Strategyproof Classification Under Constant Hypotheses: A Tale of Two Functions

Reshef Meir, Ariel D. Procaccia, and Jeffrey S. Rosenschein
School of Engineering and Computer Science
The Hebrew University of Jerusalem
{reshef24, arielpro, jeff}@cs.huji.ac.il

Abstract

We consider the following setting: a decision maker must make a decision based on reported data points with binary labels. Subsets of data points are controlled by different selfish agents, which might misreport the labels in order to sway the decision in their favor. We design mechanisms (both deterministic and randomized) that reach an approximately optimal decision and are strategyproof, i.e., agents are best off when they tell the truth. We then recast our results into a classical machine learning classification framework, where the decision maker must make a decision (choose between the constant positive hypothesis and the constant negative hypothesis) based only on a sampled subset of the agents' points.

Introduction

In the design and analysis of multiagent systems, one often cannot assume that the agents are cooperative. Rather, the agents might be self-interested, seeking to maximize their own utility, possibly at the expense of the social good. With the growing awareness of this situation, game-theoretic notions and tools are increasingly brought into play.

One such setting, which we shall consider here, arises when a decision has to be made based on data points that are controlled by multiple (possibly) selfish agents, and the decision affects all the agents. The decision maker would like to make a decision which is consistent, as much as possible, with all the available data. However, the agents might misreport their data in an attempt to influence the final decision in their favor.

Motivating Examples

Consider, for instance, a spatial sensor array (represented as points in $\mathbb{R}^3$), and assume that each agent controls a subset of the sensors (such as the ones positioned in its own territory). A sensor's output is only available, as private information, to the controlling agent. One such scenario might be battlefield acoustic sensors (Lesser and Erman 1980): every agent controls a sector, and is charged with a specific mission in this sector. An agent might be interested in retreating (and thus failing to complete its mission) only if massive enemy movement is detected *in its own sector*. However, a global decision, to proceed or retreat, has to be made. An agent may misreport its sensor readings in order to bring about a favorable allied decision.

A second example with an economic aspect might be a common central bank, such as the European Central Bank (ECB). The governing council makes decisions that are based on reports from the various national central banks (so one can think of the national central bankers as the agents). The national central bankers, in turn, collect private information, by means of their own institutions, regarding various economic indicators (these are the data points). Naturally, decisions taken at the European level (about, for instance, whether or not to support certain monetary policies) affect all national central banks. This strongly incentivizes the national central bankers to misreport their national statistics in a way that guarantees a decision they find desirable (though in this particular case, fear of discovery does incentivize truthfulness).

Overview of Models and Results

We present our results at two levels of generality. The more specific level is strongly motivated in its own right, but technically is also directly applied in order to obtain more general results in a machine learning framework.

Our specific model concerns n agents, each controlling a set of data points. Each point is labeled either as positive or negative; a positive label should be construed as implying that this data point supports some decision or proposition. Now, all agents report the labels of their points to some central authority, which in turn outputs a positive or negative decision. An agent's *risk* is (proportional to) the number of its points that the final decision mislabels, e.g., the number of negative points it controls in case of a positive decision. The decision maker is seeking to minimize the global risk, i.e., the total number of mislabeled points.

As noted above, an agent might find it advantageous to misreport the labels of its points. We are interested in designing decision-making mechanisms that are *strategyproof*: agents cannot benefit by lying. In return we only ask for approximate optimality. We put forward a simple deterministic decision-making mechanism which is group strategyproof (i.e., even coalitions of agents do not gain from lying) and gives a 3-approximation of the optimal global risk; in other words, the number of mislabeled points is at most 3 times the minimal number. Moreover, we show that no deterministic strategyproof mechanism can do better. Interestingly, we circumvent this result by designing a strategyproof *randomized* mechanism which gives a 2-approximation, and further demonstrate that this is as far as randomization can take us.

