Journal of Machine Learning Research 23 (2022) 1-24 Submitted 8/20; Revised 2/22; Published 3/22

When Hardness of Approximation Meets Hardness of Learning

Eran Malach ERAN.MALACH@MAIL.HUJI.AC.IL
School of Computer Science
The Hebrew University
Jerusalem, Israel

Shai Shalev-Shwartz SHAIS@CS.HUJI.AC.IL
School of Computer Science
The Hebrew University
Jerusalem, Israel

Editor: Gabor Lugosi

Abstract

A supervised learning algorithm has access to a distribution of labeled examples, and needs to return a function (hypothesis) that correctly labels the examples. The hypothesis of the learner is taken from some fixed class of functions (e.g., linear classifiers, neural networks etc.). A failure of the learning algorithm can occur due to two possible reasons: wrong choice of hypothesis class (hardness of *approximation*), or failure to find the best function within the hypothesis class (hardness of *learning*). Although both approximation and learnability are important for the success of the algorithm, they are typically studied separately. In this work, we show a single hardness property that implies both hardness of approximation using linear classes and shallow networks, and hardness of learning using correlation queries and gradient-descent. This allows us to obtain new results on hardness of approximation and learnability of parity functions, DNF formulas and AC^0 circuits.

Keywords: Hardness of learning, approximation, statistical-queries, gradient-descent, neural networks

1. Introduction

Given a distribution $\mathcal{D}$ over an instance space $\mathcal{X}$ and a target classification function $f : \mathcal{X} \to \{\pm 1\}$, let $f(\mathcal{D})$ be the distribution over $\mathcal{X} \times \{\pm 1\}$ obtained by sampling $x \sim \mathcal{D}$ and labeling it by $f(x)$. A learning algorithm, ALG, has access to the distribution $f(\mathcal{D})$ via an oracle, ORACLE$(f, \mathcal{D})$, and should output a hypothesis $h : \mathcal{X} \to \mathbb{R}$. The quality of h is assessed by the expected loss function:

$$L_{f(\mathcal{D})}(h) := \mathop{\mathbb{E}}_{(x,y)\sim f(\mathcal{D})}[\ell(h(x), y)] \ ,$$

where $\ell : \mathbb{R} \times \{\pm 1\} \to \mathbb{R}_+$ is some loss function. We say that the learning is ϵ-successful if ALG returns a function ALG$(f, \mathcal{D})$ such that:

$$\mathbb{E}[L_{f(\mathcal{D})}(\text{ALG}(f, \mathcal{D}))] \leq \epsilon \ ,$$

where the expectation is with respect to the randomness of the learning process. Of course, due to the well known *no-free-lunch* theorem, we cannot expect any single algorithm to succeed in this objective for all choices of $(f, \mathcal{D})$. So, we must make some assumptions on the nature of the labeled distributions observed by the algorithm. We denote by $\mathcal{A}$ the *distribution family* (or *assumption class*, as in Kearns et al. (1994)), which is some set of pairs $(f, \mathcal{D})$, where f is some function from $\mathcal{X}$ to $\{\pm 1\}$ and $\mathcal{D}$ is some distribution over $\mathcal{X}$. We say that **ALG is ϵ-successful** on a distribution family $\mathcal{A}$, if it ϵ-succeeds on every $(f, \mathcal{D}) \in \mathcal{A}$, namely:

$$\max_{(f,\mathcal{D})\in\mathcal{A}} \mathbb{E}\left[L_{f(\mathcal{D})}(\mathrm{ALG}(f,\mathcal{D}))\right] \leq \epsilon$$

The standard approach for understanding whether some algorithm is successful is using a *decomposition of error*. Let $\mathcal{H}$ be the class of functions that ALG can return (the *hypothesis class*), and note that:

$$\begin{aligned}
&\max_{(f,\mathcal{D})\in\mathcal{A}} \mathbb{E}\left[L_{f(\mathcal{D})}\left(\mathrm{ALG}(f,\mathcal{D})\right)\right] \\
&\leq \underbrace{\max_{(f,\mathcal{D})\in\mathcal{A}} \min_{h\in\mathcal{H}} L_{f(\mathcal{D})}(h)}_{\textbf{approximation} \text{ error}} + \underbrace{\max_{(f,\mathcal{D})\in\mathcal{A}} \mathbb{E}\left[L_{f(\mathcal{D})}(\mathrm{ALG}(f,\mathcal{D}))\right] - \min_{h\in\mathcal{H}} L_{f(\mathcal{D})}(h)}_{\textbf{learning} \text{ error}}
\end{aligned}$$

Similarly, it is easy to verify that

$$\max\left\{\textbf{approximation} \text{ error} \ , \ \textbf{learning} \text{ error}\right\} \ \leq \ \max_{(f,\mathcal{D})\in\mathcal{A}} \mathbb{E}\left[L_{f(\mathcal{D})}\left(\mathrm{ALG}(f,\mathcal{D})\right)\right]$$

Therefore, a sufficient condition for ALG to be ϵ-successful is that both the approximation error and learning error are at most $\epsilon/2$, and a necessary condition for ALG to be ϵ-successful is that both the approximation error and learning error are at most ϵ.

In general, we say that ALG *ϵ-learns* $\mathcal{A}$ if it achieves a learning error of at most ϵ (i.e., returns a hypothesis that is ϵ-competitive with the best hypothesis in $\mathcal{H}$), and we say that $\mathcal{H}$ *ϵ-approximates* $\mathcal{A}$ if $\mathcal{H}$ has an approximation error of at most ϵ on $\mathcal{A}$. The above inequalities show that in order for ALG to be successful, it must ϵ-learns $\mathcal{A}$ and at the same time, its hypothesis class must ϵ-approximate $\mathcal{A}$. However, the problems of *learnability* and *approximation* were typically studied separately in the literature of learning theory.

On the problem of *learnability*, there is rich literature covering possibility and impossibility results in various settings of learning. These settings can be recovered by different choices of $\mathcal{A} = \mathcal{F} \times \mathcal{P}$, where $\mathcal{F}$ is some class of Boolean[1] functions and $\mathcal{P}$ is some class of distributions over $\mathcal{X}$. The *realizable* setting is given by assuming $\mathcal{F} \subseteq \mathcal{H}$, and the *agnostic* setting is when $\mathcal{F}$ is the class of all Boolean functions. In *distribution-free* learning $\mathcal{P}$ is the class of all distributions, while *distribution-specific* learning assumes $\mathcal{P} = \{\mathcal{D}\}$, for some fixed distribution $\mathcal{D}$. The literature of learning theory also considers various choices for the oracle $\mathrm{ORACLE}(f, \mathcal{D})$. The most common choice is the examples oracle, which gives the learner access to random examples sampled i.i.d. from $f(\mathcal{D})$. Other oracles calculate statistical queries (estimating $\mathbb{E}_{(\mathbf{x},y)\sim f(\mathcal{D})}\, \psi(\mathbf{x}, y)$), membership queries (querying for labels of specific examples $\mathbf{x} \in \mathcal{X}$) or return gradient estimations.

1. In the paper, we consider Boolean expressions over $\{\pm 1\}$ instead of $\{0, 1\}$, where -1 is interpreted as 0.

The question of *approximation* has recently received a lot of attention in the machine learning community, with a growing number of works studying the limitations of linear classes, kernel methods and shallow neural networks, in terms of approximation capacity (see Section 2.1). These results often offer a separation property, showing that one hypothesis class (e.g., deep neural networks) is superior over another (e.g., shallow neural networks or linear classes). However, these questions are typically studied separately from the question of learnability.

In this work, we study the questions of learnability and approximation of general distribution families in a unified framework. We show that a single property, which we call the *variance* of the distribution family, can be used for showing both *hardness of approximation* and *hardnes of learning* results. Specifically, we show: 1) hardness of approximating $\mathcal{A}$ using any linear class with respect to a convex loss, 2) hardness of approximating some induced function using a shallow (depth-two) neural network with respect to a convex Lipschitz loss, 3) hardness of learning $\mathcal{A}$ using correlation queries and 4) hardness of learning $\mathcal{A}$ using gradient-descent.

Applying our general results to some specific choices of $\mathcal{A}$, we establish various novel results, in different settings of PAC learning:

1. Parities are hard to approximate using linear classes, under the uniform distribution.
2. The function $F(\mathbf{x}, \mathbf{z}) = \prod_i (x_i \vee z_i)$ is hard to approximate using depth-two neural networks, under the uniform distribution.
3. DNFs are hard to approximate using linear classes (*distribution-free*).
4. The function $F(\mathbf{x}, \mathbf{z}) = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{dm^2} (x_{ij} \wedge z_{ij})$ is hard to approximate using depth-two networks, for some fixed distribution.
5. Learning DNFs with correlation queries requires $2^{\Omega(n^{1/3})}$ queries (*distribution-free*).
6. Learning DNFs with noisy gradient-descent (GD) requires $2^{\Omega(n^{1/3})}$ gradient steps (*distribution-free*).

We note that the approximation in 1-4 is with respect to a wide range of convex loss functions, and 6 is shown with respect to the hinge-loss.

These results expand our understanding of learnability and approximation of various classes and algorithms. Note that despite the fact that our setting is somewhat different than traditional PAC learning, our results apply in the standard settings of PAC learning (either *distribution-specific* or *distribution-free* PAC learning). Importantly, they all follow from a single "hardness" property of the family $\mathcal{A}$. We believe this framework can be used to better understand the power and limitations of various learning algorithms.

2. Related Work

In this section, we overview different studies covering results on the approximation capacity and learnability of various classes and algorithms used in machine learning. We focus on works that are directly related to the results shown in this paper.

2.1 Hardness of Approximation

Linear Classes The problem of understanding the expressive power of a given class of functions has attracted great interest in the learning theory and theoretical computer science community over the years. A primary class of functions that has been extensively studied in the context of machine learning is the class of linear functions over a fixed embedding (for example, in Ben-David et al. (2002); Forster and Simon (2006); Sherstov (2008); Razborov and Sherstov (2010)). Notions such as margin complexity, dimension complexity and sign-rank were introduced in order to bound the minimal dimension and norm required to exactly express a class of functions using linear separators.

However, since the goal of the learner is to approximate a target function (e.g., PAC learning), and not to compute it exactly, showing hardness of exact expressivity often seems irrelevant from a machine learning perspective. Several recent studies show hardness of approximation results on linear classes (Allen-Zhu and Li, 2019, 2020; Yehudai and Shamir, 2019; Daniely and Malach, 2020). These works demonstrate a separation between linear methods and neural networks: namely, they show families of functions that are hard to approximate using linear classes, but are learnable using neural networks. A recent work by Kamath et al. (2020) gives probabilistic variants of the dimension and margin complexity in order to show hardness results on approximation using linear classes and kernel methods. We show new results on hardness of approximation using linear classes, extending prior results and techniques.

Neural Networks Another line of research that has gained a lot of attention in recent years focuses on the limitation of shallow neural networks, in terms of approximation power. The empirical success of deep neural networks sparked many questions regarding the advantage of using deep networks over shallow ones. The works of Eldan and Shamir (2016) and Daniely (2017) show examples of real valued functions that are hard to efficiently approximate using depth two (one hidden-layer) networks, but can be expressed using three layer networks, establishing a separation between these two classes of functions. The work of Martens et al. (2013) shows an example of a binary function (namely, the inner-product mod-2 function), that is hard to express using depth two networks. Other works study cases where the input dimension is fixed, and show an exponential gap in the expressive power between networks of growing depth (Delalleau and Bengio, 2011; Pascanu et al., 2013; Telgarsky, 2015, 2016; Cohen et al., 2016; Raghu et al., 2017; Montúfar, 2017; Serra et al., 2018; Hanin and Rolnick, 2019; Malach and Shalev-Shwartz, 2019). A recent work by Vardi and Shamir (2020) explores the relation between depth-separation results for neural networks, and hardness results in circuit complexity. We derive new results on hardness of approximation using shallow networks, which follow from the general hardness property introduced in the paper.

2.2 Computational Hardness

Computational Complexity Since the early works in learning theory, understanding which problems can be learned efficiently (i.e., in polynomial time) has been a key question in the field. Various classes of interest, such as DNFs, boolean formulas, decision trees, boolean circuits and neural networks are known to be computationally hard to learn, in different

settings of learning (see e.g., Kearns (1998)). In the case of DNF formulas, the best known algorithm for learning DNFs, due to Klivans and Servedio (2004), runs in time $2^{\tilde{O}(n^{1/3})}$. A work by Daniely and Shalev-Shwartz (2016) shows that DNFs are computationally hard to learn, by reduction from the random variant of the K-SAT problem. However, this work does not yet establish a computational lower bound that matches the upper bound in Klivans and Servedio (2004), and assumes some non-standard hardness assumption. Using our framework we establish a lower bound of $2^{\Omega(n^{1/3})}$, for some restricted family of algorithms, on the complexity of learning DNF formulas.

