Efficiently Learning Adversarially Robust Halfspaces with Noise

Omar Montasser [1] **Surbhi Goel** [2] **Ilias Diakonikolas** [3] **Nathan Srebro** [1]

Abstract

We study the problem of learning adversarially robust halfspaces in the distribution-independent setting. In the realizable setting, we provide necessary and sufficient conditions on the adversarial perturbation sets under which halfspaces are efficiently robustly learnable. In the presence of random label noise, we give a simple computationally efficient algorithm for this problem with respect to any ℓ_p-perturbation.

1. Introduction

Learning predictors that are robust to adversarial examples remains a major challenge in machine learning. A line of work has shown that predictors learned by deep neural networks are *not* robust to adversarial examples (Szegedy et al., 2014; Biggio et al., 2013; Goodfellow et al., 2015). This has led to a long line of research studying different aspects of robustness to adversarial examples.

In this paper, we consider the problem of distribution-independent learning of halfspaces that are robust to adversarial examples at test time, also referred to as robust PAC learning of halfspaces. Halfspaces are binary predictors of the form $h_{\boldsymbol{w}}(\boldsymbol{x}) = \text{sign}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle)$, where $\boldsymbol{w} \in \mathbb{R}^d$.

In adversarially robust PAC learning, given an instance space $\mathcal{X}$ and label space $\mathcal{Y} = \{\pm 1\}$, we formalize an adversary – that we would like to be robust against – as a map $\mathcal{U} : \mathcal{X} \mapsto 2^{\mathcal{X}}$, where $\mathcal{U}(\boldsymbol{x}) \subseteq \mathcal{X}$ represents the set of perturbations (adversarial examples) that can be chosen by the adversary at test time (i.e., we require that $\boldsymbol{x} \in \mathcal{U}(\boldsymbol{x})$). For an unknown distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, we observe m i.i.d. samples $S \sim \mathcal{D}^m$, and our goal is to learn a predictor $\hat{h} : \mathcal{X} \mapsto \mathcal{Y}$ that achieves small robust risk,

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

[1]Toyota Technological Institute at Chicago [2]University of Texas at Austin [3]University of Wisconsin-Madison. Correspondence to: Omar Montasser <omar@ttic.edu>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

The information-theoretic aspects of adversarially robust learning have been studied in recent work, see, e.g., (Schmidt et al., 2018; Cullina et al., 2018; Khim & 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 high robust accuracy. It is now known that any hypothesis class $\mathcal{H}$ with finite VC dimension is robustly learnable, though sometimes improper learning is necessary and the sample complexity may be exponential in the VC dimension (Montasser et al., 2019).

On the other hand, the computational aspects of adversarially robust PAC learning are less understood. In this paper, we take a first step towards studying this broad algorithmic question with a focus on the fundamental problem of learning adversarially robust halfspaces.

A first question to ask is whether efficient PAC learning implies efficient *robust* PAC learning, i.e., whether there is a general reduction that solves the adversarially robust learning problem. Recent work has provided strong evidence that this is not the case. Specifically, (Bubeck et al., 2019) showed that there exists a learning problem that can be learned efficiently non-robustly, but is computationally intractable to learn robustly (under plausible complexity-theoretic assumptions). There is also more recent evidence that suggests that this is also the case in the PAC model. (Awasthi et al., 2019) showed that it is computationally intractable to even weakly robustly learn degree-2 polynomial threshold functions (PTFs) with ℓ_∞ perturbations in the realizable setting, while PTFs of any constant degree are known to be efficiently PAC learnable non-robustly in the realizable setting. (Gourdeau et al., 2019) showed that there are hypothesis classes that are hard to robustly PAC learn, under the assumption that it is hard to non-robustly PAC learn.

The aforementioned discussion suggests that when studying robust PAC learning, we need to characterize which types of perturbation sets $\mathcal{U}$ admit *computationally efficient* robust PAC learners and under which noise assumptions. In the agnostic PAC setting, it is known that even weak (non-robust) learning of halfspaces is computationally intractable (Feldman et al., 2006; Guruswami & Raghavendra, 2009; Diakonikolas et al., 2011; Daniely, 2016). For ℓ_2-

perturbations, where $\mathcal{U}(x) = \{z : \|x - z\|_2 \leq \gamma\}$, it was recently shown that the complexity of proper learning is exponential in $1/\gamma$ (Diakonikolas et al., 2019b). In this paper, we focus on the realizable case and the (more challenging) case of random label noise.

We can be more optimistic in the realizable setting. Halfspaces are efficiently PAC learnable non-robustly via Linear Programming (Maass & Turán, 1994), and under the margin assumption via the Perceptron algorithm (Rosenblatt, 1958). But what can we say about robustly PAC learning halfspaces? Given a perturbation set $\mathcal{U}$ and under the assumption that there is a halfspace $h_{\boldsymbol{w}}$ that robustly separates the data, can we efficiently learn a predictor with small robust risk?

Just as empirical risk minimization (ERM) is central for non-robust PAC learning, a core component of adversarially robust learning is minimizing the *robust* empirical risk on a dataset S,

$$\hat{h} \in \mathsf{RERM}_{\mathcal{U}}(S) \triangleq \operatorname*{argmin}_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^{m} \sup_{\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})} \mathbb{1}[h(\boldsymbol{z}) \neq y].$$

In this paper, we provide necessary and sufficient conditions on perturbation sets $\mathcal{U}$, under which the robust empirical risk minimization (RERM) problem is *efficiently* solvable in the realizable setting. We show that an efficient separation oracle for $\mathcal{U}$ yields an efficient solver for $\mathsf{RERM}_{\mathcal{U}}$, while an efficient *approximate* separation oracle for $\mathcal{U}$ is necessary for even computing the robust loss $\sup_{\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})} \mathbb{1}[h_{\boldsymbol{w}}(\boldsymbol{z}) \neq y]$ of a halfspace $h_{\boldsymbol{w}}$. In addition, we relax our realizability assumption and show that under random classification noise (Angluin & Laird, 1987), we can efficiently robustly PAC learn halfspaces with respect to any ℓ_p perturbation.

Main Contributions Our main contributions can be summarized as follows:

1. In the realizable setting, the class of halfspaces is efficiently robustly PAC learnable with respect to $\mathcal{U}$, given an efficient separation oracle for $\mathcal{U}$.
2. To even compute the robust risk with respect to $\mathcal{U}$ efficiently, an efficient *approximate* separation oracle for $\mathcal{U}$ is *necessary*.
3. In the random classification noise setting, the class of halfspaces is efficiently robustly PAC learnable with respect to any ℓ_p perturbation.

1.1. Related Work

Here we focus on the recent work that is most closely related to the results of this paper. (Awasthi et al., 2019) studied the tractability of RERM with respect to ℓ_∞ perturbations, obtaining efficient algorithms for halfspaces in the realizable setting, but showing that RERM for degree-2 polynomial threshold functions is computationally intractable (assuming NP $\neq$ RP). (Gourdeau et al., 2019) studied robust learnability of hypothesis classes defined over $\{0,1\}^n$ with respect to hamming distance, and showed that monotone conjunctions are robustly learnable when the adversary can perturb only $\mathcal{O}(\log n)$ bits, but are *not* robustly learnable even under the uniform distribution when the adversary can flip $\omega(\log n)$ bits.

In this work, we take a more general approach, and instead of considering specific perturbation sets, we provide methods in terms of oracle access to a separation oracle for the perturbation set $\mathcal{U}$, and aim to characterize which perturbation sets $\mathcal{U}$ admit tractable RERM.

In the non-realizable setting, the only prior work we are aware of is by (Diakonikolas et al., 2019b) who studied the complexity of robustly learning halfspaces in the agnostic setting under ℓ_2 perturbations.

2. Problem Setup

Let $\mathcal{X} = \mathbb{R}^d$ be the instance space and $\mathcal{Y} = \{\pm 1\}$ be the label space. We consider halfspaces $\mathcal{H} = \{\boldsymbol{x} \mapsto \operatorname{sign}(\langle \boldsymbol{w}, \boldsymbol{x} \rangle) : \boldsymbol{w} \in \mathbb{R}^d\}$.

The following definitions formalize the notion of adversarially robust PAC learning in the realizable and random classification noise settings:

Definition 2.1 (Realizable Robust PAC Learning)**.** *We say $\mathcal{H}$ is robustly PAC learnable with respect to an adversary $\mathcal{U}$ in the* realizable *setting, if there exists a learning algorithm $\mathcal{A} : (\mathcal{X} \times \mathcal{Y})^* \mapsto \mathcal{Y}^{\mathcal{X}}$ with sample complexity $m : (0,1) \to \mathbb{N}$ such that: for any $\epsilon, \delta \in (0,1)$, for every data distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ where there exists a predictor $h^* \in \mathcal{H}$ with zero robust risk, $\mathrm{R}_{\mathcal{U}}(h^*; \mathcal{D}) = 0$, with probability at least $1 - \delta$ over $S \sim \mathcal{D}^m$,*

$$\mathrm{R}_{\mathcal{U}}(\mathcal{A}(S); \mathcal{D}) \leq \epsilon.$$

Definition 2.2 (Robust PAC Learning with Random Classification Noise)**.** *Let $h^* \in \mathcal{H}$ be an unknown halfspace. Let $\mathcal{D}_{\boldsymbol{x}}$ be an arbitrary distribution over $\mathcal{X}$ such that $\mathrm{R}_{\mathcal{U}}(h^*; \mathcal{D}_{h^*}) = 0$, and $\eta \leq 0 < 1/2$. A noisy example oracle, $\mathrm{EX}(h^*, \mathcal{D}_{\boldsymbol{x}}, \eta)$ works as follows: Each time $\mathrm{EX}(h^*, \mathcal{D}_{\boldsymbol{x}}, \eta)$ is invoked, it returns a labeled example $(\boldsymbol{x}, y)$, where $\boldsymbol{x} \sim \mathcal{D}_{\boldsymbol{x}}$, $y = h^*(x)$ with probability $1 - \eta$ and $y = -h^*(x)$ with probability η. Let $\mathcal{D}$ be the joint distribution on $(\boldsymbol{x}, y)$ generated by the above oracle.*

