UNIVERSITY OF CALIFORNIA

Los Angeles

Statistical Learning with Neural Networks Trained by Gradient Descent

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Statistics

by

Spencer Frei

2021

ABSTRACT OF THE DISSERTATION

Statistical Learning with Neural Networks Trained by Gradient Descent

by

Spencer Frei
Doctor of Philosophy in Statistics
University of California, Los Angeles, 2021
Professor Quanquan Gu, Co-Chair
Professor Ying Nian Wu, Co-Chair

In this thesis, we theoretically analyze the ability of neural networks trained by gradient descent to learn. The learning problem consists of an algorithmic component and a statistical component. The algorithmic question concerns the underlying optimization problem: given samples from a distribution, under what conditions can a neural network trained by gradient descent efficiently minimize the empirical risk for some loss function defined over these samples? As the underlying optimization problem is highly non-convex, standard tools from optimization theory are not applicable and thus a novel analysis is needed. The statistical question concerns the generalization problem: supposing gradient descent is successful at minimizing the empirical risk, under what conditions does this translate to a guarantee for the population risk? Contemporary neural networks used in practice are highly overparameterized and are capable of minimizing the empirical risk even when the true labels are replaced with random noise, and thus standard uniform convergence-based arguments will fail to yield meaningful guarantees for the population risk for these models.

We begin our thesis by analyzing the simplest nontrivial neural network possible: a single neuron with a nonlinear activation function under the squared loss. Even this simple network induces a highly non-convex optimization problem. By showing that an approximate surrogate risk is minimized throughout the gradient descent trajectory, we show that gradient descent is able to learn single neurons for a large class of nonlinear activation functions. Our results hold in the agnostic setting, implying that gradient descent succeeds even when the model is mis-specified.

We continue our analysis of the single neuron by examining the classification setting, where the loss of interest is the zero-one loss rather than the squared loss. As the decision boundary for single neurons in the classification setting is identical to that of linear classifiers for typical activation functions, we focus on the linear classifier setting. This reduces the problem to that of learning halfspaces with noise, a long-studied problem in computational learning theory with well-established computational hardness constraints on the learning problem due to the non-convexity of the zero-one loss. We establish connections between minimizers of convex surrogates of the zero-one loss and minimizers of the zero-one loss itself to develop the first positive guarantees for gradient descent on convex loss functions for learning halfspaces with agnostic noise.

We then establish guarantees for learning halfspaces with agnostic noise when using overparameterized SGD-trained two layer nonlinear neural networks. Our analysis requires both overcoming the non-convexity of the underlying optimization problem as well as avoiding generalization bounds that become vacuous when the number of parameters in the neural network becomes large.

In our final contribution, we derive generalization bounds for overparameterized deep residual networks trained by gradient descent. Our techniques leverage a recently developed correspondence between large, overparameterized neural networks and the tangent kernels of their infinite width approximations known as the neural tangent kernel.

The dissertation of Spencer Frei is approved.

Qing Zhou

Arash Amini

Ying Nian Wu, Committee Co-Chair

Quanquan Gu, Committee Co-Chair

University of California, Los Angeles

2021

For Jeff

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

It is an impossible task to name all of the individuals that made the completion of this dissertation possible, so please forgive the following attempt.

First and foremost, I wish to thank my two Ph.D co-supervisors, Ying Nian Wu and Quanquan Gu. Ying Nian is a constant source of inspiration for his deep intuitive understanding of so many disparate areas of statistics and machine learning. Ying Nian is one of the nicest and most down-to-earth people I have ever met.

Ying Nian introduced me to Quanquan in September of 2018 after he joined the CS department at UCLA. Shortly after our introduction, my interest and motivation for research surged dramatically, and it quickly became apparent that my particular interests in the theory of machine learning were perfectly aligned with Quanquan's research expertise. The year we met coincided with an explosion of research in the theory of deep learning, and was a very exciting time to begin to do research in the area.

I worked with Quanquan and his postdoc Yuan Cao on all of the problems contained in this thesis. They carefully read my work and helped find improvements to my arguments and presentation of results. They both have a keen sense of which research problems are worthwhile, interesting, and reasonable to pursue, which are particularly important skills in as fast-moving a field as the one we find ourselves in. I owe a significant amount of gratitude to both Quanquan and Yuan for showing me the ropes of machine learning theory and optimization, and would be a much worse researcher without their guidance and expertise. I am particularly indebted to Quanquan, who has demonstrated an enormous dedication to his students and to ensuring their success. I was extremely fortunate to have had the opportunity to work under his supervision for my Ph.D.

I am thankful to Arash Amini for providing useful guidance on a number of projects I worked on during my thesis. I want to thank Ariana Anderson for supporting me as a graduate student researcher throughout the majority of my Ph.D. I am thankful for Qing

Zhou's guidance as a member of my committee, as well as the rest of the Statistics department faculty and staff for providing an enriching learning environment during my time at UCLA.

I am thankful for my colleagues and collaborators in the statistics, mathematics, and computer science departments at UCLA and elsewhere. In particular, my officemates and academic siblings Sam Baugh, Levon Demirdjian, Junhyung Park, Gabriel Ruiz, Stephanie Stacy, Zachary Stokes, Qiaoling Ye, and Yifei Xu; and fellow machine learning researchers Niladri Chatterji, Zixiang Chen, Yoni Dukler, Surbhi Goel, Erik Nijkamp, Bo Pang, Pan Xu, Difan Zou, and Dongruo Zhou. I am also thankful to my fellow UAW 2865 members and organizers on campus and across the UCs, who made my time at UCLA a more fulfilling experience. This includes Ted Everhart, Kavitha Iyengar, Jonathan Koch, Michael McCown, Michael Stenovec, Garrett Strain, and many others.

Finally, I could not have completed this work without the steadfast support of my parents, Susan and Steve; my brother, Garret; and my partner of nearly ten years, Jeffrey Dymond. I am so thankful I have had your support and love for so long.

VITA

2013 BSc. Mathematics, McGill University, Montréal. First class honours.

2015 MSc. Mathematics, University of British Columbia, Vancouver.

2015–2020 Teaching Assitant, UCLA Department of Statistics.

2016–2021 Research Assistant, UCLA School of Medicine and UCLA Department of Psychiatry.

2016–2018 Statistical Consultant, BlackThorn Therapeutics/UCLA, Los Angeles.

2017–2019 Biostatistical Consultant, Ritter Pharmaceuticals, Los Angeles.

2020 Research Scientist Intern, Amazon Alexa AI, Cambridge, Massachusetts.

2021–2022 Postdoctoral Fellow, UC Berkeley Department of Statistics and the Simons Institute for the Theory of Computing.

PUBLICATIONS

[1] Difan Zou*, Spencer Frei*, and Quanquan Gu. Provable robustness of adversarial training for learning halfspaces with noise. In *International Conference on Machine Learning (ICML)*, 2021.

[2] Spencer Frei, Yuan Cao, and Quanquan Gu. Provable generalization of SGD-trained neural networks of any width in the presence of adversarial label noise. In *International*

Conference on Machine Learning (ICML), 2021.

[3] Spencer Frei, Yuan Cao, and Quanquan Gu. Agnostic learning of halfspaces with gradient descent via soft margins. In *International Conference on Machine Learning (ICML)* (long talk), 2021.

[4] Spencer Frei, Yuan Cao, and Quanquan Gu. Agnostic learning of a single neuron with gradient descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[5] Ariana E. Anderson, Mirella Diaz-Santos, Spencer Frei et al. Hemodynamic latency is associated with reduced intelligence across the lifespan: an fMRI DCM study of aging, cerebrovascular integrity, and cognitive ability. *Brain Structure and Function*, 2020.

[6] Spencer Frei, Yuan Cao, and Quanquan Gu. Algorithm-dependent generalization bounds for overparameterized deep residual networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[7] S. Frei and E. Perkins. A lower bound for p_c in range-R bond percolation in two and three dimensions. *Electronic Journal of Probability* 21(56), 2016.

CHAPTER 1

Introduction

Although artificial neural networks have been an object of study for statisticians and computer scientists for decades—the first edition of the now-famous Neural Information Processing Systems (NeurIPS) conference was in 1987—it is only in the past decade that these models have begun to attract significant and widespread attention in the broader scientific community. The deep learning revolution can partially be traced to the success of multilayer convolutional networks in the 2012 the ImageNet LSVRC-2012 competition. This competition is based upon an image classification task where one has access to a million samples of labeled images that can be used to learn a classifier that aims to classify colour images of around 250×250 pixels into one of 1,000 classes [DDS09]. The large dimensionality of the training data (10^6 images with over 10^5 pixels per image), together with the large number of classes, contributed to the understanding among scientists that achieving near-human level accuracy of 5% top-5 error would constitute a major breakthrough in computer vision [DDS09]. The best top-5 errors at the first two LSVRC competitions were 28.1% and 25.7%, held in 2010 and 2011 respectively. At the 2012 ImageNet LSVRC competition, the winning deep neural network classifier achieved a top-5 error of 15.3%, with the next-best competitor achieving 26.2% top-5 error using kernel-based methods [KSH12]. Future ImageNet competitions were dominated by deep neural networks, progressively lowering the top-5 error rate to 12.5% in 2013 and 8.4% in 2014, largely due to using larger and more complex neural network model classes and increased computational abilities that took advantage of the particular computing power of graphical processing units (GPUs). In 2015, deep neural networks broke the 5% 'human-level' error threshold for the first time, leading to widespread news coverage in venues like the *New York Times* [Mar15]. Progress on ImageNet has proceeded apace, with the most recent state-of-the-art model achieving 1.2% top-5 error [PDX20] with a nearly 500,000,000 parameter model. Beyond image classification tasks, deep neural network-based models have become a dominant framework in almost every conceivable machine learning task that involves learning from labeled data, in such areas as robotics, natural language understanding and automatic translation, gaming, and medical imaging.

The success of deep learning has been a surprising phenomenon to theorists in the statistics and machine learning communities. From a theoretical perspective, that modern deep neural networks (such as those that have dominated computer vision competitions) are able to perform so well is surprising in two respects: the 'optimization' or algorithmic question and the 'generalization' or statistical question. From the algorithmic perspective, the optimization problem for deep neural networks has long been known to be non-convex: there exist local minima for the objective functions of neural networks, even when the neural network consists of a single neuron with a sigmoidal nonlinearity [AHW95]. And yet the standard algorithmic framework used for learning neural networks is gradient-based optimization (e.g. stochastic gradient descent). The optimization question concerns how a local optimization method is able to find approximately global optima of the objective function, even when the underlying problem is highly nonconvex.