Statistical Queries In the general setting of PAC learning, the learner gets a sample of examples from the distribution, which is used in order to find a good hypothesis. Statistical Query (SQ) learning is a restriction of PAC learning, where instead of using a set of examples, the learner has access to estimates of statistical computations over the distribution, that are accurate up to some tolerance. This framework can be used to analyze a rich family of algorithms, showing hardness results on learning various problems from statistical-queries. Specifically, it was shown that parities, DNF formulas and decision trees cannot be learned efficiently using statistical-queries, under the uniform distribution (Kearns, 1998; Blum et al., 1994, 2003; Goel et al., 2020). In the case of DNF formulas, the number of queries required to learn this concept class under the uniform distribution is quasi-polynomial in the dimension (i.e., $n^{O(\log n)}$, and see Blum et al. (1994)). However, we are unaware of any work showing SQ lower-bounds on learning DNFs under general distributions, as we do in this work.

An interesting variant of statistical-query algorithms is algorithms that use only correlation statistical-queries (CSQ), namely — queries of the form $\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}\left[\phi(\mathbf{x})y\right]$ (Feldman, 2008). While in the distribution-specific setting, the CSQ and the SQ models are equivalent (Bshouty and Feldman, 2002), in the distribution-independent setting this is not the case. It has been shown that conjunctions are hard to *strongly* learn using CSQ in the distribution-independent setting (Feldman, 2011, 2012). We show hardness of *weak* learning using CSQ, for a general choice of distribution families.

We note that our results on CSQ learning are tightly related to previous results on learning with statistical-queries. We use a complexity measure of a distribution family (namely, the *variance* of the family) which captures the "orthogonality" between function-distribution pairs, similarly to the SQ-dimension introduced by (Blum et al., 1994). Our analysis of CSQ learning is very similar to the analysis in Yang (2005), applied to the more general setting of learning distribution families (rather than learning function classes over a fixed distribution). Our complexity measure is similar to the statistical dimension introduced in Feldman et al. (2017), and to the cross-predictability measure of (Abbe and Sandon, 2018).

Gradient-based Learning Another line of works shows hardness results for learning with gradient-based algorithms. The work of Shalev-Shwartz et al. (2017) and the work of Abbe and Sandon (2018) show that parities are hard to learn using gradient-based algorithms. The work of Shamir (2018) shows distribution-specific hardness results on learning with gradient-based algorithms. These works essentially show that gradient-descent is "stuck" at a sub-optimal point. The work of Safran and Shamir (2018) shows a natural instance of learning neural network which suffers from spurious local minima. We show

that DNF formulas are hard to learn using gradient-descent, using a generalization of the techniques used in previous works.

3. Problem Setting

We now describe in more detail the general setting for the problem of learning families of labeled distributions. Let $\mathcal{X}$ be our input space, where we typically assume that $\mathcal{X} = \{\pm 1\}^n$. We define the following:

- A (labeled) **distributions family** $\mathcal{A}$ is a set of pairs $(f, \mathcal{D})$, where f is a function from $\mathcal{X}$ to $\{\pm 1\}$, and $\mathcal{D}$ is some distribution over $\mathcal{X}$. We denote by $f(\mathcal{D})$ the distribution over $\mathcal{X} \times \{\pm 1\}$ of labeled examples $(\mathbf{x}, y)$, where $\mathbf{x} \sim \mathcal{D}$ and $y = f(\mathbf{x})$ (equivalently, $f(\mathcal{D})(\mathbf{x}, y) = \mathbf{1}_{y=f(\mathbf{x})}\mathcal{D}(\mathbf{x})$).
- A **hypothesis class** $\mathcal{H}$ is some class of functions from $\mathcal{X}$ to $\mathbb{R}$.

Throughout the paper, we analyze the approximation of the distribution family $\mathcal{A}$ with respect to some loss function $\ell : \mathbb{R} \times \{\pm 1\} \to \mathbb{R}_+$. Some popular loss functions that we consider explicitly:

- Hinge-loss: $\ell^{\text{hinge}}(\hat{y}, y) := \max\{1 - \hat{y}y, 0\}$.
- Square-loss: $\ell^{\text{sq}}(\hat{y}, y) := \frac{1}{2}(y - \hat{y})^2$.
- Zero-one loss: $\ell^{0-1}(\hat{y}, y) = \mathbf{1}\{\text{sign}(\hat{y}) = y\}$.

All of our results hold for the hinge-loss, which is commonly used for learning classification problems. Most of our results apply for other loss functions as well. The exact assumptions on the loss functions for each result are detailed in the sequel.

For some hypothesis $h \in \mathcal{H}$, some pair $(f, \mathcal{D}) \in \mathcal{A}$, and some loss function ℓ, we define the loss of h with respect to $(f, \mathcal{D})$ to be:

$$L_{f(\mathcal{D})}(h) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f(\mathcal{D})} \ell(h(\mathbf{x}), y) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} \ell(h(\mathbf{x}), f(\mathbf{x}))$$

Our primary goal is to ϵ-*succeed* on the family $\mathcal{A}$ using the hypothesis class $\mathcal{H}$. Namely, for any choice of $(f, \mathcal{D}) \in \mathcal{A}$, given access to the distribution $f(\mathcal{D})$ (via sampling, statistical-queries or gradient computations), return some hypothesis $h \in \mathcal{H}$ with $\mathbb{E}\left[L_{f(\mathcal{D})}(h)\right] \leq \epsilon$. To understand whether or not it is possible to succeed on $\mathcal{A}$ using $\mathcal{H}$, there are two basic questions that we need to account for:

- **Approximation**: for every $(f, \mathcal{D}) \in \mathcal{A}$, show that there exists $h \in \mathcal{H}$ with $L_{f(\mathcal{D})}(h) \leq \epsilon$. In other words, we want to bound:

$$\max_{(f,\mathcal{D})\in\mathcal{A}} \min_{h\in\mathcal{H}} L_{f(\mathcal{D})}(h)$$

- **Efficient Learnability**: show an algorithm s.t. for every $(f, \mathcal{D}) \in \mathcal{A}$, given access to $f(\mathcal{D})$, returns in time polynomial in $n, 1/\epsilon$ a hypothesis $h \in \mathcal{H}$ with:

$$\mathbb{E}\left[L_{f(\mathcal{D})}(h)\right] - \min_{\hat{h}\in\mathcal{H}} L_{f(\mathcal{D})}(\hat{h}) \leq \epsilon$$

Clearly, if $\mathcal{H}$ cannot approximate $\mathcal{A}$, then no algorithm, efficient or inefficient, can succeed on $\mathcal{A}$ using $\mathcal{H}$. However, even if $\mathcal{H}$ can approximate $\mathcal{A}$, it might not be efficiently learnable.

Before we move on, we give a few comments relating this setting to standard settings in learning theory:

Remark 1 *A very common assumption in learning is **realizability**, namely — assuming that for every $(f, \mathcal{D}) \in \mathcal{A}$, we have $f \in \mathcal{H}$. When assuming realizability, the question of approximation becomes trivial, and we are left with the question of learnability. In our setting, we do not assume realizability using $\mathcal{H}$, but we do assume that the family is realizable using some concept class (not necessarily the one that our algorithm uses).*

Remark 2 *There are two common settings in learning theory: **distribution-free** and **distribution-specific** learning. In the distribution-free setting, we assume that $\mathcal{A} = \mathcal{F} \times \mathcal{P}$, where $\mathcal{P}$ is the class of all distributions over $\mathcal{X}$, and $\mathcal{F}$ is some class of boolean functions over $\mathcal{X}$. In the distribution-specific setting, we assume that $\mathcal{A} = \mathcal{F} \times \{\mathcal{D}\}$, where $\mathcal{D}$ is some fixed distribution over $\mathcal{X}$ (say, the uniform distribution). In a sense, we consider a setting that generalizes both distribution-free and distribution-specific learning.*

4. Approximation and Learnability of Orthogonal Classes

First, we introduce the basic property of the family $\mathcal{A}$ that will allow us to derive the various results shown in this section. Fix some function $\phi : \mathcal{X} \to [-1, 1]$, and observe the *variance*[2] of $\langle f, \phi \rangle_{\mathcal{D}}$, over all the pairs $(f, \mathcal{D}) \in \mathcal{A}$:

$$\mathrm{Var}(\mathcal{A}, \phi) := \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\langle f, \phi \rangle_{\mathcal{D}}^2 \right]$$

We can define the "variance" of $\mathcal{A}$ by taking a supremum over all choices of ϕ, namely:

$$\mathrm{Var}(\mathcal{A}) := \sup_{\|\phi\|_\infty \leq 1} \mathrm{Var}(\mathcal{A}, \phi)$$

Now, consider the simple case of a distribution-specific family of orthogonal functions. Fix some distribution $\mathcal{D}$ over $\mathcal{X}$. We define $\langle f, g \rangle_{\mathcal{D}} = \mathbb{E}_{\mathbf{x}\sim\mathcal{D}} [f(\mathbf{x})g(\mathbf{x})]$, and let $\mathcal{F}$ be some set of functions from $\mathcal{X}$ to $\{\pm 1\}$ that are orthonormal with respect to $\langle \cdot, \cdot \rangle_{\mathcal{D}}$. Namely, for every $f, g \in \mathcal{F}$ we have $\langle f, g \rangle_{\mathcal{D}} = \mathbf{1}\{f = g\}$. For example, take $\mathcal{D}$ to be the uniform distribution over $\mathcal{X} = \{\pm 1\}^n$, and $\mathcal{F}$ to be a set of parities, i.e. functions of the form $f_I(\mathbf{x}) = \prod_{i \in I} x_i$, for $I \subseteq [n]$. Then, we observe the orthonormal family $\mathcal{A} = \mathcal{F} \times \{\mathcal{D}\}$. So, for the specific choice of orthonormal family $\mathcal{A}$, we get that, using Parseval's identity:

$$\mathrm{Var}(\mathcal{A}, \phi) = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\langle f, \phi \rangle_{\mathcal{D}}^2 \right] = \frac{1}{|\mathcal{F}|} \sum_{f \in \mathcal{F}} \langle f, \phi \rangle_{\mathcal{D}}^2 \leq \frac{\|\phi\|_{\mathcal{D}}^2}{|\mathcal{F}|} \ .$$

Since $\|\phi\|_{\mathcal{D}}^2 \leq \|\phi\|_\infty^2$, it follows that

$$\mathrm{Var}(\mathcal{A}) \leq \frac{1}{|\mathcal{F}|} \ .$$

2. This is the variance per se only in the case where $\mathbb{E} \langle f, \phi \rangle = 0$, but we will refer to this quantity as variance in other cases as well.

So, as $|\mathcal{F}|$ grows, $\mathrm{Var}(\mathcal{A})$ decreases. In fact, if we take $\mathcal{F}$ to be all the parities over $\mathcal{X}$, then $\mathrm{Var}(\mathcal{A})$ becomes exponentially small (namely, 2^{-n}). We will next show how we can use $\mathrm{Var}(\mathcal{A})$ to bound the approximation and learnability of $\mathcal{A}$ using various classes and algorithms.

