Learning with Monotone Adversarial Corruptions

Kasper Green Larsen LARSEN@CS.AU.DK
Aarhus University

Chirag Pabbaraju CPABBARA@STANFORD.EDU
Stanford University

Abhishek Shetty SHETTY@MIT.EDU
MIT

Editors: Matus Telgarsky and Jonathan Ullman

Abstract

We study the extent to which standard machine learning algorithms rely on exchangeability and independence of data by introducing a monotone adversarial corruption model. In this model, an adversary, upon looking at a "clean" i.i.d. dataset, inserts additional "corrupted" points of their choice into the dataset. These added points are constrained to be monotone corruptions, in that they get labeled according to the ground-truth target function. Perhaps surprisingly, we demonstrate that in this setting, all known optimal learning algorithms for binary classification can be made to achieve suboptimal expected error on a new independent test point drawn from the same distribution as the clean dataset. On the other hand, we show that uniform convergence-based algorithms do not degrade in their guarantees. Our results showcase how optimal learning algorithms break down in the face of seemingly helpful monotone corruptions, exposing their overreliance on exchangeability.

Keywords: Semi-random Models, Monotone Adversary.

1. Introduction

Understanding the structures that allow for generalization from seen to unseen data is perhaps the core pursuit of learning theory, and the fact that it proposes approaches addressing this accounts for both its theoretical and practical significance. A key assumption at the heart of the most basic results on generalization is that of independence, or the closely related assumption of exchangeability in the data. Given this, a natural question to ask is: how central is this assumption to the problem of generalization? This question has been studied in various guises such as in the context of domain adaptation, transfer learning, and more generally in the context of robustness. In this paper, we explore this question by studying a minimal model that violates exchangeability, while maintaining the "quality" of data, allowing us to isolate the role of these assumptions in generalization.

One motivation for the model comes from trying to understand the modern machine learning dogma of "more data is better", even when the additional data is not directly representative of the target distribution. Further, machine learning practice often involves sophisticated data curation techniques that adaptively select data points based on preliminary analysis of the data. Though these practices have shown great empirical success, their theoretical understanding is still limited. As we will see, these can break exchangeability in the data; in addition, standard techniques in machine learning do not suffice to show that such practices do not hurt generalization, even when the additional data is supposedly "clean". The key point that we want to convey using our model is that this seemingly innocuous dependence in the data can in fact provably break several approaches

to generalization such as leave-one-out and ensemble methods. On the other hand, we show that arguments based on uniform convergence are in fact robust to these perturbations, and perhaps hint at the success of loss minimization in modern machine learning.

To model this setting, we introduce the *monotone adversary* model (Definition 5), where a dataset is entirely labeled honestly, but may comprise of both representative as well as non - representative data. Here, a dataset S is prepared as follows: suppose the target ground-truth is some hypothesis $h^\star$ belonging to a class of hypotheses $\mathcal{H}$, and the representative marginal distribution of the data is $\mathcal{D}$. First, n *clean* points are drawn i.i.d. from $\mathcal{D}$ and added to S. Thereafter, a so-called "monotone adversary", who has complete knowledge about $h^\star, \mathcal{H}$ as well as $\mathcal{D}$, looks at these n points, and adds m *corrupted* points of their choice to S. The combined dataset of $n+m$ points is then entirely labeled by $h^\star$, shuffled randomly, and presented to the learning algorithm. The objective of the learning algorithm is to output a hypothesis h that has low error with respect to the representative data distribution $\mathcal{D}$.

We focus on the case when the hypothesis class $\mathcal{H}$ comprises of binary hypotheses, although the model can be defined more generally. In this case, the learnability of $\mathcal{H}$ is governed by the VC dimension d (Vapnik and Chervonenkis, 1971) of the class. In particular, the optimal expected error in the standard setting with n i.i.d. points is $\Theta(d/n)$ Haussler et al. (1994); Ehrenfeucht et al. (1989). Naturally, this is the benchmark to try and achieve in the presence of a monotone adversary.

1.1. Our Contributions

Our first main result shows that a natural perspective on generalization given by the leave-one-out principle, which is at the heart of several learning algorithms, can be completely broken by a monotone adversary. An instantiation in the binary classification setting is the celebrated *One-inclusion Graph* (OIG) algorithm (Haussler et al., 1994), which attains the optimal expected error (up to a factor of 2 (Li et al., 2001)) in the standard setting: we show that this algorithm can be forced to suffer *constant error* under monotone adversarial corruptions, even for learning a class of VC dimension 1. Our result shows that exchangeability is indeed necessary in the strongest possible sense for the guarantees of the algorithm.

Theorem 1 (Leave-one-out/OIG Lower Bound (Informal)) *There exists a monotone adversary setting for learning a binary hypothesis class of VC dimension* 1 *with* n *clean points and* n *corrupted points where the OIG algorithm suffers expected error* $1/4$.

Another reason as to why the OIG lower bound above is significant is that recent work has brought to light several natural learning settings, such as partial concept classes (Alon et al., 2022) and multiclass classes (Brukhim et al., 2022), where the OIG algorithm is the only known way to obtain a learner that attains any vanishing expected error at all. Thus, our result suggests that learnability in these settings might be fragile to strong violation of exchangeability in the data.

A different class of optimal learning algorithms in the binary setting are *ensemble/voting based* algorithms. These algorithms construct carefully chosen subsets of the training dataset, obtain a hypothesis from the underlying class $\mathcal{H}$ for each subset that is consistent with that subset (i.e., an *Empirical Risk Minimizer* for the subset), and use the majority vote of these hypotheses to make their final predictions. This class includes the first optimal learning algorithm by Hanneke (2016), as well as the more recent optimal algorithms based on Bagging (Larsen, 2023) and Majority-of-Three (Aden-Ali et al., 2024). Our next result shows that any majority voting algorithm can be made to be suboptimal in the monotone adversary model.

Theorem 2 (Majority Voting Lower Bound (Informal)) *For any majority voting algorithm that uses subsets of size at least t, there exists a monotone adversary setting for learning a binary hypothesis class of VC dimension d with n clean points and $2n/t$ corrupted points where the algorithm suffers expected error $\Omega(d\log(n/d)/n)$.*

We note that all the three optimal majority voting algorithms mentioned above satisfy $t = \Omega(n)$, and hence the lower bound holds for these algorithms with just a *constant* number of corrupted samples. Theorem 1, in conjunction with Theorem 2, establishes the fragility of all known techniques for achieving optimal algorithms, to monotone adversarial corruptions.

A key point to note here is that the lower bound above does not prove that the majority voting algorithm does not have a vanishing error rate; rather, it just shows the suboptimality of the rate. In fact, on the positive side, this suboptimality occurs just due to the fact that the algorithms are using empirical risk minimization. That is, we show that the ERM principle is robust to monotone adversarial corruptions. The intuition here is that however the adversary may try to thwart the learning algorithm, they cannot make the ground-truth hypothesis $h^\star$ look bad on the training dataset by virtue of not being able to corrupt the labels. By exploiting this observation and instantiating standard uniform convergence, we can show that in any monotone adversary model with n clean points and an *arbitrary* number of corrupted points, *every* Empirical Risk Minimizer (ERM) on the training dataset (comprising of both the clean and corrupted points) attains expected error $O(d\log(n/d)/n)$. Therefore, the suboptimality that an adversary can enforce in the monotone adversray model is at most a $\log(n/d)$ factor.

Theorem 3 (ERM Upper Bound (Informal)) *In any monotone adversary setting for learning a binary hypothesis class of VC dimension d with n clean points and an arbitrary number of corrupted points, every ERM attains expected error $O(d\log(n/d)/n)$.*

We note that the upper bound above is the best that we can hope for ERMs, since there exists a matching lower bound for ERMs in the setting where there are no corrupted points at all (Haussler et al., 1994; Auer and Ortner, 2007). The key takeaway from these results is that even though the ERM principle might not be optimal, it naturally comes with guarantees that are robust to certain forms of misspecification in the data, and perhaps hints at the success of loss minimization, as opposed to more sophisticated techniques, in modern machine learning practice.

