol{w}_S \cdot \boldsymbol{x}_i$ is bounded when $\boldsymbol{x}_i \in \{\boldsymbol{z} : c(\boldsymbol{z}; \boldsymbol{x}_i) \leq r_i\}$. Let $\delta_S = \max_{0 \leq i \leq t}\{\sup\{\boldsymbol{w}_S \cdot (\boldsymbol{z} - \boldsymbol{x}_i) : c(\boldsymbol{z}; \boldsymbol{x}_i) \leq r_i\}\}$, and $\delta = \max(1, 2\delta_S)$. Then the data set $\{\delta\boldsymbol{x}_0, \cdots, \delta\boldsymbol{x}_t\}$ can be shattered by $\mathcal{H}_d$ for any given c, R, because the classifier $(\delta\boldsymbol{w}_S, \delta b_S)$ separates the subset S and the other points regardless their strategic responses. ■

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 realizable 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 13 (Essentially Adversarial Instances) *For any strategic classification problem* $\textsc{StraC}\langle \mathcal{H}, R, c\rangle$*, let*

$$\begin{aligned} &min^- = \min\{r : (\boldsymbol{x}, y, r) \text{ with } y = -1\} \quad \text{and} \\ &max^+ = \max\{r : (\boldsymbol{x}, y, r) \text{ with } y = +1\} \end{aligned} \tag{7}$$

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 14) 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 15 shows that the SERM problem becomes NP-hard once we go slightly beyond the adversarial setups identified in Theorem 14. 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 10 and 11 where the preference set R did not play much role.

Theorem 14 *Any separable strategic linear classification instance* $\textsc{StraC}\langle \mathcal{H}_d, R, c\rangle$ *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 a 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 For any data point $(\boldsymbol{x}, y, r)$, let the manipulation cost for the data point be $c(\boldsymbol{z}; \boldsymbol{x}) = l_{\boldsymbol{x}}(\boldsymbol{z} - \boldsymbol{x})$ where $l_{\boldsymbol{x}}$ is any seminorm. Since the instance is separable, there exists a hyperplane $h : \boldsymbol{w} \cdot \boldsymbol{x} + b = 0$ that separates the given n training points $(\boldsymbol{x}_1, y_1, r_1), \cdots, (\boldsymbol{x}_n, y_n, r_n)$ under strategic behaviors. The SERM problem is thus a feasibility problem, which we

now formulate. Utilizing Lemma 12 about the signed distance from $\boldsymbol{x}_i$ to hyperplane h under cost function $c(\boldsymbol{z}; \boldsymbol{x}_i) = l_{\boldsymbol{x}_i}(\boldsymbol{z} - \boldsymbol{x}_i)$, we can formulate the SERM problem under the separability assumption. Concretely, we would like to find a hyperplane $h : \boldsymbol{w} \cdot \boldsymbol{x} + b = 0$ such that it satisfies the following for any $(\boldsymbol{x}_i, y_i, r_i)$:

1. If $y_i = 1$ and $r_i \geq 0$, we must have either $\boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq 0$ or $\boldsymbol{w} \cdot \boldsymbol{x}_i + b \leq 0$ and $\frac{-(\boldsymbol{w} \cdot \boldsymbol{x} + b)}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} \leq r_i$;
2. If $y_i = 1$ and $r_i \leq 0$, we must have $\frac{\boldsymbol{w} \cdot \boldsymbol{x} + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} \geq -r_i$ (this implies $\boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq 0$);
3. If $y_i = -1$ and $r_i \leq 0$, we must have either $\boldsymbol{w} \cdot \boldsymbol{x}_i + b \leq 0$ or $\boldsymbol{w} \cdot \boldsymbol{x}_i + b > 0$ and $\frac{\boldsymbol{w} \cdot \boldsymbol{x} + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} < -r_i$;
4. If $y_i = -1$ and $r_i \geq 0$, we must have $\frac{-(\boldsymbol{w} \cdot \boldsymbol{x} + b)}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} > r_i$ (this implies $\boldsymbol{w} \cdot \boldsymbol{x}_i + b < 0$);

Note that we classify any point on the hyperplane as $+1$ as well, which is why the strict inequality for Case 3 and 4. Case 1 can be summarized as $\frac{\boldsymbol{w} \cdot \boldsymbol{x} + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} \geq -r_i$. Similarly, Case 3 can be summarized as $\frac{\boldsymbol{w} \cdot \boldsymbol{x} + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} < -r_i$. To impose the strict inequality for Case 3 and 4, we may introduce an ϵ slack variable. These observations lead to the following formulation of the SERM problem.

$$\begin{array}{lll} \text{find} & \boldsymbol{w}, b, \epsilon > 0 & \\ \text{subject to} & \frac{\boldsymbol{w} \cdot \boldsymbol{x}_i + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} \geq -r_i, & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } y_i = 1. \\ & \frac{\boldsymbol{w} \cdot \boldsymbol{x}_i + b}{l^*_{\boldsymbol{x}_i}(\boldsymbol{w})} \leq -r_i - \epsilon, & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } y_i = -1. \end{array} \tag{8}$$

We now consider the two settings as described in the theorem statement. We first consider **Situation 1**, i.e., the essentially adversarial case with $\min^- \geq \max^+$ and an instance-invariant cost function induced by the same seminorm l, i.e., $c(\boldsymbol{z}; \boldsymbol{x}) = l(\boldsymbol{x} - \boldsymbol{z})$ for any $\boldsymbol{x}$. In this case, System (8) is equivalent to the following

$$\begin{array}{lll} \text{find} & \boldsymbol{w}, b, \epsilon > 0 & \\ \text{subject to} & \boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq -r_i, & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } y_i = 1. \\ & \boldsymbol{w} \cdot \boldsymbol{x}_i + b \leq -(r_i + \epsilon), & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } y_i = -1. \\ & l^*(\boldsymbol{w}) = 1 & \end{array} \tag{9}$$

This system is unfortunately not a convex feasibility problem. To solve System (9), we consider the following optimization program (OP), which is a relaxation of System (9) by relaxing the non-convex constraint $l^*(\boldsymbol{w}) = 1$ to the convex constraint $l^*(\boldsymbol{w}) \leq 1$.

$$\begin{array}{lll} \text{maximize} & \epsilon & \\ \text{subject to} & \boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq -r_i, & \text{for points } (\boldsymbol{x}_i, r_i) \text{ with label 1.} \\ & \boldsymbol{w} \cdot \boldsymbol{x}_i + b \leq -r_i - \epsilon, & \text{for points } (\boldsymbol{x}_i, r_i) \text{ with label -1.} \\ & l^*(\boldsymbol{w}) \leq 1 & \end{array} \tag{10}$$

Note that OP (10) is a convex program because the objective and constraints are either linear or convex. Therefore, OP (10) can be efficiently solved in polynomial time.[7] Note

7. Note that without additional assumptions on the objective and constraints, convex programs can only be solved up to precision ϵ in poly$(1/\epsilon)$ time (Nesterov et al., 2018). In this case, we simply say it can be "solved" efficiently.

that this relaxation is not tight in general as we will show later that solving System (9) is NP-hard in general.

Our main insight is that under the assumption of $\min^- \geq \max^+$, the above relaxation is tight — i.e., there always exists an optimal solution to the above problem with $l^*(\boldsymbol{w}) = 1$. This solution is then a feasible solution to System (9) as well, thus completing our proof. Concretely, given any optimal solution $(\boldsymbol{w}^*, b^*, \epsilon^*)$ to OP (10), we construct another solution $(\bar{\boldsymbol{w}}, \bar{b}, \bar{\epsilon})$ as follows:

$$\bar{\boldsymbol{w}} = \frac{\boldsymbol{w}^*}{\alpha}, \quad \bar{b} = \frac{b^*}{\alpha} + (\frac{1}{\alpha} - 1)\frac{\min^- + \max^+}{2}, \quad \bar{\epsilon} = \frac{\epsilon^*}{\alpha}, \quad \text{where } \alpha = l^*(\boldsymbol{w}^*) \leq 1.$$

We claim that the constructed solution above remains feasible to OP (10). Note that for data point with label 1, we have: (1) $\frac{\min^- + \max^+}{2} \geq r_i$ by assumption $r_i \leq \max^+ \leq \min^-$; (2) $\boldsymbol{x}_i \cdot \boldsymbol{w}^* + b^* \geq -r_i$ by the feasibility of $(\boldsymbol{w}^*, b^*, \epsilon^*)$. Therefore

$$\begin{aligned}
& \boldsymbol{x}_i \cdot \frac{\boldsymbol{w}^*}{\alpha} + \frac{b^*}{\alpha} \geq -\frac{r_i}{\alpha} \\
\implies \quad & \boldsymbol{x}_i \cdot \frac{\boldsymbol{w}^*}{\alpha} + \frac{b^*}{\alpha} + (\frac{1}{\alpha} - 1)\frac{\min^- + \max^+}{2} \geq -\frac{r_i}{\alpha} + (\frac{1}{\alpha} - 1)r_i \\
\iff \quad & \boldsymbol{x}_i \cdot \bar{\boldsymbol{w}} + \bar{b} \geq -r_i
\end{aligned}$$

This proves that the constructed solution is feasible for data points with label 1. Similar argument using the inequality $\frac{\min^- + \max^+}{2} \leq r_i$ for any negative label data point shows that it is also feasible for negative data points. It is easy to see that the solution quality is as good as the optimal solution ϵ^* since $\alpha \leq 1$. This proves the optimality of the constructed solution.