The second part of the paper recasts the first into a more general model, which deals with the classical machine learning classification framework. It is often the case that the decision maker cannot query agents regarding all their points, due to, for example, communication or privacy constraints (think of the European bank example given above; in this example, both abovementioned constraints apply, as the number of economic indicators is enormous, and economic institutions are well-aware of privacy considerations). To complicate matters, in the general model each agent holds a different distribution over the input space, which reflects the relative importance it gives to different data points. So, we assume that a mechanism receives labels of points from agents, where each agent's points are sampled from its individual distribution. The mechanism then outputs a decision: one of *two functions*, the *constant positive hypothesis* or the *constant negative hypothesis*. The goal is to guarantee that the concept returned by the algorithm gives a good approximation of the optimal risk, in expectation. Crucially, we demonstrate that the results of the previous, more specific, model can be leveraged to achieve this goal.

Related Work

Our work is closely related to the work of Dekel, Fischer and Procaccia (2008). They also investigated game-theoretic aspects of machine learning, albeit in a *regression learning* setting. Specifically, in their setting the label of each data point is a real number, and the risk of some hypothesis is the total *distance* to the correct labels. Dekel et al. put forward approximately optimal and strategyproof algorithms for some limited hypothesis classes. Our work should be seen as an extension of theirs to the world of classification, specifically under the very interesting (as will become apparent later) hypothesis class that contains only the two constant (positive and negative) functions.

Some existing work studies issues on the border of machine learning and game theory (Balcan et al. 2005; Procaccia et al. 2007). For example, some papers on multiagent learning (see, e.g., Littman (1994), or Hu and Wellman (2004)) attempt to learn a Nash equilibrium in Markov games (which model multiagent interactions), usually via reinforcement learning. That research does not consider incentives in the learning process itself, but rather investigates using learning to deal with strategic situations.

Another line of research attempts to learn in the face of noise (Littlestone 1991; Kearns and Li 1993; Goldman and Sloan 1995). Perhaps closer to our work is the paper of Dalvi et al. (2004), who model classification as a game between a classifier and an adversary. Dalvi et al. examine the optimal strategies of the classifier and adversary, given their strategic considerations. In contrast (but similarly to Dekel et al. (2008)), our research concentrates on designing strategyproof algorithms, i.e., algorithms that preclude strategic behavior *in the first place*, rather than algorithms that work well *in spite of* strategic behavior.

A Simple Setting

In this section we present a specific model, as described above: each agent controls a subset of data points; the decision maker has full information about the "identity" of the points controlled by the various agents, but does not know their labels. Rather, the labels are reported by the agents. This simple setting is strongly motivated in its own right (see the examples given above), but will also be leveraged later to obtain results in a learning-theoretic setting. In order to easily recast the results later, we introduce some learning theoretic notions already in this section.

Formally, let $I = \{1, \ldots, n\}$ be the set of agents. Let $\mathcal{X}$ be the input space, and $\{+, -\}$ be the set of labels to which points in $\mathcal{X}$ can be mapped.

For each agent $i \in I$, let $X_i = \{x_{i,1}, \ldots, x_{i,m_i}\} \subseteq \mathcal{X}^{m_i}$ be the set of points that agent i *controls*, and let $Y_i = \{y_{i,1}, \ldots, y_{i,m_i}\} \subseteq \{+, -\}^{m_i}$ the set of *labels* that are associated with these points. We refer to the pair $s_{i,j} = \langle x_{i,j}, y_{i,j} \rangle$ as an *example*. A positive label means, intuitively, that the example supports a decision, while a negative one means the example opposes it. We denote the subset of the dataset controlled by agent i with $S_i = \{s_{i,j}\}_{j=1}^{m_i}$ and the entire dataset, i.e., the multiset of all examples, by $S = \uplus_{i \in I} S_i$.