Finally, to emphasize the point that lack of exchangeability is indeed the key obstacle in obtaining optimal expected error, we show that if the monotone adversary is constrained to be *oblivious*, i.e., the corrupted points are specified independently of the clean points, we can recover optimal expected error $O(d/n)$ using the OIG algorithm.

Theorem 4 (Oblivious Adversary OIG Upper Bound (Informal)) *In any oblivious monotone adversary setting for learning a binary class of VC dimension d with n clean points and an arbitrary number of corrupted points, the OIG algorithm attains expected error $O(d/n)$.*

Our results are summarized in Table 1.

1.2. Future Directions

Before proceeding to formally describe all our results, we will outline a host of intriguing open questions that remain unsolved from our study of the monotone adversary model .

	Algorithm	# Corrupted Points	Error
Lower Bounds	One-inclusion Graph	n	$\Omega(1)$ (Theorem 17)
	Majority Voting over t-sized subsets	$2n/t$	$\Omega(d\log(n/d)/n)$ (Theorem 16)
Upper Bounds	ERM	Arbitrary	$O(d\log(n/d)/n)$ (Theorem 12)
	One-inclusion Graph	Arbitrary (oblivious)	$O(d/n)$ (Theorem 13)

Table 1: Summary of our results in the monotone adversary model with n clean points.

The foremost question that is still open is with regards to the optimal expected error that can be achieved for learning a class of binary hypotheses of VC dimension d from n clean points in the presence of a monotone adversary. While the upper bound above for ERM (Theorem 3) shows that an expected error of $O(d\log(n/d)/n)$ can always be achieved, we have not been able to obtain any algorithm that shaves the log factor. Similarly, while Theorem 1 and Theorem 2 show lower bounds for popular optimal learners, we have not been able to show a general lower bound that precludes all learners from getting error $O(d/n)$.

Open Question 1 *What is the optimal expected error that can be achieved in the monotone adversary model for learning a binary hypothesis class of VC dimension d with n clean points?*

A good place to start here is the setting where the number of corrupted points is small (like constant or $\Theta(d)$). The primary obstacle that we have faced in our numerous efforts towards resolving this question is the lack of a clean technical tool that allows us to deal with the *lack of exchangeability* in the dataset. A central ingredient in the analysis of optimal majority voting algorithms is that even if one subset of the dataset results in a poor ERM, it is unlikely that several ERMs trained on disjoint subsets of the data are simultaneously *all* bad, simply by independence of the different subsets. The adversarial corruptions preclude this attractive property by potentially correlating disjoint subsets of the dataset. It is tempting to try to reduce from an adaptive to an oblivious monotone adversary by using subsampling as in Blanc and Valiant (2025); however, such reductions typically incur polynomial sample size blowups, whereas we can't even afford a log factor. It is also not clear how techniques like *stable sample compression schemes* (Bousquet et al., 2020) may be made to work, since conditioning on any candidate compression set renders the rest of the samples to be not i.i.d. We believe there might be mileage in trying to apply the notion of a *randomized* stable compression scheme (da Cunha et al., 2024) for an optimal expected error bound, but despite several attempts, have so far been unable to succeed. Proving a general lower against arbitrary learners also appears challenging; it is worth noting that for all the lower bound instances we construct, simple learning rules like taking the majority of all hypotheses in the class attain zero expected error.

Moving beyond the question of optimal expected error for VC classes, one might ask: what can one say about Littlestone classes? The Littlestone dimension d_L of a binary class characterizes *online learnability* of binary classes (Littlestone, 1988; Ben-David et al., 2009), and is at least the VC dimension d of the class. A standard online-to-batch analysis (Chapter 5.10 in Blum et al. (2020)) allows converting any online learner with mistake bound M to a learner that has expected

error $O(M/n)$ in the setting with n i.i.d. examples with no corruptions. Since the Standard Optimal Algorithm (SOA) has mistake bound d_L for binary classes having Littlestone dimension d_L, this gives a way to obtain a learner with expected error $O(d_L/n)$, which might be better than the $O(d\log(n/d)/n)$ guarantee of ERM in certain cases. Unfortunately, standard online-to-batch analyses also seem to go awry in the presence of monotone corrupted samples.

Open Question 2 *Is it possible to obtain an expected error $O(d_L/n)$ for learning a binary hypothesis class having Littlestone dimension d_L in the monotone adversary model with n clean points?*

Finally, one can even go beyond binary hypothesis classes, and consider the settings of *partial binary classes* (Alon et al., 2022) and *multiclass classes* (Brukhim et al., 2022). Even the broader question of obtaining a learner than attains *any vanishing error rate* as n gets large is open in these settings for the monotone adversary model. Namely, since the uniform convergence principle ceases to hold in these settings (Daniely et al., 2015; Alon et al., 2022), the ERM guarantee for binary classes no longer applies. In the standard i.i.d. setting with only clean points, the learning algorithm that is the primary workhorse in these settings is the OIG algorithm. Since we are unable to analyze this algorithm at all in the (adaptive) montone adversary setting even in the binary case, it is not clear how to show a guarantee for this algorithm in the partial or multiclass settings.

Open Question 3 *Is it possible to obtain any expected error that goes to zero as n grows, for learning any learnable partial binary class or multiclass class in the monotone adversary model with n clean points?*

A final direction to explore is computational. Though we showed that ERM achieves non-trivial learning rates (in the bounded VC dimension setting), there are natural learning problems such as learning convex bodies over Gaussian space and learning monotone functions over the uniform distribution on the hypercube where ERM does not achieve a non-vacuous rate (due to unbounded VC dimension). Nevertheless, computationally efficient learning algorithms exist in these settings, though they rely on relevant distributional assumptions. In such settings, it is not clear whether computationally efficient learning algorithms can be developed for monotone adversaries.

Open Question 4 *Can we develop techniques to handle monotone adversaries in distribution-dependent computational learning settings?*

It appears to us that the resolution of the open questions above would involve novel technical tools beyond those available in the standard toolkit.

1.3. Related Work

The study of models in statistical learning beyond independence is a rich area with a long line of work, which is beyond the scope of this paper to survey in its entirety. Perhaps the strongest model considered is online learning or mistake-bounded learning (Littlestone, 1988; Cesa-Bianchi and Lugosi, 2006) where the data is assumed to be arbitrary. Unfortunately, under such weak assumptions, learnability is characterized by notions such as Littlestone dimension (Littlestone, 1988) or sequential Rademacher complexities (Rakhlin et al., 2015), which tend to be significantly larger than their counterparts in the statistical case. Though there has been a surge of work towards understanding relaxations on the arbitrariness of data (Haghtalab et al., 2022; Haghtalab et al.;

Block et al., 2024; Shetty, 2024; Montasser et al., 2025; Goel et al., 2023, 2024), these models don't directly capture the monotone adversary problem that we consider. Perhaps the most important difference is that they focus on regret while we focus on test error on the clean distribution. Another related line of work is transfer learning (Hanneke and Kpotufe, 2024) which studies how learning under one distribution transfers to another distribution. Though this line of work gives bounds that transfer from one distribution to another, to the best of our knowledge, it does not give meaningful guarantees in the monotone adversary model that we consider. Perhaps the most closely related model and an inspiration for our work is that of Goel et al. (2023), where they study an online setting with arbitrary corruptions injected in, but where the learner is allowed to abstain from making a prediction. Our data generating process is essentially an offline analogue of their setting in the unknown distribution case, where the statistical rate remains open. A similar offline analog is also studied by Blum et al. (2021); Gao et al. (2021); Hanneke et al. (2022); Chornomaz et al. (2025), where unlike in our case, the adversary's corruptions are allowed to depend on the test point as well. This makes the problem significantly more challenging, allowing for lower bounds to established.