4.1 Approximation using Linear Classes

Fix some embedding $\Psi : \mathcal{X} \to [-1,1]^N$ and consider the family of linear predictors over this embedding:

$$\mathcal{H}_\Psi = \{\mathbf{x} \mapsto \langle \Psi(\mathbf{x}), \mathbf{w} \rangle \;\; : \;\; \|\mathbf{w}\|_2 \leq B\}$$

This is often a popular choice of a hypothesis class. We start by analyzing its approximation capacity, with respect to some family $\mathcal{A}$:

Theorem 3 *Let ℓ be some convex loss function, satisfying $\ell(0,y) = \ell_0$ and $\ell'(0,y) = -y$. Then, for every family $\mathcal{A}$ and every class $\mathcal{H}_\Psi$ as defined above, the following holds:*

$$\max_{(f,\mathcal{D})\in\mathcal{A}} \min_{h\in\mathcal{H}_\Psi} L_{f(\mathcal{D})}(h) \geq \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \min_{h\in\mathcal{H}_\Psi} L_{f(\mathcal{D})}(h) \geq \ell_0 - B\sqrt{N\,Var(\mathcal{A})} \tag{1}$$

Proof We rely on a very simple observation, that will be the key of the analysis in this section. Fix some $(f,\mathcal{D}) \in \mathcal{A}$, and for every $\mathbf{w} \in \mathbb{R}^N$, define $L_{f(\mathcal{D})}(\mathbf{w}) = L_{f(\mathcal{D})}(h_\mathbf{w})$, where $h_\mathbf{w}(\mathbf{x}) = \langle \Psi(\mathbf{x}), \mathbf{w}\rangle$. Since ℓ is convex, we get that $L_{f(\mathcal{D})}$ is convex as well. Therefore, for every $\mathbf{w} \in \mathbb{R}^N$ with $\|\mathbf{w}\|_2 \leq B$ we have:

$$\begin{aligned} L_{f(\mathcal{D})}(\mathbf{w}) &\geq L_{f(\mathcal{D})}(\mathbf{0}) + \left\langle \nabla L_{f(\mathcal{D})}(\mathbf{0}), \mathbf{w} \right\rangle \\ &\geq^{C.S} L_{f(\mathcal{D})}(\mathbf{0}) - \left\|\nabla L_{f(\mathcal{D})}(\mathbf{0})\right\| \|\mathbf{w}\| \geq \ell_0 - B\left\|\nabla L_{f(\mathcal{D})}(\mathbf{0})\right\| \end{aligned}$$

This immediately gives a lower bound on the approximation of $\mathcal{A}$ using $\mathcal{H}_\Psi$:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \min_{h\in\mathcal{H}_\Psi} L_{f(\mathcal{D})}(h) = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \min_{\|\mathbf{w}\|\leq B} L_{f(\mathcal{D})}(\mathbf{w}) \geq \ell_0 - B \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left\|\nabla L_{f(\mathcal{D})}(\mathbf{0})\right\| \tag{2}$$

So, upper bounding the average gradient norm, w.r.t. a random choice of $(f,\mathcal{D}) \in \mathcal{A}$, gives a lower bound on approximating $\mathcal{A}$ with $\mathcal{H}_\Psi$. Now, using our definition of $\mathrm{Var}(\mathcal{A})$ we get:

$$\begin{aligned} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\left\|\nabla L_{f(\mathcal{D})}(\mathbf{0})\right\|^2\right] &= \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\sum_{i\in[N]} \left(\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} \ell'(0, f(\mathbf{x}))\Psi(\mathbf{x})_i\right)^2\right] \\ &= \sum_{i\in[N]} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\langle \Psi_i, f\rangle_\mathcal{D}^2\right] \leq N \cdot \mathrm{Var}(\mathcal{A}) \end{aligned}$$

Using Jensen's inequality gives $\mathbb{E}_{(f,\mathcal{D})\sim\mathcal{A}} \left\|\nabla L_{f(\mathcal{D})}(\mathbf{0})\right\| \leq \sqrt{N\mathrm{Var}(\mathcal{A})}$, and plugging in to Eq. (2) gives the required. ■

The above result is in fact quite strong: it shows a bound on approximating the class $\mathcal{A}$ using any choice of linear class (i.e., linear function over fixed embedding), and any convex loss functions (satisfying our mild assumptions). For example, it shows that any linear class $\mathcal{H}_\Psi$ of polynomial size (with B, N polynomial in n) cannot even *weakly* approximate the family of parities over $\mathcal{X}$. The loss of any linear class in this case will be effectively ℓ_0, that is — the loss of a constant-zero function. This extends the result of Kamath et al. (2020), showing a similar result for the square-loss only.

4.2 Approximation using Shallow Neural Networks

The previous result shows a hardness of approximation, and hence a hardness of learning, of any family of orthogonal functions, using a linear hypothesis class. Specifically, we showed that approximating parities over $\mathcal{X}$ is hard using any linear class. We now move to a more complex family of functions: depth-two (one hidden layer) neural networks. Given some activation σ, we define the class of depth-two networks by:

$$\mathcal{H}_{\text{2NN}} = \left\{ \mathbf{x} \mapsto \sum_{i=1}^{k} u_i \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x} \right\rangle + b_i\right) \;\; : \;\; \left\| \mathbf{w}^{(i)} \right\|_2, \|\mathbf{u}\|_2, \|\mathbf{b}\|_2 \leq R \right\}$$

It has been shown (e.g., Shalev-Shwartz et al. (2017)) that $\mathcal{H}_{\text{2NN}}$ can implement parities over $\mathcal{X}$. Therefore, together with Theorem 3 shown previously, this gives a *strong* separation between the class of depth-two networks and **any** linear class: while parities can be implemented exactly by depth-two networks, using a linear class they cannot even be approximated beyond a trivial hypothesis.

We can leverage the previous results to construct a function that **cannot** be approximated using a depth-two network. In the following results, we assume that $\mathcal{A}$ is finite, namely $|\mathcal{A}| < \infty$, and denote $M = |\mathcal{A}|$. For simplicity, we choose some arbitrary indexing of $\mathcal{A}$, and denote $\mathcal{A} = \{(f_1, \mathcal{D}_1), \ldots, (f_M, \mathcal{D}_M)\}$. Furthermore, we assume $\mathcal{X} = \{\pm 1\}^n$. [3]

Our construction takes some (finite) distribution family $\mathcal{A}$, and builds a **single** function-distribution pair $(F_{\mathcal{A}}, \mathcal{D}_{\mathcal{A}})$, *induced* by the family $\mathcal{A}$, which cannot be approximated by any depth-two network with bounded weights. In other words, we can show a lower bound on learning a "singleton" distribution family, containing only one function-distribution pair, i.e., $\mathcal{A}' = \{(F_{\mathcal{A}}, \mathcal{D}_{\mathcal{A}})\}$. Note that this is essentially different from the previous result, where we proved a lower bound against a family of labeled distributions.

We now describe our construction. Denote $m = \lceil \log M \rceil$, let $\mathcal{Z} \subseteq \{\pm 1\}^m$ be some subset such that $|\mathcal{Z}| = M$, and define some bijection $\varphi : \mathcal{Z} \to [M]$. Observe the function $F : \mathcal{X} \times \mathcal{Z} \to \{\pm 1\}$ defined as $F(\mathbf{x}, \mathbf{z}) = f_{\varphi(\mathbf{z})}(\mathbf{x})$, and the distribution $\mathcal{D}'$ over $\mathcal{X} \times \mathcal{Z}$ where $(\mathbf{x}, \mathbf{z}) \sim \mathcal{D}'$ is given by sampling $i \sim [M]$ uniformly, sampling $\mathbf{x} \sim \mathcal{D}_i$ and setting $\mathbf{z} = \varphi^{-1}(i)$. Now, we can identify the space $\{\pm 1\}^n \times \{\pm 1\}^m$ with the space $\{\pm 1\}^{n+m}$, e.g. by mapping $\mathbf{v} \in \{\pm 1\}^{n+m}$ to $(\mathbf{x}, \mathbf{w}) \in \{\pm 1\}^n \times \{\pm 1\}^m$ where $\mathbf{x} := \mathbf{v}_{1,\ldots,n}$ and $\mathbf{w} := \mathbf{v}_{n+1,\ldots,n+m}$. So, $\mathcal{D}'$ naturally induces a distribution over $\{\pm 1\}^{n+m}$, and we denote this distribution by $\mathcal{D}^{\varphi}_{\mathcal{A}}$. Similarly, we can use F to build a function $F^{\varphi}_{\mathcal{A}} : \{\pm 1\}^{n+m} \to \{\pm 1\}$, where $F^{\varphi}_{\mathcal{A}}(\mathbf{x}, \mathbf{z}) = F(\mathbf{x}, \mathbf{z})$ if $\mathbf{z} \in \mathcal{Z}$, and otherwise $F^{\varphi}_{\mathcal{A}}(\mathbf{x}, \mathbf{z}) = 1$ (since $\mathcal{D}^{\varphi}_{\mathcal{A}}$ gives zero probability outside of $\mathcal{X} \times \mathcal{Z}$, we can define $F^{\varphi}_{\mathcal{A}}$ arbitrarily outside of this set).

We consider depth-two neural-networks over $\mathbb{R}^{n+m}$. Similarly to above, we identify $\mathbb{R}^{n+m}$ with $\mathbb{R}^n \times \mathbb{R}^m$, and it will be easier to present our results on neural-networks operating over the latter space. That is, following our definition of $\mathcal{H}_{\text{2NN}}$, we define a depth-two neural-network over $\mathbb{R}^n \times \mathbb{R}^m$ by:

$$g(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{k} u_i \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x} \right\rangle + \left\langle \mathbf{v}^{(i)}, \mathbf{z} \right\rangle + b_i\right), \;\; \left\| \mathbf{w}^{(i)} \right\|_2, \left\| \mathbf{v}^{(i)} \right\|_2, \|\mathbf{u}\|_2, \|\mathbf{b}\|_2 \leq R$$

In this case, we show the following result:

3. The results can be easily extended to the case where $\mathcal{X} \subseteq \mathbb{R}^n$ for some set $\mathcal{X}$ of bounded norm.

Theorem 4 *Fix some finite distribution family $\mathcal{A}$, over the input space $\mathcal{X} = \{\pm 1\}^n$, and let φ be some bijection as defined above. Denote $m = \lceil \log |\mathcal{A}| \rceil$. Let ℓ be a 1-Lipschitz convex loss satisfying $\ell(0, y) = \ell_0$, $\ell'(0, y) = -y$. Then, every depth-two neural-network $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ with any 1-Lipschitz activation satisfies:*

$$L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(g) \geq \ell_0 - p(m, n, R, k) \cdot Var(\mathcal{A})^{1/3}$$

for some universal polynomial p.

We will start by showing this result in the case where $\mathbf{v}^{(i)}$ takes discrete values:

Lemma 5 *Assume that there exists $\Delta > 0$ such that $v^{(i)}_j \in \Delta\mathbb{Z} := \{\Delta \cdot z \ : \ z \in \mathbb{Z}\}$ for every i, j and $\left\|\mathbf{u}^{(i)}\right\|, \left\|\mathbf{w}^{(i)}\right\|, \left\|\mathbf{v}^{(i)}\right\|, \|\mathbf{b}\| < R$. Then, for every neural-network g, we have:*

$$L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(g) := \mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{z})\sim\mathcal{D}^{\varphi}_{\mathcal{A}}} \left[\ell(g(\mathbf{x}, \mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x}, \mathbf{z}))\right] \geq \ell_0 - C(m, n, R, k)\sqrt{\frac{Var(\mathcal{A})}{\Delta}}$$

for $C(m, n, R, k) = \sqrt{2k}R^{3/2}m^{1/4}\left(R(\sqrt{n} + \sqrt{m} + 1) + |\sigma(0)|\right)$.

Proof Fix some neural-network $g(\mathbf{x}, \mathbf{z}) = \sum_{i=1}^{k} u_i \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x}\right\rangle + \left\langle \mathbf{v}^{(i)}, \mathbf{z}\right\rangle + b_i\right)$. Our goal is to lower bound $L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(g)$. To do so, we reduce the problem of approximating $(F^{\varphi}_{\mathcal{A}}, \mathcal{D}^{\varphi}_{\mathcal{A}})$ using a shallow network to the problem of approximating $\mathcal{A}$ using some linear class. That is, we find some mappings $\Psi_g : \mathcal{X} \to [-1, 1]^N$ and $\mathbf{u}_g : \{\pm 1\}^m \to \mathbb{R}^N$ (mappings which depend on g) such that for every $\mathbf{x} \in \mathcal{X}$ and $\mathbf{z} \in \{\pm 1\}^{n'}$ we have $g(\mathbf{x}, \mathbf{z}) = \langle \Psi_g(\mathbf{x}), \mathbf{u}_g(\mathbf{z})\rangle$.

To get this reduction, we observe that since $\mathbf{v}^{(i)}$ is discrete, $\left\langle \mathbf{z}, \mathbf{v}^{(i)}\right\rangle$ can take only a finite number of values. In fact, we have $\left\langle \mathbf{z}, \mathbf{v}^{(i)}\right\rangle \in [-\sqrt{m}R, \sqrt{m}R] \cap \Delta\mathbb{Z}$. Indeed, fix some i and we have $\frac{1}{\Delta}\mathbf{v}^{(i)} \in \mathbb{Z}^n$, and since $\mathbf{z} \in \mathbb{Z}^n$ we have $\frac{1}{\Delta}\left\langle \mathbf{v}^{(i)}, \mathbf{z}\right\rangle = \left\langle \frac{1}{\Delta}\mathbf{v}^{(i)}, \mathbf{z}\right\rangle \in \mathbb{Z}$. So, we can map $\mathbf{x}$ to $\sigma(\left\langle \mathbf{w}^{(i)}, \mathbf{x}\right\rangle + j + b_i)$, for all choices of $j \in [-\sqrt{m}R, \sqrt{m}R] \cap \Delta\mathbb{Z}$, and get an embedding that satisfies our requirement. That is, we use the fact that $\left\langle \mathbf{w}^{(i)}, \mathbf{z}\right\rangle$ "collapses" to a small number of values to remove the dependence of $g(\mathbf{x}, \mathbf{z})$ on the exact value of $\mathbf{z}$.

