# Transductive Robust Learning Guarantees

**Omar Montasser**
Toyota Technological Institute at Chicago

**Steve Hanneke**
Purdue University

**Nathan Srebro**
Toyota Technological Institute at Chicago

## Abstract

We study the problem of adversarially robust learning in the transductive setting. For classes $\mathcal{H}$ of bounded VC dimension, we propose a simple transductive learner that when presented with a set of labeled training examples and a set of unlabeled test examples (both sets possibly adversarially perturbed), it correctly labels the test examples with a robust error rate that is linear in the VC dimension and is adaptive to the complexity of the perturbation set. This result provides an exponential improvement in dependence on VC dimension over the best known upper bound on the robust error in the inductive setting, at the expense of competing with a more restrictive notion of optimal robust error.

## 1 INTRODUCTION

We consider the problem of learning predictors that are *robust* to adversarial examples at test time. That is, we would like to be robust against a perturbation set $\mathcal{U} : \mathcal{X} \to 2^{\mathcal{X}}$, where $\mathcal{U}(x) \subseteq \mathcal{X}$ is the set of allowed perturbations that an adversary might replace $x$ with, as measured by the *robust risk*:

$$\mathrm{R}_{\mathcal{U}}(h; \mathcal{D}) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[ \sup_{z \in \mathcal{U}(x)} \mathbb{1}\left\{ h(z) \neq y \right\} \right]. \quad (1)$$

For example, $\mathcal{U}$ could be perturbations of bounded $\ell_p$-norms (Goodfellow et al., 2015).

Adversarially robust learning has been studied almost exclusively in the *inductive* setting, where the task is to learn, from (non-adversarial) training data, a *predictor* with small robust risk (Equation 1) (Montasser et al., 2019). In many applications in practice, however, test examples are available in batches and machine learning systems are tasked with classifying them all at once. *Transductive* learning refers to the learning setting where the goal is to classify a given unlabeled test set that is presented together with the training set (Vapnik, 1998).

In this paper, we study adversarially robust learning in the *transductive* setting. In this problem, $n$ i.i.d. training examples $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^n$ and $m$ separate i.i.d. test examples $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \sim \mathcal{D}^m$ are drawn from some unknown distribution $\mathcal{D}$. Then, based on all available information: $\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}$, distribution $\mathcal{D}$, perturbation set $\mathcal{U}$, and white-box access to the transductive learner $\mathbb{A} : (\mathcal{X} \times \mathcal{Y})^n \times \mathcal{X}^m \to \mathcal{Y}^m$, an adversary chooses adversarial perturbations of the test set $\tilde{z}_i \in \mathcal{U}(\tilde{x}_i) \forall i \in [m]$, which we henceforth denote by $\tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}})$. Finally, the transductive learner $\mathbb{A}$ receives as input the labeled training examples $(\boldsymbol{x}, \boldsymbol{y})$ and the perturbed test examples $\tilde{\boldsymbol{z}}$, and outputs a labeling for $\tilde{\boldsymbol{z}}$ which we denote by $\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) \in \mathcal{Y}^m$[1]. The performance of $\mathbb{A}$ is measured by the *transductive robust risk*[2]:

$$\mathrm{TR}^{n,m}_{\mathcal{U}}(\mathbb{A}; \mathcal{D}) = \mathop{\mathbb{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^m}} \left[ \sup_{\tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}})} \frac{1}{m} \sum_{i=1}^{m} \mathbb{1}\left\{ \mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})(\tilde{z}_i) \neq \tilde{y}_i \right\} \right]. \quad (2)$$

As we shall show, the transductive setting allows for much stronger results than what is known in the inductive adversarially robust setting.

How is this possible? In traditional (non-robust) learning, there are standard transductive-to-inductive and inductive-to-transductive reductions which establish that both settings are essentially equivalent statistically. However, in Section 4 we discuss how the inductive-to-transductive reduction breaks down for adverserially robust learning, opening the possibility that transductive robust learning might be inherently

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151. Copyright 2022 by the author(s).

[1] Throughout the paper, we abuse notation and use $\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})(\tilde{z}_i)$ to refer to the $i^{\text{th}}$ entry in the vector $\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$.

[2] Unless otherwise stated, in this paper we fix the test set size $m = n$.

easier than inductive robust learning. This is good news, since inductive adversarially robust learning so far seems challenging. Focusing on adversarially robust learning of VC classes (hypothesis classes with bounded VC dimension), although we know all such classes are adverserially robustly learnable, existing inductive methods require sample complexity exponential in the VC dimension, and a completely intractable and essentially non-implementable algorithm Montasser et al. (2019). In contrast, for transductive adversarially robust learning, we present a simple and straight-forward learner with sample complexity only *linear* in the VC-dimension!

So why are we interested in the transductive setting? First, if the adversarially robust transductive setting is indeed easier than its inductive counterpart, it is important to develop methods that take advantage of this setting, and could be applicable and beneficial when entire batches of test examples are processed concurrently. This paper is the first work, as far as we are aware, in this direction. Alternatively, perhaps advances in analyzing the transductive setting could potentially translate back to the inductive setting—although the standard reduction does not apply, we can still be hopeful we might close the gap through additional ideas.

**Relaxed guarantees: choice of competitor** As with most learning theory gurantees, we will show how, given enough samples, we can approach the error of some reference competitor. The best we can hope for is to compete with $\mathsf{OPT}_{\mathcal{U}} = \inf_{h\in\mathcal{H}} \Pr_{(x,y)\sim\mathcal{D}}\left[\exists z \in \mathcal{U}(x) : h(z) \neq y\right]$, which is the smallest attainable robust risk against perturbation set $\mathcal{U}$—this is the best we could do even if we knew the source distribution. In this work, we consider a weaker goal where we compete with the smallest attainable robust risk against a stronger adversary:

$$\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = \inf_{h\in\mathcal{H}} \Pr_{(x,y)\sim\mathcal{D}}\left[\exists \tilde{x} \in \mathcal{U}^{-1}(\mathcal{U})(x) : h(\tilde{x}) \neq y\right], \tag{3}$$

where $\mathcal{U}^{-1}(z) = \{x \in \mathcal{X} : z \in \mathcal{U}(x)\}$ and $\mathcal{U}^{-1}(\mathcal{U})(x) = \cup_{z\in\mathcal{U}(x)}\mathcal{U}^{-1}(z) = \{\tilde{x} \in \mathcal{X} : \mathcal{U}(x) \cap \mathcal{U}(\tilde{x}) \neq \emptyset\}$.

In words, $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$ is the smallest attainable robust risk against the larger perturbation set $\mathcal{U}^{-1}(\mathcal{U})$. In particular, when $x \in \mathcal{U}(x)$, $\mathcal{U}(x) \subseteq \mathcal{U}^{-1}(\mathcal{U})(x)$ and $\mathsf{OPT}_{\mathcal{U}} \leq \mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$. And what we will show is a transductive learner $\mathbb{A}$ with robust risk $\mathrm{TR}_{\mathcal{U}}(\mathbb{A}; \mathcal{D})$ which is competitive with the best robust risk $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$ against the larger perturbation set $\mathcal{U}^{-1}(\mathcal{U})$.

For example, consider $\mathcal{U}(x) = \mathrm{B}_\gamma(x) \triangleq \{z \in \mathcal{X} : \rho(x, z) \leq \gamma\}$ where $\gamma > 0$ and $\rho$ is some metric on $\mathcal{X}$ (e.g., $\ell_p$-balls). In this case, $\mathcal{U}^{-1}(\mathcal{U})(x) = \mathrm{B}_{2\gamma}(x)$. Furthermore, $\mathsf{OPT}_{\mathcal{U}}$ corresponds to optimal robust risk with radius $\gamma$, while $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$ corresponds to optimal robust risk with radius $2\gamma$. In this case, our guarantees will ensure robustness to perturbations within radius $\gamma$, that is almost as good as the best possible robust risk with radius $2\gamma$. In particular, our guarantees in the realizable setting ensure robustness to perturbations within radius $\gamma$ when the smallest robust risk with radius $2\gamma$ is zero, i.e.,$\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$. By way of analogy, guarantees that are similar in spirit are common in the context of bi-criteria approximation algorithms for discrete optimization problems (e.g., the sparsest cut approximation algorithm due to Arora et al. (2004)).