The choice of the name "monotone adversary" is from the long line of work on semirandom models in theoretical computer science and learning theory (Feige, 2021; Awasthi and Vijayaraghavan, 2017), where they were presented as a means of understanding the robustness of algorithms to small perturbations of the data assumptions. A salient example of this is the study of recovery thresholds for community detection where an adversary is allowed to add edges within the community and remove edges across communities (Moitra et al., 2016). Perhaps surprisingly, such changes, which one would expect to make the recovery problem easier, make the recovery problem harder. Our model can be seen as an analogue to this phenomenon in the context of learning, where the fact that the "adversarial" points are consistently labeled should make the learning problem easier, but as we show, this is not the case. More broadly, our work relates to the theme of understanding learning and statistical inference under robustness to adversarial perturbations which has seen a recent surge of interest through a computational lens (Diakonikolas and Kane, 2023).

2. Preliminaries

2.1. The Monotone Adversary

Definition 5 (Monotone Adversary) *Let $\mathcal{D}$ be a distribution over $\mathcal{X}$. Let $\mathcal{H}$ be a hypothesis class known to the learner, and let $h^\star \in \mathcal{H}$ be the target hypothesis. For $n \geq 1, m \geq 0$, let $\mathcal{A} = \mathcal{A}(\mathcal{D}, h^\star) : \mathcal{X}^n \to \mathcal{X}^m$ be a (possibly randomized) monotone adversary that has knowledge of $\mathcal{D}$, $\mathcal{H}$ and the target hypothesis $h^\star$. We denote by $\texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$ the output of the following random process: first, $x_1, \ldots, x_n \sim \mathcal{D}$ are drawn i.i.d. from $\mathcal{D}$: these are the clean samples. The adversary then takes $x_1, \ldots x_n$ as input and generates $\mathcal{A}(x_1, \ldots, x_n) = \tilde{x}_1, \ldots, \tilde{x}_m$: these are the monotone corrupted samples. The final output is a uniformly random permutation of $((x_1, h^\star(x_1)), \ldots, (x_n, h^\star(x_n)), (\tilde{x}_1, h^\star(\tilde{x}_1)), \ldots, (\tilde{x}_m, h^\star(\tilde{x}_m)))$.*

Given as input a sample $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, the aim of a learner is to output a hypothesis h that minimizes the expected error on a new test point drawn from $\mathcal{D}$, i.e.,

$$\mathrm{err}_{\mathcal{D},h^\star}(h_S) := \mathop{\mathbb{E}}_{x\sim\mathcal{D}}[\mathbb{1}[h(x) \neq h^\star(x)]] = \Pr_{x\sim\mathcal{D}}[h(x) \neq h^\star(x)]. \tag{1}$$

We note some salient features of this model. First, the entire dataset comprises of only *honest labels*. Second, even if the adversary can compose the corrupted points as a fully adaptive function

of the clean points, the benchmark for the learning algorithm is its error on a new test point drawn at random from the representative data distribution; in this sense, the test point is not in control of the adversary. Third, the introduction of the honestly labeled corrupted points nevertheless completely breaks independence and exchangeability in the data. In particular, if we condition on the corrupted points, the distribution of the clean points is no longer i.i.d. from the representative distribution.

2.1.1. OBLIVIOUS MONOTONE ADVERSARY

We also consider a more benign version of the monotone adversary that still has complete knowledge of $h^\star, \mathcal{H}$ and $\mathcal{D}$, but cannot look at the clean points while preparing the corrupted points—we term this adversary an *oblivious* monotone adversary, and will explicitly qualify it thus to distinguish it from the more powerful adaptive adversary described above.

Definition 6 (Oblivious Monotone Adversary) *Let $\mathcal{D}$ be a distribution over $\mathcal{X}$. Let $\mathcal{H}$ be a hypothesis class known to the learner, and let $h^\star \in \mathcal{H}$ be the target hypothesis. For $m \geq 0$, let $\mathcal{A} = \mathcal{A}(\mathcal{D}, h^\star) : \emptyset \to \mathcal{X}^m$ be a (possibly randomized) oblivious monotone adversary that has knowledge of $\mathcal{D}$, $\mathcal{H}$ and the target hypothesis $h^\star$. We denote by $\texttt{ObliAdv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$ the output of the following random process: first, $x_1, \ldots, x_n \sim \mathcal{D}$ are drawn i.i.d. from $\mathcal{D}$. The adversary $\mathcal{A}$ generates $\tilde{x}_1, \ldots, \tilde{x}_m$ independently without seeing $x_1, \ldots, x_n$. The final output is a uniformly random permutation of $((x_1, h^\star(x_1)), \ldots, (x_n, h^\star(x_n)), (\tilde{x}_1, h^\star(\tilde{x}_1)), \ldots, (\tilde{x}_m, h^\star(\tilde{x}_m)))$.*

2.2. Learning Algorithms

The most natural learning algorithm is an Empirical Risk Minimizer.

Definition 7 (Empirical Risk Minimizer) *An Empirical Risk Minimizer (ERM) is a learning algorithm that takes as input a training set $S = ((x_1, h^\star(x_1)), \ldots, (x_N, h^\star(x_N)))$ and produces a hypothesis $h_S \in \mathcal{H}$ with $h_S(x_i) = h^\star(x_i)$ for all $(x_i, h^\star(x_i)) \in S$.*

The following is a textbook result for the ERM algorithm.

Theorem 8 (ERM Sample Complexity (Chapter 5.6 in Blum et al. (2020))) *Let $\mathcal{H}$ be any binary hypothesis class over $\mathcal{X}$ having VC dimension d. Let $\mathcal{D}$ be any distribution over $\mathcal{X}$, and let $h^\star \in \mathcal{H}$ be the target hypothesis. With probability at least $1 - \delta$ over the draw of $S = ((x_1, y_1), \ldots, (x_N, y_N))$, where $x_1, \ldots, x_N$ are drawn i.i.d. from $\mathcal{D}$ and $y_i = h^\star(x_i)$ for every i, it holds that every $h_S \in \mathcal{H}$ that is an ERM with respect to S satisfies*

$$\mathrm{err}_{\mathcal{D},h^\star}(h_S) \leq 100 \frac{d \log(N/d) + \log(1/\delta)}{N}. \tag{2}$$

A majority voter uses ERM as a base learning algorithm in order to output a majority vote.

Definition 9 (Majority Voter) *A majority voter is a learning algorithm that takes as input an ERM algorithm and a training set $S = ((x_1, h^\star(x_1)), \ldots, (x_N, h^\star(x_N)))$. As a function of N alone, it then produces a list $L = L_1, \ldots, L_k$, where each L_i is a sequence of indices $\ell_1^i, \ldots, \ell_{t_i}^i \in \{1, \ldots, N\}$. It then constructs the training sets $S_i = ((x_{\ell_1^i}, h^\star(x_{\ell_1^i})), \ldots, (x_{\ell_{t_i}^i}, h^\star(x_{\ell_{t_i}^i})))$ corresponding to the indices in L_i. Finally, it runs the ERM algorithm on each S_i to produce hypotheses $h_1, \ldots, h_k$ and outputs the final classifier $h(x) = \mathrm{maj}(h_1(x), \ldots, h_k(x))$ where $\mathrm{maj}(\cdot)$ denotes a majority vote. We say that the majority voter uses sub-samples of size t if each list L_i has at least t distinct indices.*

We observe that previous optimal learners given by Hanneke (2016), Bagging (Larsen, 2023) as well as Majority-of-Three (Aden-Ali et al., 2024) all fall in the category of majority voters, with Hanneke's algorithm having $k = N^{\log_4 3}$ and sub-samples of size at least $N/2$, Bagging having $k = O(\log(N/\delta))$ and sub-samples of size $\Omega(N)$ (with probability $1-\exp(-\Omega(N))$), and Majority-of-Three having $k = 3$ and sub-samples of size $N/3$. We also note that the analysis of previous majority voters crucially needs sub-samples of size $\Omega(N)$.

We will also discuss a qualitatively different learning algorithm, known as the One-inclusion Graph algorithm.