We say $\mathcal{H}$ is robustly PAC learnable with respect to an adversary $\mathcal{U}$ in the random classification noise *model, if $\exists m(\epsilon, \delta, \eta) \in \mathbb{N} \cup \{0\}$ and a learning algorithm $\mathcal{A} : (\mathcal{X} \times \mathcal{Y})^* \mapsto \mathcal{Y}^{\mathcal{X}}$, such that for every distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ (generated as above by a noisy oracle), with probability at*

least $1-\delta$ *over* $S \sim \mathcal{D}^m$,

$$\mathrm{R}_{\mathcal{U}}(\mathcal{A}(S); \mathcal{D}) \leq \eta + \epsilon.$$

Sample Complexity of Robust Learning Denote by $\mathcal{L}^{\mathcal{U}}_{\mathcal{H}}$ the robust loss class of $\mathcal{H}$,

$$\mathcal{L}^{\mathcal{U}}_{\mathcal{H}} = \left\{ (x, y) \mapsto \sup_{z \in \mathcal{U}(x)} \mathbb{1}[h(z) \neq y] : h \in \mathcal{H} \right\}.$$

It was shown by (Cullina et al., 2018) that for any set $\mathcal{B} \subseteq \mathcal{X}$ that is nonempty, closed, convex, and origin-symmetric, and an adversary $\mathcal{U}$ that is defined as $\mathcal{U}(\boldsymbol{x}) = \boldsymbol{x} + \mathcal{B}$ (e.g., ℓ_p-balls), the VC dimension of the robust loss of halfspaces $\mathrm{vc}(\mathcal{L}^{\mathcal{U}}_{\mathcal{H}})$ is at most the standard VC dimension $\mathrm{vc}(\mathcal{H}) = d + 1$. Based on Vapnik's "General Learning" (Vapnik, 1982), this implies that we have uniform convergence of robust risk with $m = \mathcal{O}(\frac{d+\log(1/\delta)}{\epsilon^2})$ samples. Formally, for any $\epsilon, \delta \in (0, 1)$ and any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, with probability at least $1 - \delta$ over $S \sim \mathcal{D}^m$,

$$\forall \boldsymbol{w} \in \mathbb{R}^d, |\mathrm{R}_{\mathcal{U}}(h_{\boldsymbol{w}}; \mathcal{D}) - \mathrm{R}_{\mathcal{U}}(h_{\boldsymbol{w}}; S)| \leq \epsilon. \tag{2}$$

In particular, this implies that for any adversary $\mathcal{U}$ that satisfies the conditions above, $\mathcal{H}$ is robustly PAC learnable w.r.t. $\mathcal{U}$ by minimizing the *robust* empirical risk on S,

$$\mathsf{RERM}_{\mathcal{U}}(S) = \operatorname*{argmin}_{\boldsymbol{w} \in \mathbb{R}^d} \frac{1}{m} \sum_{i=1}^{m} \sup_{\boldsymbol{z}_i \in \mathcal{U}(\boldsymbol{x}_i)} \mathbb{1}[y_i \langle \boldsymbol{w}, \boldsymbol{z}_i \rangle \leq 0]. \tag{3}$$

Thus, it remains to efficiently solve the $\mathsf{RERM}_{\mathcal{U}}$ problem. We discuss necessary and sufficient conditions for solving $\mathsf{RERM}_{\mathcal{U}}$ in the following section.

3. The Realizable Setting

In this section, we show necessary and sufficient conditions for minimizing the robust empirical risk $\mathsf{RERM}_{\mathcal{U}}$ on a dataset $S = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_m, y_m)\} \in (\mathcal{X} \times \mathcal{Y})^m$ in the realizable setting, i.e. when the dataset S is *robustly* separable with a halfspace $h_{\boldsymbol{w}^*}$ where $\boldsymbol{w}^* \in \mathbb{R}^d$. In Theorem 3.5, we show that an efficient separation oracle for $\mathcal{U}$ yields an efficient solver for $\mathsf{RERM}_{\mathcal{U}}$. While in Theorem 3.10, we show that an efficient *approximate* separation oracle for $\mathcal{U}$ is necessary for even computing the robust loss $\sup_{\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})} \mathbb{1}[h_{\boldsymbol{w}}(\boldsymbol{z}) \neq y]$ of a halfspace $h_{\boldsymbol{w}}$.

Note that the set of allowed perturbations $\mathcal{U}$ can be non-convex, and so it might seem difficult to imagine being able to solve the $\mathsf{RERM}_{\mathcal{U}}$ problem in full generality. But, it turns out that for halfspaces it suffices to consider only convex perturbation sets due to the following observation:

Observation 3.1. *Given a halfspace* $\boldsymbol{w} \in \mathbb{R}^d$ *and an example* $(\boldsymbol{x}, y) \in \mathcal{X} \times \mathcal{Y}$. *If* $\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle > 0$, *then* $\forall z \in \mathsf{conv}(\mathcal{U}(\boldsymbol{x})), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle > 0$. *And if* $\exists z \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle \leq 0$, *then* $\exists z \in \mathsf{conv}(\mathcal{U}(\boldsymbol{x})), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle \leq 0$, *where* $\mathsf{conv}(\mathcal{U}(\boldsymbol{x}))$ *denotes the convex-hull of* $\mathcal{U}(\boldsymbol{x})$.

Observation 3.1 shows that for any dataset S that is *robustly* separable w.r.t. $\mathcal{U}$ with a halfspace $\boldsymbol{w}^*$, S is also robustly separable w.r.t. the convex hull $\mathsf{conv}(\mathcal{U})$ using the same halfspace $\boldsymbol{w}^*$, where $\mathsf{conv}(\mathcal{U})(x) = \mathsf{conv}(\mathcal{U}(x))$. Thus, in the remainder of this section we only consider perturbation sets $\mathcal{U}$ that are convex, i.e., for each $\boldsymbol{x}$, $\mathcal{U}(\boldsymbol{x})$ is convex.

Definition 3.2. *Denote by* $\mathsf{SEP}_{\mathcal{U}}$ *a separation oracle for* $\mathcal{U}$. $\mathsf{SEP}_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{z})$ *takes as input* $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$ *and either:*

- *asserts that* $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$, *or*
- *returns a separating hyperplane* $\boldsymbol{w} \in \mathbb{R}^d$ *such that* $\langle \boldsymbol{w}, \boldsymbol{z}' \rangle \leq \langle \boldsymbol{w}, \boldsymbol{z} \rangle$ *for all* $\boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})$.

Definition 3.3. *For any* $\eta > 0$, *denote by* $\mathsf{SEP}^{\eta}_{\mathcal{U}}$ *an* approximate *separation oracle for* $\mathcal{U}$. $\mathsf{SEP}^{\eta}_{\mathcal{U}}$ *takes as input* $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$ *and either:*

- *asserts that* $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})^{+\eta} \stackrel{\text{def}}{=} \{\boldsymbol{z} : \exists \boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})$ *s.t.* $\|\boldsymbol{z} - \boldsymbol{z}'\|_2 \leq \eta\}$, *or*
- *returns a separating hyperplane* $\boldsymbol{w} \in \mathbb{R}^d$ *such that* $\langle \boldsymbol{w}, \boldsymbol{z}' \rangle \leq \langle \boldsymbol{w}, \boldsymbol{z} \rangle + \eta$ *for all* $\boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})^{-\eta} \stackrel{\text{def}}{=} \{\boldsymbol{z}' : B(\boldsymbol{z}', \eta) \subseteq \mathcal{U}(\boldsymbol{x})\}$.

Definition 3.4. *Denote by* $\mathsf{MEM}_{\mathcal{U}}$ *a membership oracle for* $\mathcal{U}$. $\mathsf{MEM}_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{z})$ *takes as input* $\boldsymbol{x}, \boldsymbol{z} \in \mathcal{X}$ *and either:*

- *asserts that* $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$, *or*
- *asserts that* $\boldsymbol{z} \notin \mathcal{U}(\boldsymbol{x})$.

When discussing a separation or membership oracle for a fixed convex set K, we overload notation and write SEP_K, SEP^{η}_K, and MEM_K (in this case only one argument is required).

3.1. An efficient separation oracle for $\mathcal{U}$ is sufficient to solve $\mathsf{RERM}_{\mathcal{U}}$ efficiently

Let $\mathsf{Soln}^{\mathcal{U}}_S = \{\boldsymbol{w} \in \mathbb{R}^d : \forall (\boldsymbol{x}, y) \in S, \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle > 0\}$ denote the set of valid solutions for $\mathsf{RERM}_{\mathcal{U}}(S)$ (see Equation 3). Note that $\mathsf{Soln}^{\mathcal{U}}_S$ is not empty since we are considering the realizable setting. Although the treatment we present here is for homogeneous halfspaces (where a bias term is not needed), the results extend trivially to the non-homogeneous case.

Below, we show that we can efficiently find a solution $\boldsymbol{w} \in \mathsf{Soln}^{\mathcal{U}}_S$ given access to a separation oracle for $\mathcal{U}$, $\mathsf{SEP}_{\mathcal{U}}$.

Theorem 3.5. *Let* $\mathcal{U}$ *be an arbitrary convex adversary. Given access to a separation oracle for* $\mathcal{U}$, $\mathsf{SEP}_{\mathcal{U}}$ *that runs in time* $\mathsf{poly}(d, b)$. *There is an algorithm that finds* $\boldsymbol{w} \in \mathsf{Soln}^{\mathcal{U}}_S$

in $\mathsf{poly}(m, d, b)$ *time where* b *is an upper bound on the bit complexity of the valid solutions in* $\mathsf{Soln}_S^{\mathcal{U}}$ *and the examples and perturbations in* S.

Note that the polynomial dependence on b in the runtime is unavoidable even in standard non-robust ERM for halfspaces, unless we can solve linear programs in strongly polynomial time, which is currently an open problem.