**Main Contributions** We shed some new light on the problem of adversarially robust learning by studying the transductive setting. We propose a *simple* transductive learning algorithm with robust learning guarantees that are stronger than the known inductive guarantees in some aspects, but weaker in other aspects. Specifically, our algorithm enjoys an improved robust error rate that is at most *linear* in the VC dimension and is adaptive to the complexity of the perturbation set $\mathcal{U}$, and is also robust to adversarial perturbations in the *training* data. This comes at the expense of competing with the more restrictive $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$, where the inductive guarantees compete with $\mathsf{OPT}_{\mathcal{U}}$.

Specifically, given a class $\mathcal{H}$ and a perturbation set $\mathcal{U}$, we present a simple tansductive learner $\mathbb{A} : (\mathcal{X}\times\mathcal{Y})^n \times \mathcal{X}^n \to \mathcal{Y}^n$ (see Section 3) such that for any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$:

If Realizable, $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$,

$$\mathrm{TR}_{\mathcal{U}}(\mathbb{A}; \mathcal{D}) \leq \frac{\mathrm{vc}(\mathcal{H}) \log(2n)}{n}. \tag{4}$$

If Agnostic, $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} > 0$,

$$\mathrm{TR}_{\mathcal{U}}(\mathbb{A}; \mathcal{D}) \leq 2\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} + O\left(\sqrt{\frac{\mathrm{vc}(\mathcal{H})}{n}}\right). \tag{5}$$

Our transductive learner $\mathbb{A}$ simply asks for any predictor $h \in \mathcal{H}$ that robustly and correctly labels the training examples $(\boldsymbol{x}, \boldsymbol{y})$ with respect to $\mathcal{U}^{-1}$ and robustly labels the test examples $\boldsymbol{z}$ with respect to $\mathcal{U}^{-1}$. In Section 3, we show that our transductive learner additionally enjoys the following properties:

1. Robustness guarantees against adversarial perturbations in the *training* data. These are the first

non-trivial learning guarantees against adversarial perturbations in the training data, which has not been considered before in the literature to the best of our knowledge.

2. Adaptive robust error rates that are controlled by the complexity of $\mathcal{H}$ and the perturbation set $\mathcal{U}$ in the form of a new complexity measure that we introduce: the *relaxed $\mathcal{U}$-robust shattering dimension* $\mathrm{rdim}_{\mathcal{U}}(\mathcal{H})$ (see Definition 1). These are the first general robust learning guarantees that take the complexity of the perturbation set $\mathcal{U}$ into account.

**Practical Implications** In the context of deep learning and robustness to $\ell_p$ perturbations, and in scenarios where (adversarial) unlabeled test data is available in batches, our results suggest that to incur a low error rate on the test data it suffices to perform adversarial training (e.g., Madry et al. (2018); Zhang et al. (2019)) to find network parameters that simultaneously: (a) robustly and correctly fit the labeled training data, and (b) robustly fit the unlabeled (adversarial) test data. For instance, our transductive learner corresponds to Unsupervised Adversarial Training with Online Targets (Strategy 1 in [AUH+19]). Compared with inductive learning, where it is empirically observed that adversarial training does not always guarantee robust generalization (Schmidt et al., 2018), transductive learning offers a new perspective on adversarial robustness that highlights how unlabeled adversarial test data can inform local robustness, which perhaps is easier to achieve than global robustness.

## 2 PRELIMINARIES

Let $\mathcal{X}$ denote the instance space and $\mathcal{Y} = \{\pm 1\}$. Let $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ denote a hypothesis class and $\mathrm{vc}(\mathcal{H})$ denotes its VC dimension.Let $\mathcal{U} : \mathcal{X} \to 2^{\mathcal{X}}$ denote an arbitrary perturbation set such that for each $x \in \mathcal{X}$, $\mathcal{U}(x)$ is non-empty.Denote by $\mathcal{U}^{-1}$ the inverse image of $\mathcal{U}$, where for each $z \in \mathcal{X}$, $\mathcal{U}^{-1}(z) = \{x \in \mathcal{X} : z \in \mathcal{U}(x)\}$. Observe that for any $x, z \in \mathcal{X}$ it holds that $z \in \mathcal{U}(x) \Leftrightarrow x \in \mathcal{U}^{-1}(z)$. For an instance $x \in \mathcal{X}$, $\mathcal{U}^{-1}(\mathcal{U})(x)$ denotes the set of all *natural* examples $\tilde{x}$ that share some perturbation with $x$ according to $\mathcal{U}$, i.e., $\mathcal{U}^{-1}(\mathcal{U})(x) = \cup_{z \in \mathcal{U}(x)} \mathcal{U}^{-1}(z) = \{\tilde{x} \in \mathcal{X} : \mathcal{U}(x) \cap \mathcal{U}(\tilde{x}) \neq \emptyset\}$. For any sequence of labeled points $(\boldsymbol{x}, \boldsymbol{y}) \in (\mathcal{X} \times \mathcal{Y})^n$, any sequence of adversarial perturbations $\boldsymbol{z} \in \mathcal{X}^n$, and any predictor $h : \mathcal{X} \to \mathcal{Y}$ let $\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(h) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}\{h(x) \neq y\}$ denote the standard 0-1 error, and define

$$\mathrm{R}_{\mathcal{U}^{-1}}(h; \boldsymbol{z}, \boldsymbol{y}) = \frac{1}{n}\sum_{i=1}^{n} \sup_{\tilde{x} \in \mathcal{U}^{-1}(z_i)} \mathbb{1}\{h(\tilde{x}) \neq y_i\}, \quad (6)$$

$$\mathrm{R}_{\mathcal{U}^{-1}}(h; \boldsymbol{z}) = \frac{1}{n}\sum_{i=1}^{n} \sup_{\tilde{x} \in \mathcal{U}^{-1}(z_i)} \mathbb{1}\{h(\tilde{x}) \neq h(z_i)\}. \quad (7)$$

Our transductive robust learning guarantees (presented in Section 3) are in fact in terms of an adaptive complexity measure – that is in general tighter than the VC dimension and takes into account the complexity of both $\mathcal{H}$ and $\mathcal{U}$ – which we introduce next:

**Definition 1** (Relaxed Robust Shattering Dimension)**.** A sequence $z_1, \dots, z_k \in \mathcal{X}$ is said to be *relaxed $\mathcal{U}$-robustly shattered* by $\mathcal{H}$ if $\forall y_1, \dots, y_k \in \{\pm 1\}$ : $\exists x_1^{y_1}, \dots, x_k^{y_k} \in \mathcal{X}$ and $\exists h \in \mathcal{H}$ such that $z_i \in \mathcal{U}(x_i^{y_i})$ and $h(\mathcal{U}(x_i^{y_i})) = y_i \forall 1 \leq i \leq k$. The *relaxed $\mathcal{U}$-robust shattering dimension* $\mathrm{rdim}_{\mathcal{U}}(\mathcal{H})$ is defined as the largest $k$ for which there exist $k$ points that are relaxed $\mathcal{U}$-robustly shattered by $\mathcal{H}$.