Definition 10 (One-inclusion Graph (Alon et al., 1987; Haussler et al., 1994)) *The One-inclusion Graph (OIG) algorithm takes as input a training set $S = ((x_1, h^\star(x_1)), \ldots, (x_N, h^\star(x_N)))$, and outputs a hypothesis h (not necessarily in $\mathcal{H}$). For any $x \in \mathcal{X}$, $h(x)$ is obtained as follows. First, the algorithm constructs a graph $\mathcal{G}$, whose vertex set V is the set of projections (distinct labelings) by members of $\mathcal{H}$ onto $(x_1, \ldots, x_N, x)$, so that every vertex can be identified by a unique pattern in $\{0,1\}^{N+1}$. A vertex u connects to vertex v by an edge in the "direction" $i \in [N+1]$ if $u_i \neq v_i$, and $u_j = v_j$ for every $j \neq i$. An orientation σ of the edges in $\mathcal{G}$ maps every edge e in the graph to one of the two vertices it is connected to. The out-degree of a vertex u in the orientation σ is the number of edges connected to it which σ maps to the other end-point of the edge (which can be at most $N+1$, one for every direction $i \in [N+1]$). The OIG algorithm constructs any orientation σ of $\mathcal{G}$ which minimizes the largest out-degree of any vertex in the graph[1]. Consider potentially the two vertices of the form $(h^\star(x_1), \ldots, h^\star(x_N), 0)$ (the "0 vertex") and $(h^\star(x_1), \ldots, h^\star(x_N), 1)$ (the "1 vertex") in $\mathcal{G}$ (at least one of these vertices exists because $h^\star \in \mathcal{H}$). If only the 0 vertex exists, set $h(x) = 0$; otherwise, if only the 1 vertex exists, set $h(x) = 1$. Otherwise, if both the vertices exist, consider the edge e in direction $N+1$ connecting them. Set $h(x) = 0$ if σ orients e towards the 0 vertex, and $h(x) = 1$ if it orients it towards the 1 vertex.*

The following is a well-known structural result about orientations for one-inclusion graphs that are induced by hypothesis classes of VC dimension d.

Theorem 11 (Bounded out-degree of OIG (Haussler et al., 1994)) *Let $\mathcal{H}$ be a binary hypothesis class over $\mathcal{X}$ having VC dimension d. Then, for any $n \geq 1$ and $S = (x_1, \ldots, x_n)$, there exists an orientation of the one-inclusion graph of the projection of $\mathcal{H}$ onto S, such that the out-degree of every vertex in the orientation is at most d.*

3. Upper Bounds

First, we observe that the optimal expected error that a learning algorithm can achieve upon receiving a training dataset drawn from $\texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, where $h^\star$ belongs to a hypothesis class $\mathcal{H}$ having VC dimension d, is $O(d/n)$. This follows from the $\Omega(d/n)$ lower bound on the expected error that any learning algorithm must suffer, even in the case when there are no corruptions, and the input solely comprises of n i.i.d. draws from $\mathcal{D}$ (Ehrenfeucht et al., 1989).

The following result shows that the ERM algorithm achieves an expected error rate of $O(d\log(n/d)/n)$, even in the presence of a monotone adversary.

1. There may be multiple orientations that minimize the out-degree; it suffices to consider any of these. In this sense, the one-inclusion graph algorithm is really a *class* of algorithms.

Theorem 12 (Monotone Adversary ERM Upper Bound) *Let $\mathcal{H}$ be any binary hypothesis class over $\mathcal{X}$ having VC dimension d. Let $\mathcal{D}$ be any distribution over $\mathcal{X}$, and let $h^\star \in \mathcal{H}$ be the target hypothesis. Let $\mathcal{A}$ be any monotone adversary. With probability at least $1-\delta$ over the draw of $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, it holds that every $h_S \in \mathcal{H}$ that is an ERM with respect to S satisfies*

$$\mathrm{err}_{\mathcal{D},h^\star}(h_S) \leq 100\frac{d\log(n/d)+\log(1/\delta)}{n}. \tag{3}$$

Proof Let Π denote the uniformly random permutation that acts on the clean+corrupted data before being fed to the learner as input. For a sample $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, upon conditioning on Π, the clean data points exist as the data points $S' = (x_{i_1},y_{i_1}),\ldots,(x_{i_n},y_{i_n})$ in S, for some fixed distinct indices $i_1,\ldots,i_n$, and the corrupted dataset S'' corresponds to $S \setminus S'$. Furthermore, under this conditioning, $x_{i_1},\ldots,x_{i_n}$ are i.i.d. draws from $\mathcal{D}$, with $y_{i_j} = h^\star(x_{i_j})$ for every $j \in [n]$ (note that we are not conditioning on the corrupted data points). Now, observe that any ERM h_S with respect to all of S is also an ERM with respect to S'. This is because the labels on the corrupted points are still given by $h^\star$, and hence any ERM with respect to S must label all points in S according to $h^\star$, including S'. Letting $\varepsilon := 100\frac{d\log(n/d)+\log(1/\delta)}{n}$, we thus have that

$$\begin{aligned}
&\Pr_{S\sim\texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)}\left[\exists \text{ ERM } h_S \text{ s.t. } \mathrm{err}_{\mathcal{D},h^\star}(h_S) > \varepsilon\right] = \mathop{\mathbb{E}}_{\Pi}\left[\Pr_{S',S''}\left[\exists \text{ ERM } h_S \text{ s.t. } \mathrm{err}_{\mathcal{D},h^\star}(h_S) > \varepsilon \;\middle|\; \Pi\right]\right] \\
&\leq \mathop{\mathbb{E}}_{\Pi}\left[\Pr_{S',S''}\left[\exists \text{ ERM } h_{S'} \text{ s.t. } \mathrm{err}_{\mathcal{D},h^\star}(h_{S'}) > \varepsilon \;\middle|\; \Pi\right]\right] = \mathop{\mathbb{E}}_{\Pi}\left[\Pr_{S'}\left[\exists \text{ ERM } h_{S'} \text{ s.t. } \mathrm{err}_{\mathcal{D},h^\star}(h_{S'}) > \varepsilon \;\middle|\; \Pi\right]\right] \\
&\leq \delta.
\end{aligned}$$

The first inequality follows from our above reasoning, which implies that if there exists an ERM with respect to S that has large error, then there exists an ERM with respect to S' that has large error. The concluding inequality follows from Theorem 8, together with the fact that the conditional distribution of data points in S' is i.i.d. from $\mathcal{D}$ and labeled by $h^\star$. ■

Note that the expected error rate in Theorem 12 is optimal for ERM, even in the weaker case of oblivious adversaries. In particular, there exist hypothesis classes, where for any ERM, there exists a data distribution $\mathcal{D}$ for which the ERM, trained on n i.i.d. samples from $\mathcal{D}$ *without* any corruptions, suffers expected error $\Omega\left(\frac{d\log(n/d)}{n}\right)$ (Haussler et al., 1994; Auer and Ortner, 2007).

Nevertheless, for oblivious adversaries, it is possible to get an improved error rate using different algorithms; in particular, the following theorem shows that the OIG algorithm gets an expected error of $O(d/n)$. This is essentially because the corrupted points added by an oblivious adversary do not depend on the clean points, and hence the distribution of the clean points is i.i.d. from $\mathcal{D}$, even when we *condition* on the corrupted points. Note that this is not at all the case in the setting of an adaptive adversary that adds corrupted points as a function of the clean points.