Let C be a class of functions from $\mathcal{X}$ to $\{+, -\}$, i.e., each $c \in C$ is a classifier that maps all possible points to labels. In learning theory, C is referred to as the *concept class* of the problem. In this paper we will consider the special case where C contains only the two constant functions $\{c_+, c_-\}$ where $\forall x \in \mathcal{X}, c_+(x) = +; c_-(x) = -$, i.e., the classification mechanism may decide to classify *all* examples as positive, or all of them as negative. This should be interpreted as taking either a positive or a negative decision.

We evaluate each such classifier simply according to the number of errors it makes on the set of examples. Formally, we define the *subjective risk* associated by agent i with the classifier c as

$$R_i(c, S_i) = \frac{1}{m_i} \sum_{j=1}^{m_i} \ell(c(x_{i,j}), y_{i,j}),$$

where ℓ is the natural 0–1 loss function: $\ell(y, y')$ is 1 if $y \neq y'$ and 0 if $y = y'$. We define the *global risk* in a similar way to be the average risk with respect to all agents:

$$R(c, S) = \frac{\sum_{i \in I} m_i \, R_i(c, S_i)}{\sum_{i \in I} m_i} = \frac{1}{m} \sum_{\langle x,y \rangle \in S} \ell(c(x), y), \tag{1}$$

where $m = \sum_{i \in I} m_i$.

A *mechanism* receives as input a dataset S, and outputs one of the two concepts in C. Our goal is to design a mechanism that minimizes the global risk, i.e., a mechanism that chooses from $\{c_+, c_-\}$ the concept that makes fewer errors on S. Less formally, the decision maker would like to make either a positive or negative decision, in a way that is most consistent with the available data.

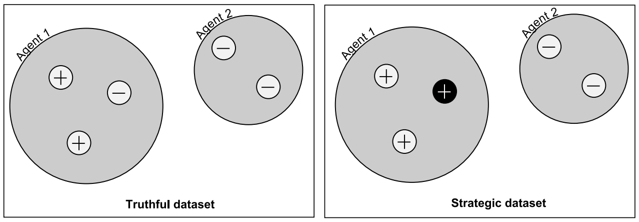

Figure 1: ERM is not strategyproof. Agent 1 changes one of its points from negative to positive, thus changing the risk minimizer from c_- to c_+, to agent 1's advantage. In this illustration, $\mathcal{X} = \mathbb{R}^2$.

In our model, agents report to the mechanism the labels of the points they control. If all agents report truthfully, the above problem is trivially solved by choosing c according to the majority of labels. This is a special case of the Empirical Risk Minimization (ERM) mechanism,[1] which by definition picks the concept in C that minimizes the risk on the set of given examples. We denote by c^* the concept returned by ERM, formally:

$$c^* = ERM(S) = \text{argmin}_{c \in C} R(c, S)$$

Unfortunately, if we choose ERM as our mechanism then agents may lie in order to decrease their subjective risk. Indeed, consider the following dataset (illustrated in Figure 1): agent 1 controls 3 examples, 2 positive and 1 negative. Agent 2 controls 2 examples, both negative. Since there is a majority of negative examples, ERM would return c_-; agent 1 would suffer a subjective risk of 2/3. On the other hand, if agent 1 reported his negative example to be positive as well, ERM would return c_+, with a subjective risk of only 1/3 for agent 1. Indeed, note that an agent's utility is measured with respect to its real labels, rather than with respect to the reported labels.

Why is truth-telling important? Once we guarantee that agents are telling the truth, we may concentrate on minimizing the risk, knowing that this is equivalent to actually maximizing the social good (i.e., making the right decision). In other words, we would like our mechanism to be *strategyproof* (SP). By definition, a mechanism is SP (in dominant strategies) if no agent may gain (i.e., lower its subjective risk) by reporting labels that differ from its real labels.

Remark 1. If we allow payments to be transferred to and from the agents, ERM can be augmented with Vickrey-Clarke-Groves (VCG) payments to achieve strategyproofness (see, e.g., (Nisan 2007) for an overview of the VCG mechanism). However, in many multiagent systems, and in particular in internet settings, such payments are often not feasible. Therefore, we concentrate throughout the paper on achieving good mechanisms *without* payments. See Dekel et al. (2008) for a discussion of this point.