Finally, we consider the **Situation 2** where the instance is adversarial, i.e, $\min^- \geq 0 \geq \max^+$. In this case, r_i in the first constraint of System (8) is always non-positive whereas r_i in the second constraint is always non-negative. After basic algebraic manipulations, the SERM problem becomes the following optimization problem.[8]

$$\begin{array}{lll}
\text{find} & \boldsymbol{w}, b, \epsilon > 0 & \\
\text{subject to} & \boldsymbol{w} \cdot \boldsymbol{x}_i + b \geq (-r_i) \cdot l^*_{\boldsymbol{x}_i}(\boldsymbol{w}), & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } r_i \leq 0. \\
& -(\boldsymbol{w} \cdot \boldsymbol{x}_i + b) \geq (r_i + \epsilon) \cdot l^*_{\boldsymbol{x}_i}(\boldsymbol{w}), & \text{for points } (\boldsymbol{x}_i, y_i, r_i) \text{ with } r_i \geq 0.
\end{array} \tag{11}$$

This is again not a convex feasibility problem due to the non-convex term $(r_i + \epsilon) \cdot l^*_{\boldsymbol{x}_i}(\boldsymbol{w})$, however for any fixed $\epsilon > 0$ both constraints are convex. Moreover, if the system is feasible for some $\epsilon_0 > 0$ then it is feasible for any $0 < \epsilon \leq \epsilon_0$. Therefore, we can determine the feasibility of the (convex) system for any fixed ϵ and then binary search for the feasible ϵ. Therefore, the feasibility problem in System (8) can be solved in polynomial time. ■

Our next result shows that the positive claim in Theorem (14) 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 (14). 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.

8. The $l^*_{\boldsymbol{x}_i}$ function in the program should be viewed as depending on the data point while not just the feature $\boldsymbol{x}_i$. However, this will not affect the proof correctness.

Theorem 15 *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. *The problem is* essentially *adversarial (*$\min^- \geq \max^+$*) and the cost function is instance-wise and induced by norms.*

Proof We start with **Situation 1**, i.e., the preferences are arbitrary but the cost function is $c(\boldsymbol{z};\boldsymbol{x}) = \|\boldsymbol{x}-\boldsymbol{z}\|_2^2$. We will show later that the second situation can be reduced from the first. In the first situation, the feasibility problem is System (9) with l as the l_2 norm. Our reduction starts by reducing this system to the following optimization problem (OP)

$$\begin{array}{lll} \text{maximize} & ||\boldsymbol{w}||_2^2 & \\ \text{subject to} & \boldsymbol{x}_i \cdot \boldsymbol{w} + b \geq -r_i, & \text{for points } (\boldsymbol{x}_i, r_i) \text{ with label } +1. \\ & \boldsymbol{x}_i \cdot \boldsymbol{w} + b \leq -r_i - \epsilon, & \text{for points } (\boldsymbol{x}_i, r_i) \text{ with label } -1. \\ & ||\boldsymbol{w}||_2^2 \leq 1 & \end{array} \tag{12}$$

Formally, we claim that for any fixed ϵ, system (9) is feasible if and only if OP (12) has optimal objective value 1. The "if" direction is simple. That is, if OP (12) has optimal objective value 1, then the optimal solution $(\boldsymbol{w}^*, b^*)$ is automatically a feasible solution to System (9) because $||\boldsymbol{w}^*||_2 = 1$. For the "only if" direction, let $(\bar{\boldsymbol{w}}, \bar{b})$ be any feasible solution to System (9), then it is easy to verify $\boldsymbol{w}^* = \frac{\bar{\boldsymbol{w}}}{||\bar{\boldsymbol{w}}||_2}$ and $b^* = \frac{\bar{b}}{||\bar{\boldsymbol{w}}||_2}$ must also be feasible to System (9). Moreover, it is an optimal solution to OP (12) with objective value 1, as desired.

We now prove that determining whether the optimal objective value of OP (12) equals 1 or not is NP-complete. We reduce from the following well-known NP-complete problem called the *partition problem*:

Given d positive integers $c_1, \cdots, c_d$, decide whether there exists a subset $S \subset [d]$ such that $\sum_{i \in S} c_i = \sum_{i \notin S} c_i$

We now reduce the above partition problem to solving OP (12). Given any instance of partition problem, construct the following SERM instance.

The Constructed Hard SERM Instance for Situation 1: We will have $n = 2d + 3$ data points with feature vectors from $\mathbb{R}^d$. For convenience, we will use $\boldsymbol{e}_i$ to denote the basis vector in $\mathbb{R}^d$ whose entries are all 0 except that the i'th is 1. For each $i \in [d]$, there is a data point $(\boldsymbol{x}, y, r) = (2\sqrt{d} \cdot \boldsymbol{e}_i, 1, 4)$ as well as a data point $(\sqrt{d} \cdot \boldsymbol{e}_i, -1, 1-\epsilon)$. The remaining three data points are $(\boldsymbol{c}, 1, 2)$, data point $(2\boldsymbol{c}, -1, 2-\epsilon)$, and data point $(3\boldsymbol{c}, 1, 2)$.

We claim that OP (12) instantiated with the above constructed instance has an optimal objective value 1 if and only if the answer to the given partition problem is *Yes*. We first prove the "if" direction. If the partition problem is a Yes instance, then there exists an S such that $\sum_{i \in S} c_i - \sum_{i \notin S} c_i = 0$. We argue that the following construction is an optimal solution to OP (12) with optimal objective value 1:

$$b^* = -2, \ w_i = \frac{1}{\sqrt{d}} \ \forall i \in S, \ w_i = -\frac{1}{\sqrt{d}} \ \forall i \notin S.$$

Clearly, $||\boldsymbol{w}^*||_2^2 = 1$. We only need to prove feasibility of $(\boldsymbol{w}^*, b^*)$. For any label 1 point $(\boldsymbol{x}, r) = (2\sqrt{d}\cdot\boldsymbol{e}_i, 4)$, we have $\boldsymbol{x}\cdot\boldsymbol{w}^*+b^* = 2\sqrt{d}\boldsymbol{e}_i\cdot\boldsymbol{w}^*-2 = -4 \geq -r$, as desired. Similarly, for any label -1 point $(\boldsymbol{x}, r) = (\sqrt{d}\cdot\boldsymbol{e}_i, 1-\epsilon)$, we have $\boldsymbol{x}\cdot\boldsymbol{w}^*+b^* = \sqrt{d}\boldsymbol{e}_i\cdot\boldsymbol{w}^*-2 = -1 \leq -r-\epsilon$. The feasibility of point $(\boldsymbol{c}, 2)$ with label 1 is argued as follows: $\boldsymbol{x}\cdot\boldsymbol{w}^*+b^* = \boldsymbol{c}\cdot\boldsymbol{w}^*-2 = -r$. Feasibility of $(2\boldsymbol{c}, 2-\epsilon)$ and $(3\boldsymbol{c}, 2)$ are similarly verified.

We now prove the "only if" direction. In particular, we prove that that if OP (12) has some optimal solution $(\boldsymbol{w}^*, b)$ with $||\boldsymbol{w}^*||_2^2 = 1$, then the partition instance must be Yes.

Let us first examine the feasibility of OP (12).

1. By the constraints with respect to positive-label data points $(2\sqrt{d}\cdot\boldsymbol{e}_i, 4)$, we have $2\sqrt{d}\boldsymbol{e}_i\cdot\boldsymbol{w}+b \geq -4$ or equivalently $w_i\sqrt{d} \geq -\frac{b}{2}-2$.