From the statistical perspective, that deep neural networks with low training error are able to achieve low *test* error (i.e. on unseen data) is curious. One-hidden-layer networks have long been known to be universal approximators of continuous functions [Cyb89], and such networks have the capacity to perfectly fit any random labeling of training data with binary labels (provided they are sufficiently wide). Typical uniform convergence-based complexity arguments, which roughly state that the greater the capacity of a model class to fit random labels the more prone the model class is to suffer from overfitting, thus fail to explain why neural networks can generalize.

The close connection between the two questions of optimization and generalization in deep learning was most clearly established in a landmark paper that appeared at the International Conference on Learning Representations (ICLR) in 2017 by Chiyuan Zhang and co-authors called 'Understanding deep learning requires rethinking generalization' [ZBH17]. The authors constructed a series of revealing experiments on the CIFAR-10 dataset, which consists of 50,000 training samples of 32×32 RGB pixel images from 10 classes, and introduced label noise to $p \in [0,1]$ fraction of the training samples, where each sample's label was assigned randomly with probability p. For each $p \in [0,1]$, they showed that SGD-training of the neural networks produce networks with 100% training accuracy. That is, SGD-training produces *interpolating* classifiers, even when the labels are replaced with pure noise, thereby showing that SGD-trained neural networks indeed have the capacity to perfectly fit randomly-labeled data. Next, they considered the generalization performance of the networks which were trained on p-corrupted data on the (uncorrupted) test set. For $p = 1$, the neural networks achieved a 10% test error rate, matching that of random guessing. Most remarkably, when trained on p-corrupted data, the SGD-trained networks achieved a test error rate that was highly correlated with the proportion of corrupted data p: data with 20% label corruption resulted in 40% test error and data with 50% label corruption resulted in 65% test error.

These experiments were, simply put, shocking. Here is a model class which demonstrably is able to interpolate randomly-labeled training data, even though the underlying optimization problem is non-convex and a local optimization method is used to learn the model. Moreover, even in settings where half of the labels are replaced with random noise, the model is still able to generalize significantly better than that of random guessing. The standard frameworks of optimization, statistics, and machine learning are not easily reconcilable with either of these phenomena in isolation; their confluence in a single problem is all the more remarkable.

And now, nine years on from "the" ImageNet competition, and almost five years on from the "Rethinking generalization" paper, we are still exploring the intricacies of both the optimization and generalization questions in deep learning. Under what conditions can neural networks trained by gradient descent provably achieve small training error? Under what conditions can such networks generalize well to unseen data? How does the presence of noise affect the answers to these questions? Do our answers to these questions require the development of new mathematical frameworks?

This thesis collects our attempts at answering these questions. We begin in Chapter 2 by focusing on the simplest non-trivial neural network possible, namely, a neural network consisting of a single neuron $x \mapsto \sigma(\langle w, x\rangle)$ with a fixed activation function $\sigma : \mathbb{R} \to \mathbb{R}$. When σ is linear, the problem of learning the best-fitting single neuron (as measured by the ℓ^2 loss) over a distribution is simple, as the model is simply a linear model and so the underlying optimization problem is convex. In particular, if we define

$$\mathsf{OPT}_\sigma^{\ell_2} := \min_{w \in \mathbb{R}^d : \|w\| \leqslant 1} \mathbb{E}_{(x,y)\sim \mathcal{D}} \left[(\sigma(\langle w, x\rangle) - y)^2 \right],$$

then for $\sigma(x) = x$, the standard gradient descent algorithm can learn the best single neuron up to risk $\mathsf{OPT}_\sigma^{\ell_2}$ in polynomial time and sample complexity. However, when σ is nonlinear, the problem becomes significantly harder, as the underlying problem is nonconvex: the empirical risk has many local minima, and so it is not clear how standard gradient-based methods would succeed in minimizing the empirical risk. Indeed, the problem of learning up to the risk $\mathsf{OPT}_\sigma^{\ell_2}$ cannot be done in polynomial time with the standard gradient descent algorithm when $\sigma(x) = \mathrm{ReLU}(x) = \max(0, x)$, even when the marginal of $\mathcal{D}$ over x is the standard Gaussian [GKK19]. In this thesis, we show that when the activation function σ is strictly increasing, then the standard gradient descent algorithm is able to learn up to risk $O(\mathsf{OPT}_\sigma^{\ell_2})$ in polynomial time, and up to risk $O\big((\mathsf{OPT}_{\mathrm{ReLU}}^{\ell_2})^{1/2}\big)$ in polynomial time when σ is the ReLU.

In Chapter 3, we then consider the problem of learning the best single neuron for classification problems, i.e. when the loss is the zero-one loss rather than the squared loss in the regression case. When σ is an odd function, since $y \cdot \sigma(\langle w, x\rangle) > 0$ iff $y \cdot \langle w, x\rangle > 0$, this is equivalent to the problem of learning the best halfspace over a distribution, and is a long-studied problem in the computational learning theory community. Let us denote the best halfspace error by

$$\mathsf{OPT}_{01} := \min_{\|w\|=1} \mathbb{P}_{(x,y)\sim \mathcal{D}}(y \neq \mathrm{sgn}(\langle w, x\rangle)).$$

Even learning up to risk $O(\mathsf{OPT}_{01})$ is known to be computationally hard without assumptions on the marginal distribution of $\mathcal{D}$ [Dan16]. Some rather complicated algorithms have been able to show that if OPT_{01} is sufficiently small, then when $\mathcal{D}_x$ satisfies certain properties (e.g. isotropic and has a log-concave probability density function), it is possible to learn halfspaces up to risk $O(\mathsf{OPT}_{01})$ [ABH15, ABH16, ABL17]. In this thesis, we show that the standard logistic regression algorithm is able to learn halfspaces up to risk $\tilde{O}(\mathsf{OPT}_{01}^{1/2})$ when $\mathcal{D}_x$ satisfies an anti-concentration property enjoyed by log-concave isotropic distributions among others. We show improved guarantees of a classification error of at most $\tilde{O}(\mathsf{OPT}_{01})$ when $\mathcal{D}_x$ comes from a noisy large margin distribution.

In Chapter 4, we consider one-hidden-layer neural networks consisting of m neurons,

$$f(x; W) = \sum_{j=1}^{m} a_j \sigma(\langle w_j, x\rangle).$$

We consider networks with leaky ReLU activations, $\sigma(z) = \max(\alpha z, z)$, for $\alpha \in (0, 1]$. We show that SGD-training of such networks will lead to classifiers that are competitive with the best halfspace over the distribution when $\mathcal{D}_x$ satisfies anti-concentration properties enjoyed by log-concave isotropic distributions. Equivalently, overparameterized one-hidden-layer neural networks are able to generalize on linearly separable data where the labels have been corrupted by an adversary. The analysis requires overcoming both (1) the highly non-convex nature of the underlying optimization problem, and (2) showing that a model which can overfit noisy training data can still generalize.

In the final chapter, we consider deep residual networks, which are defined recursively as

$$x_1 = \sigma(W_1 x), \quad x_l = x_{l-1} + \theta\sigma(W_l x_{l-1}), \quad l = 2, \ldots, L, \quad f(x; W) = W_{L+1} x_L.$$

Here, $W_1 \in \mathbb{R}^{m\times d}$, $W_l \in \mathbb{R}^{m\times m}$ for $l = 2, \ldots, L$, and $W_{L+1} \in \mathbb{R}^{1\times m}$, and θ is a scaling factor. We show that with a proper random initialization, if the underlying data distribution can be classified using an infinite-width neural network, gradient descent-trained oveparameterized networks with the above architecture will generalize well to unseen data from the distribution. This analysis relies upon the neural tangent kernel approximation, which shows that under a proper scaling, overparameterized neural networks can be approximated by their tangent kernel around their initialization. The existence of the neural tangent kernel approximation was first shown by Jacot et al. in 2018 [JGH18], with a flurry of works in 2018–2019 using the approximation to generate optimization and generalization guarantees for SGD-trained neural networks [DZP19, ADH19b, ZCZ19, ALS19, CG20, ADH19a, FCG19, COB19]. We conclude by pointing to a number of open questions that remain in understanding neural networks trained by gradient descent.

CHAPTER 2

Learning a single neuron with gradient descent

2.1 Introduction

In this chapter, we consider the problem of learning a single neuron with gradientd escent. Consider input features $x \in \mathbb{R}^d$ and output labels $y \in \mathbb{R}$ distributed according to $(x, y) \sim \mathcal{D}$. Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an activation function, and consider the problem of learning the best-fitting single neuron with activation function σ as measured by the squared loss. In particular, we define the population risk $F(w)$ associated with a set of weights w as

$$F(w) := (1/2)\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\left(\sigma(w^\top x) - y\right)^2\right]. \tag{2.1.1}$$

The activation function is assumed to be non-decreasing and Lipschitz, and includes nearly all activation functions used in neural networks such as the rectified linear unit (ReLU), sigmoid, tanh, and so on. In the agnostic PAC learning setting [KSS94], no structural assumption is made regarding the relationship of the input and the label, and so the best-fitting neuron could, in the worst case, have nontrivial population risk. Concretely, if we denote

$$v := \mathrm{argmin}_{\|w\|_2 \leqslant 1} F(w), \quad \mathsf{OPT} := F(v), \tag{2.1.2}$$

then the goal of a learning algorithm is to (efficiently) return weights w such that the population risk $F(w)$ is close to the best possible risk OPT. The agnostic learning framework stands in contrast to the *realizable* PAC learning setting, where one assumes $\mathsf{OPT} = 0$, so that there exists some v such that the labels are given by $y = \sigma(v^\top x)$.

The learning algorithm we consider in this chapter is empirical risk minimization using vanilla gradient descent. We assume we have access to a set of i.i.d. samples $\{(x_i, y_i)\}_{i=1}^n \sim \mathcal{D}^n$, and we run gradient descent with a fixed step size on the empirical risk $\widehat{F}(w) = (1/2n)\sum_{i=1}^n (\sigma(w^\top x_i) - y_i)^2$. A number of early neural network studies pointed out that the landscape of the empirical risk of a single neuron has unfavorable properties, such as a large number of spurious local minima [BRS89, AHW95], and led researchers to instead study gradient descent on a convex surrogate loss [HKW95, HKW99]. Despite this, we are able to show that gradient descent on the empirical risk itself finds weights that not only have small empirical risk but small population risk as well.