To show this formally, for every $\mathbf{z} \in \mathcal{Z}$ denote $j(\mathbf{z}) = \left\langle \mathbf{v}^{(i)}, \mathbf{z}\right\rangle$, and so from what we showed $j(\mathbf{z}) \in [-R\sqrt{m}, R\sqrt{m}] \cap \Delta\mathbb{Z}$. Define $\Psi_{i,j}(\mathbf{x}) = \frac{\sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x}\right\rangle + j + b_i\right)}{R(\sqrt{n}+\sqrt{m}+1)+|\sigma(0)|}$ for every $i \in [k]$ and $j \in [-R\sqrt{n}, R\sqrt{n}] \cap \Delta\mathbb{Z}$, and note that:

$$|\Psi_{i,j}(\mathbf{x})| \leq \frac{\left|\left\langle \mathbf{w}^{(i)}, \mathbf{x}\right\rangle + j + b_i\right| + |\sigma(0)|}{R(\sqrt{n} + \sqrt{m} + 1) + |\sigma(0)|} \leq \frac{\left\|\mathbf{w}^{(i)}\right\| \|\mathbf{x}\| + |j| + |b_i| + |\sigma(0)|}{R(\sqrt{n} + \sqrt{m} + 1) + |\sigma(0)|} \leq 1$$

where we use: $|\sigma(a)| \leq |\sigma(0)| + |\sigma(a) - \sigma(0)| \leq |\sigma(0)| + |a|$.

Notice that $|[-R\sqrt{m}, R\sqrt{m}] \cap \Delta\mathbb{Z}| \leq 2\left\lfloor \frac{R\sqrt{m}}{\Delta} \right\rfloor$, so there are at most $2\left\lfloor \frac{R\sqrt{m}}{\Delta} \right\rfloor$ choices for j. Denote $N := 2k\lfloor \frac{R\sqrt{m}}{\Delta} \rfloor$ and let $\Psi_g : \mathcal{X} \to [-1, 1]^N$ defined as $\Psi_g(\mathbf{x}) = [\Psi_{i,j}(\mathbf{x})]_{i,j}$ (in vector form). Denote $B = R^2(\sqrt{n} + \sqrt{m} + 1) + R\,|\sigma(0)|$, and from Theorem 3:

$$\mathop{\mathbb{E}}_{i\sim[M]} \left[\min_{\|\hat{\mathbf{u}}\| \leq B} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f_i(\mathcal{D}_i)} \ell(\langle \hat{\mathbf{u}}, \Psi(\mathbf{x})\rangle, y)\right] = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\min_{h \in \mathcal{H}^B_{\Psi}} L_{f(\mathcal{D})}(h)\right] \geq \ell_0 - B\sqrt{N\mathrm{Var}(\mathcal{A})}$$

Notice that $g(\mathbf{x},\mathbf{z}) = \sum_{i=1}^{k} \frac{B}{R} u_i \Psi_{i,j(\mathbf{z})}(\mathbf{x}) = \langle \mathbf{u}_g(\mathbf{z}), \Psi(\mathbf{x}) \rangle$ where:

$$\mathbf{u}_g(\mathbf{z})_{i,j} = \begin{cases} \frac{B}{R} u_i & j = j(\mathbf{z}) \\ 0 & j \neq j(\mathbf{z}) \end{cases}$$

Since $\|\mathbf{u}_g(\mathbf{z})\| \leq \frac{B}{R} \|\mathbf{u}\| \leq B$ we get that:

$$\begin{aligned}
\mathop{\mathbb{E}}_{(\mathbf{x},\mathbf{z})\sim\mathcal{D}^{\varphi}_{\mathcal{A}}} \left[\ell(g(\mathbf{x},\mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z}))\right] &= \mathop{\mathbb{E}}_{i\sim[M]} \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_i} \left[\ell(g(\mathbf{x},\varphi^{-1}(i)), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\varphi^{-1}(i)))\right] \\
&= \mathop{\mathbb{E}}_{i\sim[M]} \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}_i} \left[\ell(\left\langle \mathbf{u}_g(\varphi^{-1}(i)), \Psi_g(\mathbf{x}) \right\rangle, f_i(\mathbf{x}))\right] \\
&= \mathop{\mathbb{E}}_{i\sim[M]} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f_i(\mathcal{D}_i)} \left[\ell(\left\langle \mathbf{u}_g(\varphi^{-1}(i)), \Psi_g(\mathbf{x}) \right\rangle, y)\right] \\
&\geq \mathop{\mathbb{E}}_{i\sim[M]} \left[\min_{\|\hat{\mathbf{u}}\|\leq B} \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f_i(\mathcal{D}_i)} \ell(\langle \hat{\mathbf{u}}, \Psi_g(\mathbf{x}) \rangle, y)\right] \\
&\geq \ell_0 - B\sqrt{N \mathrm{Var}(\mathcal{A})}
\end{aligned}$$

■

Now, to prove Theorem 4, we use the fact that a network with arbitrary (bounded) weights can be approximated by a network with discrete weights.

Proof of Theorem 4. Fix some $\Delta \in (0,1)$, and let $\hat{\mathbf{v}}^{(i)} = \Delta \left\lfloor \frac{1}{\Delta} \mathbf{v}^{(i)} \right\rfloor \in \Delta\mathbb{Z}^m$, where $\lfloor \cdot \rfloor$ is taken element-wise. Notice that for every j we have:

$$\left| v_j^{(i)} - \hat{v}_j^{(i)} \right| = \left| v_j^{(i)} - \Delta \left\lfloor \frac{1}{\Delta} v_j^{(i)} \right\rfloor \right| = \Delta \left| \frac{1}{\Delta} v_j^{(i)} - \left\lfloor \frac{1}{\Delta} v_j^{(i)} \right\rfloor \right| \leq \Delta$$

Observe the following neural network:

$$\hat{g}(\mathbf{x},\mathbf{z}) = \sum_{i=1}^{k} u_i \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x} \right\rangle + \left\langle \hat{\mathbf{v}}^{(i)}, \mathbf{z} \right\rangle + b_i \right)$$

For every $\mathbf{x} \in \{\pm 1\}^n, \mathbf{z} \in \{\pm 1\}^m$, using Cauchy-Schwartz inequality, and the fact that σ is 1-Lipchitz:

$$\begin{aligned}
|g(\mathbf{x},\mathbf{z}) - \hat{g}(\mathbf{x},\mathbf{z})| &\leq \|\mathbf{u}\| \sqrt{\sum_{i=1}^{k} \left| \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x} \right\rangle + \left\langle \mathbf{v}^{(i)}, \mathbf{z} \right\rangle + b_i \right) - \sigma\left(\left\langle \mathbf{w}^{(i)}, \mathbf{x} \right\rangle + \left\langle \hat{\mathbf{v}}^{(i)}, \mathbf{z} \right\rangle + b_i \right) \right|^2} \\
&\leq \|\mathbf{u}\| \sqrt{\sum_{i=1}^{k} \left| \left\langle \mathbf{v}^{(i)}, \mathbf{z} \right\rangle - \left\langle \hat{\mathbf{v}}^{(i)}, \mathbf{z} \right\rangle \right|^2} \\
&\leq \|\mathbf{u}\| \sqrt{\sum_{i=1}^{k} \left\| \mathbf{v}^{(i)} - \hat{\mathbf{v}}^{(i)} \right\|^2 \|\mathbf{z}\|^2} \leq R\sqrt{k}\Delta m
\end{aligned}$$

Now, by Lemma 5 we have:

$$L_{F(\mathcal{D}')}(\hat{g}) \geq \ell_0 - C\sqrt{\frac{\mathrm{Var}(\mathcal{A})}{\Delta}}$$

for $C = \sqrt{2k}R^{3/2}m^{1/4}\left(R(\sqrt{n}+\sqrt{m}+1)+|\sigma(0)|\right)$. Using the fact that ℓ is 1-Lipschitz we get:

$$\begin{aligned}
L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(g) &= \mathbb{E}\left[\ell(g(\mathbf{x},\mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z}))\right] \\
&\geq \mathbb{E}\left[\ell(\hat{g}(\mathbf{x},\mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z}))\right] - \mathbb{E}\left[\left|\ell(g(\mathbf{x},\mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z})) - \ell(\hat{g}(\mathbf{x},\mathbf{z}), F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z}))\right|\right] \\
&\geq L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(\hat{g}) - \mathbb{E}\left[|g(\mathbf{x},\mathbf{z}) - \hat{g}(\mathbf{x},\mathbf{z})|\right] \geq \ell_0 - C\sqrt{\frac{\mathrm{Var}(\mathcal{A})}{\Delta}} - R\sqrt{k}\Delta m
\end{aligned}$$

This is true for any $\Delta > 0$, so we choose $\Delta = \frac{C^{2/3}}{R^{2/3}k^{1/3}m^{2/3}}\mathrm{Var}(\mathcal{A})^{1/3}$ and we get:

$$\begin{aligned}
L_{F^{\varphi}_{\mathcal{A}}(\mathcal{D}^{\varphi}_{\mathcal{A}})}(g) &\geq \ell_0 - 2C^{2/3}\mathrm{Var}(\mathcal{A})^{1/3}R^{1/3}k^{1/6}m^{1/3} \\
&= 2^{4/3}k^{1/2}R^2m^{1/2}\left(\sqrt{n}+\sqrt{m}+1+\frac{|\sigma(0)|}{R}\right)^{2/3}\mathrm{Var}(\mathcal{A})^{1/3}
\end{aligned}$$

■

4.2.1 Hardness of approximation of inner-product mod 2

We now interpret the result of Theorem 4 for the case where $\mathcal{A}$ is the family of parities over $\mathcal{X}$ with respect to the uniform distribution. For presenting this result, it is easier to define the bijection φ directly into $\mathcal{A}$ (and not by an indexing of $\mathcal{A}$, as is done in the above proof). Namely, we define $\mathcal{Z} = \mathcal{X} = \{\pm 1\}^n$ and define $\varphi : \mathcal{Z} \to \mathcal{A}$ such that $\varphi(\mathbf{z}) = (f_{\mathbf{z}}, \mathcal{D})$, where $f_{\mathbf{z}}$ is the parity such that $f_{\mathbf{z}}(\mathbf{x}) = \prod_{i\in[n],z_i=-1} x_i$. We can write the induced function as $F^{\varphi}_{\mathcal{A}}(\mathbf{x},\mathbf{z}) = \prod_{i\in[n],z_i=-1} x_i = \prod_{i\in[n]}(x_i \vee z_i)$, and the induced distribution $\mathcal{D}^{\varphi}_{\mathcal{A}}$ is simply the uniform distribution over $\mathcal{X}\times\mathcal{X}$. Using Theorem 4 and the fact that $\mathrm{Var}(\mathcal{A}) = 2^{-n}$ we get the following:

Corollary 6 *Let $F(\mathbf{x},\mathbf{z}) = \prod_{i\in[n]}(x_i \vee z_i)$, and let $\mathcal{D}$ be the uniform distribution over $\{\pm1\}^n \times \{\pm1\}^n$. Let ℓ be a 1-Lipschitz convex loss satisfying $\ell(0,y) = \ell_0$ and $\ell'(0,y) = -y$. Then, any polynomial-size network with polynomial weights and 1-Lipschitz activation, cannot (weakly) approximate F with respect to $\mathcal{D}$ and the loss ℓ.*

We note that F is similar to the inner-product mod-2 function, that has been shown to be hard to implement efficiently using depth-two networks (Martens et al., 2013). Our result shows that this function is hard to even approximate, using a polynomial-size depth-two network, under any convex loss satisfying our assumptions. Notice that F **can** be implemented using a depth-three network, and so this result gives a strong separation between the classes of depth-two and depth-three networks (of polynomial size).

4.3 Hardness of Learning with Correlation Queries

So far, we showed hardness of approximation results for linear classes and shallow (depth-two) networks. This motivates the use of algorithms that learn more complex hypothesis classes (for example, depth-three networks). We now give hardness results that are independent of the hypothesis class, but rather focus on the learning algorithm. We show

restrictions on specific classes of algorithms, for learning families $\mathcal{A}$ with small $\mathrm{Var}(\mathcal{A})$. While such results are well-known in the case of orthogonal classes, we introduce them here in a fashion that allows us to generalize such results to more general families of distributions.