Theorem 3.5 implies that for a broad family of perturbation sets $\mathcal{U}$, halfspaces $\mathcal{H}$ are efficiently robustly PAC learnable with respect to $\mathcal{U}$ in the realizable setting, as we show in the following corollary:

Corollary 3.6. *Let* $\mathcal{U} : \mathcal{X} \mapsto 2^{\mathcal{X}}$ *be an adversary such that* $\mathcal{U}(\boldsymbol{x}) = \boldsymbol{x} + \mathcal{B}$ *where* $\mathcal{B}$ *is nonempty, closed, convex, and origin-symmetric. Then, given access to an efficient separation oracle* $\mathsf{SEP}_{\mathcal{U}}$ *that runs in time* $\mathsf{poly}(d, b)$, $\mathcal{H}$ *is robustly PAC learnable w.r.t.* $\mathcal{U}$ *in the realizable setting in time* $\mathsf{poly}(d, b, 1/\epsilon, \log(1/\delta))$.

Proof. This follows from the uniform convergence guarantee for the robust risk of halfspaces (see Equation (2)) and Theorem 3.5. □

This covers many types of perturbation sets that are considered in practice. For example, $\mathcal{U}$ could be perturbations of distance at most γ w.r.t. some norm $\|\cdot\|$, such as the ℓ_∞ norm considered in many applications: $\mathcal{U}(\boldsymbol{x}) = \{\boldsymbol{z} \in \mathcal{X} : \|\boldsymbol{x} - \boldsymbol{z}\|_\infty \leq \gamma\}$. In addition, Theorem 3.5 also implies that we can solve the RERM problem for other natural perturbation sets such as translations and rotations in images (see, e.g., (Engstrom et al., 2019)), and perhaps mixtures of perturbations of different types (see, e.g., (Kang et al., 2019)), as long we have access to efficient separation oracles for these sets.

Benefits of handling general perturbation sets $\mathcal{U}$: One important implication of Theorem 3.5 that highlights the importance of having a treatment that considers general perturbation sets (and not just ℓ_p perturbations for example) is the following: for any efficiently computable feature map $\varphi : \mathbb{R}^r \to \mathbb{R}^d$, we can efficiently solve the robust empirical risk problem over the induced halfspaces $\mathcal{H}_\varphi = \left\{\boldsymbol{x} \mapsto \mathrm{sign}(\langle \boldsymbol{w}, \varphi(\boldsymbol{x})\rangle) : \boldsymbol{w} \in \mathbb{R}^d\right\}$, as long as we have access to an efficient separation oracle for the image of the perturbations $\varphi(\mathcal{U}(\boldsymbol{x}))$. Observe that in general $\varphi(\mathcal{U}(\boldsymbol{x}))$ maybe non-convex and complicated even if $\mathcal{U}(\boldsymbol{x})$ is convex, however Observation 3.1 combined with the realizability assumption imply that it suffices to have an efficient separation oracle for the convex-hull $\mathsf{conv}(\varphi(\mathcal{U}(\boldsymbol{x})))$.

Before we proceed with the proof of Theorem 3.5, we state the following requirements and guarantees for the Ellipsoid method which will be useful for us in the remainder of the section:

Lemma 3.7 (see, e.g., Theorem 2.4 in (Bubeck et al., 2015))**.** *Let* $K \subseteq \mathbb{R}^d$ *be a convex set, and* SEP_K *a separation oracle for* K. *Then, the Ellipsoid method using* $\mathcal{O}(d^2 b)$ *oracle queries to* SEP_K, *will find a* $\boldsymbol{w} \in K$, *or assert that* K *is empty. Furthermore, the total runtime is* $\mathcal{O}(d^4 b)$.

The proof of Theorem 3.5 relies on two key lemmas. First, we show that efficient robust certification yields an efficient solver for the RERM problem. Given a halfspace $\boldsymbol{w} \in \mathbb{R}^d$ and an example $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathcal{Y}$, efficient robust certification means that there is an algorithm that can *efficiently* either: (a) assert that $\boldsymbol{w}$ is robust on $\mathcal{U}(\boldsymbol{x})$, i.e. $\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z}\rangle > 0$, or (b) return a perturbation $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$ such that $y \langle \boldsymbol{w}, \boldsymbol{z}\rangle \leq 0$.

Lemma 3.8. *Let* $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$ *be a procedure that either: (a) Asserts that* $\boldsymbol{w}$ *is robust on* $\mathcal{U}(\boldsymbol{x})$, *i.e.,* $\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z}\rangle > 0$, *or (b) Finds a perturbation* $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$ *such that* $y \langle \boldsymbol{w}, \boldsymbol{z}\rangle \leq 0$. *If* $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, \boldsymbol{y}))$ *can be solved in* $\mathsf{poly}(d, b)$ *time, then there is an algorithm that finds* $\boldsymbol{w} \in \mathsf{Soln}_S^{\mathcal{U}}$ *in* $\mathsf{poly}(m, d, b)$ *time.*

Proof. Observe that $\mathsf{Soln}_S^{\mathcal{U}}$ is a convex set since

$$\begin{aligned}
\boldsymbol{w}_1, \boldsymbol{w}_2 \in \mathsf{Soln}_S^{\mathcal{U}} &\Rightarrow \forall (\boldsymbol{x}, y) \in S, \forall z \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}_1, \boldsymbol{z}\rangle > 0 \\
&\quad \text{and } y \langle \boldsymbol{w}_2, \boldsymbol{z}\rangle > 0 \\
&\Rightarrow \forall \alpha \in [0, 1], \forall (\boldsymbol{x}, y) \in S, \forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), \\
&\quad y \langle \alpha \boldsymbol{w}_1 + (1 - \alpha)\boldsymbol{w}_2, \boldsymbol{z}\rangle > 0 \\
&\Rightarrow \forall \alpha \in [0, 1], \alpha \boldsymbol{w}_1 + (1 - \alpha)\boldsymbol{w}_2 \in \mathsf{Soln}_S^{\mathcal{U}}.
\end{aligned}$$

Our goal is to find a $\boldsymbol{w} \in \mathsf{Soln}_S^{\mathcal{U}}$. Let $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$ be an efficient robust certifier that runs in $\mathsf{poly}(d, b)$ time. We will use $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$ to implement a separation oracle for $\mathsf{Soln}_S^{\mathcal{U}}$ denoted $\mathsf{SEP}_{\mathsf{Soln}_S^{\mathcal{U}}}$. Given a halfspace $\boldsymbol{w} \in \mathbb{R}^d$, we simply check if $\boldsymbol{w}$ is robustly correct on all datapoints by running $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}_i, y_i))$ on each $(\boldsymbol{x}_i, y_i) \in S$. If there is a point $(\boldsymbol{x}_i, y_i) \in S$ where $\boldsymbol{w}$ is not robustly correct, then we get a perturbation $\boldsymbol{z}_i \in \mathcal{U}(\boldsymbol{x}_i)$ where $y_i \langle \boldsymbol{w}, \boldsymbol{z}_i\rangle \leq 0$, and we return $-y_i \boldsymbol{z}_i$ as a separating hyperplane. Otherwise, we know that $\boldsymbol{w}$ is robustly correct on all datapoints, and we just assert that $\boldsymbol{w} \in \mathsf{Soln}_S^{\mathcal{U}}$.

Once we have a separation oracle $\mathsf{SEP}_{\mathsf{Soln}_S^{\mathcal{U}}}$, we can use the Ellipsoid method (see Lemma 3.7) to solve the $\mathsf{RERM}_{\mathcal{U}}(S)$ problem. More specifically, with a query complexity of $\mathcal{O}(d^2 b)$ to $\mathsf{SEP}_{\mathsf{Soln}_S^{\mathcal{U}}}$ and overall runtime of $\mathsf{poly}(m, d, b)$ (this depends on runtime of $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$), the Ellipsoid method will return a $\boldsymbol{w} \in \mathsf{Soln}_S^{\mathcal{U}}$. □

Next, we show that we can do efficient robust certification when given access to an efficient separation oracle for $\mathcal{U}$, $\mathsf{SEP}_{\mathcal{U}}$.

Lemma 3.9. *If we have an efficient separation oracle* $\mathsf{SEP}_{\mathcal{U}}$ *that runs in* $\mathrm{poly}(d, b)$ *time. Then, we can efficiently solve* $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$ *in* $\mathrm{poly}(d, b)$ *time.*

Proof. Given a halfspace $\boldsymbol{w} \in \mathbb{R}^d$ and $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathcal{Y}$, we want to either: (a) assert that $\boldsymbol{w}$ is robust on $\mathcal{U}(\boldsymbol{x})$, i.e. $\forall \boldsymbol{z} \in \mathcal{U}(\boldsymbol{x}), y \langle \boldsymbol{w}, \boldsymbol{z} \rangle > 0$, or (b) find a perturbation $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$ such that $y \langle \boldsymbol{w}, \boldsymbol{z} \rangle \leq 0$. Let $M(\boldsymbol{w}, y) = \{\boldsymbol{z}' \in \mathcal{X} : y \langle \boldsymbol{w}, \boldsymbol{z}' \rangle \leq 0\}$ be the set of all points that $\boldsymbol{w}$ mislabels. Observe that by definition $M(\boldsymbol{w}, y)$ is convex, and therefore $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$ is also convex. We argue that having an efficient separation oracle for $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$ suffices to solve our robust certification problem. Because if $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$ is not empty, then by definition, we can find a perturbation $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$ such that $y \langle \boldsymbol{w}, \boldsymbol{z} \rangle \leq 0$ with a separation oracle $\mathsf{SEP}_{\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)}$ and the Ellipsoid method (see Lemma 3.7). If $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$ is empty, then by definition, $\boldsymbol{w}$ is robustly correct on $\mathcal{U}(\boldsymbol{x})$, and the Ellipsoid method will terminate and assert that $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$ is empty.