The above complexity measure is inspired by the robust shattering dimension that was introduced by Montasser et al. (2019) and shown to lower bound the sample complexity of robust learning in the inductive setting:

**Definition 2** (Robust Shattering Dimension – Montasser et al. (2019))**.** A sequence $z_1, \dots, z_k \in \mathcal{X}$ is said to be *$\mathcal{U}$-robustly shattered* by $\mathcal{H}$ if $\exists x_1^+, x_1^-, \dots, x_k^+, x_k^- \in \mathcal{X}$ such that $\forall i \in [k], z_i \in \mathcal{U}(x_i^+) \cap \mathcal{U}(x_i^-)$ and $\forall y_1, \dots, y_k \in \{\pm 1\} : \exists h \in \mathcal{H}$ such that $h(\mathcal{U}(x_i^{y_i})) = y_i \forall 1 \leq i \leq k$. The *$\mathcal{U}$-robust shattering dimension* $\dim_{\mathcal{U}}(\mathcal{H})$ is defined as the largest $k$ for which there exist $k$ points $\mathcal{U}$-robustly shattered by $\mathcal{H}$.

We remark that for any class $\mathcal{H}$ and any perturbation set $\mathcal{U}$, it immediately follows from the definitions above that: $\dim_{\mathcal{U}}(\mathcal{H}) \leq \mathrm{rdim}_{\mathcal{U}}(\mathcal{H}) \leq \mathrm{vc}(\mathcal{H})$.

## 3 MAIN RESULTS

We obtain strong robust learning guarantees against worst-case adversarial perturbations of *both* the training data and the test data. Specifically, after training examples $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^n$ and test examples $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \sim \mathcal{D}^n$ are drawn, an adversary, which has white-box access to the learner, perturbs both training and test examples by choosing adversarial perturbations $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$ and $\tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}})$. Our transductive learner observes as input $(\boldsymbol{z}, \boldsymbol{y})$ and $\tilde{\boldsymbol{z}}$, and outputs $\hat{h}(\tilde{\boldsymbol{z}}) \in \mathcal{Y}^n$ where $\hat{h} \in \Delta_{\mathcal{H}}^{\mathcal{U}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ defined as follows:

If Realizable ($\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$),

$$\begin{aligned}\Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) = \big\{h \in \mathcal{H} : \mathrm{R}_{\mathcal{U}^{-1}}(h; \boldsymbol{z}, \boldsymbol{y}) = 0 \\ \wedge \mathrm{R}_{\mathcal{U}^{-1}}(h; \tilde{\boldsymbol{z}}) = 0\big\}.\end{aligned} \tag{8}$$

If Agnostic ($\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} > 0$),

$$\Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) = \operatorname*{argmin}_{h \in \mathcal{H}} \max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(h; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(h; \tilde{\boldsymbol{z}})\right\}. \tag{9}$$

Our transductive learner simply asks for any predictor $h \in \mathcal{H}$ that robustly and correctly labels the training examples $(\boldsymbol{z}, \boldsymbol{y})$ with respect to $\mathcal{U}^{-1}$ and robustly labels the test examples $\boldsymbol{z}$ with respect to $\mathcal{U}^{-1}$. Observe that requiring robustness on $\boldsymbol{z}$ and $\tilde{\boldsymbol{z}}$ with respect to $\mathcal{U}^{-1}$ implies, by definition of $\mathcal{U}^{-1}$, that the i.i.d. examples $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$ will be labeled in the same way as $\boldsymbol{z}$ and $\tilde{\boldsymbol{z}}$, even though the learner does not observe $\boldsymbol{x}$ and $\tilde{\boldsymbol{x}}$. This is the main insight that we rely on to obtain our transductive robust learning guarantees:

**Theorem 1** (Realizable)**.** *For any $n \in \mathbb{N}$, $\delta > 0$, class $\mathcal{H}$, perturbation set $\mathcal{U}$, and distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ satisfying $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$:*

$$\Pr_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^n}}\Bigg[\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \\ \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) : \mathrm{err}_{\tilde{\boldsymbol{z}}, \tilde{\boldsymbol{y}}}(\hat{h}) \leq \epsilon\Bigg] \geq 1 - \delta,$$

*where* $\epsilon = \frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H}) \log(2n) + \log(1/\delta)}{n} \leq \frac{\mathrm{vc}(\mathcal{H}) \log(2n) + \log(1/\delta)}{n}$.

**Theorem 2** (Agnostic)**.** *For any $n \in \mathbb{N}$, $\delta > 0$, class $\mathcal{H}$, perturbation set $\mathcal{U}$, and distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$,*

$$\Pr_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^n}}\Bigg[\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \\ \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) : \mathrm{err}_{(\tilde{\boldsymbol{z}}, \tilde{\boldsymbol{y}})}(\hat{h}) \leq \epsilon\Bigg] \geq 1 - \delta,$$

*where*

$$\epsilon = \min\Bigg\{2\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} + O\left(\sqrt{\frac{\mathrm{vc}(\mathcal{H}) + \log(1/\delta)}{n}}\right), \\ 3\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} + O\left(\sqrt{\frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H}) \log(2n) + \log(1/\delta)}{n}}\right)\Bigg\}.$$

The proofs of Theorem 1 and Theorem 2 are deferred to Section 5.

# 4 TRANSDUCTIVE VS. INDUCTIVE

For purposes of the discussion below, let $\mathbb{A}_I : (\mathcal{X} \times \mathcal{Y})^* \to \mathcal{Y}^{\mathcal{X}}$ denote an inductive learner and $\mathbb{A}_T : (\mathcal{X} \times \mathcal{Y})^n \times \mathcal{X}^m \to \mathcal{Y}^m$ denote a trnasductive learner. The *inductive* robust risk of $\mathbb{A}_I$ is defined as

$$\mathrm{IR}^n_{\mathcal{U}}(\mathbb{A}; \mathcal{D}) = \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n} \mathrm{R}_{\mathcal{U}}(\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}); \mathcal{D}).$$

For standard (non-robust) supervised learning, i.e., when $\mathcal{U}(x) = \{x\}$, there isn't much difference between the transductive and inductive settings in terms of statistical performance—an observation which has been employed in designing and analyzing inductive learning algorithms by relying on the transductive setting (Vapnik and Chervonenkis, 1974). We can always take an inductive learner $\mathbb{A}_I$ and use it transductively as $\mathbb{A}_T$ defined as

$$\forall i \in [m] : \mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{x}})(\tilde{x}_i) = \mathbb{A}_I(\boldsymbol{x}, \boldsymbol{y})(\tilde{x}_i), \tag{10}$$

and so $\mathrm{TR}^{n,m}(\mathbb{A}_T; \mathcal{D}) \leq \mathrm{IR}^n(\mathbb{A}_I; \mathcal{D})$.

In the other direction, given a transductive learner $\mathbb{A}_T$, if it's guarantee doesn't depend on the test set size $m$ (i.e., holds even when $m = 1$), we can consider an inductive learner $\mathbb{A}_I$ that outputs a predictor which just runs the transductive learner at test-time defined as

$$\forall x \in \mathcal{X} : \mathbb{A}_I(\boldsymbol{x}, \boldsymbol{y})(x) = \mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, x), \tag{11}$$

ensuring $\mathrm{IR}^n(\mathbb{A}_I; \mathcal{D}) = \mathrm{TR}^{n,1}(\mathbb{A}_T; \mathcal{D})$.