Surprisingly little is known about neural networks trained by minimizing the empirical risk with gradient descent in the agnostic PAC learning setting. We are aware of two works [ALL19, AL19] in the *improper* agnostic learning setting, where the goal is to return a hypothesis $h \in \mathcal{H}$ that achieves population risk close to $\widehat{\mathsf{OPT}}$, where $\widehat{\mathsf{OPT}}$ is the smallest possible population risk achieved by a different set of hypotheses $\widehat{\mathcal{H}}$. Another work considered the random features setting where only the final layer of the network is trained and the marginal distribution over x is uniform on the unit sphere [VW19]. But none of these address the simplest possible neural network: that of a single neuron $x \mapsto \sigma(w^\top x)$. We believe a full characterization of what we can (or cannot) guarantee for gradient descent in the single neuron setting will help us understand what is possible in the more complicated deep neural network setting. Indeed, two of the most common hurdles in the analysis of deep neural networks trained by gradient descent—nonconvexity and nonsmoothness—are also present in the case of the single neuron. We hope that our analysis in this relatively simple setup will be suggestive of what is possible in more complicated neural network models.

Our main contributions can be summarized as follows.

1) **Agnostic setting** (Theorem 2.3.3). Without any assumptions on the relationship between y and x, and assuming only boundedness of the marginal distributions of x and y, we show that for any $\varepsilon > 0$, gradient descent finds a point w_t with population risk $O(\mathsf{OPT}) + \varepsilon$ with sample complexity $O(\varepsilon^{-2})$ and runtime $O(\varepsilon^{-1})$ when $\sigma(\cdot)$ is strictly increasing and Lipschitz. When σ is ReLU, we obtain a population risk guarantee of $O(\mathsf{OPT}^{1/2}) + \varepsilon$ with sample complexity $O(\varepsilon^{-4})$ and runtime $O(\varepsilon^{-2})$ when the marginal distribution of x satisfies a nondegeneracy condition (Assumption 2.3.2). The sample and runtime complexities are independent of the input dimension for both strictly increasing activations and ReLU.

2) **Noisy teacher network setting** (Theorem 2.4.1). When $y = \sigma(v^\top x) + \xi$, where $\xi|x$ is zero-mean and sub-Gaussian (and possibly dependent on x), we demonstrate that gradient descent finds w_t satisfying $F(w_t) \leqslant \mathsf{OPT} + \varepsilon$ for activation functions that are strictly increasing and Lipschitz assuming only boundedness of the marginal distribution over x. The same result holds for ReLU under a marginal spread assumption (Assumption 2.3.2). The runtime and sample complexities are of order $\tilde{O}(\varepsilon^{-2})$, with a logarithmic dependence on the input dimension. When the noise is bounded, our guarantees are dimension independent. If we further know $\xi \equiv 0$, i.e. the learning problem is in the realizable rather than agnostic setting, we can improve the runtime and sample complexity guarantees from $O(\varepsilon^{-2})$ to $O(\varepsilon^{-1})$ by using online stochastic gradient descent (Theorem 2.9.1).

2.2 Related work

Below, we provide a high-level summary of related work in the agnostic learning and teacher network settings. Detailed comparisons with the most related works will appear after we present our main theorems in Sections 2.3 and 2.4. In Section 2.6, we provide tables that describe the assumptions and complexity guarantees of our work in comparison to related results.

Agnostic learning: The simplest version of the agnostic regression problem is to find a hypothesis that matches the performance of the best *linear* predictor. In our setting, this corresponds to σ being the identity function. This problem is completely characterized: [Sha15] showed that any algorithm that returns a linear predictor v has risk $\mathsf{OPT} + \Omega(\varepsilon^{-2} \wedge d\varepsilon^{-1})$ when the labels satisfy $|y| \leqslant 1$ and the marginal distribution over x is supported on the unit ball, matching upper bounds proved by [SST10] using mirror descent.

When σ is not the identity, related works are scarce. [GKK17] studied agnostic learning of the ReLU on distributions supported on the unit sphere but had runtime and sample complexity exponential in ε^{-1}. In another work on learning a single ReLU, [GKK19] showed that learning up to risk $\mathsf{OPT} + \varepsilon$ in polynomial time is as hard as the problem of learning sparse parities with noise, long believed to be computationally intractable. Additionally, they provided an approximation algorithm that could learn up to $O(\mathsf{OPT}^{2/3}) + \varepsilon$ risk in $\mathrm{poly}(d, \varepsilon^{-1})$ time and sample complexity when the marginal distribution over x is a standard Gaussian. In a related but incomparable set of results in the improper agnostic learning setting, [ALL19] and [AL19] showed that multilayer ReLU networks trained by gradient descent can match the population risk achieved by multilayer networks with smooth activation functions. [VW19] studied agnostic learning of a one-hidden-layer neural network when the first layer is fixed at its (random) initial values and the second layer is trained. A very recent work by [DGK20a] showed that population risk $O(\mathsf{OPT}) + \varepsilon$ can be achieved for the single ReLU neuron by appealing to gradient descent on a convex surrogate for the empirical risk.

Teacher network: The literature refers to the case of $y = \sigma(v^\top x) + \xi$ for some possible zero mean noise ξ variously as the "noisy teacher network" or "generalized linear model" (GLM) setting, and is related to the probabilistic concepts model [KS94]. In the GLM setting, σ plays the role of the inverse link function; in the case of logistic regression, σ is the sigmoid function.

The results in the teacher network setting can be broadly characterized by (1) whether they cover arbitrary distributions over x and (2) the presence of noise (or lackthereof). The GLMTron algorithm proposed by [KKK11], itself a modification of the Isotron algorithm of [KS09], is known to learn a noisy teacher network up to risk $\mathsf{OPT} + \varepsilon$ for any Lipschitz and non-decreasing σ and any distribution with bounded marginals over x. [MBM18] showed that gradient descent learns the noisy teacher network under a smoothness assumption on the activation function for a large class of distributions. [FSS18] provided a meta-algorithm for translating ε-stationary points of the empirical risk to points of small population risk in the noisy teacher network setting. A recent work by [MM20] develops a modified SGD algorithm for learning a ReLU with bounded adversarial noise on distributions where the input is bounded.

Of course, any guarantee that holds for a neural network with a single fully connected hidden layer of arbitrary width holds for the single neuron, so in this sense our work can be connected to a larger body of work on the analysis of gradient descent used for learning neural networks. The majority of such works are restricted to particular input distributions, whether it is Gaussian or uniform distributions [Sol17, Tia17, SJL19, ZYW19, GKM18, CG19b]. [DLT18] showed that in the noiseless (a.k.a. realizable) setting, a single neuron can be learned with SGD if the input distribution satisfies a certain subspace eigenvalue property. [YS20] studied the properties of learning a single neuron for a variety of increasing and Lipschitz activation functions using gradient descent, as we do in this chapter, although their analysis was restricted to the noiseless setting.

2.3 Agnostic learning setting

We begin our analysis by assuming there is no *a priori* relationship between x and y, so the population risk OPT of the population risk minimizer v defined in (2.1.2) may, in general, be a large quantity. If $\mathsf{OPT} = 0$, then $\sigma(v^\top x) = y$ a.s. and the problem is in the realizable PAC learning setting. In this case, we can use a modified proof technique to get stronger guarantees for the population risk; see Section 2.9 for the complete theorems and proofs in this setting. We will thus assume without loss of generality that $0 < \mathsf{OPT} \leqslant 1$.

The gradient descent method we use in this chapter is as follows. We assume we have a training sample $\{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathcal{D}^n$, and define the empirical risk for weight w by

$$\widehat{F}(w) = (1/2n) \textstyle\sum_{i=1}^n (\sigma(w^\top x_i) - y_i)^2.$$

We perform full-batch gradient updates on the empirical risk using a fixed step size η,

$$w_{t+1} = w_t - \eta \nabla \widehat{F}(w_t) = w_t - (\eta/n) \textstyle\sum_{i=1}^n (\sigma(w_t^\top x_i) - y_i)\sigma'(w_t^\top x_i) x_i, \qquad (2.3.1)$$

where $\sigma'(\cdot)$ is the derivative of $\sigma(\cdot)$. If σ is not differentiable at a point z, we will use its subderivative.

We begin by describing one set of activation functions under consideration in this chapter.

Assumption 2.3.1. (a) σ is continuous, non-decreasing, and differentiable almost everywhere.

(b) For any $\rho > 0$, there exists $\gamma > 0$ such that $\inf_{|z| \leqslant \rho} \sigma'(z) \geqslant \gamma > 0$. If σ is not differentiable at $z \in [-\rho, \rho]$, assume that every subderivative g on the interval satisfies $g(z) \geqslant \gamma$.

(c) σ is L-Lipschitz, i.e. $|\sigma(z_1) - \sigma(z_2)| \leqslant L|z_1 - z_2|$ for all z_1, z_2.

We note that if σ is strictly increasing and continuous, then σ satisfies Assumption 2.3.1(b) since its derivative is never zero. In particular, the assumption covers the typical activation functions in neural networks like leaky ReLU, softplus, sigmoid, tanh, etc., but excludes ReLU. [YS20] recently showed that when σ is ReLU, there exists a distribution $\mathcal{D}$ supported on the unit ball and unit length target neuron v such that *even in the realizable case* of $y = \sigma(v^\top x)$, if the weights are initialized randomly using a product distribution, there exists a constant c_0 such that with high probability, $F(w_t) \geqslant c_0 > 0$ throughout the trajectory of gradient descent. This suggests that gradient-based methods for learning ReLUs are likely to fail without additional assumptions. Because of this, they introduced the following marginal spread assumption to handle the learning of ReLU.

Assumption 2.3.2. There exist constants $\alpha, \beta > 0$ such that the following holds. For any $w \neq u$, denote by $\mathcal{D}_{w,u}$ the marginal distribution of $\mathcal{D}$ on $\mathrm{span}(w, u)$, viewed as a distribution over $\mathbb{R}^2$, and let $p_{w,u}$ be its density function. Then $\inf_{z \in \mathbb{R}^2 : \|z\| \leqslant \alpha} p_{w,u}(z) \geqslant \beta$.