Thus, it remains to implement $\mathsf{SEP}_{\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)}$. Given a point $\boldsymbol{z} \in \mathbb{R}^d$, we simply ask the separation oracle for $\mathcal{U}(\boldsymbol{x})$ by calling $\mathsf{SEP}_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{z})$ and the separation oracle for $M(\boldsymbol{w}, y)$ by checking if $y \langle \boldsymbol{w}, \boldsymbol{z} \rangle \leq 0$. If $\boldsymbol{z} \notin \mathcal{U}(\boldsymbol{x})$ the we get a separating hyperplane $\boldsymbol{c}$ from $\mathsf{SEP}_{\mathcal{U}}$ and we can use it separate $\boldsymbol{z}$ from $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$. Similarly, if $\boldsymbol{z} \notin M(\boldsymbol{w}, y)$, by definition, $\langle y\boldsymbol{w}, \boldsymbol{z} \rangle > 0$ and so we can use $y\boldsymbol{w}$ as a separating hyperplane to separate $\boldsymbol{z}$ from $\mathcal{U}(\boldsymbol{x}) \cap M(\boldsymbol{w}, y)$. The overall runtime of this separation oracle is $\mathrm{poly}(d, b)$, and so we can efficiently solve $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$ in $\mathrm{poly}(d, b)$ time using the Ellipsoid method (Lemma 3.7). □

We are now ready to proceed with the proof of Theorem 3.5.

Proof of Theorem 3.5. We want to efficiently solve $\mathsf{RERM}_{\mathcal{U}}(S)$. Given that we have a separation oracle for $\mathcal{U}$, $\mathsf{SEP}_{\mathcal{U}}$ that runs in $\mathrm{poly}(d, b)$. By Lemma 3.9, we get an efficient robust certification procedure $\mathsf{CERT}_{\mathcal{U}}(\boldsymbol{w}, (\boldsymbol{x}, y))$. Then, by Lemma 3.8, we get an efficient solver for $\mathsf{RERM}_{\mathcal{U}}$. In particular, the runtime complexity is $\mathrm{poly}(m, d, b)$. □

3.2. An efficient approximate separation oracle for $\mathcal{U}$ is necessary for computing the robust loss

Our efficient algorithm for $\mathsf{RERM}_{\mathcal{U}}$ requires a separation oracle for $\mathcal{U}$. We now show that even efficiently computing the robust loss of a halfspace $(\boldsymbol{w}, b_0) \in \mathbb{R}^d \times \mathbb{R}$ on an example $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathcal{Y}$ requires an efficient *approximate* separation oracle for $\mathcal{U}$.

Theorem 3.10. *Given a halfspace* $\boldsymbol{w} \in \mathbb{R}^d$ *and an example* $(\boldsymbol{x}, y) \in \mathbb{R}^d \times \mathcal{Y}$*, let* $\mathsf{EVAL}_{\mathcal{U}}((\boldsymbol{w}, b_0), (\boldsymbol{x}, y))$ *be a procedure that computes the robust loss* $\sup_{\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})} \mathbb{1}[y(\langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0) \leq 0]$ *in* $\mathrm{poly}(d, b)$ *time, then for any* $\gamma > 0$*, we can implement an efficient* γ*-approximate separation oracle* $\mathsf{SEP}^{\gamma}_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{z})$ *in* $\mathrm{poly}(d, b, \log(1/\gamma), \log(R))$ *time, where* $\mathcal{U}(\boldsymbol{x}) \subseteq B(0, R)$.

Proof. Let $\gamma > 0$. We will describe how to implement a γ-approximate separation oracle for $\mathcal{U}$ denoted $\mathsf{SEP}^{\gamma}_{\mathcal{U}}(\boldsymbol{x}, \boldsymbol{z})$. Fix the first argument to an arbitrary $\boldsymbol{x} \in \mathcal{X}$. Upon receiving a point $\boldsymbol{z} \in \mathcal{X}$ as input, the main strategy is to search for a halfspace $\boldsymbol{w} \in \mathbb{R}^d$ that can label all of $\mathcal{U}(\boldsymbol{x})$ with $+1$, and label the point $\boldsymbol{z}$ with -1. If $\boldsymbol{z} \notin \mathcal{U}(\boldsymbol{x})$ then there is a halfspace $\boldsymbol{w}$ that separates $\boldsymbol{z}$ from $\mathcal{U}(\boldsymbol{x})$ because $\mathcal{U}(\boldsymbol{x})$ is convex, but this is impossible if $\boldsymbol{z} \in \mathcal{U}(\boldsymbol{x})$. Since we are only concerned with implementing an *approximate* separation oracle, we will settle for a slight relaxation which is to either:

- assert that $\boldsymbol{z}$ is γ-close to $\mathcal{U}(\boldsymbol{x})$, i.e., $\boldsymbol{z} \in B(\mathcal{U}(\boldsymbol{x}), \gamma)$, or
- return a separating hyperplane $\boldsymbol{w}$ such that $\langle \boldsymbol{w}, \boldsymbol{z}' \rangle \leq \langle \boldsymbol{w}, \boldsymbol{z} \rangle$ for all $\boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})$.

Let $K = \{(\boldsymbol{w}, b_0) : \forall \boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x}), \langle \boldsymbol{w}, \boldsymbol{z}' \rangle + b_0 > 0\}$ denote the set of halfspaces that label all of $\mathcal{U}(\boldsymbol{x})$ with $+1$. Since $\mathcal{U}(\boldsymbol{x})$ is nonempty, it follows by definition that K is nonempty. To evaluate membership in K, given a query $\boldsymbol{w}_q, b_q$, we just make a call to $\mathsf{EVAL}_{\mathcal{U}}((\boldsymbol{w}_q, b_q), (\boldsymbol{x}, +))$. Let $\mathsf{MEM}_K(\boldsymbol{w}_q, b_q) = 1 - \mathsf{EVAL}_{\mathcal{U}}((\boldsymbol{w}_q, b_q), (\boldsymbol{x}, +))$. This can be efficiently computed in $\mathrm{poly}(d, b)$ time. Next, for any $\eta \in (0, 0.5)$, we can get an η-approximate separation oracle for K denoted SEP^{η}_{K} (see Definition 3.3) using $\mathcal{O}(db \log(d/\eta))$ queries to the membership oracle MEM_K described above (Lee et al., 2018). When queried with a halfspace $\tilde{\boldsymbol{w}} = (\boldsymbol{w}, b_0)$, SEP^{η}_{K} either:

- asserts that $\tilde{\boldsymbol{w}} \in K^{+\eta}$, or
- returns a separating hyperplane $\boldsymbol{c}$ such that $\langle \boldsymbol{c}, \tilde{\boldsymbol{w}}' \rangle \leq \langle \boldsymbol{c}, \tilde{\boldsymbol{w}} \rangle + \eta$ for all halfspaces $\tilde{\boldsymbol{w}}' \in K^{-\eta}$.

Observe that by definition, $K^{-\eta} \subseteq K \subseteq K^{+\eta}$. Furthermore, for any $\boldsymbol{w} \in K^{+\eta}$, by definition, $\exists \tilde{\boldsymbol{w}}' \in K$ such that $\|\tilde{\boldsymbol{w}} - \tilde{\boldsymbol{w}}'\|_2 \leq \eta$. Since, for each $\boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})$, by definition of K, we have $\langle \tilde{\boldsymbol{w}}', (\boldsymbol{z}', 1) \rangle = \langle \boldsymbol{w}', \boldsymbol{z}' \rangle + b_0 > 0$, it follows by Cauchy-Schwarz inequality that $(\forall \tilde{\boldsymbol{w}} \in K^{+\eta})\,(\forall \boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x}))$:

$$\langle \tilde{\boldsymbol{w}}, (\boldsymbol{z}', 1) \rangle = \langle \tilde{\boldsymbol{w}} - \tilde{\boldsymbol{w}}', \boldsymbol{z}' \rangle + \langle \tilde{\boldsymbol{w}}', (\boldsymbol{z}', 1) \rangle > -\eta 2R. \quad (4)$$

Let $\mathsf{SEP}^{\gamma/4R}_{K}$ be a $\frac{\gamma}{4R}$-approximate separation oracle for K. Observe that if the distance between $\boldsymbol{z}$ and $\mathcal{U}(\boldsymbol{x})$ is greater than γ, it follows that there is $(\boldsymbol{w}, b_0)$ such that:

$$\langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2 \text{ and } \langle \boldsymbol{w}, \boldsymbol{z}' \rangle + b_0 > 0\,(\forall \boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x}))\,.$$

By definition of K, this implies that $K \cap \{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$ is not empty,

which implies that the intersection $K^{+\frac{\gamma}{4R}} \cap \{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$ is nonempty. We also have the contrapositive, which is, if the intersection $K^{+\frac{\gamma}{4R}} \cap \{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$ is empty, then we know that $\boldsymbol{z} \in B(\mathcal{U}(\boldsymbol{x}), \gamma)$. To conclude the proof, we run the Ellipsoid method with the approximate separation oracle $\mathsf{SEP}_K^{\gamma/4R}$ to search over the restricted space $\{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$. Restricting the space is easily done because we will use the query point $\boldsymbol{z}$ as the separating hyperplane. Either the Ellipsoid method will find $(\boldsymbol{w}, b_0) \in K^{+\frac{\gamma}{4R}} \cap \{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$, in which case by Equation 4, $(\boldsymbol{w}, b_0)$ has the property that:

$$\langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\frac{\gamma}{2} \text{ and } \langle \boldsymbol{w}, \boldsymbol{z}' \rangle + b_0 > -\frac{\gamma}{2} \left(\forall \boldsymbol{z}' \in \mathcal{U}(\boldsymbol{x})\right),$$

and so we return $\boldsymbol{w}$ as a separating hyperplane between $\boldsymbol{z}$ and $\mathcal{U}(\boldsymbol{x})$. If the Ellipsoid terminates without finding any such $(\boldsymbol{w}, b_0)$, this implies that the intersection $K^{+\frac{\gamma}{4R}} \cap \{(\boldsymbol{w}, b_0) : \langle \boldsymbol{w}, \boldsymbol{z} \rangle + b_0 \leq -\gamma/2\}$ is empty, and therefore, by the contrapositive above, we assert that $\boldsymbol{z} \in B(\mathcal{U}(\boldsymbol{x}), \gamma)$. □

4. Random Classification Noise

In this section, we relax the realizability assumption to random classification noise (Angluin & Laird, 1987). We show that for any adversary $\mathcal{U}$ that represents perturbations of bounded norm (i.e., $\mathcal{U}(x) = x + \mathcal{B}$, where $\mathcal{B} = \left\{\delta \in \mathbb{R}^d : \|\delta\|_p \leq \gamma\right\}, p \in [1, \infty]$), the class of halfspaces $\mathcal{H}$ is efficiently robustly PAC learnable with respect to $\mathcal{U}$ in the random classification noise model.