Despite the fact that ERM is not SP, the concept that minimizes the global risk is clearly optimal. Thus we would like to use it to evaluate other concepts and mechanisms. Formally, define the optimal risk to be

$$r^* = R(c^*, S) = \min\{R(c_+, S), R(c_-, S)\}.$$

As is common in computer science, we will be satisfied with only approximate optimality (if this guarantees strategyproofness). Indeed:

Definition 2. A mechanism M is an *α-approximation* mechanism if for any dataset S it holds that $R(M(S), S) \leq \alpha \cdot r^*$.

ERM, for example, is a 1-approximation mechanism, but is not SP. On the other hand, a mechanism that always returns c_- is SP but does not give any finite approximation ratio (it is sufficient to consider a dataset with one positive example).

Remark 3. Informally we state that in our current setting, we can obtain similar approximation results even under mechanisms that are not SP, assuming agents lie only when this is beneficial to them. Nevertheless, strategyproofness gives us a very clean framework to analyze mechanisms in the face of strategic behavior. When we discuss our learning theoretic framework, where obtaining strategyproofness is next to impossible, we shall apply the former, less elegant, type of analysis.

Deterministic Mechanisms

We start with some observations. Note that the identity of each sampled point is not important, only the *number* of positive and negative points each agent controls. Thus we denote by $P_i = |\{\langle x, y\rangle \in S_i : y = +\}|, N_i = m_i - P_i = |\{\langle x, y\rangle \in S_i : y = -\}|$. For convenience we also let $P = \sum_{i \in I} P_i$, $N = \sum_{i \in I} N_i$. We emphasize that $\{P_i, N_i\}_{i \in I}$ contain all the information relevant for our problem and can thus replace S.

Now, denote by c_i the ERM on S_i, i.e., $c_i = c_+$ if $P_i \geq N_i$ and c_- otherwise. Clearly c_i is the best classifier agent i can hope for. Consider the following mechanism

Mechanism 1

1. Based on the labels of each agent P_i, N_i, calculate c_i. Define each agent as a *negative agent* if $c_i = c_-$, and as a *positive agent* if $c_i = c_+$.
2. Denote by $P' = \sum_{i:c_i=c_+} m_i$ the number of examples that belong to positive agents, and similarly $N' = \sum_{i:c_i=c_-} m_i = m - P'$.
3. If $P' \geq N'$ return c_+, otherwise return c_-.

Remark 4. Mechanism 1 can be thought of as a specialized, imported version of the Project-and-Fit mechanism of Dekel et al. (Dekel, Fischer, and Procaccia 2008). However, the results regarding Mechanism 1's guarantees do not follow from their results, since the setting is different (regression vs. classification).

We will show that this mechanism has the excellent game-theoretic property of being *group strategyproof*: no coalition of players can gain by lying. In other words, if some agent in the coalition strictly gains from the joint lie, some other agent in the coalition must strictly lose.

[1] In the current setting, there is no distinction between empirical risk and "real" risk. This distinction will become apparent in the next section.

Theorem 5. *Mechanism 1 is a 3-approximation group strategyproof mechanism.*

Proof. We first show group strategyproofness. Let $B \subseteq I$. We can assume without loss of generality that either all agents in B are positive or all of them are negative, since a positive (resp., negative) agent cannot gain from lying if the mechanism returns c_+ (resp., c_-). Again w.l.o.g., the agents are all positive. Therefore, if some agent is to benefit from lying, the mechanism has to return c_- on the truthful dataset. However, since the mechanism considers all agents in B to be positive agents when the truthful dataset is given, an agent in B can only hope to influence the outcome by reporting a majority of negative examples. However, this only increases N', reinforcing the mechanism's decision to return c_-.