2. By the constraints with respect to negative-label data points $(\sqrt{d}\cdot\boldsymbol{e}_i, 1-\epsilon)$, we have $\sqrt{d}\boldsymbol{e}_i\cdot\boldsymbol{w}+b \leq -1$ or equivalently $w_i\sqrt{d} \leq -b-1$.

3. By the constraints with respect to data point $(\boldsymbol{c}, 2)$ with label 1, we have $\boldsymbol{c}\cdot\boldsymbol{w}+b \geq -2$, or equivalently $-2-b \leq \boldsymbol{c}\cdot\boldsymbol{w}$.

4. By the constraints with respect to data point $(2\boldsymbol{c}, 2-\epsilon)$ with label -1, we have $2\boldsymbol{c}\cdot\boldsymbol{w}+b \leq -2$, or equivalently $-2-b \geq 2\boldsymbol{c}\cdot\boldsymbol{w}$.

5. By the constraints with respect to data point $(3\boldsymbol{c}, 2)$ with label 1, we have $3\boldsymbol{c}\cdot\boldsymbol{w}+b \geq -2$, or equivalently $-2-b \leq 3\boldsymbol{c}\cdot\boldsymbol{w}$.

Point 3–5 implies $2\boldsymbol{c}\cdot\boldsymbol{w} \leq -2-b \leq \min\{\boldsymbol{c}\cdot\boldsymbol{w}, 3\boldsymbol{c}\cdot\boldsymbol{w}\}$. This must imply $\boldsymbol{c}\cdot\boldsymbol{w} = 0$ as any non-zero $\boldsymbol{c}\cdot\boldsymbol{w}$ cannot satisfy $2\boldsymbol{c}\cdot\boldsymbol{w} \leq \min\{\boldsymbol{c}\cdot\boldsymbol{w}, 3\boldsymbol{c}\cdot\boldsymbol{w}\}$. As a consequence, the only feasible b value is $b = -2$. Plugging $b = -2$ into Point 1 and 2, we have

$$-\frac{1}{\sqrt{d}} \leq w_i \leq \frac{1}{\sqrt{d}}.$$

Since the optimal objective value is $1 = \sum_{i=1}^{d}(w_i^*)^2$, it is easy to see that this optimal objective is achieved only when each w_i^* equals either $-\frac{1}{\sqrt{d}}$ or $\frac{1}{\sqrt{d}}$. Now define $S = \{i : w_i^* = \frac{1}{\sqrt{d}}\}$ to be the set of i such that w_i^* is positive. It is easy to verify that S will be a solution to the partition problem, implying that it is a *Yes* instance. This proves the NP-hardness for **Situation 1** stated in the theorem.

Finally, we consider **Situation 2** which can be reduced from the first situation. In particular, the constructed hard instance above has reward preferences all being positive (in fact, drawn from only three possible values $\{1, 2, 4\}$), but do not satisfy the essentially adversarial condition. However, if we are allowed to use instant-wise cost functions, we can simply scale down the reward preference for point with label 1 but propositionally scale down its cost function so that the right-hand-side of the first constraint in System (9) remains the same. Concretely, we now modify our constructed instance above to be the follows.

The Constructed Hard SERM Instance for Situation 2: We still have $n = 2d+3$ data points with feature vectors from $\mathbb{R}^d$. For each $i \in [d]$, there is a data point $(\boldsymbol{x}, y, r) = (2\sqrt{d}\cdot\boldsymbol{e}_i, 1, 0.5)$ with cost function $c(\boldsymbol{z}; \boldsymbol{x}) = \frac{1}{8}||\boldsymbol{z}-\boldsymbol{x}||_2^2$ as well as a data point $(\sqrt{d}\cdot\boldsymbol{e}_i, -1, 1-\epsilon)$

with cost function $c(\boldsymbol{z};\boldsymbol{x}) = \|\boldsymbol{z}-\boldsymbol{x}\|_2^2$. The remaining three data points are: (1) data point $(\boldsymbol{c}, 1, 0.5)$ with cost function $c(\boldsymbol{z};\boldsymbol{x}) = \frac{1}{4}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2$; (2) data point $(2\boldsymbol{c}, -1, 2-\epsilon)$ with cost function $c(\boldsymbol{z};\boldsymbol{x}) = \|\boldsymbol{z}-\boldsymbol{x}\|_2^2$; (3) data point $(3\boldsymbol{c}, 1, 0.5)$ with cost function $c(\boldsymbol{z};\boldsymbol{x}) = \frac{1}{4}\|\boldsymbol{z}-\boldsymbol{x}\|_2^2$.

It is easy to verify that the above instance satisfy situation 1 in the theorem statement and is equivalent to the instance we constructed for the second situation and thus is also NP-hard. ■

Remark 16 *Theorem 11, Theorem 14 and Theorem 15 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 a statistical sense, but SERM is NP-hard when the cost function is instance-invariant and preferences are arbitrary.

5. The Power and Limits of Randomization

It is well-known that randomization over the classifiers does not contribute to a strictly smaller loss in standard classification. Interestingly, it turns out that randomization can be helpful in strategic classification, which is reported by Braverman and Garg (2020). However, their entire work was based on a simplified setting in the sense that they only considered one-dimensional feature space and homogeneous strategic preference. In this section, we study the power and limits of randomized linear classifiers in our generalized strategic setting. First, we define randomized classifiers as follows.

Definition 17 (Randomized Binary Classifiers) *A binary classifier H is called a randomized classifier over* any *hypothesis class $\mathcal{H}$ (not necessarily linear classifiers) if there exists a set of deterministic classifiers $\{h_1, \ldots, h_k\} \subseteq \mathcal{H}$, and a probability vector $\mathbf{p} = \{p_1, \ldots, p_k\}$ with $\sum_j p_j = 1$, such that for any input feature $\boldsymbol{x}$, $H(\boldsymbol{x}) = h_j(\boldsymbol{x})$ with probability p_j.*

The capital H notation distinguishes randomized classifiers from deterministic ones and can be viewed as a random variable. Since we always focus on binary classification, we simply say randomized classifiers henceforth. Note that randomized classifiers should not be confused with classification methods like logistic regression or ensemble methods (e.g., random forest), which ultimately still output a deterministic label. The following proposition, whose proof is deferred to Appendix B.1, shows that randomization can help in strategic linear classification. We remark that an interesting question is whether each data point can efficiently identify its best response against a general randomized (even linear) classifier in polynomial time. Due to the combinatorial nature of different possible responses, this is a non-trivial open question. However, its answer shall not affect our following result since our constructed classifier only randomizes over two linear classifiers, to which the best response is straightforward to compute.

Proposition 18 *There are strategic classification instances that are perfectly separable by a* randomized *linear classifier but not by any deterministic linear classifier. This claim is valid even when all data points have the same reward value r.*

We remark that the constructed example for Proposition 18 is in the 2-dimension Euclidean space. If the same preference r is not required, there exist simpler examples in 1-d (see Appendix B.2), though it is interesting to figure out whether there is a 1-d example satisfying the full statement of Proposition 4.

The advantages of randomized classifiers over deterministic classifiers fundamentally come from the strategic data points' tradeoff between two factors: (a) manipulation cost; (b) the gain from classification outcomes. Interestingly, *if any of these two factors is absent, randomization will not be beneficial.* First, if there is no gain from the classification outcome (i.e., $r = 0$ for any data point), the data points naturally have no benefit to move and thus will stay put. This is precisely the classic classification setup. Second, our following result shows that if data points have no moving cost, randomization does not help either in the separable case. Formally, we consider a special type of strategic classification where each data point is constrained to move within a designated region which can be *arbitrary* (even can be unconnected) and has no moving cost, i.e., $c(\boldsymbol{z}; \boldsymbol{x}) = 0$ for any feasible move. We term this the *zero-manipulation-cost* strategic classification. Note that the adversarial or robust classification setup falls into this case (Cullina et al., 2018; Awasthi et al., 2019). The following proposition shows that under the zero-manipulation-cost case, randomization over classifiers does not help that much.

Proposition 19 *Consider* zero-manipulation-cost *strategic classification. For any hypothesis class $\mathcal{H}$, if there is a randomized classifier over $\mathcal{H}$ that perfectly separates the positive points from the negative (i.e., achieving 0 loss), then there must exist a deterministic classifier in $\mathcal{H}$ that achieves so as well.*

Proof Consider any strategic classification instance $\textsc{StraC}\langle \mathcal{H}, R, c\rangle$ where the cost function is a zero-manipulation cost function, i.e., each data point has a designated feasible region to move around with cost 0. Let $\mathcal{D}$ be any distribution over data points and suppose randomized classifier H, defined by $H(\boldsymbol{x}) = h_j(\boldsymbol{x}) \in \mathcal{H}$ with probability p_j for any $j = 1, \cdots, k$, achieves perfectly separates the positive points from the negative. Assume a classifier h that randomizes over k hyperplanes $h_1, \cdots, h_k$ and achieves zero loss. We claim that any deterministic classifier h_j will also achieve zero loss and thus the randomization is not needed.