More generally, if the transductive learner $\mathbb{A}_T$ does rely on having multiple test examples, e.g., $m = n$ as in our case, we can randomly split the training set, using some of the training examples as test examples:

$$\forall x \in \mathcal{X} : \mathbb{A}_I(\boldsymbol{x}, \boldsymbol{y})(x) = \mathbb{A}_T(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{x}'' \cup \{x\})(x), \tag{12}$$

where $\boldsymbol{x}'$ and $\boldsymbol{x}''$ are random disjoint subsets of $\boldsymbol{x}$ of size $\lfloor \frac{n}{2} \rfloor$ and $\lfloor \frac{n}{2} \rfloor - 1$, and $\boldsymbol{x}'' \cup \{x\}$ is a random permutation of the concatenation. This ensures

$$\begin{aligned}\mathrm{IR}^n(\mathbb{A}_I; \mathcal{D}) &= \mathbb{E}\left[\mathbb{1}\left\{\mathbb{A}_T(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{x}'' \cup \{x\})(x) \neq y\right\}\right] \\ &= \mathop{\mathbb{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^{n/2} \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^{n/2}}} \mathop{\mathbb{E}}_{i\sim\mathrm{Unif}[n/2]} \left[\mathbb{1}\left\{\mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{x}})(\tilde{x}_i) \neq \tilde{y}_i\right\}\right] \\ &= \mathop{\mathbb{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^{n/2} \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^{n/2}}} \frac{2}{n} \sum_{i=1}^{\frac{n}{2}} \mathbb{1}\left\{\mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{x}})(\tilde{x}_i) \neq \tilde{y}_i\right\} \\ &= \mathrm{TR}^{\frac{n}{2}, \frac{n}{2}}(\mathbb{A}_T; \mathcal{D}).\end{aligned}$$

Why is the robust setting different? We can still reduce transductive to inductive just the same. Given an inductive learner $\mathbb{A}_I$, the construction of Equation 10 is still valid, and we have $\mathrm{TR}^{n,m}_{\mathcal{U}}(\mathbb{A}_T; \mathcal{D}) \leq \mathrm{IR}^n_{\mathcal{U}}(\mathbb{A}_I; \mathcal{D})$.

But what happens in the reverse direction? If transductive learner $\mathbb{A}_T$ doesn't rely on having multiple test examples, i.e., its guarantee doesn't depend on $m$

and is valid even if $m = 1$, the construction in Equation 11 can still be used, and we have $\mathrm{IR}^n_{\mathcal{U}}(\mathbb{A}_I; \mathcal{D}) = \mathrm{TR}^{n,1}_{\mathcal{U}}(\mathbb{A}_T; \mathcal{D})$. This reduction has the potential of aiding in designing robust inductive learning methods, but it relies on the transductive method not depending on the number of test examples, or equivalently referred to as 1-point transductive learners (e.g., the one-inclusion graph prediction algorithm due to Haussler et al. (1994)). Unfortunately, this is not the case for our transductive learner $\mathbb{A}_T$ (presented in Section 3) which requires $m = n$.

Trying to apply the other reduction from Equation 12 and its analysis in the transductive setting, we would need to implement: $\mathbb{A}_T(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{z}'' \cup \{z\})$. To apply our transductive learner $\mathbb{A}_T$, we would need not the subset of training points $\boldsymbol{x}''$, but rather their perturbations $\boldsymbol{z}''$. But how could we obtain this? This is not given to us. If $x \in \mathcal{U}(x)$, we can try using $z''_i = x''_i$, i.e. $\mathbb{A}_T(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{x}'' \cup \{z\})$, for which we get the following inductive error:

$$\mathrm{IR}^n_{\mathcal{U}}(\mathbb{A}_I; \mathcal{D}) = \mathop{\mathbb{E}}_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^{n/2} \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^{n/2}}} \Bigg[ \mathop{\mathbb{E}}_{i\sim\mathrm{Unif}[n/2]}$$

$$\sup_{\tilde{z}_i \in \mathcal{U}(\tilde{x}_i)} \mathbb{1}\left\{ \mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{x}}_{1:i}, \tilde{z}_i, \tilde{\boldsymbol{x}}_{i+1:\frac{n}{2}})(\tilde{z}_i) \neq y_i \right\} \Bigg].$$

But the right hand side here is only a (loose) upper bound on $\mathrm{TR}^{n/2,n/2}_{\mathcal{U}}(\mathbb{A}_T; \mathcal{D})$. Specifically, in the above, the supremum representing the adversary, comes in after knowing which one of the $\frac{n}{2}$ points we are evaluating on. On the other hand, in a true transductive setting, i.e., in the definition of $\mathrm{TR}^{n/2,n/2}_{\mathcal{U}}$, the adversary needs to commit to a perturbation that would be bad (for us) on all $\frac{n}{2}$ of the points. In a sense, the fact that the adversary must perturb all points, and we can leverage knowledge of these perturbations, is what restricts the power of the adversary in the transductive setting, and allows us to better protect against adversarial attacks affecting many examples (we might still get a few examples wrong, but that's OK).

**Proper vs. Improper** Another issue where we see a difference between inductive and transductive adversarially robust learning is with regards to whether the learning can be proper. Montasser et al. (2019) showed that learning some VC classes in the inductive setting necessarily requires improper learning. Specifically, there are classes $\mathcal{H}$ with constant VC dimension that are not robustly PAC learnable with any inductive proper learner $\mathbb{A}_I : (\mathcal{X} \times \mathcal{Y})^* \to \mathcal{H}$, where proper learning is defined as outputting a predictor in $\mathcal{H}$. Even in the case of robust realizability with respect to $\mathcal{U}^{-1}(\mathcal{U})$, i.e., $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$, we can still adapt the construction of Montasser et al. (2019) to conclude that improper learning is needed in the inductive setting, whereas for transductive learning, our learner from Section 3 is proper. But this isn't surprising, and also in the standard (non-robust) setting we can expect differences in properness between transductive and inductive.

We mentioned that any transductive non-robust learner can be transformed to an inductive learner, using the reduction in Equation 12. But even if the transductive learner is proper, the resulting inductive learner is not. And furthermore, any improper transductive learner, whether non-robust or robust, can be transformed to a proper transductive learner. Specifically, for any transductive learner $\mathbb{A}_T$ and any input $(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) \in (\mathcal{X} \times \mathcal{Y})^n \times \mathcal{X}^m$, we can project the labeling $\mathbb{A}_T(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ to the closest *proper* labeling in the set $\Gamma_{\mathcal{H}}(\tilde{\boldsymbol{z}}) = \{(h(\tilde{\boldsymbol{z}})) : h \in \mathcal{H}\}$. In the realizable setting, when $\exists h \in \mathcal{H}$ s.t. $\mathrm{R}_{\mathcal{U}}(h; \mathcal{D}) = 0$, we are guaranteed that whenever $\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ has $\epsilon$ error then the *proper* labeling has $2\epsilon$ error. In the agnostic setting, we are guaranteed that the *proper* labeling will incur robust error at most $3 \inf_{h\in\mathcal{H}} \mathrm{R}_{\mathcal{U}}(h; \mathcal{D}) + 2\epsilon$ whenever $\mathbb{A}(\boldsymbol{x}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ has robust error of $\inf_{h\in\mathcal{H}} \mathrm{R}_{\mathcal{U}}(h; \mathcal{D}) + \epsilon$. We therefore see that in the transductive setting, improperness can never buy a significant advantage, and we should not be surprised that learning that must be improper in the inductive setting can be proper in the transductive one.

## 5 PROOFS

We start with stating a helpful lemma that extends the classic Sauer-Shelah-Perles lemma for the robust setting (the proof is deferred to Appendix A).

**Lemma 3** (Sauer's lemma for $\mathrm{rdim}_{\mathcal{U}}(\mathcal{H})$)**.** *For any class $\mathcal{H}$, any perturbation set $\mathcal{U}$, and any sequence of points $z_1, \ldots, z_n \in \mathcal{X}$,*

$$\left|\Pi^{\mathcal{U}}_{\mathcal{H}}(z_1, \ldots, z_n)\right| \triangleq$$

$$\left|\left\{ (h(z_1), \ldots, h(z_n)) \Big|_{\substack{\exists x_1,\ldots,x_n \in \mathcal{X}, \exists h \in \mathcal{H}: \\ z_i \in \mathcal{U}(x_i) \wedge h(\mathcal{U}(x_i)) = h(z_i) \forall 1 \leq i \leq n}} \right\}\right|$$

$$\leq \binom{n}{\leq \mathrm{rdim}_{\mathcal{U}}(\mathcal{H})} \triangleq \sum_{i=0}^{\mathrm{rdim}_{\mathcal{U}}(\mathcal{H})} \binom{n}{i}.$$

### 5.1 Realizable Setting

*Proof.* (Proof of Theorem 1) It suffices to show that

$$\Pr_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n \\ (\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^n}} \Bigg[ \exists \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \exists \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}),$$

$$\exists \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) : \mathrm{err}_{\tilde{\boldsymbol{z}},\tilde{\boldsymbol{y}}}(\hat{h}) > \epsilon \Bigg] \leq \delta.$$

Observe that since $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 0$, it holds by definition of $\Delta^{\mathcal{U}}_{\mathcal{H}}$ (see Equation 8) that the set $\Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ is non-empty with probability 1.