It remains to demonstrate that the approximation ratio is as claimed. We assume without loss of generality that the mechanism returned c_+, i.e., $P' \geq N'$. We first prove that if the mechanism returned the positive concept, at least $1/4$ of the examples are indeed positive.

Lemma 6. $P \geq \frac{1}{4}m$.

Proof. Clearly $P' \geq \frac{m}{2} \geq N'$ otherwise we would get $c = c_-$. Now, if an agent is *positive* ($c_i = c_+$), at least half of its examples are also positive. Thus

$$P = \sum_{i \in I} P_i \geq \sum_{i : c_i = c_+} P_i \geq \sum_{i : c_i = c_+} \frac{m_i}{2} = \frac{P'}{2},$$

and so:

$$P \geq \frac{P'}{2} \geq \frac{m}{4}$$

□

Now, we know that $P + N = m$, so:

$$N = m - P \leq m - (\frac{m}{4}) = \frac{3m}{4} \leq 3P$$

Clearly if the mechanism decided "correctly", i.e., $P \geq m/2$, then

$$R(c, S) = R(c_+, S) = \frac{N}{m} = r^*.$$

Otherwise, if $P < m/2$, then

$$R(c, S) = R(c_+, S) = \frac{N}{m} \leq 3\frac{P}{m} = 3R(c_-, S) = 3r^*.$$

In any case we have that $R(c, S) \leq 3r^*$, proving that Mechanism 1 is indeed a 3-approximation mechanism. □

As 3-approximation is achieved by such a trivial mechanism, we would naturally like to know whether it is possible to get a better approximation ratio, without waiving the SP property. We show that this is *not* the case by proving a matching lower bound on the best possible approximation ratio achievable by an SP mechanism. Note that the lower bound only requires strategyproofness, not group strategyproofness.

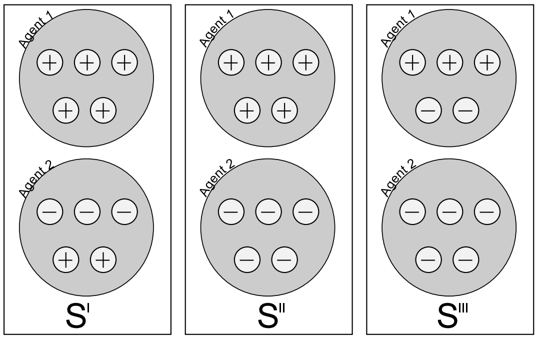

Figure 2: The examples of each agent in the three datasets are shown (for $k = 2$). Agent 1 can make dataset II look like dataset III and vice versa by reporting false labels. The same goes for agent 2 regarding datasets I and II.

Theorem 7. *Let $\epsilon > 0$. There is no $(3 - \epsilon)$-approximation strategyproof mechanism.*

Proof. To prove the bound, we present 3 different datasets. We show that any SP mechanism must return the same result on all of them, while neither concept in C yields an approximation ratio of $(3 - \epsilon)$ in all three.

Let $\epsilon > 0$. We will use $I = \{1, 2\}$, and an integer $k = k(\epsilon)$ to be defined later. Note that in all 3 datasets $m_1 = m_2 = 2k + 1$. We define the three datasets as follows (see Figure 2 for an illustration):

- S^I: $P_1 = 2k + 1, N_1 = 0$; $P_2 = k, N_2 = k + 1$
- S^{II}: $P_1 = 2k + 1, N_1 = 0$; $P_2 = 0, N_2 = 2k + 1$
- S^{III}: $P_1 = k + 1, N_1 = k$; $P_2 = 0, N_2 = 2k + 1$

Let M be some strategyproof mechanism. Then it must hold that $M(S^I) = M(S^{II})$. Indeed, otherwise assume first that $M(S^I) = c_+$ and $M(S^{II}) = c_-$. Notice that the only difference between the two settings is agent 2's labels. If agent 2's truthful labels are as in S^I, his subjective ERM is c_-. Therefore, he can report his labels to be as in S^{II} (i.e., all negative) and obtain c_-. Now, if $M(S^I) = c_-$ and $M(S^{II}) = c_+$, agent 2 can gain by deviating from S^{II} to S^I. A symmetric argument, with respect to agent 1 (that in all settings prefers c_+) shows that $M(S^{II}) = M(S^{III})$.