Theorem 4.1. *Let $\mathcal{U} : \mathcal{X} \mapsto 2^{\mathcal{X}}$ be an adversary such that $\mathcal{U}(\boldsymbol{x}) = \boldsymbol{x} + \mathcal{B}$ where $\mathcal{B} = \left\{\delta \in \mathbb{R}^d : \|\delta\|_p \leq \gamma\right\}$ and $p \in [1, \infty]$. Then, $\mathcal{H}$ is robustly PAC learnable w.r.t $\mathcal{U}$ under random classification noise in time* $\mathsf{poly}(d, 1/\varepsilon, 1/\gamma, 1/(1-2\eta), \log(1/\delta))$.

The proof of Theorem 4.1 relies on the following key lemma. We show that the structure of the perturbations $\mathcal{B}$ allows us to relate the robust loss of a halfspace $\boldsymbol{w} \in \mathbb{R}^d$ with the γ-margin loss of $\boldsymbol{w}$. Before we state the lemma, recall that the dual norm of $\boldsymbol{w}$ denoted $\|\boldsymbol{w}\|_*$ is defined as $\sup\{\langle \boldsymbol{u}, \boldsymbol{w} \rangle : \|\boldsymbol{u}\| \leq 1\}$.

Lemma 4.2. *For any $\boldsymbol{w}, \boldsymbol{x} \in \mathbb{R}^d$ and any $y \in \mathcal{Y}$,*

$$\sup_{\delta \in \mathcal{B}} \mathbb{1}\{h_{\boldsymbol{w}}(\boldsymbol{x} + \delta) \neq y\} = \mathbb{1}\left\{y \left\langle \frac{\boldsymbol{w}}{\|\boldsymbol{w}\|_*}, \boldsymbol{x} \right\rangle \leq \gamma\right\}.$$

Proof. First observe that

$$\begin{aligned}\sup_{\delta \in \mathcal{B}} \mathbb{1}\{h_{\boldsymbol{w}}(\boldsymbol{x} + \delta) \neq y\} &= \sup_{\delta \in \mathcal{B}} \mathbb{1}\{y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle \leq 0\} \\ &= \mathbb{1}\left\{\inf_{\delta \in \mathcal{B}} y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle \leq 0\right\}.\end{aligned}$$

This holds because when $\inf_{\delta \in \mathcal{B}} y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle > 0$, by definition $\forall \delta \in \mathcal{B}, y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle > 0$, which implies that $\sup_{\delta \in \mathcal{B}} \mathbb{1}\{h_{\boldsymbol{w}}(\boldsymbol{x} + \delta) \neq y\} = 0$. For the other direction, when $\sup_{\delta \in \mathcal{B}} \mathbb{1}\{h_{\boldsymbol{w}}(\boldsymbol{x} + \delta) \neq y\} = 1$, by definition $\exists \delta \in \mathcal{B}$ such that $y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle \leq 0$, which implies that $\inf_{\delta \in \mathcal{B}} y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle \leq 0$. To conclude the proof, by definition of the set $\mathcal{B}$ and the dual norm $\|\cdot\|_*$, we have

$$\begin{aligned}\inf_{\delta \in \mathcal{B}} y \langle \boldsymbol{w}, \boldsymbol{x} + \delta \rangle &= y \langle \boldsymbol{w}, \boldsymbol{x} \rangle - \sup_{\delta \in \mathcal{B}} \langle -y\boldsymbol{w}, \delta \rangle \\ &= y \langle \boldsymbol{w}, \boldsymbol{x} \rangle - \|\boldsymbol{w}\|_* \gamma.\end{aligned}$$

□

Lemma 4.2 implies that for any distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, to solve the γ-robust learning problem

$$\operatorname*{argmin}_{w \in \mathbb{R}^d} \mathop{\mathbb{E}}_{(\boldsymbol{x}, y) \sim \mathcal{D}} \left[\sup_{\delta \in \mathcal{B}} \mathbb{1}\{h_{\boldsymbol{w}}(\boldsymbol{x} + \delta) \neq y\}\right], \tag{5}$$

it suffices to solve the γ-margin learning problem

$$\operatorname*{argmin}_{\|\boldsymbol{w}\|_* = 1} \mathop{\mathbb{E}}_{(\boldsymbol{x}, y) \sim \mathcal{D}} \left[\mathbb{1}\{y \langle \boldsymbol{w}, \boldsymbol{x} \rangle \leq \gamma\}\right]. \tag{6}$$

We will solve the γ-margin learning problem in Equation (6) in the random classification noise setting using an appropriately chosen convex surrogate loss. Our convex surrogate loss and its analysis build on a convex surrogate that appears in the appendix of (Diakonikolas et al., 2019a) for learning large ℓ_2-margin halfspaces under random classification noise w.r.t. the 0-1 loss. We note that the idea of using a convex surrogate to (non-robustly) learn large margin halfspaces in the presence of random classification noise is implicit in a number of prior works, starting with (Bylander, 1994).

Our robust setting is more challenging for the following reasons. First, we are not interested in only ensuring small 0-1 loss, but rather ensuring small γ-margin loss. Second, we want to be able to handle all ℓ_p norms, as opposed to just the ℓ_2 norm. As a result, our analysis is somewhat delicate.

Let

$$\phi(s) = \begin{cases} \lambda(1 - \frac{s}{\gamma}), & s > \gamma \\ (1 - \lambda)(1 - \frac{s}{\gamma}), & s \leq \gamma \end{cases}.$$

We will show that solving the following convex optimization problem:

$$\operatorname*{argmin}_{\|\boldsymbol{w}\|_* \leq 1} G_\lambda^\gamma(\boldsymbol{w}) \stackrel{\text{def}}{=} \mathop{\mathbb{E}}_{(\boldsymbol{x}, y) \sim \mathcal{D}} \left[\phi(y \langle \boldsymbol{w}, \boldsymbol{x} \rangle)\right], \tag{7}$$

where $\lambda = \frac{\varepsilon\gamma/2 + \eta}{1 + \varepsilon\gamma}$, suffices to solve the γ-margin learning problem in Equation (6). Intuitively, the idea here is that for $\lambda = 0$, the ϕ objective is exactly a scaled hinge loss, which gives a learning guarantee w.r.t to the γ-margin loss when

there is no noise ($\eta = 0$). When the noise $\eta > 0$, we slightly adjust the slopes, such that even correct prediction encounters a loss. The choice of the slope is based on λ which will depend on the noise rate η and the ε-suboptimality that is required for Equation (6).

We can solve Equation (7) with a standard first-order method through samples using Stochastic Mirror Descent, when the dual norm $\|\cdot\|_*$ is an ℓ_q-norm. We state the following properties of Mirror Descent we will require:

Lemma 4.3 (see, e.g., Theorem 6.1 in (Bubeck et al., 2015))**.** *Let $G(\boldsymbol{w}) \stackrel{\text{def}}{=} \mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\ell(\boldsymbol{w},(\boldsymbol{x},y))]$ be a convex function that is L-Lipschitz w.r.t. $\|\boldsymbol{w}\|_q$ where $q > 1$. Then, using the potential function $\psi(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|_q^2$, a suitable step-size η, and a sequence of iterates $\boldsymbol{w}^k$ computed by the following update:*

$$\boldsymbol{w}^{k+1} = \Pi^{\psi}_{B_q}\left(\nabla\psi^{-1}\left(\nabla\psi(\boldsymbol{w}^k) - \eta\boldsymbol{g}^k\right)\right) ,$$

Stochastic Mirror Descent with $\mathcal{O}(L^2/(q-1)\varepsilon^2)$ stochastic gradients $\boldsymbol{g}$ of G, will find an ε-suboptimal point $\hat{w}$ such that $\|\hat{\boldsymbol{w}}\|_q \le 1$ and $G(\hat{\boldsymbol{w}}) \le \inf_{\boldsymbol{w}} G(\boldsymbol{w}) + \varepsilon$.

Remark 4.4. *When $q = 1$, we will use the entropy potential function $\psi(\boldsymbol{w}) = \sum_{i=1}^d w_i \log w_i$. In this case, Stochastic Mirror Descent will require $\mathcal{O}(\frac{L^2 \log d}{\varepsilon^2})$ stochastic gradients.*

We are now ready to state our main result for this section:

Theorem 4.5. *Let $\mathcal{X} = \left\{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\|_p \le 1\right\}$. Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ such that there exists a halfspace $\boldsymbol{w}^* \in \mathbb{R}^d$ with $\mathbf{Pr}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[|\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle| > \gamma] = 1$ and y is generated by $h_{\boldsymbol{w}^*}(\boldsymbol{x}) := \mathrm{sign}(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)$ corrupted by RCN with noise rate $\eta < 1/2$. An application of Stochastic Mirror Descent on $G^\gamma_\lambda(\boldsymbol{w})$, returns, with high probability, a halfspace $\boldsymbol{w}$ where $\|\boldsymbol{w}\|_q \le 1$ with $\gamma/2$-robust misclassification error $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}[\mathbb{1}\{y\langle \boldsymbol{w}, \boldsymbol{x}\rangle \le \gamma/2\}] \le \eta + \varepsilon$ in $\mathsf{poly}(d, 1/\varepsilon, 1/\gamma, 1/(1-2\eta))$ time.*

With Theorem 4.5, the proof of Theorem 4.1 immediately follows.