We will first start with a standard observation stating that sampling two iid sequences of length $n$, $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^n$ and $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \sim \mathcal{D}^n$, is equivalent to sampling a single iid sequence of length $2n$, $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^{2n}$, and then randomly splitting it into two sequences of length $n$ (using a permutation $\sigma$ of $\{1, \dots, 2n\}$ sampled uniformly at random). Thus, it follows that

$$\Pr_{\substack{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^n\\(\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}})\sim\mathcal{D}^n}}\Bigg[\exists \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \exists \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \exists \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) : \\ \mathrm{err}_{\tilde{\boldsymbol{z}},\tilde{\boldsymbol{y}}}(\hat{h}) > \epsilon\Bigg] = \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{D}^{2n}}\left[\Pr_{\sigma}\left[E_{\sigma,\boldsymbol{z}} | (\boldsymbol{x}, \boldsymbol{y})\right]\right],$$

where $\sigma$ is a permutation of $[2n]$ sampled uniformly at random and $E_{\sigma,\boldsymbol{z}}$ is defined as:

$$E_{\sigma,\boldsymbol{z}} = \Bigg\{\exists \boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)}), \\ \exists \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)}) : \\ \mathrm{err}_{\boldsymbol{z}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon\Bigg\}.$$

**High error on $\boldsymbol{z}$'s implies high error on $\boldsymbol{x}$'s.** It suffices to show that for any $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^{2n}$ such that $\exists h^* \in \mathcal{H}$ with $h^*(\mathcal{U}^{-1}(\mathcal{U})(\boldsymbol{x})) = \boldsymbol{y}$ (which occurs with probability one): $\Pr_{\sigma}\left[E_{\sigma,\boldsymbol{z}} | (\boldsymbol{x}, \boldsymbol{y})\right] \leq \delta$. To this end, we will start by showing that the event $E_{\sigma,\boldsymbol{z}}$ implies the following event $E_{\sigma,\boldsymbol{x}}$:

$$E_{\sigma,\boldsymbol{x}} = \Bigg\{\exists \boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)}), \\ \exists \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)}) : \\ \mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon\Bigg\}.$$

In words, in case there are adversarial perturbations $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$ and a predictor $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)})$ with many mistakes on the adversarial perturbations: $\mathrm{err}_{\boldsymbol{z}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon$, then this implies that $\hat{h}$ makes many mistakes on the original non-adversarial test sequence: $\mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon$. This is because for any $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$, by definition of $\Delta^{\mathcal{U}}_{\mathcal{H}}$ (see Equation 8), any $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)})$ robustly labels the perturbations $\boldsymbol{z}_{\sigma(1:2n)}$: $\hat{h}(\mathcal{U}^{-1}(\boldsymbol{z}_{\sigma(1:2n)})) = \hat{h}(\boldsymbol{z}_{\sigma(1:2n)})$. That is,

$$(\forall 1 \leq i \leq 2n)\left(\forall \tilde{x} \in \mathcal{U}^{-1}(z_{\sigma(i)})\right) : \hat{h}(\tilde{x}) = \hat{h}(z_{\sigma(i)}).$$

By definition of $\mathcal{U}^{-1}$, it holds that $\boldsymbol{x}_{\sigma(1:2n)} \in \mathcal{U}^{-1}(\boldsymbol{z}_{\sigma(1:2n)})$. Thus, it follows that $\hat{h}(\boldsymbol{z}_{\sigma(n+1:2n)}) = \hat{h}(\boldsymbol{x}_{\sigma(n+1:2n)})$, and therefore, event $E_{\sigma,\boldsymbol{z}}$ implies event $E_{\sigma,\boldsymbol{x}}$.

**Finite robust labelings on $\boldsymbol{x}$'s** Based on the above, it suffices now to show that: $\Pr_{\sigma}\left[E_{\sigma,\boldsymbol{x}} | (\boldsymbol{x}, \boldsymbol{y})\right] \leq \delta$. To this end, we will show that for any permutation $\sigma$, any $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$, and any $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)})$ it holds that the labeling $\hat{h}(\boldsymbol{x}_{\sigma(1:2n)})$ is included in a finite set of possible behaviors $\Pi^{\mathcal{U}}_{\mathcal{H}}$ defined on the entire sequence $\boldsymbol{x} = (x_1, \dots, x_{2n})$ by:

$$\Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n}) \\ = \left\{(h(\boldsymbol{x}_{1:2n}))\,\middle|\,\substack{\exists z_1 \in \mathcal{U}(x_1), \dots, z_{2n} \in \mathcal{U}(x_{2n}),\\ \exists h \in \mathcal{H}: h(\mathcal{U}^{-1}(z_i)) = h(x_i) \forall 1 \leq i \leq 2n}\right\}.$$

Consider an arbitrary permutation $\sigma$ and an arbitrary $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$. For any $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)})$, by definition of $\Delta^{\mathcal{U}}_{\mathcal{H}}$, it holds that $\hat{h}\left(\mathcal{U}^{-1}(\boldsymbol{z}_{\sigma(1:n)})\right) = \boldsymbol{y}_{\sigma(1:n)}$ and $\hat{h}(\mathcal{U}^{-1}(\boldsymbol{z}_{\sigma(n+1:2n)})) = \hat{h}(\boldsymbol{z}_{\sigma(n+1:2n)})$. Therefore, $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$ and the predictor $\hat{h} \in \mathcal{H}$ are witnesses that satisfy $\forall 1 \leq i \leq 2n$:

$$z_{\sigma(i)} \in \mathcal{U}(x_{\sigma(i)}) \wedge \hat{h}(\mathcal{U}^{-1}(z_{\sigma(i)})) = \hat{h}(x_{\sigma(i)}).$$

Thus, by definition of $\Pi^{\mathcal{U}}_{\mathcal{H}}$, it holds that $\hat{h}(\boldsymbol{x}_{\sigma(1:2n)}) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n})$. This allows us to establish that the event $E_{\sigma,\boldsymbol{x}}$ implies the event that there exists a labeling $\hat{h}(\boldsymbol{x}_{\sigma(1:2n)}) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n})$ that achieves zero loss on the training examples $\mathrm{err}_{\boldsymbol{x}_{\sigma(1:n)},\boldsymbol{y}_{\sigma(1:n)}}(\hat{h}) = 0$, but makes error more than $\epsilon$ on the test examples $\mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon$. Specifically,

$$\Pr_{\sigma}\left[E_{\sigma,\boldsymbol{x}}\right] \leq \\ \Pr_{\sigma}\Bigg[\exists \hat{h} \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n}) : \mathrm{err}_{\boldsymbol{x}_{\sigma(1:n)},\boldsymbol{y}_{\sigma(1:n)}}(\hat{h}) = 0 \\ \wedge\, \mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon\Bigg] \\ \overset{(i)}{\leq} \left|\Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n})\right| 2^{\lceil -\epsilon n \rceil} \\ \overset{(ii)}{\leq} (2n)^{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})}\, 2^{\lceil -\epsilon n \rceil},$$

where inequality $(i)$ follows from applying a union bound over labelings $\hat{h} \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \dots, x_{2n})$, and observing that for any such fixed $\hat{h}$:

$$\Pr_{\sigma}\Bigg[\mathrm{err}_{\boldsymbol{x}_{\sigma(1:n)},\boldsymbol{y}_{\sigma(1:n)}}(\hat{h}) = 0 \\ \wedge\, \mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)},\boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h}) > \epsilon\Bigg] \leq 2^{-\lceil \epsilon n \rceil}.$$

To see this, suppose that $s = \sum_{i=1}^{2n} \mathbb{1}\{\hat{h}(x_i) \neq y_i\} \geq \lceil \epsilon n \rceil$ (otherwise, the probability of the event above is zero). Now, when sampling a random permutation $\sigma$, the chance that all of the mistakes fall into the test split is at most $2^{-s} \leq 2^{-\lceil \epsilon n \rceil}$. Because if we pair the $s$ mistakes and any $s$ out of the $2n - s$ non-mistakes while fixing the remaining non-mistakes to be in the training split, then the chance that all the $s$ mistakes appear in the test split is at most $2^{-s}$.