Consider any data point $(\boldsymbol{x}, y, r) \sim \mathcal{D}$. Given H, let $\Delta_c(\boldsymbol{x}, r; H)$ denote the optimal manipulated feature that the data point will be moving to. By assumption of zero loss, we know that for any $j \in [k]$, h_j will classify $\Delta_c(\boldsymbol{x}, r; H)$ correctly, meaning $h_j(\Delta_c(\boldsymbol{x}, r; H))$ equals the true label of y. This however did not prove our result yet since after deploying the deterministic classifier h_j, the optimal manipulated feature should have been $\Delta_c(\boldsymbol{x}, r; h_j)$ while not the $\Delta_c(\boldsymbol{x}, r; H)$.

We now prove $h_j(\Delta_c(\boldsymbol{x}, r; h_j)) = h_j(\Delta_c(\boldsymbol{x}, r; H)) = y$. Suppose, for the sake of contradiction, that $h_j(\Delta_c(\boldsymbol{x}, r; h_j)) \neq y$. Since both $\Delta_c(\boldsymbol{x}, r; h_j), \Delta_c(\boldsymbol{x}, r; H)$ are feasible moves for the data point, the reason that the data point now strictly prefers $\Delta_c(\boldsymbol{x}, r; h_j)$ over $\Delta_c(\boldsymbol{x}, r; H)$ must because $h_j(\Delta_c(\boldsymbol{x}, r; h_j)) = \text{sgn}(r)$ and $h_j(\Delta_c(\boldsymbol{x}, r; H)) = -\text{sgn}(r)$, which

equals y. Since H achieves perfect classification, all h_j's must classify $\Delta_c(\boldsymbol{x}, r; H)$ as the same label $-\text{sgn}(r)$. That means, the move of the data point's feature from $\boldsymbol{x}$ to $\Delta_c(\boldsymbol{x}, r; H)$ leads to the data point being classified as the label $-\text{sgn}(r)$ that it does not prefer. This however conflicts with the assumption that $\Delta_c(\boldsymbol{x}, r; H)$ is an optimal manipulated feature since moving to $\Delta_c(\boldsymbol{x}, r; h_j)$ instead would at least lead to the data point being classified as $\text{sgn}(r)$ by h_j. Therefore, we must have $h_j(\Delta_c(\boldsymbol{x}, r; h_j)) = h_j(\Delta_c(\boldsymbol{x}, r; H)) = y$ for any j, concluding the proof.

■

In adversarial classification, it is usually assumed that each data point can move within a δ-ball around it with no cost. An immediate corollary of Theorem 19 is the following.

Corollary 20 *Randomization does not help in any perfectly separable adversarial classification problems.*

We remark that perfectly separable assumption in Theorem 19 is necessary. That is, even for zero-manipulation-cost strategic classification problem, there are non-separable examples where randomized linear classifier can achieve strictly larger accuracy than any deterministic linear classifier, as shown in the following proposition with proof deferred to Appendix B.3.

Proposition 21 *There exists zero-manipulation-cost* non-separable *strategic classification instances in $\mathbb{R}^2$ where the minimum risk of any deterministic linear classifier is strictly larger than some randomized linear classifier.*

6. 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 seminorms. A future direction is to generalize the theory to cost function induced by asymmetric seminorms 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. It is still interesting to understand the behavior of the optimal classifier in the partial information strategic setting.

We also find that the randomization over linear classifiers can strictly enhance the accuracy compared to deterministic ones. This observation is interesting because simple randomization over a set of classifiers is not helpful in standard classification problems in general. It might suggest that the difficulties of learning under standard and strategic settings differ by nature. Another interesting follow-up question is how we can efficiently compute the optimal randomized linear classifier for strategic classification. It is challenging because it is unclear how to compute the best response of a data point against such a randomized classifier. We identify it as an intriguing future direction.

Acknowledgement

We would like to thank the editor and the anonymous reviewers for their insightful comments that helped greatly improved the paper. 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.

Appendix A. Omitted Proofs from Section

A.1 Proof of Proposition 6

Proof The adversarial VC-dimension defined in (Cullina et al., 2018) relies on an auxiliary definition of *corrupted classifier* $\tilde{h} = \kappa_R(h)$ of any classifier h for the standard non-adversarial setting such that $\tilde{h}(\boldsymbol{x}) = h(\boldsymbol{x})$ if all the points in $N(\boldsymbol{x})$ have the same label as $\boldsymbol{x}$ and otherwise, $\tilde{h}(\boldsymbol{x}) = \perp$. Recall that $N(\boldsymbol{x}) = \{\boldsymbol{z} \in \mathcal{X} : (\boldsymbol{z}; \boldsymbol{x}) \in \mathcal{B}\} = \{\boldsymbol{z} \in \mathcal{X} : c(\boldsymbol{z}; \boldsymbol{x}) \leq r\}$ denotes the set of all possible adversarial features $\boldsymbol{x}$ can move to. Given this auxiliary definition, the adversarial VC-dimension is defined as $\mathrm{AVC}(\mathcal{H}, \mathcal{B}) = \sup\{n : \sigma_n(\mathcal{F}, \mathcal{B}) = 2^n\}$, where

$$\sigma_n(\mathcal{F}, \mathcal{B}) = \max_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}^n \times \{+1, -1\}^n} |\{(f(\boldsymbol{x}_1, y_1; h), \ldots, f(\boldsymbol{x}_n, y_n; h)) : h \in \mathcal{H}\}| \tag{13}$$

is the shattering coefficient, and $f(\boldsymbol{x}_i, y_i) = \mathbb{I}(\tilde{h}(\boldsymbol{x}_i) \neq y_i)$ is the *loss function* of the corrupted classifier $\tilde{h} = \kappa_R(h)$.

Since $\mathcal{B}$ and c are r-consistent, we have $\mathcal{B} = \{(\boldsymbol{z}; \boldsymbol{x}) : c(\boldsymbol{z}; \boldsymbol{x}) \leq r\}$. Let $R = \{+r, -r\}$. We now prove the proposition by arguing

$$\sup\{n \in \mathbb{N} : \sigma_n(\mathcal{H}, R, c) = 2^n\} = \sup\{n : \sigma_n(\mathcal{F}, \mathcal{B}) = 2^n\}. \tag{14}$$

1. If $\sup\{n \in \mathbb{N} : \sigma_n(\mathcal{H}, R, c) = 2^n\} = n$, by Definition 1, there exists $(\boldsymbol{x}'_i, r'_i) \in \mathcal{X} \times R, i = 1, \cdots, n$ such that $|\{(h(\Delta_c(\boldsymbol{x}'_1, r'_1; h)), \cdots, h(\Delta_c(\boldsymbol{x}'_n, r'_n; h)) : h \in \mathcal{H}\}| = 2^n$. Since Definition 1 does not rely on the true labels of $\boldsymbol{x}'_i$, we may let the true labels of $\boldsymbol{x}'_i$ be $y'_i = -r'_i/r$ for any i. In this case, each $\boldsymbol{x}'_i$'s strategic preference is against its true label, which corresponds to the loss function f in Equation (13) for the adversarial

setting. Therefore, taking $(\boldsymbol{x}_i, y_i) = (\boldsymbol{x}'_i, y'_i)$ in Equation (13) gives $\sigma_n(\mathcal{F}, \mathcal{B}) = 2^n$. This implies $\sup\{n \in \mathbb{N} : \sigma_n(\mathcal{H}, R, c) = 2^n\} \leq \sup\{n : \sigma_n(\mathcal{F}, \mathcal{B}) = 2^n\}$.

2. Conversely, if $\sup\{n : \sigma_n(\mathcal{F}, \mathcal{B}) = 2^n\} = n$, from Equation (13), there exists $(\boldsymbol{x}_i, y_i) \in \mathcal{X} \times R, i = 1, \cdots, n$ such that $|\{(f(\boldsymbol{x}_1, y_1), \ldots, f(\boldsymbol{x}_n, y_n)) : f \in \mathcal{F}\}| = 2^n$. Similarly, taking $r_i = -ry_i \in R$ gives $\sigma_n(\mathcal{H}, R, c) = 2^n$, which implies $\sup\{n \in \mathbb{N} : \sigma_n(\mathcal{H}, R, c) = 2^n\} \geq \sup\{n : \sigma_n(\mathcal{F}, \mathcal{B}) = 2^n\}$.