Theorem 13 (Oblivious Monotone Adversary OIG Upper Bound) *Let $\mathcal{H}$ be any binary hypothesis class over $\mathcal{X}$ having VC dimension d. Let $\mathcal{D}$ be any distribution over $\mathcal{X}$, and let $h^\star \in \mathcal{H}$ be the target hypothesis. Let $\mathcal{A}$ be any oblivious monotone adversary. Then, for $S \sim \texttt{ObliAdv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, the hypothesis h_S ouput by the OIG algorithm on S satisfies*

$$\mathop{\mathbb{E}}_{S\sim\texttt{ObliAdv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)}\left[\mathrm{err}_{\mathcal{D},h^\star}(h_S)\right] \leq \frac{d}{n+1}. \tag{4}$$

Proof Let us again condition on the uniformly random permutation Π that acts on the clean+corrupted data before it is given to the learner. For a sample $S \sim \texttt{ObliAdv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, upon conditioning on Π, the clean data points exist as the data points $S' = (x_{i_1}, y_{i_1}), \ldots, (x_{i_n}, y_{i_n})$ in S, for some fixed distinct indices $i_1, \ldots, i_n$, and the corrupted dataset S'' corresponds to $S \setminus S'$. Let us further condition on the corrupted points in S''. Since S'' was specified by an oblivious adversary, it is independent of S'. Thus, conditioned on both Π and S'', the distribution of S' is that of n i.i.d. draws from $\mathcal{D}$ labeled by $h^\star$. We have that

$$\mathop{\mathbb{E}}_{S\sim\texttt{ObliAdv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)}\left[\mathrm{err}_{\mathcal{D},h^\star}(h_S)\right] = \mathop{\mathbb{E}}_{\Pi,S''}\left[\mathop{\mathbb{E}}_{S'}\left[\mathrm{err}_{\mathcal{D},h^\star}(h_S) \;\middle|\; \Pi, S''\right]\right].$$

By our reasoning above, and writing out the definition of $\mathrm{err}_{\mathcal{D},h^\star}(h_S)$, the inner expectation is equal to

$$\mathop{\mathbb{E}}_{S'}\left[\mathrm{err}_{\mathcal{D},h^\star}(h_S) \;\middle|\; \Pi, S''\right] = \mathop{\mathbb{E}}_{\substack{x_{i_1},\ldots,x_{i_n}\sim\mathcal{D}^n \\ S'=((x_{i_1},h^\star(x_{i_1})),\ldots,(x_{i_n},h^\star(x_{i_n})))}}\left[\mathop{\mathbb{E}}_{x\sim\mathcal{D}}\left[\mathbb{1}[h_S(x)\neq h^\star(x)] \;\middle|\; \Pi, S''\right]\right].$$

Now, let i_{n+1} stand for a separate index, e.g., $i_{n+1} = n+m+1$. Let $j \sim [n+1]$ be a uniformly random index drawn from $[n+1]$. Then, recalling that $S = (S', S'')$ and using exhangeability, we have that the expectation above is equal to

$$\mathop{\mathbb{E}}_{\substack{x_{i_1},\ldots,x_{i_{n+1}}\sim\mathcal{D}^{n+1} \\ \bar{S}=((x_{i_1},h^\star(x_{i_1})),\ldots,(x_{i_{n+1}},h^\star(x_{i_{n+1}})))}}\left[\mathop{\mathbb{E}}_{j\sim[n+1]}\left[\mathbb{1}[h_{(\bar{S}_{-i_j},S'')}(x_{i_j})\neq h^\star(x_{i_j})] \;\middle|\; \Pi, S''\right]\right]$$

$$= \frac{1}{n+1}\mathop{\mathbb{E}}_{\substack{x_{i_1},\ldots,x_{i_{n+1}}\sim\mathcal{D}^{n+1} \\ \bar{S}=((x_{i_1},h^\star(x_{i_1})),\ldots,(x_{i_{n+1}},h^\star(x_{i_{n+1}})))}}\left[\sum_{j=1}^{n+1}\mathbb{1}[h_{(\bar{S}_{-i_j},S'')}(x_{i_j})\neq h^\star(x_{i_j})] \;\middle|\; \Pi, S''\right].$$

In the above, $\bar{S}_{-i_j}$ is shorthand for $\bar{S} \setminus (x_{i_j}, h^\star(x_{i_j}))$. We will argue that for every fixed $\bar{S}$, the summation inside the expectation is at most d, which will give us the desired result. Let $S'' = (x_{\ell_1}, \ldots, x_{\ell_m})$, and observe that conditioned on Π and S'', we can further upper bound the summation as

$$\sum_{j=1}^{n+1}\mathbb{1}[h_{(\bar{S}_{-i_j},S'')}(x_{i_j})\neq h^\star(x_{i_j})] \leq \sum_{j=1}^{n+1}\mathbb{1}[h_{(\bar{S}_{-i_j},S'')}(x_{i_j})\neq h^\star(x_{i_j})] + \sum_{j=1}^{m}\mathbb{1}[h_{(\bar{S},S''_{-\ell_j})}(x_{\ell_j})\neq h^\star(x_{\ell_j})].$$

But now observe that in order to make a prediction in any of the summations above, the OIG algorithm constructs the same graph, because the combined set of training+test points is the same, namely $(\bar{S}, S'')$. Furthermore, by definition of the OIG algorithm which predicts on a point according to the orientation of the edge in the direction of that point, the summation above is precisely equal to the out-degree of the vertex that is the projection of $h^\star$ in this graph. From Theorem 11, we know that the out-degree of every vertex in the orientation that the algorithm constructs is at most d. This concludes the proof. ■

We remark that the guarantee above holds for any one-inclusion graph algorithm, i.e., no matter how it orients the one-inclusion graph, so long as the orientation minimizes the max out-degree.

4. Lower Bounds

In this section, we present our lower bounds which show how optimal learning strategies in the standard i.i.d. setting may become suboptimal with monotone adversarial corruptions. We will require using the following auxiliary lemma that is based on the coupon collector problem. For completeness, we give a proof in Section A.

Lemma 14 (Coupon Collector) *Consider a set of $r = cn/\log(n/d)$ elements $x_1, \ldots, x_r$, where $c > 0$ is a sufficiently large constant, $n \geq cd$ and $d \geq 1$. Then for a set S of n i.i.d. uniform samples from $x_1, \ldots, x_r$, it holds with probability at least $1/2$ that there are at least d elements x_i not in S.*

4.1. Lower Bounds against Majority Voting Strategies

Recall that one class of optimal learners in the standard i.i.d. setting with no adversary is based on majority voting among ERMs. This includes the algorithms given by Hanneke (2016), Bagging (Larsen, 2023) and Majority-of-Three (Aden-Ali et al., 2024). In this section, we prove a lower bound generally against the class of majority voting strategies, showing that a monotone adversary may force such strategies into obtaining suboptimal error.

We first prove a lower bound for majority voting invoked with *some* ERM, i.e., where we explicitly design an adversarial ERM strategy that causes suboptimal error. Note that an optimal error bound of $O(d/n)$ holds for the above mentioned voting algorithms *regardless* of which ERM algorithm they use as the base learner. With our adversarial ERM, the lower bound we prove holds for these algorithms even with a constant number of adversarial samples $m = O(1)$.

The adversarial ERM is however quite unnatural, and is explicitly designed to force a failure of majority voters. We therefore also consider a more natural ERM strategy, in which a uniformly random *consistent* hypothesis from $\mathcal{H}$ is returned as the base learner. Here, a hypothesis is consistent if it correctly labels all training samples. We extend the previous lower bound to this case as well, still needing only a constant number of adversarial samples for the optimal algorithms mentioned above.

We now formally state our first lower bound.

Theorem 15 (Majority Voting Lower Bound for Fixed ERM) *For any $d \geq 1$ and $n \geq cd$ for sufficiently large constant $c > 0$, there exists an ERM E, a hypothesis class $\mathcal{H}$ of VC dimension d over a finite domain $\mathcal{X}$, a target $h^\star \in \mathcal{H}$ and a distribution $\mathcal{D}$ over $\mathcal{X}$, such that for any majority voter with sub-samples of size at least t, there is a monotone adversary $\mathcal{A}$ that uses n clean and $m = 2n/t$ adversarial samples, such that the expected error of the majority voter on $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$ and base learner E is $\Omega(d\log(n/d)/n)$.*

Proof Let $r = cn/\log(n/d)$ for a sufficiently large constant $c > 0$ and assume $r \geq d$ (which is the case for $n \geq cd$). We define the input domain $\mathcal{X} = \{x_1, \ldots, x_r, y_1, \ldots, y_{\binom{r}{d}}\}$. We think of each y_i, instead, as y_T, corresponding to a d-sized subset T of $\{1, \ldots, r\}$. The hypothesis class $\mathcal{H}$ contains every hypothesis predicting 1 on precisely d of the r points in $\{x_1, \ldots, x_r\}$ and 0 elsewhere. Finally, $\mathcal{H}$ also contains the target hypothesis $h^\star$ which is the constant 0 function on the entire domain $\mathcal{X}$. Note that the VC dimension of $\mathcal{H}$ is d. The distribution $\mathcal{D}$ is uniform over $\{x_1, \ldots, x_r\}$.