Finally, inequality *(ii)* follows from applying Sauer's lemma on our introduced relaxed notion of robust shattering dimension (Definition 1). Setting $(2n)^{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})} 2^{\lceil -\epsilon n \rceil} \leq \delta$ and solving for $\epsilon$ yields the stated bound. □

### 5.2 Agnostic Setting

*Proof.* (Proof of Theorem 2) Let $n \in \mathbb{N}$. For notational brevity, we write $\mathsf{OPT} = \mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$. We will assume that $\mathsf{OPT}$ (see Equation 3) is attained by some predictor $h^* \in \mathcal{H}$.[3] Let $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^n$, $(\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \sim \mathcal{D}^n$ be independent.

We will establish that with high probability over the drawings of $(\boldsymbol{x}, \boldsymbol{y}), (\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \sim \mathcal{D}^n$: for any adversarial perturbations $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$, $\tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}})$, and any predictor $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$:

1. $\hat{h}$ achieves low robust error on the training examples: $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}) \leq \mathsf{OPT} + \epsilon_0$.
2. $\hat{h}$ is robust (but not necessarily correct) on many of the test examples: $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}}) \leq \mathsf{OPT} + \epsilon_0$.

We will then combine properties (1) and (2) above with a standard guarantee from VC theory to show that $\hat{h}$ achieves low error on the test examples $\tilde{\boldsymbol{z}}$.

We now begin with showing (1) and (2). For the above fixed $h^*$, observe that by a standard Hoeffding bound, for $\epsilon_0 = \sqrt{\frac{\ln(2/\delta)}{2n}}$, it holds that

$$\Pr\Big[ \left(\mathrm{R}_{\mathcal{U}^{-1}(\mathcal{U})}(h^*; \boldsymbol{x}, \boldsymbol{y}) \leq \mathsf{OPT} + \epsilon_0\right) \\ \wedge \left(\mathrm{R}_{\mathcal{U}^{-1}(\mathcal{U})}(h^*; \tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}) \leq \mathsf{OPT} + \epsilon_0\right) \Big] \geq 1 - \delta.$$

By definition of $\mathrm{R}_{\mathcal{U}^{-1}(\mathcal{U})}$, this implies that

$$\Pr\Big[ (\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}) : \mathrm{R}_{\mathcal{U}^{-1}}(h^*; \boldsymbol{z}, \boldsymbol{y}) \leq \mathsf{OPT} + \epsilon_0) \\ \wedge (\forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}) : \mathrm{R}_{\mathcal{U}^{-1}}(h^*; \tilde{\boldsymbol{z}}) \leq \mathsf{OPT} + \epsilon_0) \Big] \geq 1 - \delta.$$

This implies that

$$\Pr\Big[ \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}) : \\ \min_{h \in \mathcal{H}} \max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(h; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(h; \tilde{\boldsymbol{z}})\right\} \leq \mathsf{OPT} + \epsilon_0 \Big] \\ \geq 1 - \delta.$$

By Equation 9, we have

$$\Pr\Big[ \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}}) : \\ \max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}})\right\} \leq \mathsf{OPT} + \epsilon_0 \Big] \\ \geq 1 - \delta.$$

**VC Guarantee** Next, to show that $\hat{h}$ achieves low error on the test examples $\tilde{\boldsymbol{z}}$, we will combine the properties above with a standard guarantee in the transductive setting from VC theory, which states that for $\epsilon = O\left(\sqrt{\frac{\mathrm{vc}(\mathcal{H}) + \log(1/\delta)}{n}}\right)$:

$$\Pr\left[\forall h \in \mathcal{H} : \left|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(h) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(h)\right| \leq \epsilon\right] \geq 1 - \delta.$$

Thus, for $\epsilon = O\left(\sqrt{\frac{\mathrm{vc}(\mathcal{H}) + \log(1/\delta)}{n}}\right)$:

$$\Pr\Big[ \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) : \\ \max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}})\right\} \leq \mathsf{OPT} + \epsilon \\ \wedge \left|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h})\right| \leq \epsilon \Big] \geq 1 - 2\delta.$$

Finally, observe that for any predictor $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}, \boldsymbol{y}, \tilde{\boldsymbol{z}})$ that satisfies $\max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}})\right\} \leq \mathsf{OPT} + \epsilon$ and $|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h})| \leq \epsilon$, we can deduce that:

- $\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) \leq \mathsf{OPT} + \epsilon$ (since $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}) \leq \mathsf{OPT} + \epsilon$ and $\boldsymbol{x} \in \mathcal{U}^{-1}(\boldsymbol{z})$). Therefore, $\mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h}) \leq \mathsf{OPT} + 2\epsilon$ (since $|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h})| \leq \epsilon$).
- Since $\mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h}) \leq \mathsf{OPT} + 2\epsilon$ and $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}) \leq \mathsf{OPT} + \epsilon$, this implies that $\mathrm{err}_{\boldsymbol{z},\tilde{\boldsymbol{y}}}(\hat{h}) \leq 2\mathsf{OPT} + 3\epsilon$.

**A refined bound** We will show that

$$\Pr\Big[ \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \forall \tilde{\boldsymbol{z}} \in \mathcal{U}(\tilde{\boldsymbol{x}}), \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) : \\ \max\left\{\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}, \boldsymbol{y}), \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}})\right\} \leq \mathsf{OPT} + \epsilon \\ \wedge \left|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h})\right| \leq \tilde{\epsilon} \Big] \geq 1 - \delta.$$

[3] Otherwise, we can always choose a predictor $h^* \in \mathcal{H}$ attaining $\mathsf{OPT} + \epsilon'$ for any small $\epsilon' > 0$.

for a different $\tilde{\epsilon}$ that scales with $\mathsf{OPT}$ and $\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})$ (instead of $\mathrm{vc}(\mathcal{H})$). To this end, it suffices to show that for any fixed $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathcal{D}^{2n}$:

$$\Pr_{\sigma}\Big[\forall \boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)}), \forall \hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)}) :$$
$$\max\Big\{\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}), \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}_{\sigma(n+1,2n)})\Big\} \leq$$
$$\mathsf{OPT} + \tilde{\epsilon} \wedge \Big|\mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) - \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h})\Big| \leq \tilde{\epsilon}\Big] \geq 1 - \delta,$$

where $\sigma$ is a permutation of $\{1, 2, 3, \ldots, 2n\}$ sampled uniformly at random.