This assumption covers, for instance, log-concave distributions like the Gaussian and uniform distribution with $\alpha, \beta = O(1)$ [LV07]. We note that a similar assumption was used in recent work on learning halfspaces with Massart noise [DKT20a]. We will use this assumption for all of our results when σ is ReLU. Additionally, although the ReLU is not differentiable at the origin, we will denote by $\sigma'(0)$ its subderivative, with the convention that $\sigma'(0) = 1$. Such a convention is consistent with the implementation of ReLUs in modern deep learning software packages.

With the above in hand, we can describe our main theorem.

Theorem 2.3.3. Suppose the marginals of $\mathcal{D}$ satisfy $\|x\|_2 \leqslant B_X$ a.s. and $|y| \leqslant B_Y$ a.s. Let $a := (|\sigma(B_X)| + B_Y)^2$. When σ satisfies Assumption 2.3.1, let $\gamma > 0$ be the constant corresponding to $\rho = 2B_X$ and fix a step size $\eta \leqslant (1/8)\gamma L^{-3} B_X^{-2}$. For any $\delta > 0$, with probability at least $1 - \delta$, gradient descent initialized at the origin and run for $T = \lceil \eta^{-1}\gamma^{-1}L^{-1}B_X^{-1}[\mathsf{OPT} + an^{-1/2}\log^{1/2}(4/\delta)]^{-1}\rceil$ iterations finds weights w_t, $t < T$, such that

$$F(w_t) \leqslant C_1\mathsf{OPT} + C_2 n^{-1/2}, \tag{2.3.2}$$

where $C_1 = 12\gamma^{-3}L^3 + 2$ and $C_2 = O(L^3 B_X^2\sqrt{\log(1/\delta)} + C_1 a\sqrt{\log(1/\delta)})$.

When σ is ReLU, further assume that $\mathcal{D}_x$ satisfies Assumption 2.3.2 for constants $\alpha, \beta > 0$, and let $\nu = \alpha^4\beta/8\sqrt{2}$. Fix a step size $\eta \leqslant (1/4)B_X^{-2}$. For any $\delta > 0$, with probability at least $1 - \delta$, gradient descent initialized at the origin and run for $T = \lceil \eta^{-1}B_X^{-1}[\mathsf{OPT} + an^{-1/2}\log^{1/2}(4/\delta)]^{-1/2}\rceil$ iterations finds a point w_t such that

$$F(w_t) \leqslant C_1\mathsf{OPT}^{1/2} + C_2 n^{-1/4} + C_3 n^{-1/2}, \tag{2.3.3}$$

where $C_1 = O(B_X\nu^{-1})$, $C_2 = O(C_1 a^{1/2}\log^{1/4}(1/\delta))$, and $C_3 = O(B_X^2\nu^{-1}\log^{1/2}(1/\delta))$.

We remind the reader that the optimization problem for the empirical risk is highly nonconvex [AHW95] and thus any guarantee for the empirical risk, let alone the population risk, is nontrivial. This makes us unsure if the suboptimal guarantee of $O(\mathsf{OPT}^{1/2})$ for ReLU is an artifact of our analysis or a necessary consequence of nonconvexity.

In comparison to recent work, [GKK19] considered the agnostic setting for the ReLU activation when the marginal distribution over x is a standard Gaussian and showed that learning up to risk $\mathsf{OPT} + \varepsilon$ is as hard as learning sparse parities with noise. By using an approximation algorithm of [ABL17], they were able to show that one can learn up to

$O(\mathsf{OPT}^{2/3}) + \varepsilon$ with $O(\mathrm{poly}(d, \varepsilon^{-1}))$ runtime and sample complexity. In a very recent work, [DGK20a] improved the population risk guarantee for the ReLU to $O(\mathsf{OPT}) + \varepsilon$ when the features are sampled from an isotropic log-concave distribution by analyzing gradient descent on a convex surrogate loss. Projected gradient descent on this surrogate loss produces the weight updates of the GLMTron algorithm of [KKK11]. Using the solution found by gradient descent on the surrogate loss, they proposed an improper learning algorithm that improves the population risk guarantee from $O(\mathsf{OPT}) + \varepsilon$ to $(1 + \delta)\mathsf{OPT} + \varepsilon$ for any $\delta > 0$.

By contrast, we show that gradient descent on the empirical risk learns up to a population risk of $O(\mathsf{OPT}) + \varepsilon$ for *any* joint distribution with bounded marginals when σ is strictly increasing and Lipschitz, even though the optimization problem is nonconvex. In the case of ReLU, our guarantee holds for the class of bounded distributions over x that satisfy the marginal spread condition of Assumption 2.3.2 and hence covers (bounded) log-concave distributions, although the guarantee is $O(\mathsf{OPT}^{1/2})$ in this case. For all activation functions we consider, the runtime and sample complexity guarantees do not have (explicit) dependence on the dimension.[1] Moreover, we shall see in the next section that if the data is known to come from a noisy teacher network, the guarantees of gradient descent improve to $\mathsf{OPT} + \varepsilon$ for both strictly increasing activations and ReLU.

In the remainder of this section we will prove Theorem 2.3.3. Our proof relies upon the following auxiliary errors for the true risk F:

$$\begin{aligned} G(w) &:= (1/2)\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\left(\sigma(w^\top x) - \sigma(v^\top x)\right)^2\right], \\ H(w) &:= (1/2)\mathbb{E}_{(x,y)\sim\mathcal{D}}\left[\left(\sigma(w^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w^\top x)\right]. \end{aligned} \tag{2.3.4}$$

We will denote the corresponding empirical risks by $\widehat{G}(w)$ and $\widehat{H}(w)$. We first note that G trivially upper bounds F: this follows by a simple application of Young's inequality and, when $\mathbb{E}[y|x] = \sigma(v^\top x)$, by using iterated expectations.

[1] We note that for some distributions, the B_X term may hide an implicit dependence on d; more detailed comments on this are given in Section 2.6.

Claim 2.3.4. For any joint distribution $\mathcal{D}$, for any vector u, and any continuous activation function σ, $F(u) \leqslant 2G(u) + 2F(v)$. If additionally we know that $\mathbb{E}[y|x] = \sigma(v^\top x)$, we have $F(u) = G(u) + F(v)$.

This claim shows that in order to show the population risk is small, it suffices to show that G is small. It is easy to see that if $\inf_{z\in\mathbb{R}} \sigma'(z) \geqslant \gamma > 0$, then $H(w) \leqslant \varepsilon$ implies $G(w) \leqslant \gamma^{-1}\varepsilon$, but the only typical activation function that satisfies this condition is the leaky ReLU. Fortunately, when σ satisfies Assumption 2.3.1, or when σ is ReLU and $\mathcal{D}$ satisfies Assumption 2.3.2, Lemma 2.3.5 below shows that H is still an upper bound for G. The proof is deferred to Appendix 2.7.

Lemma 2.3.5. If σ satisfies Assumption 2.3.1, $\|x\|_2 \leqslant B$ a.s., and $\|w\|_2 \leqslant W$, then for γ corresponding to $\rho = WB$, $H(w) \leqslant \varepsilon$ implies $G(w) \leqslant \gamma^{-1}\varepsilon$. If σ is ReLU and $\mathcal{D}$ satisfies Assumption 2.3.2 for some constants $\alpha, \beta > 0$, and if for some $\varepsilon > 0$ the bound $\overline{F}(w) \leqslant \beta\alpha^4\varepsilon/8\sqrt{2}$ holds, then $\|w - v\|_2 \leqslant 1$ implies $G(w) \leqslant \varepsilon$.

Claim 2.3.4 and Lemma 2.3.5 together imply that if gradient descent finds a point with auxiliary error $H(w_t) \leqslant O(\mathsf{OPT}^\alpha)$ for some $\alpha \leqslant 1$, then gradient descent achieves population risk $O(\mathsf{OPT}^\alpha)$. In the remainder of this section, we will show that this is indeed the case. In Section 2.3.1, we first consider activations satisfying Assumption 2.3.1, for which we are able to show $H(w_t) \leqslant O(\mathsf{OPT})$. In Section 2.3.2, we show $H(w_t) \leqslant O(\mathsf{OPT}^{1/2})$ for the ReLU.

2.3.1 Strictly increasing activations

In Lemma 2.3.6 below, we show that $\widehat{H}(w_t)$ is a natural quantity of the gradient descent algorithm that in a sense tells us how good a direction the gradient is pointing at time t, and that $\widehat{H}(w_t)$ can be as small as $O(\widehat{F}(v))$. Our proof technique is similar to that of [KKK11], who studied the GLMTron algorithm in the (non-agnostic) noisy teacher network setup.

Lemma 2.3.6. Suppose that $\|x\|_2 \leqslant B_X$ a.s. under $\mathcal{D}_x$. Suppose σ satisfies Assumption 2.3.1, and let γ be the constant corresponding to $\rho = 2B_X$. Assume $\widehat{F}(v) > 0$. Gradient descent with fixed step size $\eta \leqslant (1/8)\gamma L^{-3} B_X^{-2}$ initialized at $w_0 = 0$ finds weights w_t satisfying $\widehat{H}(w_t) \leqslant 6L^3\gamma^{-2}\widehat{F}(v)$ within $T = \lceil \eta^{-1}\gamma^{-1}L^{-1}B_X^{-1}\widehat{F}(v)^{-1} \rceil$ iterations, with $\|w_t - v\|_2 \leqslant 1$ for each $t = 0, \ldots, T-1$.

Before beginning the proof, we first note the following fact, which allows us to connect terms that appear in the gradient to the square loss.

Fact 2.3.7. If σ is strictly increasing on an interval $[a, b]$ with $\sigma'(z) \geqslant \gamma > 0$ for all $z \in [a, b]$, and if $z_1, z_2 \in [a, b]$, then, it holds that

$$\gamma(z_1 - z_2)^2 \leqslant (\sigma(z_1) - \sigma(z_2))(z_1 - z_2). \tag{2.3.5}$$

Proof of Lemma 2.3.6. The proof comes from the following induction statement. We claim that for every $t \in \mathbb{N}$, either (a) $\widehat{H}(w_\tau) \leqslant 6L^3\gamma^{-2}\widehat{F}(v)$ for some $\tau < t$, or (b) $\|w_t - v\|_2^2 \leqslant \|w_{t-1} - v\|_2^2 - \eta L\widehat{F}(v)$ holds. If this claim is true, then until gradient descent finds a point where $\widehat{H}(w_t) \leqslant 6L^3\gamma^{-2}\widehat{F}(v)$, the squared distance $\|w_t - v\|_2^2$ decreases by $\eta L\widehat{F}(v)$ at every iteration. Since $\|w_0 - v\|_2^2 = 1$, this means there can be at most $1/(\eta L\widehat{F}(v)) = \eta^{-1}L^{-1}\widehat{F}(v)^{-1}$ iterations until we reach $\widehat{H}(w_t) \leqslant 6L^3\gamma^{-2}\widehat{F}(v)$.