We now define the adversary $\mathcal{A}(\mathcal{D}, h^\star)$. On a clean sample $C \sim \mathcal{D}^n$, it checks if there is a subset T of d indices in $\{1, \ldots, r\}$ for which $x_i \notin C$, $\forall i \in T$. If so, it picks an arbitrary such T and

outputs m copies of the sample y_T (recall each y_T corresponds to a d-sized subset of $\{1,\ldots,r\}$). Otherwise, it simply adds m copies of the sample x_1.

We next define an ERM. On any labeled sample S, it checks if S contains any point y_T. If so, it obtains the d-sized subset T of $\{1,\ldots,r\}$. If S does not contain any x_i for $i \in T$, it then outputs the hypothesis in $\mathcal{H}$ predicting 1 on x_i for every $i \in T$ and 0 elsewhere. In all other cases, it outputs $h^\star$. Note that this is a valid ERM as its output hypothesis correctly predicts the label of all samples in its input.

We finally lower bound the expected error of any majority voter that is given input $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$, when m is chosen to be sufficiently large as a function of n and the majority voter's sub-sample size t. Here, we observe that if more than half of the subsets $S_1,\ldots,S_k$ constructed by the majority voter from S contain at least one copy of an adversarial sample y_T, corresponding to a set $T \subset \{1,\ldots,r\}$, then the majority voter predicts 1 on x_i for every $i \in T$, resulting in an error of $|T|/r = \Omega(d\log(n/d)/n)$. We thus show that this event happens with constant probability. Recall from Definition 9 that the lists L_i of indices that determine the subsets $S_1,\ldots,S_k$ are determined by the majority voter as a function of the size of the input set S alone ($n+m$ in our case), and hence, the lists are independent of the random shuffling in $\texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$. So, if the adversary added m adversarial samples y_T to the initial input, then the probability that a sub-sample S_i contains no adversarial samples is $\binom{n}{t}/\binom{n+m}{t} \leq (n/(n+m))^t \leq \exp(-mt/(n+m))$. For $m \geq 2n/t$, this is at most e^{-1}. Thus, conditioned on the initial clean sample $C \sim \mathcal{D}^n$ satisfying that there are at least d indices in $\{1,\ldots,r\}$ for which $x_i \notin C$, $\forall i \in T$, we get that the expected number of sub-samples S_i with no adversarial sample is at most k/e. By Markov's inequality, we get that with a constant probability, more than $k/2$ sub-samples contain an adversarial sample y_T and thus the error is $\Omega(d\log(n/d)/n)$. Therefore, all that remains to show is that there are at least at least d indices in $\{1,\ldots,r\}$ for which $x_i \notin C$, $\forall i \in T$ with constant probability. This follows immediately from our choice of r and Lemma 14. ■

We next consider the ERM strategy in which a uniformly random consistent hypothesis from $\mathcal{H}$ is returned on a given input sample. Denote this ERM by E_{rand}.

Theorem 16 (Majority Voting Lower Bound for Random ERM) *For any $d \geq 1$ and $n \geq cd$ for sufficiently large constant $c > 0$, there exists a hypothesis class $\mathcal{H}$ of VC dimension d over a finite input domain $\mathcal{X}$, a target $h^\star \in \mathcal{H}$ and a distribution $\mathcal{D}$ over $\mathcal{X}$, such that for any majority voter with sub-samples of size at least t, there is a monotone adversary $\mathcal{A}$ that uses n clean and $m = 2n/t$ adversarial samples, such that the expected error of the majority voter on $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n,m)$ and base learner E_{rand} is $\Omega(d\log(n/d)/n)$.*

Proof Our proof extends the proof of Theorem 15 and we only explain the modifications. We thus assume that the reader has read the proof of Theorem 15.

First, we expand the universe by adding 1000 additional points $\{z_1,\ldots,z_{1000}\}$. The target hypothesis is still the constant 0 function. For each subset T of d indices among $\{1,\ldots,r\}$, we will first specify a template hypothesis h_T that determines labels on all points except $z_1,\ldots,z_{1000}$. Concretely, $h_T(x_i) = 1$ if $i \in T$, and $h_T(x_i) = 0$ otherwise. We set $h_T(y_T) = 0$ for the y_T that corresponds to T (see the previous proof), and set $h(y_{T'}) = 1$ otherwise. Finally, we add 1000 near-identical copies of h_T to $\mathcal{H}$. The copy $h_{T,j}$ takes the same values as h_T on $\{x_1,\ldots,x_r,y_1,\ldots,y_{\binom{r}{d}}\}$. On the points $z_1,\ldots,z_{1000}$ it takes the value 1 on z_j and 0 elsewhere. In all, $\mathcal{H}$ comprises of all the

$h_{T,j}$ functions across all T and j, together with $h^\star$. Observe that the VC dimension of $\mathcal{H}$ is at most $d+2$.

We still consider the distribution $\mathcal{D}$ to be uniform over $\{x_1,\dots,x_r\}$, and employ the same adversarial strategy as in the proof of Theorem 15. Following the analysis there further, there is a constant probability that more than $51k/100$ of the sub-samples $S_1,\dots,S_k$ contain a copy of an adversarial sample y_T with ground-truth label 0; so, we condition on this event. Since $h_{T'}(y_T) = 1$ for any $T' \neq T$, the only hypotheses in $\mathcal{H}$ consistent with these sub-samples are $h^\star$ and $h_{T,1},\dots,h_{T,1000}$. Thus, for each of these (at least $51k/100$ many) sub-samples, E_{rand} returns a hypothesis predicting 1 on all of T with probability $1000/1001$. In particular, if we consider the first $51k/100$ of these sub-samples, the expected number of hypotheses returned by E_{rand} that predict 1 on all of T is at least $510k/1001$. So, by Markov's inequality, with a constant probability, we have that at least $k/2$ many of the returned hypotheses predict 1 on all of T, meaning that the expected error of the majority vote is at least $\Omega(d\log(n/d)/n)$ as required. ■

4.2. Lower Bound against One-Inclusion Graph Algorithms

We now turn to proving lower bounds for OIG algorithms. Since the predictions made by an OIG algorithm depend critically on the orientation of the edges chosen by the algorithm, we need to define which orientation strategy we will prove lower bounds against. In the standard i.i.d. setting, any orientation with maximum out-degree of t gives an expected error of $O(t/n)$ (Haussler et al., 1994). Combined with Theorem 11, it follows that for a class of VC dimension d, we can compute an orientation with max out-degree d, and thus obtain an optimal expected error of $O(d/n)$.