Therefore, we have $\text{AVC}(\mathcal{H}, \mathcal{B}) = \text{SVC}(\mathcal{H}, \{+r, -r\}, c)$ for any r-consistent pair $(\mathcal{B}, c)$. ■

A.2 Proof of Corollary 7

Proof Since $\{+r, -r\} \subseteq \mathcal{B}$, we have $\sigma_n(\mathcal{H}, R, c) \geq \sigma_n(\mathcal{H}, \{+r, -r\}, c)$ by Definition 1. As a result, $\text{SVC}(\mathcal{H}, R, c) \geq \text{SVC}(\mathcal{H}, \{+r, -r\}, c)$. Then by applying Proposition 6 we have $\text{SVC}(\mathcal{H}, R, c) \geq \text{SVC}(\mathcal{H}, \{+r, -r\}, c) = \text{AVC}(\mathcal{H}, \mathcal{B})$. ■

A.3 Proof of Proposition 8

Given any positive integer n, let $[n]$ denotes $\{1, 2, \cdots, n\}$, and $\mathcal{S}$ be the power set of $[n]$, i.e., the set that contains all the subsets of $[n]$. Let $\mathcal{X} = [n] \cup \mathcal{S}$ be the sample space of size $n + 2^n$, and the hypothesis class $\mathcal{H}$ is the set of all the point classifiers with points from $\mathcal{S}$, i.e., $\mathcal{H} = \{h_s : s \in \mathcal{S}\}$, where point classifier h_s only classifies the point $s \in \mathcal{S}$ as positive. The cost function $c(z; x)$ is a metric defined as follows. Since metric is symmetric, i.e., $c(z; x) = c(x; z)$, we will use the notation $c(x, z)$ instead throughout this proof.

$$c(x, z) = \begin{cases} x, & \text{if} \quad x \in [n], z \in \mathcal{S}, x \in z \\ x + 1, & \text{if} \quad x \in [n], z \in \mathcal{S}, x \notin z \\ c(z, x), & \text{if} \quad x \in \mathcal{S}, z \in [n] \\ x + z, & \text{if} \quad x, z \in [n], x \neq z \\ 1, & \text{if} \quad x, z \in \mathcal{S}, x \neq z \\ 0, & \text{if} \quad x = z, \end{cases} \tag{15}$$

and R is set to be $[-n, -1] \cup [1, n]$.

First, we verify that $c(\cdot, \cdot)$ is indeed a metric. Given definition (15), it is easy to see that $c(x, z) = 0$ iff $x = z$, and $c(x, z) = c(z, x), \forall x, z \in \mathcal{X}$. It remains to check the triangle inequality, i.e., for any $x, y, z \in \mathcal{X}$, $c(x, y) + c(y, z) \geq c(x, z)$. Consider the case when x, y, z are different elements in $\mathcal{X}$. By enumerating all the possibility that whether each x, y, z is in $[n]$ or $\mathcal{S}$, it suffices to discuss the following $8 (= 2^3)$ cases:

1. if $x, y, z \in [n]$, $c(x, y) + c(y, z) = x + y + y + z > x + z = c(x, z)$.
2. if $x, y, z \in \mathcal{S}$, $c(x, y) + c(y, z) = 2 > 1 = c(x, z)$.
3. if $x, z \in [n], y \in \mathcal{S}$, then $c(x, y) \geq x, c(y, z) \geq z$. $\implies c(x, y) + c(y, z) \geq x + z = c(x, z)$.

4. if $x, y \in [n], z \in \mathcal{S}$, we need to show that $c(x,y) \geq c(x,z) - c(y,z)$. Conditioned on the relationship between x, y and set z, the maximum value of $c(x,z) - c(y,z)$ is $x - y + 1$ when $y \in z, x \notin z$. Therefore, $c(x,y) = x + y \geq x - y + 1 \geq c(x,z) - c(y,z)$.

5. if $x, z \in \mathcal{S}, y \in [n]$, then $c(x,y) + c(y,z) \geq y + y > 1 \geq c(x,z)$.

6. if $x, y \in \mathcal{S}, z \in [n]$, then the maximum value for $c(x,z) - c(y,z)$ is $z + 1 - z = 1$ when $z \notin x, z \in y$. Therefore, $c(x,y) \geq 1 \geq c(x,z) - c(y,z)$.

7. if $x \in \mathcal{S}, y, z \in [n]$, it is equivalent to case 4.

8. if $y, z \in \mathcal{S}, x \in [n]$, it is equivalent to case 6.

Next, we show $\text{VC}(\mathcal{H}) = 1$, $\text{AVC}(\mathcal{H}, \mathcal{B}_c(r)) = 1$, and $\text{SVC}(\mathcal{H}, R, c) \geq n$. Observe that $\text{VC}(\mathcal{H}) = 1$ follows easily since no point classifier $h_s \in \mathcal{H}$ can generate the label pattern $(+1, +1)$ for any pair of distinct data points.

Next we prove $\text{AVC}(\mathcal{H}, \mathcal{B}_c(r)) = 1$. We first show $\text{AVC}(\mathcal{H}, \mathcal{B}_c(r)) \leq 1$ by arguing that under binary nearness relation $\mathcal{B}_c(r) = \{(z; x) : c(z,x) \leq r\}$ with $r \geq 1$, any two elements x_1, x_2 in $\mathcal{X}$ cannot be shattered by $\mathcal{H}$.

1. If at least one of r_1, r_2 equals $-r$, e.g., $r_1 = -r$, we show that x_1 can never be classified as $+1$ by contradiction. Suppose some $h_s \in \mathcal{H}$ classifies $(x_1, -r)$ as $+1$: if $x_1 \neq s$, since $r_1 = -r < 0$, x_1 will not manipulate its feature and be classified as -1; if $x_1 = s$, x_1 can move to any $z \in \mathcal{S}$ with cost $1 \leq r$, and will also be classified as -1. Therefore, (x_1, x_2) can not be shattered.

2. If $r_1 = r_2 = r$, consider the following two cases:

   (a) If at least one of x_1, x_2 belongs to $\mathcal{S}$, e.g., $x_1 \in \mathcal{S}$, then x_1 can move to any $s \in \mathcal{S}$ as $c(x_1, s) = 1 \leq r$ for any $s \in \mathcal{S}$. Therefore x_1 can never be classified as -1 by any point classifier in $\mathcal{H}$.

   (b) if $x_1, x_2 \in [n]$, we may w.l.o.g. assume $x_1 < x_2$, i.e., $x_1 + 1 \leq x_2$. Observe that when $r < x_1$, any $h_s \in \mathcal{H}$ will classify x_1 as -1 because $c(x_1, s) = x_1 > r, \forall s \in \mathcal{S}$; when $r \geq x_1 + 1$, any $h_s \in \mathcal{H}$ will classify x_1 as $+1$ because $c(x_1, s) = x_1 + 1 \leq r, \forall s \in \mathcal{S}$. Therefore, in order to shatter (x_1, x_2), r must lie in the interval $[x_1, x_1 + 1) \cap [x_2, x_2 + 1) = \emptyset$, which draws the contradiction.

To see that $\text{AVC}(\mathcal{H}, \mathcal{B}_c(r)) \geq 1$, for any $x \in [n]$ with $r > 0$, it can be classified as either $+1$ or -1 as long as $r \in [x, x + 1)$. We thus have $\text{AVC}(\mathcal{H}, c) = 1$.

Finally, we prove that $\text{SVC}(\mathcal{H}, R, c) = n$. Consider the subset $[n] \subset \mathcal{X}$ of size n, with each element i equipped with a strategic preference $r_i = i$. For any label pattern $\mathcal{L} \in \{+1, -1\}^n$, let $s_{\mathcal{L}} = \{i \in [n] : \mathcal{L}_i = +1\}$ be an element in $\mathcal{S}$. We claim that $h_{s_{\mathcal{L}}} \in \mathcal{H}$ gives exactly the label pattern $\mathcal{L}$ on $[n]$. To see this, consider any $i \in [n]$:

1. If $i \in s_{\mathcal{L}}$, i will move to $s_{\mathcal{L}} \in \mathcal{S}$ and be classified as $+1$, as the cost $c(i, s_{\mathcal{L}}) = i \leq r_i = i$.