So let us now suppose the induction hypothesis holds for t, and consider the case $t+1$. If (a) holds, then we are done. So now consider the case that for every $\tau \leqslant t$, we have $\widehat{H}(w_\tau) > 6L^3\gamma^{-2}\widehat{F}(v)$. Since (a) does not hold, $\|w_\tau - v\|_2^2 \leqslant \|w_{\tau-1} - v\|_2^2 - \eta L\widehat{F}(v)$ holds for each $\tau = 1, \ldots, t$, and so $\|w_0 - v\|_2 = 1$ implies

$$\|w_\tau - v\|_2 \leqslant 1 \ \forall \tau \leqslant t. \tag{2.3.6}$$

In particular, $\|w_\tau\|_2 \leqslant 1 + \|v\|_2 \leqslant 2$ holds for all $\tau \leqslant t$. By Cauchy–Schwarz, this implies $|w_\tau^\top x| \vee |v^\top x| \leqslant 2B_X$ a.s. By defining $\rho = 2B_X$ and letting γ be the constant from Assumption 2.3.1, this implies $\sigma'(z) \geqslant \gamma > 0$ for all $|z| \leqslant 2B_X$. Fact 2.3.7 therefore implies

$$\sigma'(w_\tau^\top x) \geqslant \gamma > 0 \quad \text{and} \quad (\sigma(w_\tau^\top x) - \sigma(v^\top x)) \cdot (w_\tau^\top x - v^\top x) \geqslant \gamma(w_\tau^\top x - v^\top x)^2 \quad \forall \tau \leqslant t. \tag{2.3.7}$$

We proceed with the proof by demonstrating an appropriate lower bound for the quantity

$$\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 = 2\eta \left\langle \nabla \widehat{F}(w_t), w_t - v \right\rangle - \eta^2 \left\| \nabla \widehat{F}(w_t) \right\|_2^2 .$$

We begin with the inner product term. We have

$$\begin{aligned}
\left\langle \nabla \widehat{F}(w_t), w_t - v \right\rangle &= (1/n) \sum_{i=1}^n \left(\sigma(w_t^\top x_i) - \sigma(v^\top x_i) \right) \sigma'(w_t^\top x_i)(w_t^\top x_i - v^\top x_i) \\
&\quad + (1/n) \sum_{i=1}^n \left(\sigma(v^\top x_i) - y_i \right) \gamma^{-1/2} \cdot \gamma^{1/2} \sigma'(w_t^\top x_i)(w_t^\top x_i - v^\top x_i) \\
&\geqslant (\gamma/n) \sum_{i=1}^n \left(w_t^\top x_i - v^\top x_i \right)^2 \sigma'(w_t^\top x_i) \\
&\quad - \frac{\gamma^{-1}}{2n} \sum_{i=1}^n \left(\sigma(v^\top x_i) - y_i \right)^2 \sigma'(w_t^\top x_i) - \frac{\gamma}{2n} \sum_{i=1}^n \left(w_t^\top x_i - v^\top x_i \right)^2 \sigma'(w_t^\top x_i) \\
&\geqslant \frac{\gamma}{2} \sum_{i=1}^n (w_t^\top x_i - v^\top x_i)^2 \sigma'(w_t^\top x_i) - L\gamma^{-1} \widehat{F}(v) \\
&\geqslant \gamma L^{-2} \widehat{H}(w_t) - L\gamma^{-1} \widehat{F}(v). \qquad (2.3.8)
\end{aligned}$$

In the first inequality we used (2.3.7) for the first term and Young's inequality for the second (and that $\sigma' \geqslant 0$). For the final two inequalities, we use that σ is L-Lipschitz.

For the gradient upper bound,

$$\begin{aligned}
\left\| \nabla \widehat{F}(w) \right\|^2 &\leqslant 2 \left\| \frac{1}{n} \sum_{i=1}^n (\sigma(w^\top x_i) - \sigma(v^\top x_i)) \sigma'(w^\top x_i) x_i \right\|^2 \\
&\quad + 2 \left\| \frac{1}{n} \sum_{i=1}^n (\sigma(v^\top x_i) - y_i) \sigma'(w^\top x_i) x_i \right\|^2 \\
&\leqslant \frac{2}{n} \sum_{i=1}^n (\sigma(w^\top x_i) - \sigma(v^\top x_i))^2 \sigma'(w^\top x_i)^2 \|x_i\|_2^2 \\
&\quad + \frac{2}{n} \sum_{i=1}^n (\sigma(v^\top x_i) - y_i)^2 \sigma'(w^\top x_i)^2 \|x_i\|_2^2 \\
&\leqslant \frac{2LB_X^2}{n} \sum_{i=1}^n (\sigma(w^\top x_i) - \sigma(v^\top x_i))^2 \sigma'(w^\top x_i) + 4L^2 B_X^2 \widehat{F}(v) \\
&= 4LB_X^2 \widehat{H}(w) + 4L^2 B_X^2 \widehat{F}(v). \qquad (2.3.9)
\end{aligned}$$

The first inequality is due to Young's inequality, and the second is due to Jensen's inequality. The last inequality holds because σ is L-Lipschitz and $\|x\|_2 \leqslant B_X$ a.s. Putting (2.3.8) and (2.3.9) together and taking $\eta \leqslant (1/8)L^{-3}B_X^{-2}\gamma$,

$$\begin{aligned}\|w_t - v\|^2 - \|w_{t+1} - v\|^2 &\geqslant 2\eta(\gamma L^{-2}\widehat{H}(w_t) - L\gamma^{-1}\widehat{F}(v)) - 4\eta^2(LB_X^2\widehat{H}(w_t) + L^2B_X^2\widehat{F}(v)) \\ &\geqslant 2\eta\left(\frac{\gamma L^{-2}}{2}\widehat{H}(w_t) - \frac{5}{2}L\gamma^{-1}\widehat{F}(v)\right) \\ &\geqslant \eta\gamma L\widehat{F}(v).\end{aligned}$$

The last inequality uses the induction assumption that $\widehat{H}(w_t) \geqslant 6L^3\gamma^{-2}\widehat{F}(v)$, completing the proof. □

Since the auxiliary error $\widehat{H}(w)$ is controlled by $\widehat{F}(v)$, we need to bound $\widehat{F}(v)$. When the marginals of $\mathcal{D}$ are bounded, Lemma 2.3.8 below shows that $\widehat{F}(v)$ concentrates around $F(v) = \mathsf{OPT}$ at rate $n^{-1/2}$ by Hoeffding's inequality; for completeness, the proof is given in Section 2.10.

Lemma 2.3.8. If $\|x\|_2 \leqslant B_X$ and $|y| \leqslant B_Y$ a.s. under $\mathcal{D}_x$ and $\mathcal{D}_y$ respectively, and if σ is non-decreasing, then for $a := (|\sigma(B_X)| + B_Y)^2$ and $\|v\|_2 \leqslant 1$, we have with probability at least $1 - \delta$,

$$|\widehat{F}(v) - \mathsf{OPT}| \leqslant 3a\sqrt{n^{-1}\log(2/\delta)}.$$

The final ingredient to the proof is translating the bounds for the empirical risk to one for the population risk. Since $\mathcal{D}_x$ is bounded and we showed in Lemma 2.3.6 that $\|w_t - v\|_2 \leqslant 1$ throughout the gradient descent trajectory, we can use standard properties of Rademacher complexity to get it done. The proof for Lemma 2.3.9 can be found in Section 2.10.

Lemma 2.3.9. Suppose σ is L-Lipschitz and $\|x\|_2 \leqslant B_X$ a.s. Denote $\ell(w; x)$ by the loss $(1/2)\left(\sigma(w^\top x) - \sigma(v^\top x)\right)^2$. For a training set $S \sim \mathcal{D}^n$, let $\mathfrak{R}_S(\mathcal{G})$ denote the empirical Rademacher complexity of the following function class

$$\mathcal{G} := \{x \mapsto w^\top x : \|w - v\|_2 \leqslant 1,\ \|v\|_2 = 1\}.$$

Then we have

$$\mathfrak{R}(\ell \circ \sigma \circ \mathcal{G}) = \mathbb{E}_{S \sim \mathcal{D}^n} \mathfrak{R}_S(\ell \circ \sigma \circ \mathcal{G}) \leqslant 2L^3 B_X^2/\sqrt{n}.$$

With Lemmas 2.3.6, 2.3.8 and 2.3.9 in hand, the bound for the population risk follows in a straightforward manner.