We will show that for any permutation $\sigma$, any $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$, and any $\hat{h} \in \Delta^{\mathcal{U}}_{\mathcal{H}}(\boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}, \boldsymbol{z}_{\sigma(n+1:2n)})$ it holds that the labeling $\hat{h}(\boldsymbol{x}_{\sigma(1:2n)})$ is included in a finite set of possible behaviors $\Pi^{\mathcal{U}}_{\mathcal{H}}$ defined on the entire sequence $\boldsymbol{x} = (x_1, \ldots, x_{2n})$ by:

$$\Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \ldots, x_{2n}) =$$
$$\left\{(h(x_1), h(x_2), \ldots, h(x_{2n}))\middle|\substack{\exists I \subseteq [2n], |I| \geq (1-\mathsf{OPT}-\epsilon_0)2n, \\ \forall i \in I, \exists z_i \in \mathcal{U}(x_i), \\ \exists h \in \mathcal{H}: h(\mathcal{U}^{-1}(z_i)) = h(x_i) \forall i \in I}\right\}.$$

When it holds that $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}) \leq \mathsf{OPT} + \epsilon_0$ and $\mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \boldsymbol{z}_{\sigma(n+1:2n)}) \leq \mathsf{OPT} + \epsilon_0$, by definition of $\mathrm{R}_{\mathcal{U}^{-1}}$ (see Equation 6), it follows that the predictor $\hat{h} \in \mathcal{H}$ and $\boldsymbol{z}_{\sigma(1:2n)} \in \mathcal{U}(\boldsymbol{x}_{\sigma(1:2n)})$ are witnesses that satisfy $(\exists I \subseteq [2n], |I| \geq (1 - \mathsf{OPT} - \epsilon_0)2n)\,(\forall i \in I)$:

$$z_{\sigma(i)} \in \mathcal{U}(x_{\sigma(i)}) \wedge \hat{h}(\mathcal{U}^{-1}(z_{\sigma(i)})) = \hat{h}(x_{\sigma(i)}).$$

Thus, by definition of $\Pi^{\mathcal{U}}_{\mathcal{H}}$, it holds that $\hat{h}(\boldsymbol{x}_{\sigma(1:2n)}) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \ldots, x_{2n})$. Then, observe that for any $\tilde{\epsilon} > 0$

$$\Pr_{\sigma}\Big[\exists \hat{h} \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \ldots, x_{2n}) :$$
$$\Big|\mathrm{err}_{\boldsymbol{x}_{\sigma(1:n)}, \boldsymbol{y}_{\sigma(1:n)}}(\hat{h}) - \mathrm{err}_{\boldsymbol{x}_{\sigma(n+1:2n)}, \boldsymbol{y}_{\sigma(n+1:2n)}}(\hat{h})\Big| > \tilde{\epsilon}\Big]$$
$$\overset{(i)}{\leq} \left|\Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \ldots, x_{2n})\right| e^{-\tilde{\epsilon}^2 n}$$
$$\overset{(ii)}{\leq} \binom{2n}{\leq (\mathsf{OPT} + \epsilon_0)2n}\binom{(1 - \mathsf{OPT} - \epsilon_0)2n}{\leq \mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})} e^{-\tilde{\epsilon}^2 n}$$
$$\overset{(iii)}{\leq} 2^{H(\mathsf{OPT}+\epsilon_0)2n}\left((1 - \mathsf{OPT} - \epsilon_0)2n\right)^{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})} e^{-\tilde{\epsilon}^2 n},$$

where inequality $(i)$ follows from applying a union bound over labelings $\hat{h} \in \Pi^{\mathcal{U}}_{\mathcal{H}}(x_1, \ldots, x_{2n})$, and a standard Hoeffding bound. Inequality $(ii)$ follows from the definition of $\Pi^{\mathcal{U}}_{\mathcal{H}}$ and applying Sauer's lemma on our introduced relaxed notion of robust shattering dimension (Lemma 3). Inequality $(iii)$ follows from bounds on the binomial coefficients, where $H$ is the entropy function. Let $p = \mathsf{OPT} + \epsilon_0$. Setting $2^{H(p)2n}\left((1-p)2n\right)^{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})} e^{-\tilde{\epsilon}^2 n}$ less than $\frac{\delta}{2}$ and solving for $\tilde{\epsilon}$ yields:

$$\tilde{\epsilon} \leq \sqrt{2\ln(2)H(p) + \frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})\ln\left((1-p)2n\right) + \ln(1/\delta)}{n}}$$
$$\leq \sqrt{2\ln(2)H(p)} + \sqrt{\frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})\ln\left((1-p)2n\right) + \ln(1/\delta)}{n}}$$
$$\leq p + \sqrt{\frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})\ln\left(2n\right) + \ln(1/\delta)}{n}}$$

Combining both events from above, we get that $\mathrm{err}_{\tilde{\boldsymbol{z}},\tilde{\boldsymbol{y}}}(\hat{h}) \leq \mathrm{err}_{\tilde{\boldsymbol{x}},\tilde{\boldsymbol{y}}}(\hat{h}) + \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}}) \leq \mathrm{err}_{\boldsymbol{x},\boldsymbol{y}}(\hat{h}) + \tilde{\epsilon} + \mathrm{R}_{\mathcal{U}^{-1}}(\hat{h}; \tilde{\boldsymbol{z}}) \leq \mathsf{OPT} + \epsilon_0 + \tilde{\epsilon} + \mathsf{OPT} + \epsilon_0 = 2\mathsf{OPT} + 2\epsilon_0 + \tilde{\epsilon} \leq 3\mathsf{OPT} + 3\epsilon_0 + \sqrt{\frac{\mathrm{rdim}_{\mathcal{U}^{-1}}(\mathcal{H})\ln(2n) + \ln(1/\delta)}{n}}$. □

# 6 DISCUSSION

**Related Work** Adversarially robust learning has been mainly studied in the inductive setting (see e.g., Schmidt et al. (2018); Cullina et al. (2018); Khim and Loh (2018); Bubeck et al. (2019); Yin et al. (2019); Montasser et al. (2019). This includes studying what learning rules should be used for robust learning and how much training data is needed to guarantee low robust error.

In transductive learning, the learner is given unlabeled test examples to classify all at once or in batches, rather than individually (Vapnik, 1998). Without robustness guarantees, it is known that $\mathsf{ERM}$ is nearly minimax optimal in the transductive setting (Vapnik and Chervonenkis, 1974; Blumer et al., 1989; Tolstikhin and Lopez-Paz, 2016). In particular, additional unlabeled test data does not offer any help from a minimax perspective. More recently, (Goldwasser et al., 2020) gave a transductive learning algorithm that takes as input labeled training examples from a distribution $\mathcal{D}$ and *arbitrary* unlabeled test examples (chosen by an unbounded adversary, not necessarily according to perturbation set $\mathcal{U}$). For classes $\mathcal{H}$ of bounded VC dimension, their algorithm guarantees low error rate on the test examples but it might *abstain* from classifying some (or perhaps even all) of them. This is different from the guarantees we present in this work, where we restrict the adversary to choose from a perturbation set $\mathcal{U}$ but we do *not* abstain from classifying.

On the empirical side, Wu et al. (2020) recently proposed a method that leverages unlabeled test data for adversarial robustness in the context of deep neural networks. However, Chen et al. (2021) later proposed an empirical attack that breaks their defense. Furthermore, Chen et al. (2021) proposed another empirical transductive defense but with no theoretical guarantees. It would be interesting to empirically evaluate the adaptive attacks developed in (Chen et al.,

2021) against our proposed transductive learner. In semi-supervised learning, recent works (Alayrac et al., 2019; Carmon et al., 2019) have shown that (non-adversarial) unlabeled test data can improve adversarially robust generalization in practice, and there is also theoretical work quantifying the benefit of unlabeled data for robust generalization (Ashtiani et al., 2020).

**Open Problems** Can we design transductive learners that compete with $\mathsf{OPT}_{\mathcal{U}}$ instead of $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$? We note that this will likely require more sophistication, in the sense that we can construct classes $\mathcal{H}$ with $\mathrm{vc}(\mathcal{H}) = 1$ (similar construction to Montasser et al. (2019)) and distributions $\mathcal{D}$ where $\mathsf{OPT}_{\mathcal{U}} = 0$ but $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})} = 1$, and moreover, our simple transductive learner fails and finding a robust labeling on the test examples no longer suffices.

At the expense of competing with $\mathsf{OPT}_{\mathcal{U}^{-1}(\mathcal{U})}$, can we obtain stronger robust learning guarantees in the inductive setting, similar to the transductive guarantees established in this work? As we discussed in Section 4, we can not obtain such guarantees by directly reducing to the transductive problem, and we need improper learning because proper learning will not work.