So, without loss of generality assume that $c = M(S^I) = M(S^{II}) = M(S^{III}) = c_+$ (otherwise, symmetric arguments yield the same result). Therefore:

$$R(c, S^{III}) = R(c_+, S^{III}) = \frac{N_1 + N_2}{m} = \frac{3k + 1}{4k + 2} \qquad (2)$$

On the other hand, the negative concept is much better:

$$r^* = R(c_-, S^{III}) = \frac{k + 1}{4k + 2}$$

By combining the last two equations:

$$\frac{R(c, S^{III})}{r^*} = \frac{\frac{3k+1}{4k+2}}{\frac{k+1}{4k+2}} = \frac{3k + 1}{k + 1}$$

Let us set $k > \frac{3}{\epsilon}$; then the last expression is strictly greater than $3 - \epsilon$, and thus $R(c, S^{III}) > (3 - \epsilon)r^*$. We conclude that any SP mechanism cannot have an approximation ratio of $3 - \epsilon$. □

Randomized mechanisms

What if we let our mechanism flip coins? Can we find an SP randomized mechanism that beats (in expectation) the 3-approximation deterministic lower bound? To answer the question we first need to formally define the risk of such a mechanism, since it may return different concepts on the same dataset. We do this by simply by taking the *expected* risk over all possible outcomes.

Definition 8. Let M be a randomized mechanism, which returns each concept $c \in C$ with probability $p_M(c|S)$.

$$R(M(S), S) = \mathbb{E}[R(c, S)] = \sum_{c \in C} p_M(c|S) \cdot R(c, S)$$

For our simple concept class $C = \{c_+, c_-\}$, a randomized mechanism is defined only by the probability of returning a positive or negative concept, given S. Accordingly, the risk is

$$R(M(S), S) = p_M(c_+|S)R(c_+, S) + p_M(c_-|S)R(c_-, S)$$

We start our investigation of SP randomized mechanisms by establishing a lower bound of 2 on their approximation ratio.

Theorem 9. *Let $\epsilon > 0$. There is no $(2 - \epsilon)$-approximation strategyproof randomized mechanism.*

The remaining proofs are omitted due to space limitations.

We presently put forward a randomized SP 2-approximation mechanism, thereby matching the lower bound with an upper bound. We will calculate P' and N' as in our deterministic Mechanism 1. The natural thing to do would be simply to select c_+ with probability P'/m and c_- with probability N'/m. Unfortunately, this simple randomization (which is clearly SP) cannot even beat the deterministic bound of $3 - \epsilon$.[2]

Crucially, a more sophisticated (and less intuitive) randomization can do the trick.

Mechanism 2

1. Compute P' and N' as in Mechanism 1.
2. If $P' \geq N'$, set $t = \frac{N'}{m}$; return c_+ with probability $\frac{2-3t}{2-2t}$, and c_- with probability $\frac{t}{2-2t}$.
3. Else if $N' > P'$, set $t = \frac{P'}{m}$; return c_- with probability $\frac{2-3t}{2-2t}$, and c_+ with probability $\frac{t}{2-2t}$.

Theorem 10. *Mechanism 2 is a group strategyproof 2-approximation randomized mechanism.*

A Learning Theoretic Setting

In this section we extend our simple setting to a more general machine learning framework. Our previous results will be leveraged to obtain powerful learning theoretic results.