Proof of Theorem 2.3.3 for strictly increasing activations. By Lemma 2.3.6, there exists w_t with $t < T$ and $\|w_t - v\|_2 \leqslant 1$ such that $\widehat{H}(w_t) \leqslant 6L^3\gamma^{-2}\widehat{F}(v)$. By Lemmas 2.3.5 and Lemma 2.3.8, this implies that with probability at least $1 - \delta/2$,

$$\widehat{G}(w_t) \leqslant 6L^3\gamma^{-3}\left(\mathsf{OPT} + 3an^{-1/2}\log^{1/2}(4/\delta)\right). \tag{2.3.10}$$

Since $\|w - v\|_2 \leqslant 1$ implies $\ell(w; x) = (1/2)(\sigma(w^\top x) - \sigma(v^\top x))^2 \leqslant L^2 B_X^2/2$, standard results from Rademacher complexity (e.g., Theorem 26.5 of [SB14]) imply that with probability at least $1 - \delta/2$,

$$G(w_t) \leqslant \widehat{G}(w_t) + \mathbb{E}_{S\sim\mathcal{D}^n}\mathfrak{R}_S(\ell \circ \sigma \circ \mathcal{G}) + 2L^2 B_X^2\sqrt{\frac{2\log(8/\delta)}{n}},$$

where ℓ is the loss and $\mathcal{G}$ is the function class defined in Lemma 2.3.9. We can combine (2.3.10) with Lemma 2.3.9 and a union bound to get that with probability at least $1 - \delta$,

$$G(w_t) \leqslant 6L^3\gamma^{-3}\left(\mathsf{OPT} + 3a\sqrt{\frac{\log(4/\delta)}{n}}\right) + \frac{2L^3 B_X^2}{\sqrt{n}} + \frac{2L^2 B_X^2\sqrt{2\log(8/\delta)}}{\sqrt{n}}.$$

This shows that $G(w_t) \leqslant O(\mathsf{OPT} + n^{-1/2})$. By Claim 2.3.4, we have

$$F(w_t) \leqslant 2G(w_t) + 2\mathsf{OPT} \leqslant O(\mathsf{OPT} + n^{-1/2}),$$

completing the proof for those σ satisfying Assumption 2.3.1. □

2.3.2 ReLU activation

The proof above crucially relies upon the fact that σ is strictly increasing so that we may apply Fact 2.3.7 in the proof of Lemma 2.3.6. In particular, it is difficult to show a strong lower bound for the gradient direction term in (2.3.8) if it is possible for $(z_1 - z_2)^2$ to be arbitrarily large when $(\sigma(z_1) - \sigma(z_2))^2$ is small. To get around this, we will use the same proof technique wherein we show that the gradient lower bound involves a term that relates the auxiliary error $\widehat{H}(w_t)$ to $\widehat{F}(v)$, but our bound will involve a term of the form $O(\widehat{F}(v)^{1/2})$ rather than $O(\widehat{F}(v))$. To do so, we will use the following property of non-decreasing Lipschitz functions.

Fact 2.3.10. If σ is non-decreasing and L-Lipschitz, then for any z_1, z_2 in the domain of σ, it holds that $(\sigma(z_1) - \sigma(z_2))(z_1 - z_2) \geqslant L^{-1}(\sigma(z_1) - \sigma(z_2))^2$.

With this fact we can present the analogue to Lemma 2.3.6 that holds for a general non-decreasing and Lipschitz activation and hence includes the ReLU.

Lemma 2.3.11. Suppose that $\|x\|_2 \leqslant B_X$ a.s. under $\mathcal{D}_x$. Suppose σ is non-decreasing and L-Lipschitz. Assume $\widehat{F}(v) \in (0,1)$. Gradient descent with fixed step size $\eta \leqslant (1/4)L^{-2}B_X^{-2}$ initialized at $w_0 = 0$ finds weights w_t satisfying $\widehat{H}(w_t) \leqslant 2L^2 B_X \widehat{F}(v)^{1/2}$ within $T = \lceil \eta^{-1} L^{-1} B_X^{-1} \widehat{F}(v)^{-1/2} \rceil$ iterations, with $\|w_t - v\|_2 \leqslant 1$ for each $t = 0, \ldots, T-1$.

Proof. Just as in the proof of Lemma 2.3.6, the lemma is proved if we can show that for every $t \in \mathbb{N}$, either (a) $\widehat{H}(w_\tau) \leqslant 2L^2 B_X \widehat{F}(v)^{1/2}$ for some $\tau < t$, or (b) $\|w_t - v\|_2^2 \leqslant \|w_{t-1} - v\|_2^2 - \eta L B_X \widehat{F}(v)^{1/2}$ holds. To this end we assume the induction hypothesis holds for some $t \in \mathbb{N}$, and since we are done if (a) holds, we assume (a) does not hold and thus for every $\tau \leqslant t$, we have $\widehat{H}(w_\tau) > 2L^2 B_X \widehat{F}(v)^{1/2}$. Since (a) does not hold, $\|w_\tau - v\|_2^2 \leqslant \|w_{\tau-1} - v\|_2^2 - \eta L B_X \widehat{F}(v)^{1/2}$ holds for each $\tau = 1, \ldots, t$ and hence the identity

$$\|w_\tau - v\|_2 \leqslant 1 \quad \forall \tau \leqslant t, \tag{2.3.11}$$

holds. We now proceed with showing the analogues of (2.3.8) and (2.3.9). We begin with the lower bound,

$$\begin{aligned}\left\langle \nabla \widehat{F}(w_t), w_t - v\right\rangle &= (1/n)\sum_{i=1}^n \left(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\right)\sigma'(w_t^\top x_i)(w_t^\top x_i - v^\top x_i)\\ &\quad + \left\langle (1/n)\sum_{i=1}^n \left(\sigma(v^\top x_i) - y_i\right)\sigma'(w_t^\top x_i)x_i, w_t - v\right\rangle \qquad (2.3.12)\\ &\geqslant (1/Ln)\sum_{i=1}^n \left(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\right)^2 \sigma'(w_t^\top x_i)\\ &\quad - \|w_t - v\|_2 \left\| (1/n)\sum_{i=1}^n \left(\sigma(v^\top x_i) - y_i\right)\sigma'(w_t^\top x_i)x_i\right\|_2\\ &\geqslant 2L^{-1}\widehat{H}(w_t) - LB_X\widehat{F}(v)^{1/2}. \qquad (2.3.13)\end{aligned}$$

In the first inequality, we have used Fact 2.3.10 and that $\sigma'(z) \geqslant 0$ for the first term. For the second term, we use Cauchy–Schwarz. The last inequality is a consequence of (2.3.11), Cauchy–Schwarz, and that $\sigma'(z) \leqslant L$ and $\|x\|_2 \leqslant B_X$. As for the gradient upper bound at w_t, the bound (2.3.9) still holds since it only uses that σ is L-Lipschitz. The choice of $\eta \leqslant (1/4)L^{-2}B_X^{-2}$ then ensures

$$\begin{aligned}\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 &\geqslant 2\eta\left(2L^{-1}\widehat{H}(w_t) - LB_X\widehat{F}(v)^{1/2}\right)\\ &\quad - \eta^2\left(4B_X^2 L\widehat{H}(w_t) + 4L^2B_X^2\widehat{F}(v)\right)\\ &\geqslant \eta\left(3L^{-1}\widehat{H}(w_t) - 3LB_X\left(\widehat{F}(v) \vee \widehat{F}(v)^{1/2}\right)\right)\\ &\geqslant \eta LB_X\widehat{F}(v)^{1/2}, \qquad (2.3.14)\end{aligned}$$

where the last line comes from the induction hypothesis that $\widehat{H}(w_t) \geqslant 2L^2B_X\widehat{F}(v)^{1/2}$ and since $\widehat{F}(v) \in (0,1)$. This completes the proof. □

With this lemma in hand, the proof of Theorem 2.3.3 follows just as in the strictly increasing case.

Proof of Theorem 2.3.3 for ReLU. We highlight here the main technical differences with the proof for the strictly increasing case. Although Lemma 2.3.9 applies to the loss function

$\ell(w;x) = (1/2)\left(\sigma(w^\top x) - \sigma(v^\top x)\right)^2$, the same results hold for the loss function $\tilde{\ell}(w;x) = \ell(w;x)\sigma'(w^\top x)$ for ReLU, since $\nabla\sigma'(w^\top x) \equiv 0$ a.e. Thus $\tilde{\ell}$ is still B_X-Lipschitz, and we have

$$\mathbb{E}_{S\sim\mathcal{D}^n}\mathfrak{R}_S\left(\tilde{\ell}\circ\sigma\circ\mathcal{G}\right) \leqslant \frac{2B_X^2}{\sqrt{n}}. \tag{2.3.15}$$

With this in hand, the proof is essentially identical: By Lemmas 2.3.11 and 2.3.8, with probability at least $1-\delta/2$ gradient descent finds a point with

$$\widehat{H}(w_t) \leqslant 2B_X\widehat{F}(v)^{1/2} \leqslant 2B_X\left(\mathsf{OPT}^{1/2} + \frac{\sqrt{3a}\log^{1/4}(4/\delta)}{n^{1/4}}\right). \tag{2.3.16}$$

We can then use (2.3.15) to get that with probability at least $1-\delta$,

$$H(w_t) \leqslant 2B_X\left(\mathsf{OPT}^{1/2} + \frac{\sqrt{3a}\log^{1/4}(4/\delta)}{n^{1/4}}\right) + \frac{2B_X^2}{\sqrt{n}} + 2B_X^2\sqrt{\frac{2\log(8/\delta)}{n}}. \tag{2.3.17}$$

Since $\mathcal{D}_x$ satisfies Assumption 2.3.2 and $\|w_t - v\|_2 \leqslant 1$, Lemma 2.3.5 yields $G(w_t) \leqslant 8\sqrt{2}\alpha^{-4}\beta^{-1}H(w_t)$. Then applying Claim 2.3.4 completes the proof. □

Remark 2.3.12. An examination of the proof of Theorem 2.3.3 shows that when σ satisfies Assumption 2.3.1, any initialization with $\|w_0 - v\|_2$ bounded by a universal constant will suffice. In particular, if we use Gaussian initialization $w_0 \sim N(0, \tau^2 I_d)$ for $\tau^2 = O(1/d)$, then by concentration of the chi-square distribution the theorem holds with (exponentially) high probability over the random initialization. For ReLU, initialization at the origin greatly simplifies the proof since Lemma 2.3.11 shows that $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$ for all t. When $w_0 = 0$, this implies $\|w_t - v\|_2 \leqslant 1$ and allows for an easy application of Lemma 2.3.5. For isotropic Gaussian initialization, one can show that with probability approaching 1/2 that $\|w_0 - v\|_2 < 1$ provided its variance satisfies $\tau^2 = O(1/d)$ (see e.g. Lemma 5.1 of [YS20]). In this case, the theorem will hold with constant probability over the random initialization.

2.4 Noisy teacher network setting

In this section, we consider the teacher network setting, where the joint distribution of $(x,y) \sim \mathcal{D}$ is given by a target neuron v (with $\|v\|_2 \leqslant 1$) plus zero-mean s-sub-Gaussian noise,

$$y|x \sim \sigma(v^\top x) + \xi, \quad \mathbb{E}\xi|x = 0.$$

We assume throughout this section that $\xi \not\equiv 0$; we deal with the realizable setting separately (and achieve improved sample complexity) in Section 2.9. We note that this is precisely the setup of the generalized linear model with (inverse) link function σ. We further note that we only assume that $\mathbb{E}[y|x] = \sigma(v^\top x)$, i.e., the noise is *not* assumed to be independent of the input x, and thus falls into the probabilistic concept learning model of [KS94].