In light of the above, we consider the following natural orientation strategy: for a one-inclusion graph $\mathcal{G}$ with vertex set V, let τ be the smallest achievable maximum out-degree, i.e.

$$\tau = \min_{\sigma} \max_{v\in V} \text{out-degree}_{\sigma}(v),$$

where $\text{out-degree}_\sigma(v)$ denotes the out-degree of v under the orientation σ. We consider the orientation strategy $\mathcal{O}^\star_{\text{rand}}$ that picks a uniformly random orientation σ among all orientations that satisfy $\max_{v\in V}\text{out-degree}_\sigma(v) = \tau$, i.e. an optimal max out-degree. For this orientation strategy, we show:

Theorem 17 (Lower Bound for OIG) *For any $n \geq c$ for sufficiently large constant $c > 0$, there exists a hypothesis class $\mathcal{H}$ of VC dimension 1 over a finite input domain $\mathcal{X}$, a target $h^\star \in \mathcal{H}$ and a distribution $\mathcal{D}$ over $\mathcal{X}$, such that for the OIG algorithm with orientation strategy $\mathcal{O}^\star_{rand}$, there is a monotone adversary that uses n clearn and $m = n$ adversarial samples, such that the expected error of the OIG algorithm is at least $1/4$. Furthermore, for any $1 \leq k \leq c^{-1}n$, there is a monotone adversary that uses n clean and $m = cn/\log(n/k)$ adversarial samples, such that the expected error of the OIG algorithm with orientation strategy $\mathcal{O}^\star_{rand}$ is $\Omega(k\log(n/k)/n)$.*

Choosing e.g., $k = n^{1-\varepsilon}$ for an arbitrarily small constant $\varepsilon > 0$ gives an adversary that uses $O(n/\log(n))$ adversarial samples and causes the OIG algorithm with orientation strategy $\mathcal{O}^\star_{\text{rand}}$ to suffer an expected error of $\Omega(n^{-\varepsilon})$.

Proof We start by presenting the lower bound with an adversary that uses n adversarial samples and then generalize it to $cn/\log(n/k)$ samples.

Consider the domain $\mathcal{X} = \{x_1, \ldots, x_{2n}, y_1, \ldots y_{2n}\}$. Let $\mathcal{H}$ be the hypothesis class that for every $i \in \{1, \ldots, 2n\}$ has a hypothesis h_i with $h_i(x_j) = h_i(y_j) = 1$ if $j = i$ and 0 otherwise. We let the target hypothesis $h^\star$ be the constant 0 function and add it to $\mathcal{H}$ as well. The VC dimension of this hypothesis class is 1. We consider the distribution $\mathcal{D}$ that is uniform over $\{x_1, \ldots, x_{2n}\}$.

We now define the adversary $\mathcal{A}(\mathcal{D}, h^\star)$. On a clean sample $C \sim \mathcal{D}^n$, it picks all $x_i \in C$ and adds in the corresponding point y_i.

To analyze the expected error of the OIG algorithm with orientation strategy $\mathcal{O}^\star_{\text{rand}}$ on the above distribution and adversary, assume the algorithm receives a training sample $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n, m)$ and needs to make a prediction on a fresh point $x_j \notin S$ (if $x_j \in S$, the OIG algorithm will be correct on x_j). The event $x_j \notin S$ happens with probability at least $1/2$ since $\mathcal{D}$ is uniform over $2n$ points. Conditioned on this event, consider all the vertices V in the graph $\mathcal{G}$ constructed by the algorithm, which correspond to projections of all the hypotheses in $\mathcal{H}$ on the (unlabeled) points in $S \cup \{x_j\}$. For any $h_i \in \mathcal{H}$, observe that if $x_i \in S$, then h_i predicts 1 on at least two points in S, namely x_i and y_i. This implies that there is no edge between the all-0 vertex and the vertex in V corresponding to h_i. Similarly, the vertex corresponding to h_i also does not share an edge with any vertex corresponding to h_ℓ, where $\ell \neq i$ and $x_\ell \in S$. Next, consider the hypothesis h_j: since $x_j \notin S$, h_j predicts 1 on precisely x_j, and 0 on all other points in S. There is thus an edge between the vertex corresponding to h_j and the all-0 vertex, but no edge between the vertex corresponding to h_j and any vertex corresponding to h_ℓ, where $\ell \neq j$ and $x_\ell \in S$. Finally, for any h_i with $x_i \notin S$ and $i \neq j$, observe that h_i predicts 0 on all of $S \cup \{x_j\}$, hence collapsing to the all-0 vertex. We have thus argued that the only edge in the graph is the edge between the all-0 vertex and the vertex corresponding to h_j — every other vertex in the graph is isolated. An orientation σ simply has to orient this single edge, and for either of the two ways that it can orient it, the max out-degree of a vertex in the graph is 1. Since σ is chosen uniformly at random among all orientations with smallest max out-degree, with probability $1/2$, this edge will be oriented away from the all-0 vertex, in which case the algorithm mispredicts the label on x_j. We conclude that the expected error is at least $1/4$.

To extend the above lower bound to $r = cn/\log(n/k)$ adversarial samples, we consider the domain $\mathcal{X} = \{x_1, \ldots, x_r, y_1, \ldots y_r\}$ with $\mathcal{D}$ uniform over $\{x_1, \ldots, x_r\}$. We let $\mathcal{H}$ be as above, so that $\mathcal{H}$ contains a hypothesis h_i for $i = 1, \ldots, r$ making predictions $h_i(x_j) = h_i(y_j) = 1$ if $j = i$ and 0 otherwise. We let $h^\star$ be all-0.

We now define the adversary $\mathcal{A}(\mathcal{D}, h^\star)$. On a clean sample $C \sim \mathcal{D}^n$ it picks all $i \in \{1, \ldots, r\}$ such that x_i appears at least once in C and adds in the point y_i. This specifies the draw of $S \sim \texttt{Adv}_{\mathcal{D},h^\star,\mathcal{A}}(n, m)$ for $m = r$.

By the same arguments as above, we see that if the OIG algorithm is asked to predict the label of a point $x_j \notin S$, then the one-inclusion graph has precisely one edge in it connecting the all-0 vertex and the vertex corresponding to the hypothesis h_j. We thus have that the OIG algorithm errs with probability at least $1/2$ on such x_j. By the coupon collector argument in Lemma 14, we have that with probability at least $1/2$ over the draw of S, there are at least k points x_i that do not appear in S. Each of these may be drawn from $\mathcal{D}$ with probability $1/r$, and causes the OIG algorithm to incur an error $1/2$. We conclude that the expected error of the OIG algorithm is thus $\Omega(k/r) = \Omega(k \log(n/k)/n)$. ■

Acknowledgments

CP was supported by Gregory Valiant's and Moses Charikar's Simons Investigator Awards, and a Google PhD Fellowship. KGL is funded by the European Union (ERC, TUCLA, 101125203). AS is supported in part by ARO award W911NF-21-1-0328, the Simons Foundation, NSF award DMS-2031883, a DARPA AIQ award, and an NSF FODSI Postdoctoral Fellowship. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council. Neither the European Union nor the granting authority can be held responsible for them. AS would further like to acknowledge various fruitful conversations with Jason Gaitonde, Surbhi Goel and Noah Golowich.

References

Ishaq Aden-Ali, Mikael Møller Høgsgaard, Kasper Green Larsen, and Nikita Zhivotovskiy. Majority-of-three: The simplest optimal learner? In *The Thirty Seventh Annual Conference on Learning Theory*, pages 22–45. PMLR, 2024.

Noga Alon, David Haussler, and Emo Welzl. Partitioning and geometric embedding of range spaces of finite vapnik-chervonenkis dimension. In *Proceedings of the third annual symposium on Computational geometry*, pages 331–340, 1987.

Noga Alon, Steve Hanneke, Ron Holzman, and Shay Moran. A theory of pac learnability of partial concept classes. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 658–671. IEEE, 2022.

Peter Auer and Ronald Ortner. A new pac bound for intersection-closed concept classes. *Machine Learning*, 66(2):151–163, 2007.

Pranjal Awasthi and Aravindan Vijayaraghavan. Clustering semi-random mixtures of gaussians, 2017. URL https://arxiv.org/abs/1711.08841.

Shai Ben-David, Dávid Pál, and Shai Shalev-Shwartz. Agnostic online learning. In *COLT*, volume 3, page 1, 2009.

Guy Blanc and Gregory Valiant. Adaptive and oblivious statistical adversaries are equivalent. In *Proceedings of the 57th Annual ACM Symposium on Theory of Computing*, pages 2031–2042, 2025.