Instead of looking at a fixed set of examples and selecting the concept that fits them best, each agent $i \in I$ now has a private function $Y_i : \mathcal{X} \rightarrow \{+, -\}$, which assigns a label to every point in the input space. In addition, every agent holds a (known) distribution ρ_i over the input space, which reflects the relative importance it attributes to each point. The new definition of the subjective risk naturally extends the previous setting by expressing the errors a concept makes when compared to Y_i, given the distribution ρ_i:

$$R_i(c) = \mathbb{E}_{x \sim \rho_i}[\ell(c(x), Y_i(x))]$$

The global risk is calculated similarly to the way it was before. For ease of exposition, we will assume in this section that all agents have equal weight.[3] $(n = |I|)$

$$R(c) = \sum_{i \in I} \frac{1}{n} \cdot R_i(c)$$

Since we cannot directly evaluate the risk in this learning theoretic framework, we may only sample points from the agents' distributions and ask the agents to label them. We then try to minimize the *real* global risk, using the *empirical risk* as a proxy. The empirical risk is the risk on the sampled dataset, as defined in the previous section.

Mechanism 3

1. For each agent $i \in I$, sample m points i.i.d. from ρ_i. Denote i's set of points as $X_i = \{x_{i1}, \ldots, x_{im}\}$.
2. For every $i \in I$, $j = 1, \ldots, m$, ask agent i to label x_{ij}. Denote $\overline{S}_i = \{\langle x_{i,j}, y_{i,j} \rangle\}_{j=1}^{m}$.
3. Use Mechanism 2 on $\overline{S} = \{\overline{S}_1, \ldots, \overline{S}_n\}$, and return the result.

We presently establish a theorem that explicitly states the number of examples we need to sample in order to properly estimate the real risk. We will get that, in expectation (taken over the randomness of the sampling procedure and Mechanism 2's randomization), Mechanism 3 yields close to a 2-approximation with relatively few examples, even in the face of strategic behavior. The subtle point here is that Mechanism 3 is not strategyproof. Indeed, even if an agent gives greater weight to negative points (according to Y_i and ρ_i), it might be the case that (by miserable chance) the agent's sampled dataset only contains positive points.

However, since Mechanism 2 is SP in the previous section's setting, if an agent's sampled dataset faithfully represents its true distribution, and the agent is strongly inclined towards c_+ or c_-, the agent still cannot benefit by lying. If an agent is almost indifferent between c_+ and c_-, it might wish to lie—but crucially, such an agent contributes little to the global risk.

[2] We will not prove this formally, but shortly consider $P_1 = k + 1$, $N_1 = k$; $N_2 = m_2 = k(2k + 1)$ as k increases.

[3] The results can be generalized to varying weights by sampling for each agent a number of points proportional to its weight, yet still large enough.

Our game theoretic assumption in the theorem is that agents that cannot gain by lying will tell the truth (so under this assumption, some agents may tell the truth even if they gain by lying). This is a weaker assumption than the common assumption that all agents are utility maximizing (i.e., simply wish to minimize their subjective risk). It is useful to employ the weaker version, as in many settings it might be the case that some of the agents are centrally designed, and so are bound to tell the truth regardless (even if they can gain by lying).

Remark 11. Consider the following simple mechanism: sample one point per agent, and let the agent label this single point. If the agent labels the point positively, the agent is positive; otherwise it is negative. Now apply Mechanism 2. Under the latter (strong) assumption this mechanism provides good guarantees, but under the former (weak) assumption it provides bad guarantees (since truthful agents might be assigned datasets that do not reflect their risk)—unlike Mechanism 3, as will become apparent momentarily.

One can also consider a mechanism that just asks each agent to report whether it prefers c_+ or c_-. Such a mechanism, though, is not consistent with our learning theoretic framework, and so is outside the scope of this paper.

Theorem 12. *Given sampled datasets, assume that agents are truthful if they cannot gain by lying. Let $R(M_3)$ denote the expected risk of Mechanism 3, where the expectation is taken over the randomness of the sampling and Mechanism 2. For any $\epsilon > 0$, there is an m (polynomial in ln(n) and $\frac{1}{\epsilon}$) such that by sampling m points for each agent, it holds that*

$$R(M_3) \leq 2r^* + \epsilon.$$

Specifically, sampling $m > 50\frac{1}{\epsilon^2}\ln(\frac{10n}{\epsilon})$ will suffice.