Proof of Theorem 4.1. This follows from Lemma 4.2 and Theorem 4.5. □

Remark 4.6. *In Theorem 4.5, we get a $\gamma/2$-robustness guarantee assuming γ-robust halfspace $\boldsymbol{w}^*$ that is corrupted with random classification noise. This can be strengthened to get a guarantee of $(1-c)\gamma$-robustness for any constant $c > 0$.*

The rest of this section is devoted to the proof of Theorem 4.5. The high-level strategy is to show that an ε'-suboptimal solution to Equation (7) gives us an ε-suboptimal solution to Equation (6) (for a suitably chosen ε'). In Lemma 4.8, we bound from above the $\gamma/2$-margin loss in terms of our convex surrogate objective G^γ_λ, and in Lemma 4.9 we show that there are minimizers of our convex surrogate G^γ_λ such that it is sufficiently small. These are the two key lemmas that we will use to piece everything together.

For any $\boldsymbol{w}, \boldsymbol{x} \in \mathbb{R}^d$, consider the contribution of the objective G^γ_λ of $\boldsymbol{x}$, denoted by $G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})$. This is defined as $G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x}) = \mathbb{E}_{y\sim\mathcal{D}_y(\boldsymbol{x})}[\phi(y\langle \boldsymbol{w}, \boldsymbol{x}\rangle)] = \eta\phi(-z) + (1-\eta)\phi(z)$ where $z = h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle$. In the following lemma, we provide a decomposition of $G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})$ that will help us in proving Lemmas 4.8 and 4.9.

Lemma 4.7. *For any $\boldsymbol{w}, \boldsymbol{x} \in \mathbb{R}^d$, let $z = h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle$. Then, we have that:*

$$\begin{aligned} G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x}) = (\eta - \lambda)\left(\frac{z}{\gamma}\right) + \lambda + \eta - 2\lambda\eta \\ + \mathbb{1}\{-\gamma \le z \le \gamma\}(1-\eta)(1-2\lambda)\left(1 - \frac{z}{\gamma}\right) \\ + \mathbb{1}\{z < -\gamma\}(1-2\lambda)\left(1 - 2\eta - \frac{z}{\gamma}\right). \end{aligned}$$

Proof. Based on the definition of the surrogate loss, it suffices to consider three cases:

Case $z > \gamma$:

$$\begin{aligned} l_1(z) &\stackrel{\text{def}}{=} \eta\phi(-z) + (1-\eta)\phi(z) \\ &= \eta(1-\lambda)(1 + z/\gamma) + (1-\eta)\lambda(1 - z/\gamma) \\ &= (\eta(1-\lambda) - (1-\eta)\lambda)\left(\frac{z}{\gamma}\right) + (1-\eta)\lambda \\ &\quad + \eta(1-\lambda) \\ &= (\eta - \lambda)\left(\frac{z}{\gamma}\right) + \lambda + \eta - 2\lambda\eta. \end{aligned}$$

Case $-\gamma \le z \le \gamma$:

$$\begin{aligned} l_2(z) &\stackrel{\text{def}}{=} \eta\phi(-z) + (1-\eta)\phi(z) \\ &= \eta(1-\lambda)(1 + z/\gamma) \\ &\quad + (1-\eta)(1-\lambda)(1 - z/\gamma) \\ &= -(1-2\eta)(1-\lambda)\left(\frac{z}{\gamma}\right) + 1 - \lambda. \end{aligned}$$

Case $z < -\gamma$:

$$\begin{aligned} l_3(z) &\stackrel{\text{def}}{=} \eta\lambda(1 + z/\gamma) + (1-\eta)(1-\lambda)(1 - z/\gamma) \\ &= (\eta\lambda - (1-\eta)(1-\lambda))\left(\frac{z}{\gamma}\right) + \eta\lambda \\ &\quad + (1-\eta)(1-\lambda) \\ &= (\eta + \lambda - 1)\left(\frac{z}{\gamma}\right) + 1 - \eta - \lambda + 2\eta\lambda. \end{aligned}$$

Considering the three cases above, we can write

$$G(z) = l_1(z) + \mathbb{1}\{-\gamma \le z \le \gamma\}\left(l_2(z) - l_1(z)\right) + \mathbb{1}\{z < -\gamma\}\left(l_3(z) - l_1(z)\right) .$$

Then, we calculate $l_2(z) - l_1(z)$ and $l_3(z) - l_1(z)$ as follows:

$$\begin{aligned} l_2(z) - l_1(z) &= -(1-2\eta)(1-\lambda)\left(\frac{z}{\gamma}\right) + 1 - \lambda \\ &\quad - (\eta - \lambda)\left(\frac{z}{\gamma}\right) - \lambda - \eta + 2\lambda\eta \\ &= -((1-2\eta)(1-\lambda) + \eta - \lambda)\left(\frac{z}{\gamma}\right) \\ &\quad + (1-\eta)(1-2\lambda) \\ &= (1-\eta)(1-2\lambda)\left(1 - \frac{z}{\gamma}\right) , \text{ and} \\ l_3(z) - l_1(z) &= (\eta + \lambda - 1)\left(\frac{z}{\gamma}\right) + 1 - \eta - \lambda + 2\eta\lambda \\ &\quad - (\eta - \lambda)\left(\frac{z}{\gamma}\right) - \lambda - \eta + 2\lambda\eta \\ &= (2\lambda - 1)\left(\frac{z}{\gamma}\right) + (1-2\eta)(1-2\lambda) \\ &= (2\lambda - 1)\left(\frac{z}{\gamma} + 2\eta - 1\right) . \end{aligned}$$

Using the above, we have that

$$G(z) = l_1(z) + \mathbb{1}\{-\gamma \le z \le \gamma\}(1-\eta)(1-2\lambda)\left(1 - \frac{z}{\gamma}\right) + \mathbb{1}\{z < -\gamma\}(1-2\lambda)\left(1 - 2\eta - \frac{z}{\gamma}\right) .$$

□

The following lemma allows us to bound from below our convex surrogate $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})]$ in terms of the $\gamma/2$-margin loss of $\boldsymbol{w}$.

Lemma 4.8. *Assume that λ is chosen such that $\lambda < 1/2$ and $\eta < \lambda$. Then, for any $\boldsymbol{w} \in \mathbb{R}^d$, $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})] \ge \frac{\eta-\lambda}{\gamma} + \frac{1}{2}(1-2\lambda)(1-\eta)\,\mathbb{E}_x\left[\mathbb{1}\left\{z \le \frac{\gamma}{2}\right\}\right] + \lambda + \eta - 2\lambda\eta$.*

Proof. By Lemma 4.7, and linearity of expectation, we have that

$$\begin{aligned} &\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})] = (\eta - \lambda) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\frac{z}{\gamma}\right] + \lambda + \eta - 2\lambda\eta \\ &+ (1-\eta)(1-2\lambda) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{-\gamma \le z \le \gamma\}\left(1 - \frac{z}{\gamma}\right)\right] \\ &+ (1-2\lambda) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{z < -\gamma\}\left(1 - 2\eta - \frac{z}{\gamma}\right)\right] . \end{aligned}$$

First, observe that for any $\boldsymbol{x}$, $-1 \le z = h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle \le 1$ and since $\eta \le \lambda$, we have

$$(\eta - \lambda) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\frac{z}{\gamma}\right] \ge \frac{\eta - \lambda}{\gamma} .$$

Then we observe that whenever $z < -\gamma$, $1 - \frac{z}{\gamma} - 2\eta > 2(1-\eta) > (1-\eta)/2$ and $\lambda \le 1/2$, thus we can bound from below the third term

$$(1-2\lambda) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{z < -\gamma\}\left(1 - 2\eta - \frac{z}{\gamma}\right)\right] \ge \frac{1}{2}(1-2\lambda)(1-\eta) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{z < -\gamma\}\right] .$$

Next we note that whenever $-\gamma \le z \le \gamma$, $1 - \frac{z}{\gamma} \ge 0$. This implies that instead of considering $\mathbb{1}\{-\gamma \le z \le \gamma\}$, we can relax this and consider the subset $\mathbb{1}\left\{-\gamma \le z \le \frac{\gamma}{2}\right\}$, and on this subset $1 - \frac{z}{\gamma} \ge 1/2$. Thus, we can bound the second term from below as follows:

$$(1-2\lambda)(1-\eta) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{-\gamma \le z \le \gamma\}\left(1 - \frac{z}{\gamma}\right)\right] \ge \frac{1}{2}(1-2\lambda)(1-\eta) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\left\{-\gamma \le z \le \frac{\gamma}{2}\right\}\right] .$$

Combining the above, we obtain

$$\begin{aligned} &\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})] \ge \frac{\eta - \lambda}{\gamma} \\ &+ \frac{1}{2}(1-2\lambda)(1-\eta)\,\mathop{\mathbb{E}}_{x}\left[\mathbb{1}\left\{-\gamma \le z \le \frac{\gamma}{2}\right\} + \mathbb{1}\{z < -\gamma\}\right] \\ &+ \lambda + \eta - 2\lambda\eta \\ &\ge \frac{\eta - \lambda}{\gamma} + \frac{1}{2}(1-2\lambda)(1-\eta)\,\mathop{\mathbb{E}}_{x}\left[\mathbb{1}\left\{z \le \frac{\gamma}{2}\right\}\right] \\ &+ \lambda + \eta - 2\lambda\eta , \end{aligned}$$

as desired. □

We now show that there exist minimizers of the convex surrogate G^γ_λ such that it is sufficiently small, which will be useful later in choosing the suboptimality parameter ϵ'.