2. If $i \notin s_{\mathcal{L}}$, i will not move to $s_{\mathcal{L}} \in \mathcal{S}$ and be classified as -1, as the cost $c(i, s_{\mathcal{L}}) = i + 1 > r_i = i$.

Therefore, any label pattern $\mathcal{L} \in \{+1, -1\}^n$ can be achieved by some $h_{s_\mathcal{L}} \in \mathcal{H}$. This implies $\text{SVC}(\mathcal{H}, R, c) \geq n$. On the other hand, it's easy to see $\mathcal{H}$ cannot shatter $n+1$ points, because any subset of size $n+1$ must contain an element s_0 in $\mathcal{S}$, and no matter what strategic preference s_0 has, it will either be classified as $+1$ by all $h_s \in \mathcal{H}$, or be classified as $+1$ by only one classifier in $\mathcal{H}$, i.e., h_{s_0}. Either case renders the shattering for $n+1$ points impossible.

A.4 Proof of Proposition 9

Proof Define the adversarial region for an adversary $(\boldsymbol{x}, r)$ as $N(\boldsymbol{x}, r) = \{\boldsymbol{z} \in \mathcal{X} : c_2(\boldsymbol{z}) \leq c_1(\boldsymbol{x}) + |r|\} \supseteq \{\boldsymbol{x}\}$. Since staying with the same feature has no cost, this implies $c(\boldsymbol{x}; \boldsymbol{x}) = 0$ or equivalently $c_2(\boldsymbol{x}) \leq c_1(\boldsymbol{x})$ for any $\boldsymbol{x} \in \mathcal{X}$. Then, the best response function for $(\boldsymbol{x}, r)$ can be characterized by

1. if $h(\boldsymbol{x}) = \text{sgn}(r)$, then $h(\Delta(\boldsymbol{x}, r; h)) = \text{sgn}(r)$;
2. if $h(\boldsymbol{x}) = -\text{sgn}(r)$, then

$$h(\Delta(\boldsymbol{x}, r; h)) = \begin{cases} -\text{sgn}(r), & \forall \boldsymbol{z} \in N(\boldsymbol{x}, r) : h(\boldsymbol{z}) = -\text{sgn}(r) \\ \text{sgn}(r), & \exists \boldsymbol{z} \in N(\boldsymbol{x}, r) : h(\boldsymbol{z}) = \text{sgn}(r) \end{cases} \tag{16}$$

Suppose there are three points $\{(x_i, r_i)\}_{i=1}^3$ that can be shattered by $\mathcal{H}$. Let $b_i = c_1(\boldsymbol{x}_i) + r_i$ and w.l.o.g. let $b_1 \leq b_2 \leq b_3$. From $b_1 \leq b_2 \leq b_3$, we have $N(\boldsymbol{x}_1, r_1) \subseteq N(\boldsymbol{x}_2, r_2) \subseteq N(\boldsymbol{x}_3, r_3)$.

By Pigeonhole principle, there must exists two elements in $\{r_1, r_2, r_3\}$ which have the same sign. Suppose these two elements are r_1, r_2 and consider the following two cases:

1. $r_1 > 0, r_2 > 0$. From Equation 16, for any $h \in \mathcal{H}$, $h(\Delta(\boldsymbol{x}_2, r_2; h)) = -1$ means $h(\boldsymbol{z}) = -1, \forall \boldsymbol{z} \in N(\boldsymbol{x}_2, r_2)$. Note that $N(\boldsymbol{x}_1, r_1) \subseteq N(\boldsymbol{x}_2, r_2)$, we also have $h(\boldsymbol{z}) = -1, \forall \boldsymbol{z} \in N(\boldsymbol{x}_1, r_1)$. As a result, $h(\Delta(\boldsymbol{x}_1, r_1; h)) = -1$, meaning the sign pattern $\{+, -\}$ cannot be achieved by any $h \in \mathcal{H}$ for $\{(\boldsymbol{x}_1, r_1), (\boldsymbol{x}_2, r_2)\}$.
2. $r_1 < 0, r_2 < 0$. From Equation 16, for any $h \in \mathcal{H}$, $h(\Delta(\boldsymbol{x}_2, r_2; h)) = 1$ means $h(\boldsymbol{z}) = 1, \forall \boldsymbol{z} \in N(\boldsymbol{x}_2, r_2)$. Similarly, from $N(\boldsymbol{x}_1, r_1) \subseteq N(\boldsymbol{x}_2, r_2)$ we conclude $h(\boldsymbol{z}) = 1, \forall \boldsymbol{z} \in N(\boldsymbol{x}_1, r_1)$ and $h(\Delta(\boldsymbol{x}_1, r_1; h)) = 1$, meaning the sign pattern $\{-, +\}$ cannot be achieved by any $h \in \mathcal{H}$ for $\{(\boldsymbol{x}_1, r_1), (\boldsymbol{x}_2, r_2)\}$.

Therefore, $\{(x_i, r_i)\}_{i=1}^3$ cannot be shattered by $\mathcal{H}$, which implies $\text{SVC}(\mathcal{H}, R, c) \leq 2$. ■

Appendix B. Omitted Proofs in Section 5

B.1 Proof of Proposition 18

Consider an example with data points as depicted in Figure 3. Both plots show exactly the same strategic classification instance. Here, all solid points have positive labels, whereas

all other points have negative labels. Three data points are of interest in our analysis, i.e., point A, B, C, and they are on the same line. The left plot draws the optimal deterministic classifier, and the right plot draws a randomized classifier that picks hyperline h_1, h_2 uniformly at random. The cost function here is the Euclidean distance $c(\boldsymbol{z}; \boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{z}\|_2$. Let $r = \sqrt{2}$ for all the data points.

Under the squared Euclidean distance cost function and the condition that $r = \sqrt{2}$ is the same for all data points, any deterministic linear classifier is "strategically" equivalent to another linear classifier in the following sense: the classification outcome for linear classifier h in the strategic setup is identical to the classification outcome for linear classifier h' in the non-strategic setup where h' is obtained by shifting h towards the negative direction by $r = \sqrt{2}$. It is easy to see that no linear classifier can strictly separate the positive points from the negatives in the truthful setting, and thus this impossibility also holds in the strategic setting. As a result, the best deterministic linear classifier makes at least one mistake. One such optimal deterministic linear classifier is the hyperplane h as depicted in the left plot of Figure 3 which is parallel to line ABC but just above line ABC by 1.5. As a result, to move to the positive side of the classifier, point A, B, C need to suffer a cost $1.5 > \sqrt{2}$ which does not balance the benefit they gain. Therefore, all data points will remain truthful, and the classifier makes one mistake at data point A.

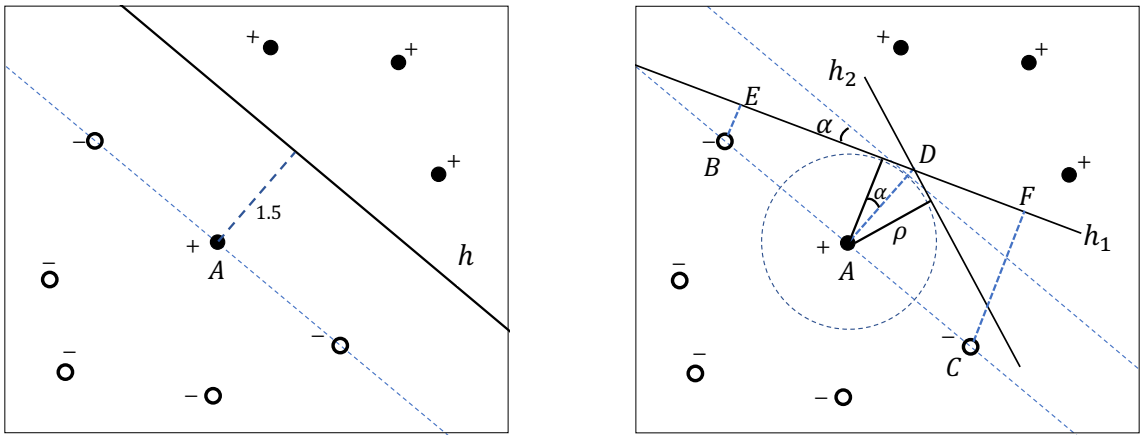


Figure 3: A strategic classification instance where randomized linear classifier beats any deterministic classifier. Cost function $c(\boldsymbol{z}; \boldsymbol{x}) = \|\boldsymbol{x} - \boldsymbol{z}\|_2$ is the Euclidean distance. Left: an optimal deterministic classifier. Right: a randomized classifier with perfect precision which picks hyperplane h_1, h_2 uniformly at random.