With the additional structural assumption of a noisy teacher, we can improve the agnostic result from $O(\mathsf{OPT}) + \varepsilon$ (for strictly increasing activations) and $O(\mathsf{OPT}^{1/2}) + \varepsilon$ (for ReLU) to $\mathsf{OPT} + \varepsilon$. The key difference from the proof in the agnostic setting is that when trying to show the gradient points in a good direction as in (2.3.8) and (2.3.12), since we know $\mathbb{E}[y|x] = \sigma(v^\top x)$, the average of terms of the form $a_i(\sigma(v^\top x_i) - y_i)$ with fixed and bounded coefficients a_i will concentrate around zero. This allows us to improve the lower bound from $\langle \nabla \widehat{F}(w_t), w_t - v\rangle \geqslant C(\widehat{H}(w) - \widehat{F}(v)^\alpha)$ to one of the form $\geqslant C(\widehat{H}(w) - \varepsilon)$, where C is an absolute constant. The full proof of Theorem 2.4.1 is given in Section 2.8.

Theorem 2.4.1. Suppose $\mathcal{D}_x$ satisfies $\|x\|_2 \leqslant B_X$ a.s. and $\mathbb{E}[y|x] = \sigma(v^\top x)$ for some $\|v\|_2 \leqslant 1$. Assume that $\sigma(v^\top x) - y$ is s-sub-Gaussian. Assume gradient descent is initialized at $w_0 = 0$ and fix a step size $\eta \leqslant (1/4)L^{-2}B_X^{-2}$. If σ satisfies Assumption 2.3.1, let γ be the constant corresponding to $\rho = 2B_X$. There exists an absolute constant $c_0 > 0$ such that for any $\delta > 0$, with probability at least $1 - \delta$, gradient descent for $T = \eta^{-1}\sqrt{n}/(c_0 L B_x s\sqrt{\log(4d/\delta)})$ iterations finds weights w_t, $t < T$, satisfying

$$F(w_t) \leqslant \mathsf{OPT} + C_1 n^{-1/2} + C_2 n^{-1/2}\sqrt{\log(8/\delta)} + C_3 n^{-1/2}\sqrt{\log(4d/\delta)}, \tag{2.4.1}$$

where $C_1 = 4L^3B_X^2$, $C_2 = 2\sqrt{2}L^2B_X^2\sqrt{2}$, and $C_3 = 4c_0\gamma^{-1}L^2 s B_X$. When σ is ReLU, further assume that $\mathcal{D}_x$ satisfies Assumption 2.3.2 for constants $\alpha, \beta > 0$, and let $\nu = \alpha^4\beta/8\sqrt{2}$. Then (2.4.1) holds for $C_1 = B_X^2\nu^{-1}$, $C_2 = 2C_1$, and $C_3 = 4c_0 s\nu^{-1}B_X$.

We first note that although (2.4.1) contains a $\log(d)$ term, the dependence on the dimension can be removed if we assume that the noise is bounded rather than sub-Gaussian; details for this are given in Section 2.8. As mentioned previously, if we are in the realizable setting, i.e. $\xi \equiv 0$, we can improve the sample and runtime complexities to $O(\varepsilon^{-1})$ by using online SGD and a martingale Bernstein bound. For details on the realizable case, see Section 2.9.

In comparison with existing literature, [KKK11] proposed GLMTron to show the learnability of the noisy teacher network for a non-decreasing and Lipschitz activation σ when the noise is bounded.[2] In GLMTron, updates take the form $w_{t+1} = w_t - \eta \tilde{g}_t$ where $\tilde{g}_t = (\sigma(w_t^\top x) - y)x$, while in gradient descent, the updates take the form $w_{t+1} = w_t - \eta g_t$ where $g_t = \tilde{g}_t \sigma'(w_t^\top x)$. Intuitively, when the weights are in a bounded region and σ is strictly increasing and Lipschitz, the derivative satisfies $\sigma'(w_t^\top x) \in [\gamma, L]$ and so the additional σ' factor will not significantly affect the algorithm. For ReLU this is more complicated as the gradient could in the worst case be zero in a large region of the input space, preventing effective learnability using gradient-based optimization, as was demonstrated in the negative result of [YS20]. For this reason, a type of nondegeneracy condition like Assumption 2.3.2 is essential for gradient descent on ReLUs.

In terms of other results for ReLU, recent work by [MM20] introduced another modified version of SGD, where updates now take the form $w_{t+1} = w_t - \eta \widehat{g}_t$, with $\widehat{g}_t = \tilde{g}_t \sigma'(y > \theta)$, and θ is an upper bound for an adversarial noise term. They showed that this modified SGD recovers the parameter v of the teacher network under the nondegeneracy condition that the matrix $\mathbb{E}_x[xx^\top \mathbb{1}(v^\top x \geqslant 0)]$ is positive definite. A similar assumption was used by [DLT18] in the realizable setting.

Our GLM result is also comparable to recent work by [FSS18], where the authors provide a meta-algorithm for translating guarantees for ε-stationary points of the empirical risk to

[2] A close inspection of the proof shows that sub-Gaussian noise can be handled with the same concentration of norm sub-Gaussian random vectors that we use for our results.

guarantees for the population risk provided that the population risk satisfies the so-called "gradient domination" condition and the algorithm can guarantee that the weights remain bounded (see their Proposition 3). By considering GLMs with bounded, strictly increasing, Lipschitz activations, they show the gradient domination condition holds, and any algorithm that can find a stationary point of an ℓ^2-regularized empirical risk objective is guaranteed to have a population risk bound. In contrast, our result concretely shows that vanilla gradient descent learns the GLM, even in the ReLU setting.

2.5 Conclusion and remaining open problems

In this work, we considered the problem of learning a single neuron with the squared loss by using gradient descent on the empirical risk. We first analyzed this in the agnostic PAC learning framework and showed that if the activation function is strictly increasing and Lipschitz, then gradient descent finds weights with population risk $O(\mathsf{OPT}) + \varepsilon$, where OPT is the smallest possible population risk achieved by a single neuron. When the activation function is ReLU, we showed that gradient descent finds a point with population risk at most $O(\mathsf{OPT}^{1/2}) + \varepsilon$. Under the more restricted noisy teacher network setting, we showed the population risk guarantees improve to $\mathsf{OPT} + \varepsilon$ for both strictly increasing activations and ReLU.

Our work points towards a number of open problems. Does gradient descent on the empirical risk provably achieve population risk with a better dependence on OPT than we have shown in this work, or are there distributions for which this is impossible? Recent work by [GGK20] provides a statistical query lower bound for learning a sigmoid with respect to the correlation loss $\mathbb{E}[\ell(y\sigma(w^\top x))]$, but we are not aware of lower bounds for learning non-ReLU single neurons under the squared loss. It thus remains a possibility that gradient descent (or another algorithm) can achieve $\mathsf{OPT} + \varepsilon$ risk for such activation functions. For ReLU, [DGK20a] showed that gradient descent on a convex surrogate for the empirical risk can achieve $O(\mathsf{OPT}) + \varepsilon$ population risk for log concave distributions; it would be interesting if such bounds could be shown for gradient descent on the empirical risk itself.

2.6 Detailed comparisons with related work

Here, we describe comparisons of our results to those in the literature and give detailed comments on the specific rates we achieve. In Table 2.1, we compare our agnostic learning results. We note the guarantees for the population risk in the fourth column, the marginal distributions over x for which the bounds hold in the fifth column, and the sample complexity required to reach the specified level of risk plus some $\varepsilon > 0$ in the final column. Our results in this setting come from Theorem 2.3.3. The Big-O notation hides constants that may depend on the parameters of the distribution or activation function, but does not hide explicit dependence on the dimension d. However, the parameters of the distribution itself may have *implicit* dependence on the dimension. In particular, for bounded distributions that satisfy $\|x\|_2 \leqslant B_X$, the $O()$ hides multiplicative factors that depend on B_X. This means that if B_X depends on d, so will our bounds. For ReLU, the $O()$ hides polynomial factors in B_X. For non-ReLU, the worst-case activation functions under consideration in Assumption 2.3.1 (e.g. the sigmoid) can have $\gamma \sim \exp(-B_X)$, making the runtime and sample complexity depend on $\gamma^{-1} \sim \exp(B_X)$, in which case it is preferable for B_X to be a constant independent of the dimension. We note that the sample complexity for [DGK20a] for the $(1+\delta)\mathsf{OPT}$ guarantee is $O(\varepsilon^{-2}[d\delta^{-3}\nu^{-2}]^{\delta^{-3}})$ when $\mathcal{D}_x$ is ν sub-Gaussian for some $\nu = O(1)$, and thus the exact dependence on the dimension depends on the sub-Gaussian norm and error threshold desired.

In Table 2.2, we provide comparisons of our noisy teacher network setting, where $y = \sigma(v^\top x) + \xi$ for some zero mean noise ξ. Our results in this setting come from Theorem 2.4.1. The complexity column here denotes the sample complexity required to reach population risk $\mathsf{OPT} + \varepsilon$. The subspace eigenvalue assumption given by [MM20] is that $\mathbb{E}[xx^\top \mathbb{1}(v^\top x \geqslant$

$0)] \succ 0$. We note that the result of Mukherjee and Muthukumar holds for any bounded noise distribution and thus is in the more general adversarial noise (but not agnostic[3]) setting.

Finally, in Table 2.3, we provide comparisons with results in the realizable setting ($\xi \equiv 0$). (Our results in this setting are given in Theorem 2.9.1 in Section 2.9.) For G.D. and S.G.D., the complexity column denotes the sample complexity required to reach population risk ε. For G.D. or gradient flow on the population risk, it refers to the runtime complexity only as there are no samples in this setting. For [DLT18], the subspace eigenvalue assumption is that for any w and for the target neuron v, it holds that $\mathbb{E}[xx^\top \mathbb{1}(w^\top x \geqslant 0, v^\top x \geqslant)] \succ 0$. This is a nondegeneracy assumption that is related to the marginal spread condition given in Assumption 2.3.2, in the sense that it allows for one to show that H is an upper bound for G. Finally, we note that any result in the agnostic or noisy teacher network settings applies in the realizable setting as well.

2.7 Proof of Lemma 2.3.5

To prove Lemma 2.3.5, we use the following result of [YS20].