We consider learnability using statistical-query algorithms. A statistical-query algorithm has access to an oracle $\mathrm{STAT}(f, \mathcal{D})$ which, given a query $\psi : \mathcal{X} \times \{\pm 1\} \to [-1, 1]$, returns a response v such that $|\mathbb{E}_{\mathbf{x}\sim\mathcal{D}}\, \psi(\mathbf{x}, f(\mathbf{x})) - v| \leq \tau$, for some tolerance parameter $\tau > 0$. Specifically, we focus on algorithms that use only correlation queries, i.e. queries of the form $\psi(\mathbf{x}, y) = y\phi(\mathbf{x})$ for some $\phi : \mathcal{X} \to [-1, 1]$. We say that a statistical-query algorithm *learns* a distribution family $\mathcal{A}$ to loss ϵ using m queries, if it achieves a loss $< \epsilon$ after performing m queries, with respect to worst-case (adversarial) choice of $(f, \mathcal{D}) \in \mathcal{A}$. This is similar to the notion of learnability with statistical-queries studied in previous works (e.g., Kearns (1998); Blum et al. (1994); Yang (2005)).

We show the following result on learning with correlation queries:

Theorem 7 *Let $\ell \in \{\ell^{\mathrm{hinge}}, \ell^{\mathrm{sq}}, \ell^{0-1}\}$. Fix some family $\mathcal{A}$. Then, for any $\tau > 0$, any statistical-query algorithm that makes only correlation queries needs to make at least $\frac{\tau^2}{Var(\mathcal{A})} - 1$ queries of tolerance τ to learn $\mathcal{A}$ with loss $< a_\ell - b_\ell\tau$, for some universal constants $a_\ell, b_\ell > 0$ that depend on the loss function.*

We note that this result is essentially an adaptation of Theorem 2 from Yang (2005) to the setting presented in this paper. The main difference between our result and the results in Yang (2005) is the analysis of learning distribution families (i.e., sets of function-distribution pairs), rather than function classes with a fixed distribution, as is analyzed in Yang (2005). Another difference is the choice of loss functions, where our results apply for a larger family of losses (beyond the zero-one loss). We start with the following lemma:

Lemma 8 *Fix some loss $\ell \in \{\ell^{\mathrm{hinge}}, \ell^{\mathrm{sq}}, \ell^{0-1}\}$, and let $h : \mathcal{X} \to \mathbb{R}$ be some hypothesis. Let $\tilde{h} = clamp \circ h$, where $clamp(\hat{y}) = \max\{\min\{\hat{y}, 1\}, -1\}$. Then, for every function $f : \mathcal{X} \to \{\pm 1\}$ and distribution $\mathcal{D}$ over $\mathcal{X}$, if $\left|\left\langle \tilde{h}, f \right\rangle_{\mathcal{D}}\right| \leq \tau$ then $L_{f(\mathcal{D})}(h) \geq a_\ell - b_\ell\tau$, for some universal constants $a_\ell, b_\ell > 0$ which depend on the loss function.*

Proof We claim that the following holds for every $\ell \in \{\ell^{\mathrm{hinge}}, \ell^{\mathrm{sq}}, \ell^{0-1}\}$:

$$L_{f(\mathcal{D})}(h) = \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f(\mathcal{D})} [\ell(h(\mathbf{x}), y)] \geq \mathop{\mathbb{E}}_{(\mathbf{x},y)\sim f(\mathcal{D})} \left[\ell(\tilde{h}(\mathbf{x}), y)\right] \geq a_\ell - b_\ell \left\langle f, \tilde{h} \right\rangle_{\mathcal{D}} \geq a_\ell - b_\ell\tau$$

The first inequality holds for all choices of ℓ, by definition of $\tilde{h}$. We show the second inequality separately for the different loss functions:

- If $\ell = \ell^{\mathrm{hinge}}$ then since $\tilde{h}(\mathbf{x}) \in [-1, 1]$ we have $\ell(\tilde{h}(\mathbf{x}), y) = 1 - y\tilde{h}(\mathbf{x})$.
- If $\ell = \ell^{\mathrm{sq}}$ then we have: $\ell(\tilde{h}(\mathbf{x}), y) = \frac{1}{2}\tilde{h}(\mathbf{x})^2 - \tilde{h}(\mathbf{x})y + \frac{1}{2}y^2 \geq \frac{1}{2} - \tilde{h}(\mathbf{x})y$.
- If $\ell = \ell^{0-1}$ then we have: $\ell(\tilde{h}(\mathbf{x}), y) = \mathbf{1}\{\tilde{h}(\mathbf{x}) \neq y\} = \frac{1}{2} - \frac{1}{2}\tilde{h}(\mathbf{x})y$.

■

The proof of Theorem 7 is very similar to the proof presented in Yang (2005).

Proof Assume the family $\mathcal{A}$ can be learned to loss $< a_\ell - b_\ell \tau$ using q queries, where a_ℓ, b_ℓ are the constants from Lemma 8. Consider an adversarial oracle that, for every query ϕ, returns 0 if there exists $(f, \mathcal{D}) \in \mathcal{A}$ such that $|\langle f, \phi \rangle_{\mathcal{D}}| \leq \tau$. Let $\phi_1, \ldots, \phi_q : \mathcal{X} \to [-1, 1]$ be the queries of the algorithm, and let $h : \mathcal{X} \to \mathbb{R}$ be the hypothesis returned by the algorithm. Denote $\phi_{q+1} = \text{clamp} \circ h$.

Claim: For every $(f, \mathcal{D}) \in \mathcal{A}$, there exists some $i \in [q+1]$ such that $|\langle f, \phi_i \rangle_{\mathcal{D}}| > \tau$.

Proof: Assume otherwise, and so the adversary can choose the target to be $(f, \mathcal{D}) \in \mathcal{A}$ such that $|\langle f, \phi_i \rangle_{\mathcal{D}}| \leq \tau$ for every $i \in [q+1]$. In this case, it holds that $\left|\langle f, \phi_{q+1} \rangle_{\mathcal{D}}\right| \leq \tau$, and so from Lemma 8 we get that $L_{f(\mathcal{D})}(h) \geq a_\ell - b_\ell \tau$, contradicting the assumption.

Using the above claim, we get the following:

$$(q+1)\text{Var}(\mathcal{A}) \geq \sum_{i=1}^{q+1} \text{Var}(\mathcal{A}, \phi_i) = \sum_{i=1}^{q+1} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\langle f, \phi_i \rangle_{\mathcal{D}}^2\right] = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\sum_{i=1}^{q+1} \langle f, \phi_i \rangle_{\mathcal{D}}^2\right] \geq \tau^2$$

and therefore the required follows. ■

Remark 9 *Observe that in the case where $\mathcal{A}$ is a distribution-specific family, any statistical-query algorithm can be modified to use only correlation queries, as shown in a work by Bshouty and Feldman (2002). However, this is not true for general families of distributions.*

When $\mathcal{A}$ is a family of parities with respect to the uniform distribution, Theorem 7 along with the fact that $\text{Var}(\mathcal{A}) = 2^{-n}$ recovers the well-known result on hardness of learning parities using statistical-queries (Kearns, 1998).

4.4 Hardness of Learning with Gradient-Descent

We showed that families $\mathcal{A}$ with small $\text{Var}(\mathcal{A})$ are hard to learn using correlation queries. We now turn to analyze a specific popular algorithm: the gradient-descent algorithm. In this part we take the loss function to be the hinge-loss, so $\ell = \ell^{\text{hinge}}$.

Observe the following formulation of gradient-descent: let $\mathcal{H}$ be a parametric hypothesis class, such that $\mathcal{H} = \{h_{\mathbf{w}} : \mathbf{w} \in \mathbb{R}^N\}$. In the gradient-descent algorithm, we initialize $\mathbf{w}_0$ (possibly randomly), and perform the following updates:

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta \nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\mathbf{w}_{t-1}}) \qquad \text{(GD Update)}$$

However, assuming we have access to the exact value of $\nabla_{\mathbf{w}} L_{f(\mathcal{D})}$ is often unrealistic. For example, when running gradient-descent on a machine with bounded precision, we can expect the value of the gradient to be accurate only up to some fixed precision. So, we consider instead a variant of the gradient-descent algorithm which has access to gradients that are accurate up to a fixed precision $\Delta > 0$. The Δ-*approximate gradient-descent* algorithm performs the following updates:

$$\mathbf{w}_t = \mathbf{w}_{t-1} - \eta \mathbf{v}_t \qquad \text{(Approx. GD Update)}$$

with $\mathbf{v}_t \in \Delta\mathbb{Z}^N$ (where $\Delta\mathbb{Z} := \{\Delta z : z \in \mathbb{Z}\}$) satisfying $\left\|\mathbf{v}_t - \nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\mathbf{w}_{t-1}})\right\|_\infty \leq \frac{\Delta}{2}$ (i.e., $\mathbf{v}_t$ is the result of rounding $\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\mathbf{w}_{t-1}})$ to $\Delta\mathbb{Z}^N$).

Notice that if the gradients are smaller than the machine precision, they will be rounded to zero, and so the gradient-descent algorithm will be "stuck". We show that if $\mathrm{Var}(\mathcal{A})$ is small, then for most choices of $(f,\mathcal{D}) \in \mathcal{A}$ the initial gradient will indeed be extremely small. The key for showing this is the following lemma:

Lemma 10 *Fix some $\mathbf{w} \in \mathbb{R}^N$ satisfying $|h_{\mathbf{w}}(\mathbf{x})| \leq 1$ and $\|\nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x})\| \leq B$, for every $\mathbf{x} \in \mathcal{X}$. Then:*

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\mathbf{w}})\right\|_2^2 \leq B^2 N\, Var(\mathcal{A})$$

Proof Denote $\phi_i(\mathbf{x}) = \frac{1}{B}\frac{\partial}{\partial w_i} h_{\mathbf{w}}(\mathbf{x})$ and note that $\phi_i(\mathbf{x}) \in [-1,1]$. Note that since $h_{\mathbf{w}}(\mathbf{x}) \in [-1,1]$ for every $\mathbf{x}$, we have, for every $(f,\mathcal{D}) \in \mathcal{A}$:

$$L_{f(\mathcal{D})}(h_{\mathbf{w}}) = \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} \ell(h_{\mathbf{w}}(\mathbf{x}), f(\mathbf{x})) = 1 - \mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} h_{\mathbf{w}}(\mathbf{x}) f(\mathbf{x})$$

where we use the fact that the hinge-loss satisfies $\ell(\hat{y}, y) = 1 - \hat{y}y$ for every $\hat{y} \in [-1,1]$. Therefore, we get:

$$\begin{aligned}
\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\mathbf{w}})\right\|_2^2 &= \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \sum_{i=1}^{N} \left(\frac{\partial}{\partial w_i} L_{f(\mathcal{D})}(h_{\mathbf{w}})\right)^2 \\
&= \sum_{i=1}^{N} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left(\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}} f(\mathbf{x}) \frac{\partial}{\partial w_i} h_{\mathbf{w}}(\mathbf{x})\right)^2 \\
&= \sum_{i=1}^{N} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} B^2 \langle f, \phi_i\rangle_{\mathcal{D}}^2 \leq NB^2 \mathrm{Var}(\mathcal{A})
\end{aligned}$$

■

Using the above, we can show that running gradient-descent with Δ-approximate gradients has high loss on average, unless Δ is very small:

Theorem 11 *Assume we initialize $\mathbf{w}_0$ from some distribution $\mathcal{W}$ such that almost surely, for every $\mathbf{x} \in \mathcal{X}$ we have $|h_{\mathbf{w}_0}(\mathbf{x})| \leq 1$ and $\|\nabla_{\mathbf{w}} h_{\mathbf{w}_0}(\mathbf{x})\| \leq B$ for some $B > 0$. Then, if $\Delta \geq 6\sqrt{2B^2 N\, Var(\mathcal{A})}$, there exists some $(f,\mathcal{D}) \in \mathcal{A}$ such that for every $T > 0$, running Δ-approximate gradient-descent for T steps returns a hypothesis $h_{\mathbf{w}_T}$ which satisfies:*

$$\mathop{\mathbb{E}}_{\mathbf{w}_0\sim\mathcal{W}} L_{f(\mathcal{D})}(h_{\mathbf{w}_T}) \geq \frac{3}{4}\left(1 - \sqrt{8\, Var(\mathcal{A})}\right)$$

In the proof of the Theorem, we rely on the following Lemma:

Lemma 12 *Fix some family $\mathcal{A}$, some $\delta > 0$, and let $\phi : \mathcal{X} \to [-1,1]$. Then:*

$$\mathop{\mathbb{P}}_{(f,\mathcal{D})\sim\mathcal{A}} [|\langle f, \phi\rangle|_{\mathcal{D}} \geq \delta] \leq \frac{Var(\mathcal{A})}{\delta^2}$$

Proof By definition of $\mathrm{Var}(\mathcal{A})$ we have that:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left[\langle f, \phi\rangle_{\mathcal{D}}^2\right] \leq \mathrm{Var}(\mathcal{A})$$

So, using Markov's inequality we get:

$$\mathop{\mathbb{P}}_{(f,\mathcal{D})\sim\mathcal{A}}\left[|\langle f,\phi\rangle_{\mathcal{D}}|\geq\delta\right]=\mathop{\mathbb{P}}_{(f,\mathcal{D})\sim\mathcal{A}}\left[\langle f,\phi\rangle_{\mathcal{D}}{}^2\geq\delta^2\right]\leq\frac{\mathbb{E}_{(f,\mathcal{D})\sim\mathcal{A}}\left[\langle f,\phi\rangle_{\mathcal{D}}^2\right]}{\delta^2}\leq\frac{\mathrm{Var}(\mathcal{A})}{\delta^2}$$

■

Proof of Theorem 11. Fix some $\mathbf{w}_0$ satisfying $|h_{\mathbf{w}_0}(\mathbf{x})|\leq 1$ and $\|\nabla_{\mathbf{w}}h_{\mathbf{w}_0}(\mathbf{x})\|\leq B$ for every $\mathbf{x}\in\mathcal{X}$. So, from Lemma 10 we have:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}}\left\|\nabla_{\mathbf{w}}L_{f(\mathcal{D})}(h_{\mathbf{w}_0})\right\|_2^2\leq NB^2\mathrm{Var}(\mathcal{A})$$

From Markov's inequality, we get that with probability at least $1-\frac{9B^2N\mathrm{Var}(\mathcal{A})}{\Delta^2}$ over the choice of $(f,\mathcal{D})\sim\mathcal{A}$ we have $\left\|\nabla_{\mathbf{w}}L_{f(\mathcal{D})}(h_{\mathbf{w}_0})\right\|_2\leq\frac{\Delta}{3}$, and in this case $\mathbf{v}_t=0$. So, for every $(f,\mathcal{D})\in\mathcal{A}$ with $\left\|\nabla_{\mathbf{w}}L_{f(\mathcal{D})}(h_{\mathbf{w}_0})\right\|_2\leq\frac{\Delta}{3}$, gradient-descent will return $h_{\mathbf{w}_0}$. Since $h_{\mathbf{w}_0}(\mathbf{x})\in[-1,1]$, by application of Lemma 12 with $\delta=\sqrt{8\mathrm{Var}(\mathcal{A})}$, with probability at least $7/8$ over the choice of $(f,\mathcal{D})\sim\mathcal{A}$ we have $|\langle f,h_{\mathbf{w}_0}\rangle|\leq\sqrt{8\mathrm{Var}(\mathcal{A})}$ and so:

$$L_{f(\mathcal{D})}(h_{\mathbf{w}_0})=1-\langle f,h_{\widetilde{\mathbf{w}}_0}\rangle_{\mathcal{D}}\geq 1-\sqrt{8\mathrm{Var}(\mathcal{A})}$$

Using the union bound, with probability at least $\frac{7}{8}-\frac{9B^2N\mathrm{Var}(\mathcal{A})}{\Delta^2}\geq\frac{3}{4}$ over the choice of $(f,\mathcal{D})\sim\mathcal{A}$, gradient-descent algorithm will return a hypothesis with loss at least $1-\sqrt{8\mathrm{Var}(\mathcal{A})}$, and so:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}}L_{f(\mathcal{D})}(h_{\mathbf{w}_T})\geq\frac{3}{4}\left(1-\sqrt{8\mathrm{Var}(\mathcal{A})}\right)$$

Applying the above for a random choice of $\mathbf{w}_0\sim\mathcal{W}$ we get:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}}\mathop{\mathbb{E}}_{\mathbf{w}_0\sim\mathcal{W}}L_{f(\mathcal{D})}(h_{\mathbf{w}_T})=\mathop{\mathbb{E}}_{\mathbf{w}_0\sim\mathcal{W}}\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}}L_{f(\mathcal{D})}(h_{\mathbf{w}_T})\geq\frac{3}{4}\left(1-\sqrt{8\mathrm{Var}(\mathcal{A})}\right)$$

And so the required follows ■

So far, we analyzed the gradient-descent algorithm with respect to an approximation of the population gradient. The above result shows that if the initial gradient is very small, then gradient-descent is "stuck" on the first iteration. In practice, however, gradient-descent uses stochastic estimation of the population gradient. In this case, the stochastic noise due to the gradient estimation will cause non-zero update steps, even if the population gradient is zero. To account for this setting, we consider a noisy version of gradient-descent. The *σ-noisy gradient-descent* performs the same update as in (Approx. GD Update), with $\mathbf{v}_t\in\Delta\mathbb{Z}^N$ which satisfies:

$$\left\|\mathbf{v}_t-\left(\nabla_{\mathbf{w}}L_{f(\mathcal{D})}(h_{\mathbf{w}_{t-1}})+\xi_t\right)\right\|_\infty\leq\frac{\Delta}{2}$$

where $\xi_1,\ldots,\xi_T$ are i.i.d. random noise variables with $\xi_t\in\Delta\mathbb{Z}^N$ and $\|\xi_t\|_2\leq\sigma$. For the *noisy gradient-descent* algorithm, we get the following hardness result:

Theorem 13 *Assume we initialize $\mathbf{w}_0$ from some distribution $\mathcal{W}$ such that almost surely, for every $\mathbf{x} \in \mathcal{X}$ we have $|h_{\mathbf{w}_0}(\mathbf{x})| \leq \frac{1}{2}$. Assume that for every $\mathbf{w}$, $h_{\mathbf{w}} \in \mathcal{H}$ is differentiable and satisfies $\|\nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x})\| \leq B$ for all $\mathbf{x} \in \mathcal{X}$. Then, there exists some $(f, \mathcal{D}) \in \mathcal{A}$ such that running the noisy gradient-descent algorithm for at most $T \leq \frac{\Delta^2}{72B^2 N Var(\mathcal{A})}$ steps with $\eta \leq \frac{1}{2\sigma BT}$, returns a hypothesis $h_{\mathbf{w}_T}$ satisfying:*

$$\mathop{\mathbb{E}}_{\mathbf{w}_0, \xi_1, \ldots, \xi_T} L_{f(\mathcal{D})}(h_{\mathbf{w}_T}) \geq \frac{3}{4}\left(1 - \sqrt{8\,Var(\mathcal{A})}\right)$$

Proof Fix some $\xi_1, \ldots, \xi_T \in \Delta\mathbb{Z}^N$ such that $\|\xi_t\|_2 \leq \sigma$ for every $t \in [T]$. Fix some $\mathbf{w}_0 \in \Delta\mathbb{Z}^N$ with $\|h_{\mathbf{w}_0}\|_\infty \leq \frac{1}{2}$. We define $\widetilde{\mathbf{w}}_t := \mathbf{w}_0 + \eta \sum_{i=1}^t \xi_i$, and observe that for every t, by application of the mean value theorem we have for every $\mathbf{x}$:

$$|h_{\widetilde{\mathbf{w}}_t}(\mathbf{x}) - h_{\mathbf{w}_0}(\mathbf{x})| \leq \sup_{\mathbf{w}} \|\nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x})\|_2 \|\widetilde{\mathbf{w}}_t - \mathbf{w}_0\|_2 \leq B\eta \left\|\sum_{i=1}^t \xi_i\right\| \leq B\eta\sigma T \leq \frac{1}{2}$$

where in the last inequality we use the fact that $\eta \leq \frac{1}{2\sigma BT}$. So, we have:

$$|h_{\widetilde{\mathbf{w}}_t}(\mathbf{x})| \leq |h_{\mathbf{w}_0}(\mathbf{x})| + |h_{\widetilde{\mathbf{w}}_t}(\mathbf{x}) - h_{\mathbf{w}_0}(\mathbf{x})| \leq 1$$

Therefore, using Lemma 10 we get:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \frac{1}{T}\sum_{t=0}^{T-1} \left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\widetilde{\mathbf{w}}_t})\right\|_2^2 = \mathop{\mathbb{E}}_{t\sim[T-1]} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\widetilde{\mathbf{w}}_t})\right\|_2^2 \leq B^2 N \mathrm{Var}(\mathcal{A})$$

From Markov's inequality, with probability at least 7/8 over the choice of $(f, \mathcal{D}) \sim \mathcal{A}$, we have $\sum_{t=0}^{T-1} \left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\widetilde{\mathbf{w}}_t})\right\|_2^2 \leq 8TB^2 N \mathrm{Var}(\mathcal{A}) \leq \frac{\Delta^2}{9}$. For every such $(f, \mathcal{D})$, we have $\left\|\nabla_{\mathbf{w}} L_{f(\mathcal{D})}(h_{\widetilde{\mathbf{w}}_t})\right\|_2 \leq \frac{\Delta}{3}$ for every $t \in [T-1]$, and so $\mathbf{v}_t = \xi_t$ for every t, in which case we have the updates $\mathbf{w}_t = \widetilde{\mathbf{w}}_t$, and the noisy gradient-descent algorithm will output $h_{\widetilde{\mathbf{w}}_T}$. Since $h_{\widetilde{\mathbf{w}}_T}(\mathbf{x}) \in [-1, 1]$, by application of Lemma 12 with $\delta = \sqrt{8\mathrm{Var}(\mathcal{A})}$, with probability at least 7/8 over the choice of $(f, \mathcal{D}) \sim \mathcal{A}$ we have:

$$L_{f(\mathcal{D})}(h_{\widetilde{\mathbf{w}}_T}) = 1 - \left\langle f, h_{\widetilde{\mathbf{w}}_T} \right\rangle_{\mathcal{D}} \geq 1 - \sqrt{8\mathrm{Var}(\mathcal{A})}$$

All in all, using the union bound, w.p. at least 3/4 over the choice of $(f, \mathcal{D}) \sim \mathcal{A}$ the gradient-decent algorithm returns a hypothesis $h_{\mathbf{w}_T}$ with loss $\geq 1 - \sqrt{8\mathrm{Var}(\mathcal{A})}$, and so:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} L_{f(\mathcal{D})}(h_{\mathbf{w}_T}) \geq \frac{3}{4}\left(1 - \sqrt{8\mathrm{Var}(\mathcal{A})}\right)$$

Now, for a random choice of $\mathbf{w}_0, \xi_1, \ldots, \xi_T$ we have:

$$\mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \mathop{\mathbb{E}}_{\mathbf{w}_0,\xi_1,\ldots,\xi_T} L_{f(\mathcal{D})}(h_{\mathbf{w}_T}) = \mathop{\mathbb{E}}_{\mathbf{w}_0,\xi_1,\ldots,\xi_T} \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} L_{f(\mathcal{D})}(h_{\mathbf{w}_T}) \geq \frac{3}{4}\left(1 - \sqrt{8\mathrm{Var}(\mathcal{A})}\right)$$

and therefore the required follows. ■

Applying the previous theorem for the family of uniform parities, we get that gradient-descent fails to reach non-trivial loss on the class of parities, unless the approximation tolerance is exponentially small or the number of steps is exponentially large. This result is similar to the results of Shalev-Shwartz et al. (2017) and Abbe and Sandon (2018).

5. General Distribution Families

In the previous section, we showed various hardness of learning and approximation results, all derived from the measure $\mathrm{Var}(\mathcal{A})$. We showed the application of such hardness results to the case of parities, or more generally — families of orthogonal functions. However, note that the measure $\mathrm{Var}(\mathcal{A})$ can be applied to any family of distributions, and therefore all of our results can be derived for the very general setting of learning arbitrary families of labeled distributions. In this section, we interpret these results for general distribution families, and show how to derive bounds on $\mathrm{Var}(\mathcal{A})$ in the general case. Using this, we show novel results on hardness of approximation and learnability of DNFs and AC^0 circuits.