Lemma 4.9. *Assume that λ is chosen such that $\lambda < 1/2$ and $\eta < \lambda$. Then we have that*

$$\inf_{\boldsymbol{w}\in\mathbb{R}^d} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})] \le 2\eta(1-\lambda).$$

Proof. By definition, we have that $\inf_{\boldsymbol{w}\in\mathbb{R}^d} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}, \boldsymbol{x})] \le \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}[G^\gamma_\lambda(\boldsymbol{w}^*, \boldsymbol{x})]$. By assumption, with probability 1 over $\boldsymbol{x} \sim \mathcal{D}_{\boldsymbol{x}}$, we have $h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle > \gamma$. Thus, by Lemma 4.7, we have

$$\begin{aligned} \mathop{\mathbb{E}}_{\boldsymbol{x}}[G^\gamma_\lambda(\boldsymbol{w}^*, \boldsymbol{x})] &= (\eta - \lambda)\left(\frac{z}{\gamma}\right) + \lambda + \eta - 2\lambda\eta \\ &\le \eta - \lambda + \lambda + \eta - 2\lambda\eta \\ &= 2\eta(1-\lambda) , \end{aligned}$$

where the last inequality follows from the fact that $\eta < \lambda$ and the fact that $\mathbb{E}_{\boldsymbol{x}}\left[\frac{z}{\gamma}\right] > 1$. □

Using the above lemmas, we are now able to bound from above the $\gamma/2$-margin loss of a halfspace $\boldsymbol{w}$ that is ε'-suboptimal for our convex optimization problem (see Equation (7)).

Lemma 4.10. *For any $\varepsilon' \in (0,1)$ and any $\boldsymbol{w} \in \mathbb{R}^d$ such that $\mathbb{E}_{x\sim\mathcal{D}_x}[G^{\gamma}_{\lambda}(\boldsymbol{w},\boldsymbol{x})] \leq \mathbb{E}_{x\sim\mathcal{D}_x}[G^{\gamma}_{\lambda}(\boldsymbol{w}^*,\boldsymbol{x})] + \varepsilon'$, the $\gamma/2$-missclassification error of $\boldsymbol{w}$ satisfies*

$$\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathbb{1}\{y\langle \boldsymbol{w},\boldsymbol{x}\rangle \leq \gamma/2\}\right] \leq \eta + \frac{2}{(1-2\lambda)}\left(\varepsilon' + (\lambda-\eta)\left(\frac{1}{\gamma}-1\right)\right) .$$

Proof. By Lemma 4.8 and Lemma 4.9, we have

$$\frac{\eta-\lambda}{\gamma} + \frac{1}{2}(1-2\lambda)(1-\eta)\mathop{\mathbb{E}}_{x}\left[\mathbb{1}\left\{z \leq \frac{\gamma}{2}\right\}\right] + \lambda + \eta - 2\lambda\eta \leq 2\eta(1-\lambda) + \varepsilon'.$$

This implies

$$(1-\eta)\mathop{\mathbb{E}}_{x}\left[\mathbb{1}\left\{z \leq \frac{\gamma}{2}\right\}\right] \leq \frac{2}{(1-2\lambda)}\left(\varepsilon' + (\lambda-\eta)\left(\frac{1}{\gamma}-1\right)\right)$$

Since $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathbb{1}\{y\langle \boldsymbol{w},\boldsymbol{x}\rangle \leq \gamma/2\}\right] \leq \eta + (1-\eta)\,\mathbb{E}_x\left[\mathbb{1}\left\{z \leq \frac{\gamma}{2}\right\}\right]$, we get the desired result. □

We are now ready to prove Theorem 4.5.

Proof of Theorem 4.5. Based on Lemma 4.10, we will choose λ, ε' such that

$$\frac{2}{(1-2\lambda)}\left(\varepsilon' + (\lambda-\eta)\left(\frac{1}{\gamma}-1\right)\right) \leq \varepsilon .$$

By setting $\varepsilon' = \lambda - \eta$, this condition reduces to

$$\frac{2(\lambda-\eta)}{(1-2\lambda)} \leq \gamma\varepsilon .$$

This implies that we need $\lambda \leq \frac{\varepsilon\gamma/2+\eta}{1+\varepsilon\gamma}$. We will choose $\lambda = \frac{\varepsilon\gamma/2+\eta}{1+\varepsilon\gamma}$. Note that our analysis relied on having $\lambda \leq 1/2$ and $\eta \leq \lambda$. These conditions combined imply that we should choose λ such that $\eta \leq \lambda \leq 1/2$. Our choice of $\lambda = \frac{\varepsilon\gamma/2+\eta}{1+\varepsilon\gamma}$ satisfies these conditions, since

$$\frac{\varepsilon\gamma/2+\eta}{1+\varepsilon\gamma} - \eta = \frac{\varepsilon\gamma/2+\eta-\eta(1+\varepsilon\gamma)}{1+\varepsilon\gamma} = \frac{\varepsilon\gamma(1/2-\eta)}{1+\varepsilon\gamma} \geq 0 ,$$

and

$$\frac{\varepsilon\gamma/2+\eta}{1+\varepsilon\gamma} - \frac{1}{2} = \frac{\varepsilon\gamma/2+\eta-1/2(1+\varepsilon\gamma)}{1+\varepsilon\gamma} = \frac{\eta-1/2}{1+\varepsilon\gamma} \leq 0 .$$

By our choice of λ, we have that $\varepsilon' = \lambda - \eta = \frac{\varepsilon\gamma(1/2-\eta)}{1+\varepsilon\gamma} = \frac{\varepsilon\gamma(1-2\eta)}{2(1+\varepsilon\gamma)}$. By the guarantees of Stochastic Mirror Descent (see Lemma 4.3), our theorem follows with $\mathcal{O}\left(\frac{1}{\varepsilon^2\gamma^2(1-2\eta)^2(q-1)}\right)$ samples for $q > 1$ and $\mathcal{O}\left(\frac{\log d}{\varepsilon^2\gamma^2(1-2\eta)^2}\right)$ samples for $q = 1$. □

5. Conclusion

In this paper, we provide necessary and sufficient conditions for perturbation sets $\mathcal{U}$, under which we can efficiently solve the robust empirical risk minimization (RERM) problem. We give a polynomial time algorithm to solve RERM given access to a polynomial time separation oracle for $\mathcal{U}$. In addition, we show that an efficient *approximate* separation oracle for $\mathcal{U}$ is necessary for even computing the robust loss of a halfspace. As a corollary, we show that halfspaces are efficiently robustly PAC learnable for a broad range of perturbation sets. By relaxing the realizability assumption, we show that under random classification noise, we can efficiently robustly PAC learn halfspaces with respect to any ℓ_p perturbations. An interesting direction for future work is to understand the computational complexity of robustly PAC learning halfspaces under stronger noise models, including Massart noise and agnostic noise.

Acknowledgments

We thank Sepideh Mahabadi for many insightful and helpful discussions, and Brian Bullins for helpful discussions on Mirror Descent. We also thank the anonymous reviewers for their helpful feedback. This work was initiated when the authors were visiting the Simons Institute for the Theory of Computing as part of the Summer 2019 program on *Foundations of Deep Learning*. Work by N.S. and O.M. was partially supported by NSF award IIS-1546500 and by DARPA[1] cooperative agreement HR00112020003. I.D. was supported by NSF Award CCF-1652862 (CAREER), a Sloan Research Fellowship, and a DARPA Learning with Less Labels (LwLL) grant. S.G. was supported by the JP Morgan AI Research PhD Fellowship.

Appendix: Another Approach to Large Margin Learning under Random Classification Noise

We remark that γ-margin learning of halfspaces has been studied in earlier work, and we have algorithms such as Margin Perceptron (Balcan et al., 2008) and SVM. The Margin Perceptron (for ℓ_2 margin) and other ℓ_p margin algorithms have been also implemented in the SQ model

[1]This paper does not reflect the position or the policy of the Government, and no endorsement should be inferred.

(Feldman et al., 2017). But no explicit connection has been made to adversarial robustness.

We present here a simple approach to learn γ-margin halfspaces under random classification noise using only a convex surrogate loss and Stochastic Mirror Descent. The construction of the convex surrogate is based on learning generalized linear models with a suitable link function $u : \mathbb{R} \to \mathbb{R}$. To the best of our knowledge, the result of this section is not explicit in prior work.

Theorem 5.1. *Let $\mathcal{X} = \left\{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\|_p \leq 1\right\}$. Let $\mathcal{D}$ be a distribution over $\mathcal{X} \times \mathcal{Y}$ such that there exists a halfspace $\boldsymbol{w}^* \in \mathbb{R}^d, \|\boldsymbol{w}^*\|_q = 1$ with $\mathbf{Pr}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[|\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle| > \gamma\right] = 1$ and y is generated by $h_{\boldsymbol{w}^*}(\boldsymbol{x}) := \mathrm{sign}(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)$ corrupted with random classification noise rate $\eta < 1/2$. Then, running Stochastic Mirror Descent on the following convex optimization problem:*

$$\min_{\boldsymbol{w}\in\mathbb{R}^d, \|\boldsymbol{w}\|_q \leq 1} \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}} \left[\ell(\boldsymbol{w}, (\boldsymbol{x}, y))\right]$$

where the convex loss function ℓ is defined in Equation 8, returns with high probability, a halfspace $\boldsymbol{w}$ with $\gamma/2$-robust misclassification error $\mathbb{E}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathbb{1}\{y\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}\right] \leq \eta + \varepsilon$.

We prove Theorem 4.5 in the remainder of this section. We will connect our problem to that of solving generalized linear models. We define the link function as follows,

$$u(s) = \begin{cases} \eta & s < -\gamma \\ \frac{1-2\eta}{2\gamma}s + \frac{1}{2} & -\gamma \leq s \leq \gamma \\ 1-\eta & s > \gamma \end{cases}.$$

Observe that u is monotone and $\frac{1-2\eta}{2\gamma}$-Lipschitz.