Next, we show that the optimal randomized classifier makes no mistakes and thus is strictly better. The randomized classifier is depicted in the right plot of Figure 3 with carefully chosen parameters r, α and it randomizes over h_1, h_2 uniformly at random. The geometry of the constructed randomized classifier is as follows. Data points A, B, C lie on the same line, and the length of segment AB and AC both equal d. The angle between line ABC and line h_1 is α and h_1 is tangent to the circle centered at point A with radius l. Note that these two conditions uniquely determine the position of the line h_1. The parameters d, α, ρ will be determined later. Similarly, line h_2 also rotates from line ABC with angle α and is tangent to the same circle. h_1 and h_2 intersect at point D. Note that D is outside

the circle. The projection of point B to line h_1 is E whereas the projection of point C to h_1 is F.

We start with some geometric calculation. First, the angle between ABC and h_1 is α. Therefore, their normal vectors must also have angle α, which is precisely the angle between AD and the normal vector of h_1, as depicted in the left plot. As a consequence, the length $|AD|$ equals $\frac{\rho}{\cos\alpha}$. Since $|AB| = |AC|$, we know that $|BE|+|CF| = 2\rho$. Moreover, $|CF| - |BE| = 2d \cdot \sin\alpha$. This yields $|BE| = \rho - d\sin\alpha$ and $|CF| = \rho + d\sin\alpha$. For our construction, we will select parameters to satisfy the following conditions

$$\begin{aligned} |AD| &= \frac{\rho}{\cos\alpha} < \sqrt{2} \\ |BE| &= \rho - d\sin\alpha > \sqrt{2}/2 \\ |CF| &= \rho + d\sin\alpha > \sqrt{2}. \\ d &> l \end{aligned} \tag{17}$$

There are many ways to pick the parameters to satisfy Equation (17). For example, it can be verified that $\rho = 1.38, \alpha = 0.05\pi$ and $d = 2.23$ will satisfy all these constraints.

We now claim that the randomized classifier with any parameters satisfying the above constraints will make a perfect prediction. In particular, point A has the incentive to manipulate its feature to point D (or slightly beyond) because the point suffers a cost less than $\sqrt{2}$ by the first constraint in Equation (17), but now is able to make both classifiers to predict it with a positive label, increasing the prediction utility by $\sqrt{2}$. Moreover, point B does not have any incentive to manipulate its feature. This is due to the following reason. To get one of the classifiers to classify B with label 1, the manipulation cost is at least $|BE|$, which is strictly greater than $\sqrt{2}/2$ whereas the expected utility from prediction is $\sqrt{2}/2$ since only half of the time the randomized classifier will classify B as 1. On the other hand, to get the randomized classifier to always classify B as positive, the manipulation cost of B is at least the distance of B from h_2, which equals $|CF| > \sqrt{2}$ by symmetry, whereas the benefit from classification outcome is only $\sqrt{2}$. As a consequence, point B does not have any incentive to manipulate its feature. Similarly, point C will not manipulate its feature as well. Overall, the randomized classifier makes a perfect prediction due to the manipulation of point A.

B.2 An Additional 1-d Example for Proposition 18

Now we present an additional example in 1-d without the requirement of the same r. We have four data points $A = -1.0, B = -0.8, C = 0.8, D = 1.0$ as depicted in Figure 4. Let $r = 0$ for B, C, $r = 1.2$ for A, and $r = -1.2$ for D. The cost function here is the Euclidean distance $c(z; x) = |x - z|$. Both plots show exactly the same strategic classification instance, and all solid points have positive labels, whereas all other points have negative labels. The left plot draws the optimal deterministic classifier $h = 2\mathbb{I}[x \geq 0] - 1$, and the right plot draws a randomized classifier that picks hyperline $h_1 = 2\mathbb{I}[x \geq -0.3] - 1, h_2 = 2\mathbb{I}[x \geq 0.3] - 1$ uniformly at random.

First of all, we argue that any deterministic classifier cannot perfectly separate $\{A, B, C, D\}$. Otherwise, there exists an h that makes no mistake, which implies that it must have the form $h = 2\mathbb{I}[x \geq \theta] - 1, \theta \in (-0.8, 0.8]$, as it can separate B and C perfectly. However, when

$\theta \in (-0.8, 0.8]$, the distance between h's decision boundary and one of A and D is at most $1.0 < r = 1.2$. Therefore, at least one point in $\{A, D\}$ wants to move across the decision boundary of h and the classification result cannot be perfect.

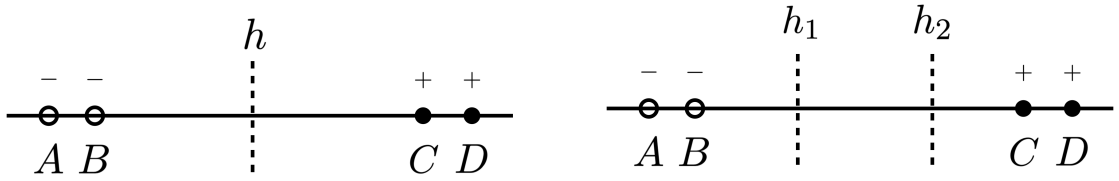


Figure 4: A strategic classification instance where randomized linear classifier beats any deterministic classifier. Cost function $c(z; x) = |x - z|$ is the Euclidean distance. Left: an optimal deterministic classifier. Right: a randomized classifier with perfect precision which picks hyperplane h_1, h_2 uniformly at random.

Next, we show that the randomized classifier $\{(h_1, 0.5), (h_2, 0.5)\}$ shown in the right panel of Figure 4 makes no mistakes and thus is strictly better. First, it is obvious that $\{(h_1, 0.5), (h_2, 0.5)\}$ outputs the true labels for data points B, C who do not have any incentive to deviate. For points A, D, their situations are symmetric when facing the classifier so we only need to show that A does not have enough incentive to manipulate its feature without loss of generality. In order to gain a positive utility, A needs to move across at least one decision boundary of the randomized classifier. However, since the distance between 0.3 and $A = -1.0$ exceeds 1.2, A can only move across -0.3. When A move across -0.3, it yields an expected utility of $0.6 = 0.5 \times 1.2$ which is less than the minimum moving cost $0.7 = -0.3 - (-1.0)$. As a result, A will stay put and accept the true classification result.

B.3 Proof of Proposition 21

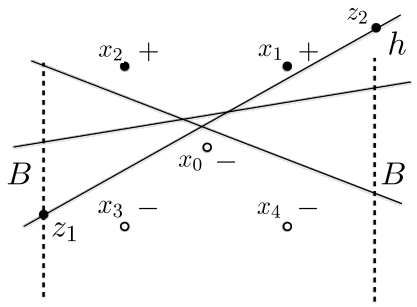
The problem instance is constructed as follows. There are five labeled data points on $\mathbb{R}^2$, defined as $(\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4) = ((0,0), (1,1), (-1,1), (-1,-1), (1,-1)), (y_0, y_1, y_2, y_3, y_4) = (-1, +1, +1, -1, -1)$. $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4$ are non-strategic while $\boldsymbol{x}_0$ prefers to be labeled as $+1$ and can move within the region $B = \{(x, y) : x \in \{-2, 2\}, y \in [-2, 1]\}$, as shown in Figure 5.

First, we prove that any $h \in \mathcal{H}$ that perfectly separate $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^4$ must incorrectly classify x_0, so that the minimum empirical risk on $\mathcal{H}$ is at least $\frac{1}{5}$. Suppose $h = 2\mathbb{I}(ax + by + c \geq 0) - 1$ is a linear classifier that perfectly separate $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^4$. Since $\boldsymbol{x}_1, \boldsymbol{x}_4$ and $\boldsymbol{x}_2, \boldsymbol{x}_3$ both have opposite labels, $ax + by + c = 0$ must intersect with both segments $[\boldsymbol{x}_1, \boldsymbol{x}_4)$ and $[\boldsymbol{x}_2, \boldsymbol{x}_3)$ (with a slight abuse of notation, we use the notation $[\boldsymbol{x}_i, \boldsymbol{x}_j)$ to represent the segment formed by endpoints x_i, x_j with x_j excluded). Assume the two intersection points are $(1, p)$ and $(-1, q)$, where $p, q \in (-1, 1]$. Thus, h can be written as $2\mathbb{I}((p-q)x - 2y + p + q \geq 0) - 1$ and h intersects the two lines $x = -2, x = 2$ at $\boldsymbol{z}_1 = (-2, \frac{-p+3q}{2})$ and $\boldsymbol{z}_2 = (2, \frac{3p-q}{2})$. We claim that at least one of $\boldsymbol{z}_1, \boldsymbol{z}_2$ lies in B, because:

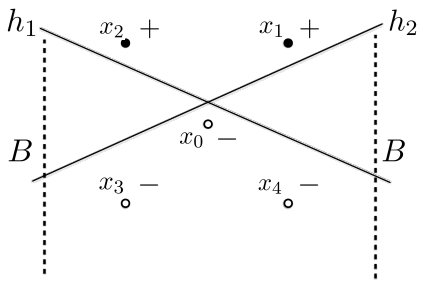
1. $p, q \in (-1, 1]$, we have $\frac{-p+3q}{2} > -2$, $\frac{3p-q}{2} > -2$.
2. note that $\frac{-p+3q}{2} + \frac{3p-q}{2} = p + q \leq 2$, at least one of $\frac{-p+3q}{2}$, $\frac{3p-q}{2}$ is not greater than 1.