We start by showing a general method for bounding $\mathrm{Var}(\mathcal{A})$. Let $M(\mathcal{A})$ be the linear operator from $\mathbb{R}^{\mathcal{X}}$ to $\mathbb{R}^{\mathcal{A}}$, such that for every $\phi : \mathcal{X} \to \mathbb{R}$, $M(\mathcal{A})(\phi)_{(f,\mathcal{D})} = \langle f, \phi \rangle_{\mathcal{D}}$. The linearity of $M(\mathcal{A})$ follows from the bi-linearity of the inner product $\langle \cdot, \cdot \rangle_{\mathcal{D}}$. Note that when $\mathcal{X}$ and $\mathcal{A}$ are finite (as we assume in this work), $M(\mathcal{A})$ can be written in a matrix form, where $M(\mathcal{A}) \in \mathbb{R}^{\mathcal{A} \times \mathcal{X}}$ and $M(\mathcal{A})_{(f,\mathcal{D}),\mathbf{x}} = f(\mathbf{x})\mathcal{D}(\mathbf{x})$. In this case, we identify $\phi : \mathcal{X} \to \mathbb{R}$ with a vector $\mathbf{v}(\phi) \in \mathbb{R}^{\mathcal{X}}$ with $\mathbf{v}(\phi)_{\mathbf{x}} = \phi(\mathbf{x})$, and we get:

$$M(\mathcal{A})\mathbf{v}(\phi) = [\sum_{\mathbf{x} \in \mathcal{X}} f(\mathbf{x})\phi(\mathbf{x})\mathcal{D}(\mathbf{x})]_{(f,\mathcal{D})} = [\langle f, \phi \rangle_{\mathcal{D}}]_{(f,\mathcal{D})}$$

Now, observe that for every $\phi : \mathcal{X} \to [-1, 1]$ we have:

$$\mathrm{Var}(\mathcal{A}, \phi) = \mathop{\mathbb{E}}_{(f,\mathcal{D}) \sim \mathcal{A}} \left[\langle f, \phi \rangle_{\mathcal{D}}^2 \right] = \frac{1}{|\mathcal{A}|} \left\| M(\mathcal{A})\phi \right\|_2^2 \leq \frac{1}{|\mathcal{A}|} \left\| M(\mathcal{A}) \right\|_2^2 \left\| \phi \right\|_2^2 \leq \frac{|\mathcal{X}|}{|\mathcal{A}|} \left\| M(\mathcal{A}) \right\|_2^2 \quad (3)$$

Where $\|M(\mathcal{A})\|_2$ is the L_2 operator norm of $M(\mathcal{A})$. Hence, Eq. (3) gives a general bound for $\mathrm{Var}(\mathcal{A})$, in terms of the operator norm of the matrix $M(\mathcal{A})$.

5.1 Operator Norm of the AND-OR-AND Function

In this part, we give a concrete family of distributions, generated by an AND-OR-AND type function, and analyze its variance. Specifically, we show that its variance decays like $2^{-O(n^{1/3})}$. The key for showing this result is bounding the operator norm of the relevant matrix, along with the analysis introduced in Razborov and Sherstov (2010), which bounds the norm of a similar matrix. The main result is the following:

Theorem 14 *For large enough m, there exist subsets $\mathcal{X}, \mathcal{Z} \subseteq \{\pm 1\}^{dm^3}$, for some universal constant $d > 0$, and a family $\mathcal{A}$ over $\mathcal{X}$ such that:*

- *For each $(f, \mathcal{D}) \in \mathcal{A}$, it holds that $f(\mathbf{x}) = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{dm^2} (x_{ij} \wedge z_{ij})$, for some $\mathbf{z} \in \mathcal{Z}$.*
- *$Var(\mathcal{A}) \leq 17^{-2m}$.*

For the proof of the theorem, we need the following simple result:

Definition 15 *Let $\mathcal{A}$ be some distribution family over an input space $\mathcal{X}$, and let $\mathcal{A}'$ be some distribution family over another input space $\mathcal{X}'$. $\mathcal{A}$ and $\mathcal{A}'$ are **isomorphic** if there exists a bijection $\Psi : \mathcal{X} \to \mathcal{X}'$ such that $\mathcal{A} = \{(f \circ \Psi, \mathcal{D} \circ \Psi) \ : \ (f, \mathcal{D}) \in \mathcal{A}'\}$.*

Lemma 16 *If $\mathcal{A}$ and $\mathcal{A}'$ are isomorphic distribution families, then $Var(\mathcal{A}) = Var(\mathcal{A}')$.*

Proof Fix $\phi : \mathcal{X} \to [-1,1]$, and observe that:

$$\begin{aligned}\mathrm{Var}(\mathcal{A},\phi) &= \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}} \langle f,\phi\rangle_{\mathcal{D}}^2 = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}'_n} \langle f\circ\Psi,\phi\rangle_{\mathcal{D}\circ\Psi}^2 = \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}'_n}\left(\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}\circ\Psi}[f(\Psi(\mathbf{x}))\phi(\mathbf{x})]\right)^2 \\ &= \mathop{\mathbb{E}}_{(f,\mathcal{D})\sim\mathcal{A}'_n}\left(\mathop{\mathbb{E}}_{\mathbf{x}\sim\mathcal{D}}\left[f(\mathbf{x})\phi(\Psi^{-1}(\mathbf{x}))\right]\right)^2 = \mathrm{Var}(\mathcal{A}',\phi\circ\Psi^{-1}) \le \mathrm{Var}(\mathcal{A}')\end{aligned}$$

Therefore $\mathrm{Var}(\mathcal{A}) \le \mathrm{Var}(\mathcal{A}')$, and the required follows from symmetry. ■

Now, the key for bounding the operator norm related to the family $\mathcal{A}$, is using the pattern-matrix technique, as used in Razborov and Sherstov (2010).

Pattern Matrix Let n, N be two integers, and let $\mathcal{V}(N,n)$ be the family of subsets $V \subset [N]$ of size $|V| = n$, with one element from each block of size N/n from $[N]$. Define the projection onto V by $x|_V = (x_{i_1},\ldots,x_{i_n})$ where $i_1 < \cdots < i_n$ are the elements of V.

Definition 17 *For $\phi : \{\pm1\}^n \to \mathbb{R}$, the (N,n,ϕ)-pattern matrix is the matrix:*

$$A = [\phi(\mathbf{x}|_V \oplus \mathbf{w})]_{\mathbf{x}\in\{\pm1\}^N,(V,\mathbf{w})\in\mathcal{V}(N,n)\times\{\pm1\}^n}$$

Theorem 18 *(Razborov and Sherstov (2010)) Let $n = 4m^3$ and $N = 17^6 n$. Let $MP_m(\mathbf{x}) = \bigwedge_{i=1}^m \bigvee_{j=1}^{4m^2} x_{ij}$, and let M be the (N,n,MP_m)-pattern matrix. There exists a distribution $\mu : \{\pm1\}^n \to \mathbb{R}_+$, such that the (N,n,μ)-pattern matrix P satisfies $\|M\odot P\|_2 \le 17^{-m}2^{-n}\sqrt{2^{N+n}(\frac{N}{n})^n}$ (where $\odot$ denotes the Hadamard product).*

Proof of Theorem 14.

Let $\mu : \{\pm1\}^n \to \mathbb{R}$ be the distribution from Theorem 18, and let N, n as defined in the Theorem. Denote $\mathcal{X}' = \{\pm1\}^N$, $\mathcal{Z}' = \mathcal{V}(N,n)\times\{\pm1\}^n$. Fix some $(V,\mathbf{w}) \in \mathcal{Z}'$ denote $f_{V,\mathbf{w}}(\mathbf{x}) = MP_m(\mathbf{x}|_V \oplus \mathbf{w})$, and define the distribution $\mathcal{D}_{V,w}$ over $\mathcal{X}'$ such that $\mathcal{D}_{V,w}(\mathbf{x}) = \frac{1}{2^{N-n}}\mu(\mathbf{x}|_V \oplus \mathbf{w})$. $\mathcal{D}_{V,\mathbf{w}}$ indeed defines a distribution over $\mathcal{X}'$, since:

$$\sum_{\mathbf{x}\in\mathcal{X}'}\mathcal{D}_{V,\mathbf{w}}(\mathbf{x}) = \sum_{\mathbf{z}\in\{\pm1\}^n}\sum_{\mathbf{x}\in\mathcal{X}',\mathbf{x}|_V=\mathbf{z}}\frac{1}{2^{N-n}}\mu(\mathbf{z}\oplus\mathbf{w}) = \sum_{\mathbf{z}\in\{\pm1\}^n}\mu(\mathbf{z}\oplus\mathbf{w}) = 1$$

Define the family $\mathcal{A}' = \{(f_{V,\mathbf{w}},\mathcal{D}_{V,\mathbf{w}}) : (V,\mathbf{w})\in\mathcal{Z}'\}$ and recall that we defined $M(\mathcal{A}') = [f_{V,\mathbf{w}}(\mathbf{x})\mathcal{D}_{V,\mathbf{w}}(\mathbf{x})]_{\mathbf{x}\in\mathcal{X}',(V,\mathbf{w})\in\mathcal{Z}}$. Let M be the (N,n,MP_m)-pattern matrix and let P be the (N,n,μ)-pattern matrix, and so $M(\mathcal{A}') = \frac{1}{2^{N-n}}M\odot P$, and from Thm 18 we have:

$$\left\|M(\mathcal{A}')\right\|_2 = 2^{n-N}\left\|M\odot P\right\|_2 \le 2^{n-N}17^{-m}2^{-n}\sqrt{2^{N+n}\left(\frac{N}{n}\right)^n} = 17^{-m}\sqrt{2^{n-N}\left(\frac{N}{n}\right)^n}$$

And from Eq. 3 we get:

$$\mathrm{Var}(\mathcal{A}') \le \frac{|\mathcal{X}'|}{|\mathcal{A}'|}\left\|M(\mathcal{A}')\right\|_2^2 \le \frac{2^N}{(N/n)^n 2^n}17^{-2m}2^{n-N}(N/n)^n = 17^{-2m}$$

Observe that $\mathcal{A}'$ is a class of distributions labeled by CNF formulas, with $\mathrm{Var}(\mathcal{A}') \leq 17^{-2m}$ as required. However, to get the first condition of Theorem 14 we need to present each CNF as a three-level AND-OR-AND formula. To do so, we rewrite each function $f_{V,\mathbf{w}}$ as a function over a slightly larger space, which can then be used to achieve the above form.

So, we identify $\mathcal{X}'$ and $\mathcal{Z}'$ with subsets of $\{\pm 1\}^{n'}$, for $n' = m \cdot 4m^2 \cdot N/n \cdot 2 = 8m^3 \frac{N}{n}$. Denote $\Psi : \mathcal{X}' \to \{\pm 1\}^{n'}$ such that $\Psi(\mathbf{x})_{ijk\epsilon} = \mathbf{1}\{x_{ijk} = \epsilon\} = (x, \neg x)$, where $\neg$ denotes negation. Denote $\Phi : \mathcal{Z}' \to \{\pm 1\}^{n'}$ such that $\Phi(V, \mathbf{w})_{ijk\epsilon} = \mathbf{1}\{w_{ij} \neq \epsilon\} \wedge \mathbf{1}\{V_{ij} = k\}$, where $V_{ij} \in [N/n]$ indicates which elements from the ij block was selected by V. Now, note that:

$$f_{V,\mathbf{w}}(\mathbf{x}) = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{4m^2} (\mathbf{x}|_V \oplus \mathbf{w})_{i,j} = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{4m^2} \bigvee_{k=1}^{N/n} \bigvee_{\epsilon \in \{\pm 1\}} ((x_{ijk} = \epsilon) \wedge (w_{ij} \neq \epsilon) \wedge (V_{ij} = k))$$

$$= \bigwedge_{i=1}^{m} \bigvee_{j=1}^{4m^2} \bigvee_{k=1}^{N/n} \bigvee_{\epsilon \in \{\pm 1\}} (\Psi(\mathbf{x})_{ijk\epsilon} \wedge \Phi(V, \mathbf{w})_{ijk\epsilon})$$

Finally, we define $\mathcal{X} = \Psi(\mathcal{X}')$ and $\mathcal{Z} = \Phi(\mathcal{Z}')$, and define

$$\mathcal{A} = \{(\mathcal{D}_{\Phi^{-1}(\mathbf{z})}, f_{\Phi^{-1}(\mathbf{z})}) \ : \ \mathbf{z} \in \mathcal{Z}\} = \{\left(f_{V,w} \circ \Psi^{-1}, \mathcal{D}_{V,w} \circ \Psi^{-1}\right) \ : \ (f_{V,w}, \mathcal{D}_{V,w}) \in \mathcal{A}'\}$$

Observe that $\mathcal{A}$ and $\mathcal{A}'$ are isomorphic, and therefore from Lemma 16 we get $\mathrm{Var}(\mathcal{A}) = \mathrm{Var}(\mathcal{A}') \leq 17^{-2m}$. ■

In the rest of this section, we show how Theorem 14 can be used to derive hardness results on approximation and learnability of DNFs and AC^0.