First, we will relate our loss of interest, which is the $\gamma/2$-margin loss with the squared loss defined in terms of the link function u,

Lemma 5.2. *For any $\boldsymbol{w} \in \mathbb{R}^d$,*

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}\right] \leq \frac{16}{(1-2\eta)^2} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - u(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)\right)^2\right].$$

Proof. Let $E^+ = \{h_{\boldsymbol{w}^*}(\boldsymbol{x}) = +\}$ and $E^- = \{h_{\boldsymbol{w}^*}(\boldsymbol{x}) = -\}$. Let $U(\boldsymbol{x}) = \left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - u(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)\right)^2$. By law of total expectation and definition of the link function u, we have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[U(\boldsymbol{x})\right] = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[U(\boldsymbol{x})\mathbb{1}\left\{E^+\right\}\right] + \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[U(\boldsymbol{x})\mathbb{1}\left\{E^-\right\}\right] \\
&= \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - (1-\eta)\right)^2 \mathbb{1}\left\{E^+\right\}\right] \\
&+ \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - \eta\right)^2 \mathbb{1}\left\{E^-\right\}\right].
\end{aligned}$$

We will lower bound both terms:

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - (1-\eta)\right)^2 \mathbb{1}\left\{E^+\right\}\right] \\
&\geq \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(a - (1-\eta)\right)^2 \mathbb{1}\left\{E^+\right\}\mathbb{1}\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \leq a\}\right], \\
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - \eta\right)^2 \mathbb{1}\left\{E^-\right\}\right] \\
&\geq \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(b - \eta\right)^2 \mathbb{1}\left\{E^-\right\}\mathbb{1}\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \geq b\}\right].
\end{aligned}$$

Then, observe that the event $\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}$ implies the event $\left\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \leq \frac{3-2\eta}{4}\right\}$, and similarly the event $\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \geq -\gamma/2\}$ implies the event $\left\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \geq \frac{2\eta+1}{4}\right\}$. This means that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(\frac{3-2\eta}{4} - (1-\eta)\right)^2 \mathbb{1}\left\{E^+\right\}\mathbb{1}\left\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \leq \frac{3-2\eta}{4}\right\}\right] \\
&\geq \left(\frac{3-2\eta}{4} - (1-\eta)\right)^2 \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\left\{E^+\right\}\mathbb{1}\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}\right], \text{ and} \\
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(\frac{2\eta+1}{4} - \eta\right)^2 \mathbb{1}\left\{E^-\right\}\mathbb{1}\left\{u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) \geq \frac{2\eta+1}{4}\right\}\right] \\
&\geq \left(\frac{2\eta+1}{4} - \eta\right)^2 \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\left\{E^-\right\}\mathbb{1}\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \geq -\gamma/2\}\right].
\end{aligned}$$

We combine these observations to conclude the proof,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - u(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)\right)^2\right] \geq \frac{(1-2\eta)^2}{16} \times \\
&\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}(E^+)\mathbb{1}\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\} + \mathbb{1}(E^-)\mathbb{1}\{\langle \boldsymbol{w}, \boldsymbol{x}\rangle \geq -\gamma/2\}\right] \\
&\geq \frac{(1-2\eta)^2}{16} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}\right].
\end{aligned}$$

□

But note that the squared loss is non-convex and so it may not be easy to optimize. Luckily, we can get a tight upper-bound with the following surrogate loss (see (Kanade, 2018)):

$$\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) = \int_0^{\langle \boldsymbol{w}, \boldsymbol{x}\rangle} (u(s) - y)ds. \tag{8}$$

Note that $\ell(\boldsymbol{w}, (\boldsymbol{x}, y))$ is convex w.r.t $\boldsymbol{w}$ since the Hessian $\nabla^2_{\boldsymbol{w}}\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) = u'(\langle \boldsymbol{w}, \boldsymbol{x}\rangle)\boldsymbol{x}\boldsymbol{x}^T$ is positive semi-definite.

Assuming our labels y have been transformed to $\{0, 1\}$ from $\{\pm 1\}$, observe that $\mathbb{E}[y|x] = u(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)$. We now have the following guarantee (see e.g., (Cohen, 2014; Kanade, 2018)) for $U(\boldsymbol{x}) = \left(u(\langle \boldsymbol{w}, \boldsymbol{x}\rangle) - u(\langle \boldsymbol{w}^*, \boldsymbol{x}\rangle)\right)^2$:

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[U(\boldsymbol{x})\right] \leq \frac{1-2\eta}{\gamma} \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) - \ell(\boldsymbol{w}^*, (\boldsymbol{x}, y))\right]. \tag{9}$$

Proof of Theorem 5.1. Combining Lemma 5.2 and Equation 9, we get the following guarantee for any $\boldsymbol{w} \in \mathbb{R}^d$,

$$(1-2\eta) \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\langle \boldsymbol{w}, \boldsymbol{x}\rangle \leq \gamma/2\}\right] \leq \frac{16}{\gamma} \mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\ell(\boldsymbol{w}, (\boldsymbol{x}, y)) - \ell(\boldsymbol{w}^*, (\boldsymbol{x}, y))\right].$$

Thus, running Stochastic Mirror Descent with $\varepsilon' = (\varepsilon\gamma(1-2\eta))/16$ and $O(1/\varepsilon'^2)$ samples (labels transformed to $\{0,1\}$), returns with high probability, a halfspace $\boldsymbol{w}$ such that

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] \le \varepsilon. \tag{10}$$

Then, to conclude the proof, observe that

$$\begin{aligned}
&\mathop{\mathbb{E}}_{(\boldsymbol{x},y)\sim\mathcal{D}}\left[\mathbb{1}\{y\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] = \eta \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{-h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] \\
&+ (1-\eta)\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] \\
&= \eta(1-\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le -\gamma/2\}\right]) \\
&+ (1-\eta)\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] \\
&\overset{(i)}{\le} \eta + (1-\eta)\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le \gamma/2\}\right] \\
&\overset{(ii)}{\le} \eta + \varepsilon,
\end{aligned}$$

where (i) follows from that fact that $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_{\boldsymbol{x}}}\left[\mathbb{1}\{h_{\boldsymbol{w}^*}(\boldsymbol{x})\left\langle \boldsymbol{w},\boldsymbol{x}\right\rangle \le -\gamma/2\}\right] \ge 0$, and (ii) follows from Equation 10. □

References

Angluin, D. and Laird, P. D. Learning from noisy examples. *Mach. Learn.*, 2(4):343–370, 1987. doi: 10.1007/BF00116829. URL https://doi.org/10.1007/BF00116829.

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

Balcan, M., Blum, A., and Srebro, N. A theory of learning with similarity functions. *Machine Learning*, 72(1-2):89–112, 2008. doi: 10.1007/s10994-008-5059-5. URL https://doi.org/10.1007/s10994-008-5059-5.

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

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

Bubeck, S. et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Bylander, T. Learning linear threshold functions in the presence of classification noise. In Warmuth, M. K. (ed.), *Proceedings of the Seventh Annual ACM Conference on Computational Learning Theory, COLT 1994, New Brunswick, NJ, USA, July 12-15, 1994*, pp. 340–347. ACM, 1994. doi: 10.1145/180139.181176. URL https://doi.org/10.1145/180139.181176.

Cohen, A. *Surrogate Loss Minimization*. PhD thesis, Hebrew University of Jerusalem, 2014.

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

Daniely, A. Complexity theoretic limitations on learning halfspaces. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 105–117, 2016.

Diakonikolas, I., O'Donnell, R., Servedio, R. A., and Wu, Y. Hardness results for agnostically learning low-degree polynomial threshold functions. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011*, pp. 1590–1606. SIAM, 2011.

Diakonikolas, I., Gouleakis, T., and Tzamos, C. Distribution-independent pac learning of halfspaces with massart noise. In *Advances in Neural Information Processing Systems*, pp. 4751–4762, 2019a.

Diakonikolas, I., Kane, D., and Manurangsi, P. Nearly tight bounds for robust proper learning of halfspaces with a margin. In *Advances in Neural Information Processing Systems*, pp. 10473–10484, 2019b.

Engstrom, L., Tran, B., Tsipras, D., Schmidt, L., and Madry, A. Exploring the landscape of spatial robustness. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 1802–1811, Long Beach, California, USA, 09–15 Jun 2019. PMLR. URL http://proceedings.mlr.press/v97/engstrom19a.html.

Feldman, V., Gopalan, P., Khot, S., and Ponnuswami, A. K. New results for learning noisy parities and halfspaces. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pp. 563–574. IEEE, 2006.

Feldman, V., Guzman, C., and Vempala, S. S. Statistical query algorithms for mean vector estimation and stochastic convex optimization. In Klein, P. N. (ed.), *Proceedings*

of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19, pp. 1265–1277. SIAM, 2017. doi: 10.1137/1.9781611974782.82. URL https://doi.org/10.1137/1.9781611974782.82.

Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6572.

Gourdeau, P., Kanade, V., Kwiatkowska, M., and Worrell, J. On the hardness of robust classification. In *Advances in Neural Information Processing Systems*, pp. 7444–7453, 2019.

Guruswami, V. and Raghavendra, P. Hardness of learning halfspaces with noise. *SIAM Journal on Computing*, 39 (2):742–765, 2009.

Kanade, V. Computational learning theory notes - 8 : Learning real-valued functions, 2018.

Kang, D., Sun, Y., Hendrycks, D., Brown, T., and Steinhardt, J. Testing robustness against unforeseen adversaries. *CoRR*, abs/1908.08016, 2019. URL http://arxiv.org/abs/1908.08016.

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

Lee, Y. T., Sidford, A., and Vempala, S. S. Efficient convex optimization with membership oracles. In *Conference On Learning Theory*, pp. 1292–1294, 2018.

Maass, W. and Turán, G. How fast can a threshold gate learn? In *Proceedings of a workshop on Computational learning theory and natural learning systems (vol. 1): constraints and prospects: constraints and prospects*, pp. 381–414, 1994.

Montasser, O., Hanneke, S., and Srebro, N. Vc classes are adversarially robustly learnable, but only improperly. In Beygelzimer, A. and Hsu, D. (eds.), *Proceedings of the Thirty-Second Conference on Learning Theory*, volume 99 of *Proceedings of Machine Learning Research*, pp. 2512–2530, Phoenix, USA, 25–28 Jun 2019. PMLR.

Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

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

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. Intriguing properties of neural networks. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6199.

Vapnik, V. *Estimation of Dependencies Based on Empirical Data*. Springer-Verlag, New York, 1982.

Yin, D., Ramchandran, K., and Bartlett, P. L. Rademacher complexity for adversarially robust generalization. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *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*, pp. 7085–7094. PMLR, 2019. URL http://proceedings.mlr.press/v97/yin19b.html.