Remark 13. In our current learning theoretic setting there are no reasonable SP mechanisms. Indeed, even dictatorship, i.e., choosing some fixed agent's best classifier given its reported examples, is not SP, as one can sample a majority of positive examples when the agent in fact prefers c_-. In their Theorem 5.1, Dekel et al. (2008) do not obtain SP in the (regression) learning theoretic setting, but rather ϵ-SP: agents cannot gain more than ϵ by lying—with high probability, given enough examples. We circumvent the strategyproofness issue with a more complicated assumption, and thereby obtain a far stronger result (which is not true in the Dekel et al. setting). On the other hand, our result only holds for the very small hypothesis class $\{c_+, c_-\}$, while theirs is more general.

Conclusions

We explored the problem of making a decision based on labeled data, under the assumption that the labels are not directly accessible. Rather, they are reported by agents that may lie in order to bias the final decision in their favor.

Using the classic definition of optimal risk as the minimal number of mislabeled data points, we presented a very simple deterministic strategyproof mechanism whose risk is at most three times optimal. Moreover, we demonstrated that no deterministic mechanism can do better while maintaining the strategyproofness property. We further showed that the deterministic 3-approximation bound can be improved to a 2-approximation using the notion of expected risk and a nonintuitive randomized mechanism. Finally, in the last section we demonstrated how to reformulate this mechanism in a learning theoretic setting, where the mechanism essentially learns a constant concept based on sampled data that is controlled by selfish agents.

Our mechanisms can serve human and automated decision makers that wish to maximize social welfare in the face of data that is biased by conflicting interests. Crucially, our results in the learning theoretic setting constitute first steps in designing classifiers that can function well in noncooperative environments; in the future we intend to extend the results to richer concept classes.

Acknowledgments

This work was partially supported by Israel Science Foundation grant #898/05. Ariel Procaccia is supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

References

Balcan, M.-F.; Blum, A.; Hartline, J. D.; and Mansour, Y. 2005. Mechanism design via machine learning. In *The 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005)*, 605–614.

Dalvi, N.; Domingos, P.; Mausam; Sanghai, S.; and Verma, D. 2004. Adversarial classification. In *Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, 99–108.

Dekel, O.; Fischer, F.; and Procaccia, A. D. 2008. Incentive compatible regression learning. In *The ACM-SIAM Symposium on Discrete Algorithms (SODA 2008)*, 277–286.

Goldman, S. A., and Sloan, R. H. 1995. Can PAC learning algorithms tolerate random attribute noise? *Algorithmica* 14(1):70–84.

Hu, J., and Wellman, M. 2004. Nash Q-learning for general-sum stochastic games. *Journal of Machine Learning Research* 4:1039–1069.

Kearns, M., and Li, M. 1993. Learning in the presence of malicious errors. *SIAM J. on Computing* 22(4):807–837.

Lesser, V. R., and Erman, L. D. 1980. Distributed interpretation: A model and experiment. *IEEE Transactions on Computers* 29(12):1144–1163.

Littlestone, N. 1991. Redundant noisy attributes, attribute errors, and linear-threshold learning using Winnow. In *COLT*, 147–156.

Littman, M. L. 1994. Markov games as a framework for multi-agent reinforcement learning. In *ICML*, 157–163.

Nisan, N. 2007. Introduction to mechanism design (for computer scientists). In Nisan, N.; Roughgarden, T.; Tardos, E.; and Vazirani, V., eds., *Algorithmic Game Theory*. Cambridge University Press. chapter 9.

Procaccia, A. D.; Zohar, A.; Peleg, Y.; and Rosenschein, J. S. 2007. Learning voting trees. In *The National Conference on Artificial Intelligence (AAAI 2007)*, 110–115.