Thus, at least one of $\frac{-p+3q}{2}, \frac{3p-q}{2}$ falls in $(-2, 1]$, meaning at least one of $\boldsymbol{z}_1, \boldsymbol{z}_2$ lies in B. Suppose $\boldsymbol{z}_1 \in B$, as shown in Figure 5a. Consequently, $\boldsymbol{x}_0$ can always move to $\boldsymbol{z}_1$ and be classified as $+1$ against its true label. Therefore, any $h \in \mathcal{H}$ makes at least one mistake on the data set $\{(\boldsymbol{x}_i, y_i)\}_{i=0}^4$, and we conclude that the minimum empirical risk on $\mathcal{H}$ is at least $\frac{1}{5}$.

Next, we construct a randomized classifier $H \in \tilde{\mathcal{H}}$ and show that H attains an empirical risk smaller than $\frac{1}{5}$. Let $h_1 = 2\mathbb{I}(4x+7y-2 \geq 0)-1, h_2 = 2\mathbb{I}(-4x+7y-2 \geq 0)-1$, and $H = \{(h_1, 0.5), (h_2, 0.5)\}$, as shown in Figure 5b. It's easy to see that H still perfectly separates $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4$, and $\boldsymbol{x}_0$ can manipulate its feature to mislead either h_1 or h_2. However, since the region $\{(x, y) : h_1(x, y) = 1, h_2(x, y) = 1\}$ does not intersect with B, $\boldsymbol{x}_0$ cannot alter its feature to mislead both h_1 and h_2. This implies H will correctly classify $\boldsymbol{x}_0$ as -1 with probability 0.5. Thus, the empirical risk of H is $\frac{1}{5} \cdot \frac{1}{2} = \frac{1}{10} < \frac{1}{5}$, which concludes the proof.



(a) Any deterministic linear classifier that perfectly separates $\{x_1, x_2, x_3, x_4\}$ must intersect with x_0's manipulation region B. As a result, x_0 can always move to a point in B and thus be misclassified.



(b) We can construct a randomized classifier $H = \{(h_1, 0.5), (h_2, 0.5)\}$ such that H not only correctly separates $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4$ but also classifies $\boldsymbol{x}_0$ to its true label $(+1)$ with probability 0.5.

Figure 5: A zero-manipulation-cost strategic classification instance where a randomized classifier beats all deterministic classifiers. Data points x_1, x_2 have true label $+1$ and x_0, x_3, x_4 have true label -1. $\boldsymbol{x}_0$ is the only strategic point and its manipulation region B contains two segments marked with dashed lines. Left: any deterministic classifier incurs an empirical risk at least $\frac{1}{5}$. Right: there exists a randomized classifier that obtains a better empirical risk $\frac{1}{10}$.

References

Saba Ahmadi, Hedyeh Beyhaghi, Avrim Blum, and Keziah Naggita. The strategic perceptron. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 6–25, 2021.

Emrah Akyol, Cedric Langbort, and Tamer Basar. Price of transparency in strategic machine learning. *arXiv*, pages arXiv–1610, 2016.

Kareem Amin, Afshin Rostamizadeh, and Umar Syed. Learning prices for repeated auctions with strategic buyers. In *Advances in Neural Information Processing Systems*, pages 1169–1177, 2013.

Pranjal Awasthi, Abhratanu Dutta, and Aravindan Vijayaraghavan. On robustness to adversarial examples and polynomial optimization. In *Advances in Neural Information Processing Systems*, pages 13737–13747, 2019.

Yahav Bechavod, Katrina Ligett, Zhiwei Steven Wu, and Juba Ziani. Causal feature discovery through strategic modification. *arXiv preprint arXiv:2002.07024*, 2020.

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

Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. Evasion attacks against machine learning at test time. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 387–402. Springer, 2013.

Mark Braverman and Sumegha Garg. The role of randomness and noise in strategic classification. In *1st Symposium on Foundations of Responsible Computing (FORC 2020)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2020.

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

Miles Bryan and Keystone Crossroads. 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/.

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

Yiling Chen, Chara Podimata, Ariel D. Procaccia, and Nisarg Shah. Strategyproof linear regression in high dimensions. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, page 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`.

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

Danielle Keats Citron and Frank A. Pasquale. 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.

Daniel Cullina, Arjun Nitin Bhagoji, and Prateek Mittal. Pac-learning in the presence of adversaries. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 230–241. Curran Associates, Inc., 2018.

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

Jinshuo Dong, Aaron Roth, Zachary Schutzman, Bo Waggoner, and Zhiwei Steven Wu. Strategic classification from revealed preferences. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, EC '18, page 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.

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

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

Bryce Goodman and Seth Flaxman. 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.

Aniko Hannak, Gary Soeller, David Lazer, Alan Mislove, and Christo Wilson. Measuring price discrimination and steering on e-commerce web sites. In *Proceedings of the 2014 conference on internet measurement conference*, pages 305–318, 2014.

Moritz Hardt, Nimrod Megiddo, Christos Papadimitriou, and Mary Wootters. Strategic classification. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, ITCS '16, page 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.

Lily Hu, Nicole Immorlica, and Jennifer Wortman Vaughan. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 259–268, 2019a.

Lily Hu, Nicole Immorlica, and Jennifer Wortman Vaughan. The disparate effects of strategic manipulation. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 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.

Matthew Jagielski, Alina Oprea, Battista Biggio, Chang Liu, Cristina Nita-Rotaru, and Bo Li. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. *2018 IEEE Symposium on Security and Privacy (SP)*, pages 19–35, 2018.

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

Bo Li and Yevgeniy Vorobeychik. Feature cross-substitution in adversarial classification. In *Advances in neural information processing systems*, pages 2087–2095, 2014.

John Miller, Smitha Milli, and Moritz Hardt. Strategic classification is causal modeling in disguise. *arXiv*, pages arXiv–1910, 2019.

Smitha Milli, John Miller, Anca D. Dragan, and Moritz Hardt. The social cost of strategic classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, FAT* '19, page 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.

Mehryar Mohri and Andres Munoz. Revenue optimization against strategic buyers. In *Advances in Neural Information Processing Systems*, pages 2530–2538, 2015.

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

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

Yurii Nesterov et al. *Lectures on convex optimization*, volume 137. Springer, 2018.

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

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

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

Benjamin I.P. Rubinstein, Blaine Nelson, Ling Huang, Anthony D. Joseph, Shing-hon Lau, Satish Rao, Nina Taft, and J. D. Tygar. 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.

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

Yonadav Shavit, Benjamin Edelman, and Brian Axelrod. Causal strategic linear regression. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 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/.

Stratis Tsirtsis, Behzad Tabibian, Moein Khajehnejad, Adish Singla, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. Optimal decision making under strategic behavior. *arXiv preprint arXiv:1905.09239*, 2019.

Berk Ustun, Alexander Spangher, and Yang Liu. Actionable recourse in linear classification. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 10–19, 2019.

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

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

Jane Williams and Bridget Haire. 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.

Hanrui Zhang and Vincent Conitzer. Incentive-aware pac learning. *AAAI 2021*, 2021.

Hanrui Zhang, Yu Cheng, and Vincent Conitzer. Distinguishing distributions when samples are strategically transformed. In *Advances in Neural Information Processing Systems*, pages 3193–3201, 2019a.

Hanrui Zhang, Yu Cheng, and Vincent Conitzer. When samples are strategically selected. In *International Conference on Machine Learning*, pages 7345–7353, 2019b.