5.2 Hardness of Approximating DNFs using Linear Classes

Observe the family $\mathcal{A}$ as defined in Theorem 14. Note that for every $(f, \mathcal{D}) \in \mathcal{A}$, the function $\neg f$ is in fact a DNF. Using the fact that $\mathrm{Var}(\mathcal{A}) \leq 17^{-2m}$, along with Theorem 3, we get that for every mapping $\Psi : \mathcal{X} \to [-1, 1]^N$ we have:

$$\max_{(f,\mathcal{D}) \in \mathcal{A}} \min_{h \in \mathcal{H}_\Psi} L_{f(\mathcal{D})}(h) \geq \ell_0 - B\sqrt{N} 17^{-m}$$

Therefore, the following result is immediate:

Corollary 19 *Let ℓ be some convex loss function, satisfying $\ell(0, y) = \ell_0$ and $\ell'(0, y) = -y$. For every mapping $\Psi : \{\pm 1\}^n \to [-1, 1]^N$, there exists some DNF f and a distribution $\mathcal{D}$ over $\{\pm 1\}^n$ s.t. for all $\mathbf{w} \in \mathbb{R}^N$, the function $h_{\mathbf{w}}(\mathbf{x}) = \langle \mathbf{w}, \Psi(\mathbf{x}) \rangle$ has loss lower-bounded by:*

$$L_{f(\mathcal{D})}(h_{\mathbf{w}}) \geq \ell_0 - \frac{\|\mathbf{w}\| \sqrt{N}}{2^{\Omega(n^{1/3})}}$$

Note that DNFs are known to be learnable using polynomial threshold functions of degree $\tilde{O}(n^{1/3})$, which can be implemented using a linear classifier over a mapping $\Psi : \{\pm 1\}^n \to [-1, 1]^N$, for $N = 2^{\tilde{O}(n^{1/3})}$ (Klivans and Servedio, 2004). Our result shows that no linear class can approximate DNFs unless $N = 2^{\Omega(n^{1/3})}$, regardless of the choice of Ψ. This extends the result in Razborov and Sherstov (2010), which shows a similar bound on the dimension of a linear class that is required to exactly express DNFs.

5.3 Hardness of Approximating AC^0 using Shallow Networks

In Theorem 14 we showed a family $\mathcal{A}$ over $\mathcal{X} \subseteq \{\pm 1\}^n$, where every $(f, \mathcal{D}) \in \mathcal{A}$ is identified with $\mathbf{z} \in \mathcal{Z}$, with $\mathcal{Z} \subseteq \{\pm 1\}^n$, such that $f(\mathbf{x}) = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{dm^2}(x_{ij} \wedge z_{ij})$. So, we can define the function $F : \mathcal{X} \times \mathcal{Z} \to \{\pm 1\}$, induced from the family $\mathcal{A}$, such that $F(\mathbf{x}, \mathbf{z}) = \bigwedge_{i=1}^{m} \bigvee_{j=1}^{dm^2}(x_{ij} \wedge z_{ij})$. Observe that $F \in \mathrm{AC}^0$, where AC^0 is the class of polynomial-size constant-depth AND/OR circuits. Then, Theorem 4 implies the following:

Corollary 20 *Let $\mathcal{X} = \{\pm 1\}^n$, and let ℓ be a 1-Lipschitz convex loss satisfying $\ell(0, y) = \ell_0$ and $\ell'(0, y) = -y$. There exists a function $F : \mathcal{X} \to \{\pm 1\}$ such that $F \in \mathrm{AC}^0$, and a distribution $\mathcal{D}'$ over $\mathcal{X}$, such that for every neural-network g, with 1-Lipschitz activation, using k neurons with weights of L_2-norm at most R, has loss lower-bounded by:*

$$L_{F(\mathcal{D}')}(g) \geq \ell_0 - \frac{\sqrt{k} R^2 n^{5/6}}{2^{\Omega(n^{1/3})}}$$

This extends the result in Razborov and Sherstov (2010), which shows that such function cannot be exactly implemented using a threshold circuit, unless its size is $2^{\Omega(n^{1/3})}$.

5.4 Hardness of Learning DNFs

As noted, the family $\mathcal{A}$ from Theorem 14 defines a family of distributions labeled by DNF formulas. In Theorem 7, we showed that families with small variance are hard to learn from correlation queries. Therefore, we get an exponential lower bound on the number of queries required for learning DNF formulas from correlation queries, with respect to the hinge-loss:

Corollary 21 *For any $\tau > 0$, any statistical-query algorithm that makes only correlation queries needs at least $\tau^2 2^{\Omega(n^{1/3})}$ queries to achieve hinge-loss $< 1 - \tau$, square loss $< 1 - 2\tau$ or zero-one loss $< \frac{1}{2} - \frac{1}{2}\tau$, on DNF formulas of dimension n.*

Following these results, using Theorem 13 shows that any hypothesis class (with bounded gradients), optimized with gradient-descent on the hinge-loss, will need at least $\Omega(n^{1/3})$ gradient-iterations to approximate the family of DNF formulas:

Corollary 22 *Assume we initialize $\mathbf{w}_0$ from some distribution $\mathcal{W}$ such that almost surely, for every $\mathbf{x} \in \mathcal{X}$ we have $|h_{\mathbf{w}_0}(\mathbf{x})| \leq \frac{1}{2}$. Assume that every $h_{\mathbf{w}} \in \mathcal{H}$ is differentiable and satisfies $\|\nabla_{\mathbf{w}} h_{\mathbf{w}}(\mathbf{x})\| \leq B$ for all $\mathbf{x} \in \mathcal{X}$. Then, there exists some DNF f and distribution $\mathcal{D}$ such that the noisy gradient-descent algorithm with the hinge-loss requires at least $2^{\Omega(n^{1/3})} \frac{\Delta^2}{B^2 N}$ steps to approximate f with respect to $\mathcal{D}$.*

Note that these hardness results match the currently known upper bound of learning DNFs in time $2^{\tilde{O}(n^{1/3})}$, due to Klivans and Servedio (2004). While these results apply only to a restricted family of algorithms (namely, correlation query algorithms and gradient-descent), we hope similar techniques can be used to show such hardness results for a broader family of algorithms.

References

Emmanuel Abbe and Colin Sandon. Provable limitations of deep learning. *arXiv preprint arXiv:1812.06369*, 2018.

Zeyuan Allen-Zhu and Yuanzhi Li. What can resnet learn efficiently, going beyond kernels? In *Advances in Neural Information Processing Systems*, pages 9017–9028, 2019.

Zeyuan Allen-Zhu and Yuanzhi Li. Backward feature correction: How deep learning performs deep learning. *arXiv preprint arXiv:2001.04413*, 2020.

Shai Ben-David, Nadav Eiron, and Hans Ulrich Simon. Limitations of learning via embeddings in euclidean half spaces. *Journal of Machine Learning Research*, 3(Nov):441–461, 2002.

Avrim Blum, Merrick Furst, Jeffrey Jackson, Michael Kearns, Yishay Mansour, and Steven Rudich. Weakly learning dnf and characterizing statistical query learning using fourier analysis. In *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing*, pages 253–262, 1994.

Avrim Blum, Adam Kalai, and Hal Wasserman. Noise-tolerant learning, the parity problem, and the statistical query model. *Journal of the ACM (JACM)*, 50(4):506–519, 2003.

Nader H Bshouty and Vitaly Feldman. On using extended statistical queries to avoid membership queries. *Journal of Machine Learning Research*, 2(Feb):359–395, 2002.

Nadav Cohen, Or Sharir, and Amnon Shashua. On the expressive power of deep learning: A tensor analysis. In *Conference on learning theory*, pages 698–728, 2016.

Amit Daniely. Depth separation for neural networks. *arXiv preprint arXiv:1702.08489*, 2017.

Amit Daniely and Eran Malach. Learning parities with neural networks. *arXiv preprint arXiv:2002.07400*, 2020.

Amit Daniely and Shai Shalev-Shwartz. Complexity theoretic limitations on learning dnf's. In *Conference on Learning Theory*, pages 815–830, 2016.

Olivier Delalleau and Yoshua Bengio. Shallow vs. deep sum-product networks. In *Advances in neural information processing systems*, pages 666–674, 2011.

Ronen Eldan and Ohad Shamir. The power of depth for feedforward neural networks. In *Conference on learning theory*, pages 907–940, 2016.

Vitaly Feldman. Evolvability from learning algorithms. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 619–628, 2008.

Vitaly Feldman. Distribution-independent evolvability of linear threshold functions. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 253–272, 2011.

Vitaly Feldman. A complete characterization of statistical query learning with applications to evolvability. *Journal of Computer and System Sciences*, 78(5):1444–1459, 2012.

Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh S Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *Journal of the ACM (JACM)*, 64(2):1–37, 2017.

Jürgen Forster and Hans Ulrich Simon. On the smallest possible dimension and the largest possible margin of linear arrangements representing given concept classes. *Theoretical Computer Science*, 350(1):40–48, 2006.

Surbhi Goel, Aravind Gollakota, Zhihan Jin, Sushrut Karmalkar, and Adam Klivans. Superpolynomial lower bounds for learning one-layer neural networks using gradient descent. *arXiv preprint arXiv:2006.12011*, 2020.

Boris Hanin and David Rolnick. Complexity of linear regions in deep networks. *arXiv preprint arXiv:1901.09021*, 2019.

Pritish Kamath, Omar Montasser, and Nathan Srebro. Approximate is good enough: Probabilistic variants of dimensional and margin complexity. *arXiv preprint arXiv:2003.04180*, 2020.

Michael Kearns. Efficient noise-tolerant learning from statistical queries. *Journal of the ACM (JACM)*, 45(6):983–1006, 1998.

Michael J Kearns, Robert E Schapire, and Linda M Sellie. Toward efficient agnostic learning. *Machine Learning*, 17(2-3):115–141, 1994.

Adam R Klivans and Rocco A Servedio. Learning dnf in time $2^{\tilde{o}(n^{1/3})}$. *Journal of Computer and System Sciences*, 68(2):303–318, 2004.

Eran Malach and Shai Shalev-Shwartz. Is deeper better only when shallow is good? In *Advances in Neural Information Processing Systems*, pages 6429–6438, 2019.

James Martens, Arkadev Chattopadhya, Toni Pitassi, and Richard Zemel. On the representational efficiency of restricted boltzmann machines. In *Advances in Neural Information Processing Systems*, pages 2877–2885, 2013.

Guido Montúfar. Notes on the number of linear regions of deep neural networks. *Sampling Theory Appl., Tallinn, Estonia, Tech. Rep*, 2017.

Razvan Pascanu, Guido Montufar, and Yoshua Bengio. On the number of response regions of deep feed forward networks with piece-wise linear activations. *arXiv preprint arXiv:1312.6098*, 2013.

Maithra Raghu, Ben Poole, Jon Kleinberg, Surya Ganguli, and Jascha Sohl-Dickstein. On the expressive power of deep neural networks. In *international conference on machine learning*, pages 2847–2854, 2017.

Alexander A Razborov and Alexander A Sherstov. The sign-rank of AC^0. *SIAM Journal on Computing*, 39(5):1833–1855, 2010.

Itay Safran and Ohad Shamir. Spurious local minima are common in two-layer relu neural networks. In *International Conference on Machine Learning*, pages 4433–4441, 2018.

Thiago Serra, Christian Tjandraatmadja, and Srikumar Ramalingam. Bounding and counting linear regions of deep neural networks. In *International Conference on Machine Learning*, pages 4558–4566, 2018.

Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Failures of gradient-based deep learning. *arXiv preprint arXiv:1703.07950*, 2017.

Ohad Shamir. Distribution-specific hardness of learning neural networks. *The Journal of Machine Learning Research*, 19(1):1135–1163, 2018.

Alexander A Sherstov. Halfspace matrices. *Computational Complexity*, 17(2):149–178, 2008.

Matus Telgarsky. Representation benefits of deep feedforward networks. *arXiv preprint arXiv:1509.08101*, 2015.

Matus Telgarsky. Benefits of depth in neural networks. *arXiv preprint arXiv:1602.04485*, 2016.

Gal Vardi and Ohad Shamir. Neural networks with small weights and depth-separation barriers. *arXiv preprint arXiv:2006.00625*, 2020.

Ke Yang. New lower bounds for statistical query learning. *Journal of Computer and System Sciences*, 70(4):485–509, 2005.

Gilad Yehudai and Ohad Shamir. On the power and limitations of random features for understanding neural networks. In *Advances in Neural Information Processing Systems*, pages 6598–6608, 2019.