### Acknowledgements

This work was done as part of the NSF/Simons sponsored Collaboration on the Theoretical Foundations of Deep Learning (deepfoundations.ai). This work was supported in part by DARPA under cooperative agreement HR00112020003.[4]

## References

Alayrac, J., Uesato, J., Huang, P., Fawzi, A., Stanforth, R., and Kohli, P. (2019). Are labels required for improving adversarial robustness? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 12192–12202.

Arora, S., Rao, S., and Vazirani, U. V. (2004). Expander flows, geometric embeddings and graph partitioning. In Babai, L., editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing, Chicago, IL, USA, June 13-16, 2004*, pages 222–231. ACM.

Ashtiani, H., Pathak, V., and Urner, R. (2020). Black-box certification and learning under adversarial perturbations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 388–398. PMLR.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. (1989). Learnability and the Vapnik-Chervonenkis dimension. *Journal of the Association for Computing Machinery*, 36(4):929–965.

Bubeck, S., Lee, Y. T., Price, E., and Razenshteyn, I. (2019). Adversarial examples from computational constraints. In *International Conference on Machine Learning*, pages 831–840.

Carmon, Y., Raghunathan, A., Schmidt, L., Duchi, J. C., and Liang, P. (2019). Unlabeled data improves adversarial robustness. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 11190–11201.

Chen, J., Guo, Y., Wu, X., Li, T., Lao, Q., Liang, Y., and Jha, S. (2021). Towards adversarial robustness via transductive learning. *CoRR*, abs/2106.08387.

Cullina, D., Bhagoji, A. N., and Mittal, P. (2018). Pac-learning in the presence of adversaries. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 228–239.

Goldwasser, S., Kalai, A. T., Kalai, Y., and Montasser, O. (2020). Beyond perturbations: Learning guarantees with arbitrary adversarial test examples. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.*

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.*

Haussler, D., Littlestone, N., and Warmuth, M. K. (1994). Predicting \0,1\-functions on randomly drawn points. *Inf. Comput.*, 115(2):248–292.

Khim, J. and Loh, P.-L. (2018). Adversarial risk bounds for binary classification via function transformation. *arXiv preprint arXiv:1810.09519.*

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations,*

[4] The views expressed in this work do not necessarily reflect the position or the policy of the Government and no official endorsement should be inferred. Approved for public release; distribution is unlimited.

*ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Montasser, O., Hanneke, S., and Srebro, N. (2019). Vc classes are adversarially robustly learnable, but only improperly. In Beygelzimer, A. and Hsu, D., editors, *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pages 2512–2530, Phoenix, USA. PMLR.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. (2018). Adversarially robust generalization requires more data. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 5019–5031.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge university press.

Tolstikhin, I. O. and Lopez-Paz, D. (2016). Minimax lower bounds for realizable transductive classification. *CoRR*, abs/1602.03027.

Vapnik, V. and Chervonenkis, A. (1974). *Theory of Pattern Recognition*. Nauka, Moscow.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley-Interscience.

Wu, Y., Yuan, C., and Wu, S. (2020). Adversarial robustness via runtime masking and cleansing. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 10399–10409. PMLR.

Yin, D., Ramchandran, K., and Bartlett, P. L. (2019). Rademacher complexity for adversarially robust generalization. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7085–7094. PMLR.

Zhang, H., Yu, Y., Jiao, J., Xing, E. P., Ghaoui, L. E., and Jordan, M. I. (2019). Theoretically principled trade-off between robustness and accuracy. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7472–7482. PMLR.

## A Proof of Lemma 3

*Proof.* The proof will follow a standard argument that is used to prove Sauer-Shela-Perles lemma (see for e.g., Shalev-Shwartz and Ben-David (2014)). Specifically, it suffices to prove the following stronger claim:

$$\left|\Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n)\right| \leq \left|\{S \subseteq \{z_1,\dots,z_n\} : S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right|. \tag{13}$$

This is because

$$\left|\{S \subseteq \{z_1,\dots,z_n\} : S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right| \leq \binom{n}{\leq \mathrm{rdim}_{\mathcal{U}}(\mathcal{H})}.$$

We will prove Equation 13 by induction on $n$. When $n = 1$, both sides of Equation 13 either evaluate to 1 or 2 (the empty set is always considered to be relaxed $\mathcal{U}$-robustly shattered by $\mathcal{H}$). When $n > 1$, assume that Equation 13 holds for sequences of length $k < n$. Let $C = \{z_1,\dots,z_n\}$ and $C' = \{z_2,\dots,z_n\}$. Consider the following two sets:

$$Y_0 = \left\{(y_2,\dots,y_n) : (+1, y_2,\dots,y_n) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n) \vee (-1, y_2,\dots,y_n) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n)\right\},$$

and

$$Y_1 = \left\{(y_2,\dots,y_n) : (+1, y_2,\dots,y_n) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n) \wedge (-1, y_2,\dots,y_n) \in \Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n)\right\}.$$

Observe that $\left|\Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n)\right| = |Y_0| + |Y_1|$. Additionally, note that by definition of $Y_0$, $Y_0 \subseteq \Pi^{\mathcal{U}}_{\mathcal{H}}(z_2,\dots,z_n)$. Thus, by the inductive assumption,

$$\begin{aligned}|Y_0| \leq \left|\Pi^{\mathcal{U}}_{\mathcal{H}}(z_2,\dots,z_n)\right| &\leq \left|\{S \subseteq C' : S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right| \\ &= \left|\{S \subseteq C : z_1 \notin S \wedge S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right|.\end{aligned}$$

Next, define $\mathcal{H}' \subseteq \mathcal{H}$ to be

$$\mathcal{H}' = \left\{h \in \mathcal{H} : \exists h' \in \mathcal{H}, \boldsymbol{x}_{2:n}, \tilde{\boldsymbol{x}}_{2:n} \in \mathcal{U}^{-1}(\boldsymbol{z}_{2:n}) \text{ s.t. } h(\mathcal{U}(x_1)) = -h'(\mathcal{U}(x_1)) \wedge h(\mathcal{U}(\boldsymbol{x}_{2:n})) = h'(\mathcal{U}(\tilde{\boldsymbol{x}}_{2:n}))\right\}.$$

Observe that if a set $S \subseteq C'$ is relaxed $\mathcal{U}$-robustly shattered by $\mathcal{H}'$, then $S \cup \{z_1\}$ is also relaxed $\mathcal{U}$-robustly shattered by $\mathcal{H}'$ and vice versa. Observe also that, by definition, $Y_1 = \Pi^{\mathcal{U}}_{\mathcal{H}'}(z_2,\dots,z_n)$. By applying the inductive assumption on $\mathcal{H}'$ and $C'$ we obtain that

$$\begin{aligned}|Y_1| = \left|\Pi^{\mathcal{U}}_{\mathcal{H}'}(z_2,\dots,z_n)\right| &\leq \left|\{S \subseteq C' : S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}'\}\right| \\ &= \left|\{S \subseteq C' : S \cup \{z_1\} \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}'\}\right| \\ &= \left|\{S \subseteq C : z_1 \in S \wedge S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}'\}\right| \\ &\leq \left|\{S \subseteq C : z_1 \in S \wedge S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right|.\end{aligned}$$

Overall, we have shown that

$$\begin{aligned}\left|\Pi^{\mathcal{U}}_{\mathcal{H}}(z_1,\dots,z_n)\right| &= |Y_0| + |Y_1| \\ &\leq \left|\{S \subseteq C : z_1 \notin S \wedge S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right| \\ &\qquad + \left|\{S \subseteq C : z_1 \in S \wedge S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right| \\ &= \left|\{S \subseteq \{z_1,\dots,z_n\} : S \text{ is relaxed } \mathcal{U}\text{-robustly shattered by } \mathcal{H}\}\right|,\end{aligned}$$

which concludes our proof. □