Adam Block, Alexander Rakhlin, and Abhishek Shetty. On the performance of empirical risk minimization with smoothed data. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 596–629. PMLR, 2024.

Avrim Blum, John Hopcroft, and Ravindran Kannan. *Foundations of data science*. Cambridge University Press, 2020.

Avrim Blum, Steve Hanneke, Jian Qian, and Han Shao. Robust learning under clean-label attack. In *Conference on Learning Theory*, pages 591–634. PMLR, 2021.

Olivier Bousquet, Steve Hanneke, Shay Moran, and Nikita Zhivotovskiy. Proper learning, helly number, and an optimal svm bound. In *Conference on Learning Theory*, pages 582–609. PMLR, 2020.

Nataly Brukhim, Daniel Carmon, Irit Dinur, Shay Moran, and Amir Yehudayoff. A characterization of multiclass learnability. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 943–955. IEEE, 2022.

Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

Bogdan Chornomaz, Yonatan Koren, Shay Moran, and Tom Waknine. Agnostic learning under targeted poisoning: Optimal rates and the role of randomness, 2025. URL https://arxiv.org/abs/2506.03075.

Arthur da Cunha, Kasper Green Larsen, and Martin Ritzert. Boosting, voting classifiers and randomized sample compression schemes. *arXiv preprint arXiv:2402.02976*, 2024.

Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. Multiclass learnability and the erm principle. *J. Mach. Learn. Res.*, 16(1):2377–2404, 2015.

Ilias Diakonikolas and Daniel M Kane. *Algorithmic high-dimensional robust statistics*. Cambridge university press, 2023.

Andrzej Ehrenfeucht, David Haussler, Michael Kearns, and Leslie Valiant. A general lower bound on the number of examples needed for learning. *Information and Computation*, 82(3):247–261, 1989.

Uriel Feige. *Introduction to Semirandom Models*, page 189–211. Cambridge University Press, 2021.

Ji Gao, Amin Karbasi, and Mohammad Mahmoody. Learning and certification under instance-targeted poisoning, 2021. URL https://arxiv.org/abs/2105.08709.

Surbhi Goel, Steve Hanneke, Shay Moran, and Abhishek Shetty. Adversarial resilience in sequential prediction via abstention. *Advances in Neural Information Processing Systems*, 36, 2023.

Surbhi Goel, Abhishek Shetty, Konstantinos Stavropoulos, and Arsen Vasilyan. Tolerant algorithms for learning with arbitrary covariate shift. *Advances in Neural Information Processing Systems*, 37:124979–125018, 2024.

Nika Haghtalab, Yanjun Han, Abhishek Shetty, and Kunhe Yang. Oracle-efficient online learning for smoothed adversaries. In *Advances in Neural Information Processing Systems*.

Nika Haghtalab, Tim Roughgarden, and Abhishek Shetty. Smoothed analysis with adaptive adversaries. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 942–953. IEEE, 2022.

Steve Hanneke. The optimal sample complexity of pac learning. *Journal of Machine Learning Research*, 17(38):1–15, 2016.

Steve Hanneke and Samory Kpotufe. A more unified theory of transfer learning. *arXiv e-prints*, pages arXiv–2408, 2024.

Steve Hanneke, Amin Karbasi, Mohammad Mahmoody, Idan Mehalel, and Shay Moran. On optimal learning under targeted data poisoning. *Advances in Neural Information Processing Systems*, 35: 30770–30782, 2022.

David Haussler, Nick Littlestone, and Manfred K Warmuth. Predicting $\{0, 1\}$-functions on randomly drawn points. *Information and Computation*, 115(2):248–292, 1994.

Kasper Green Larsen. Bagging is an optimal pac learner. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 450–468. PMLR, 2023.

Yi Li, Philip M Long, and Aravind Srinivasan. The one-inclusion graph algorithm is near-optimal for the prediction model of learning. *IEEE Transactions on Information Theory*, 47(3):1257–1261, 2001.

Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine learning*, 2(4):285–318, 1988.

Ankur Moitra, William Perry, and Alexander S. Wein. How robust are reconstruction thresholds for community detection?, 2016. URL https://arxiv.org/abs/1511.01473.

Omar Montasser, Abhishek Shetty, and Nikita Zhivotovskiy. Beyond worst-case online classification: Vc-based regret bounds for relaxed benchmarks. *arXiv preprint arXiv:2504.10598*, 2025.

Alexander Rakhlin, Karthik Sridharan, and Ambuj Tewari. Online learning via sequential complexities. *J. Mach. Learn. Res.*, 16(1):155–186, 2015.

Abhishek Vasantha Shetty. *Learning in a Changing World: Covariate Shift, Subset Selection and Optimal PAC Bounds*. PhD thesis, University of California, Berkeley, 2024.

VN Vapnik and A Ya Chervonenkis. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability & Its Applications*, 16(2):264–280, 1971.

Appendix A. Proofs of Auxiliary Results

Proof [Proof of Lemma 14] Recall that $r = cn/\log(n/d)$ for sufficiently large constant $c > 0$. Define an indicator X_i for each x_i, taking the value 1 if $x_i \notin S$. Then $\Pr[X_i = 1] = (1 - 1/r)^n$. For two distinct $i \neq j$ we have $\Pr[X_j = 1 \mid X_i = 1] = (1 - 1/(r-1))^n$. Hence $\mathbb{E}[X_i X_j] =$

$(1-1/r)^n(1-1/(r-1))^n$. We have $\mathbb{E}[\sum_i X_i] = r(1-1/r)^n$ and

$$\mathbb{E}\left[\left(\sum_i (X_i - (1-1/r)^n)\right)^2\right] =$$
$$\sum_i \left(\mathbb{E}[X_i^2] - (1-1/r)^{2n}\right) + \sum_{i\neq j}\left(\mathbb{E}[X_iX_j] - (1-1/r)^{2n}\right) =$$
$$\sum_i \left((1-1/r)^n - (1-1/r)^{2n}\right) + \sum_{i\neq j}\left((1-1/r)^n(1-1/(r-1))^n - (1-1/r)^{2n}\right) =$$
$$r(1-1/r)^n - r(1-1/r)^{2n} + \sum_{i\neq j}(1-1/r)^{2n}\left[\left(\frac{(r-1)^2}{r(r-2)}\right)^n - 1\right] \leq$$
$$r(1-1/r)^n + r^2(1-1/r)^{2n}\left(\left(1+\frac{1}{r^2-2r}\right)^n - 1\right) \leq$$
$$r(1-1/r)^n + r^2(1-1/r)^{2n}\left(\exp\left(\frac{n}{r^2-2r}\right) - 1\right).$$
(using $1+x \leq e^x$)

For c sufficiently large, we have $n \leq (1/2)(r^2-2r)$ and thus $\exp(n/(r^2-2r)) \leq 1+2n/(r^2-2r)$ (using that $\exp(x) \leq 1+2x$ for $0 \leq x \leq 1/2$); we conclude

$$\mathbb{E}\left[\left(\sum_i (X_i - (1-1/r)^n)\right)^2\right] \leq$$
$$r(1-1/r)^n + r^2(1-1/r)^{2n}\cdot\frac{2n}{r^2-2r} \leq$$
$$r(1-1/r)^n + r^2(1-1/r)^{2n}/16.$$

Here again, the last inequality holds for c sufficiently large. By Chebyshev's inequality, we have that

$$\begin{aligned}\Pr\left[\sum_i X_i \leq (1/2)r(1-1/r)^n\right] &\leq \frac{r(1-1/r)^n + r^2(1-1/r)^{2n}/16}{(1/4)r^2(1-1/r)^{2n}} \\ &= \frac{4}{r(1-1/r)^n} + 1/4 \\ &\leq \exp(n/r)/r + 1/4 \\ &= \exp(\log(n/d)c^{-1})/r + 1/4 \\ &\leq 1/2.\end{aligned}$$

where the last inequality holds for large enough constant $c > 0$. We conclude by noting that $d \leq (1/2)r(1-1/r)^n$ under the assumptions of the lemma, and for large enough c. ■