Lemma 2.7.1 (Lemma B.1, [YS20])**.** Under Assumption 2.3.2, for any two vectors $a, b \in \mathbb{R}^2$

[3] Agnostic learning results typically require i.i.d. samples, and adversarial noise may depend on other samples in malicious ways. Even in the i.i.d. case, trouble arises if one wishes to use parameter recovery to show that a given algorithm competes with the population risk minimizer. Consider the ReLU with labels given by $y = \sigma(v^\top x) + \xi$ where $\xi = -\sigma(v^\top x)$. The zero vector minimizes the population risk, and so any algorithm that returns the target neuron $\sigma(v^\top x)$ has large population risk. A similar phenomenon occurs for $\xi = \sigma(v^\top x)$.

[4] Although their result is stated for the ReLU and isotropic log-concave distributions, their results also apply for L-Lipschitz activations satisfying $\inf_z \sigma'(z) \geqslant \gamma > 0$ for isotropic distributions that satisfy our Assumption 2.3.2. In this setting, one can show that the Chow parameters satisfy $\|\chi(\sigma_u) - \chi(\sigma_w)\|^2 \geqslant \gamma L^{-1}\mathbb{E}[(\sigma(u^\top x) - \sigma(v^\top x))^2]$, from which the result follows easily.

Table 2.1: Comparison of single neuron results in the agnostic setting

Algorithm	Activations	Pop. risk	$\mathcal{D}_x$	Sample Complexity
Halfspace reduction [GKK19]	ReLU	$O(\mathsf{OPT}^{2/3})$	standard Gaussian	$O(\mathrm{poly}(d, \varepsilon^{-1}))$
Convex surrogate G.D. [DGK20a][4]	ReLU	$O(\mathsf{OPT})$	isotropic +log-concave	$O(d\varepsilon^{-2})$
Convex surrogate G.D. + Domain Partition [DGK20a]	ReLU	$(1+\delta)\mathsf{OPT}$	sub-Gaussian	$O(d^c\varepsilon^{-2})$
Gradient Descent (This chapter)	strictly increasing + Lipschitz	$O(\mathsf{OPT})$	bounded	$O(\varepsilon^{-2})$
Gradient Descent (This chapter)	ReLU	$O(\mathsf{OPT}^{1/2})$	bounded + marginal spread	$O(\varepsilon^{-4})$

satisfying $\theta(a,b) \leqslant \pi - \delta$ for $\delta \in (0,\pi]$, it holds that

$$\inf_{u\in\mathbb{R}^2:\ \|u\|=1} \int (u^\top y)^2 \mathbb{1}(a^\top y \geqslant 0,\ b^\top y \geqslant 0,\ \|y\| \leqslant \alpha)dy \geqslant \frac{\alpha^4}{8\sqrt{2}} \sin^3(\delta/4).$$

Proof of Lemma 2.3.5. We first consider the case when σ satisfies Assumption 2.3.1. By assumption,

$$\overline{F}(w) = (1/2)\mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w_t^\top x)\right] \leqslant \varepsilon.$$

Table 2.2: Comparison of single neuron results in the noisy teacher network setting

Algorithm	Activations	$\mathcal{D}_x$	Sample Complexity
GLMTron [KKK11]	increasing + Lipschitz	bounded	$O(\varepsilon^{-2})$
Modified SGD [MM20]	ReLU	bounded + subspace eigenvalue	$O(\log(1/\varepsilon))$
Meta-algorithm [FSS18]	strictly increasing + Lipschitz + σ' Lipschitz	bounded	$O(\varepsilon^{-2} \wedge d\varepsilon^{-1})$
Gradient Descent [MBM18]	strictly increasing + diff'ble + Lipschitz + σ' Lipschitz + σ'' Lipschitz	centered + sub-Gaussian + $\mathbb{E}[xx^\top] \succ 0$	$O(d\varepsilon^{-1})$
Gradient Descent (This chapter)	strictly increasing + Lipschitz	bounded	$O(\varepsilon^{-2})$
Gradient Descent (This chapter)	ReLU	bounded + marginal spread	$O(\varepsilon^{-2})$

Table 2.3: Comparison of single neuron results in the realizable setting

Algorithm	Activations	$\mathcal{D}_x$	Sample Complexity
SGD [DLT18]	ReLU	bounded + subspace eigenvalue	$O(\log(1/\varepsilon))$
Projected Regularized GD [Sol17]	ReLU	standard Gaussian	$O(\log(1/\varepsilon))$
Population GD[YS20]	$\inf_{z\in\mathbb{R}} \sigma'(z) > 0$	bounded + $\mathbb{E}[xx^\top] \succ 0$	$O(\log(1/\varepsilon))$
Population GD [YS20]	$\inf_{0<z<\alpha} \sigma'(z) > 0$ + Lipschitz	bounded + marginal spread	$O(\log(1/\varepsilon))$
Population Gradient Flow [YS20]	ReLU	marginal spread + spherical symmetry	$O(\log(1/\varepsilon))$
SGD [YS20]	$\inf_{0<z<\alpha} \sigma'(z) > 0$ + Lipschitz	bounded + marginal spread	$\tilde{O}(\varepsilon^{-2})$
Population GD + SGD (This work)	strictly increasing + Lipschitz	bounded	$O(\varepsilon^{-1})$
Population GD + SGD (This work)	ReLU	bounded + marginal spread	$O(\varepsilon^{-1})$

Since the term in the expectation is nonnegative, restricting the integral to a smaller set only decreases its value, so that

$$(1/2)\mathbb{E}\left[\left(\sigma(w_t^\top x)-\sigma(v^\top x)\right)^2\sigma'(w_t^\top x)\mathbb{1}(|w_t^\top x|\leqslant\rho)\right]\leqslant\varepsilon. \tag{2.7.1}$$

For $\rho = BW$, since $\|w\|_2 \leqslant W$, the inclusion $\{\|x\|_2 \leqslant \rho/W\} = \{\|x\|_2 \leqslant B\} \subset \{|w_t^\top x| \leqslant \rho\}$ holds. This means we can lower bound (2.7.1) by substituting the indicator $\mathbb{1}(|w_t^\top x| \leqslant \rho)$ with $\mathbb{1}(\|x\|_2 \leqslant B)$, which is identically one by assumption. Since $H(w) \leqslant \varepsilon$, this implies

$$\frac{\gamma}{2}\mathbb{E}\left[\left(\sigma(w_t^\top x)-\sigma(v^\top x)\right)^2\right]\leqslant(1/2)\mathbb{E}\left[\left(\sigma(w_t^\top x)-\sigma(v^\top x)\right)^2\sigma'(w_t^\top x)\mathbb{1}(\|x\|_2\leqslant B)\right]\leqslant\varepsilon.$$

Dividing both sides by γ completes this part of the proof.

For ReLU, let us assume that $\overline{F}(w) \leqslant \varepsilon$, and denote the event

$$K_{w,v} := \{w^\top x \geqslant 0, v^\top x \geqslant 0\},$$

and define $\zeta := \beta\alpha^4/8\sqrt{2}$. Since $\overline{F}(w) = \mathbb{E}[(\sigma(w^\top x) - \sigma(v^\top x))^2\mathbb{1}(w^\top x \geqslant 0)] \leqslant \zeta\varepsilon$, it holds that

$$\mathbb{E}\left[\left(\sigma(w^\top x)-\sigma(v^\top x)\right)^2\mathbb{1}(K_{w,v})\right]\leqslant\zeta\varepsilon. \tag{2.7.2}$$

Denote $\widehat{w}$ and $\widehat{v}$ as the projections of w and v respectively onto the two dimensional subspace $\operatorname{span}(w, v)$. Using a proof similar to that of [YS20], we have

$$\begin{aligned}
\mathbb{E}_{x\sim\mathcal{D}}\left[\left(w^\top x-v^\top x\right)^2\mathbb{1}(K_{w,v})\right] &= \|w-v\|_2^2\,\mathbb{E}_{x\sim\mathcal{D}}\left[\left(\left(\frac{w-v}{\|w-v\|_2}\right)^\top x\right)^2\mathbb{1}(K_{w,v})\right]\\
&\geqslant \|w-v\|_2^2\inf_{u\in\operatorname{span}(w,v),\ \|u\|=1}\mathbb{E}_x\left[\mathbb{1}(u^\top x)^2\mathbb{1}(K_{w,v})\right]\\
&= \|w-v\|_2^2\inf_{u\in\mathbb{R}^2,\ \|u\|=1}\mathbb{E}_{y\sim\mathcal{D}_{w,v}}\left[(u^\top y)^2\mathbb{1}(\widehat{w}^\top y\geqslant 0,\ \widehat{v}^\top y\geqslant 0)\right]\\
&\geqslant \|w-v\|_2^2\inf_{u\in\mathbb{R}^2,\ \|u\|=1}\int(u^\top y)^2\mathbb{1}(\widehat{w}^\top y\geqslant 0,\ \widehat{v}^\top y\geqslant 0,\ \|y\|_2\leqslant\alpha)p_{w,v}(y)dy\\
&\geqslant \beta\|w-v\|_2^2\inf_{u\in\mathbb{R}^2,\ \|u\|=1}\int(u^\top y)^2\mathbb{1}(\widehat{w}^\top y\geqslant 0,\ \widehat{v}^\top y\geqslant 0,\ \|y\|_2\leqslant\alpha)dy.
\end{aligned} \tag{2.7.3}$$

By assumption, $\|w - v\|_2 \leqslant 1$. Since

$$1 \geqslant \|w-v\|_2^2 = \|w\|_2\left(\|w\|_2 - 2\cos\theta(w,v)\right) + 1,$$

we must have either $w = 0$ or $\theta(w, v) \in [0, \pi/2]$. To see that $w = 0$ is impossible, suppose for the contradiction that $w = 0$ and so $\overline{F}(w) = \overline{F}(0) \leqslant \zeta\varepsilon$. Let z be any vector orthogonal to v, so that $\theta(v, z) = \pi/2$. Then,

$$\begin{aligned}
\zeta\varepsilon &\geqslant \overline{F}(0) \\
&= \mathbb{E}_{x\sim\mathcal{D}}\left[(v^\top x)^2 \mathbb{1}(v^\top x \geqslant 0)\right] \\
&= \mathbb{E}_{y\sim\mathcal{D}_{0,v}}\left[(\widehat{v}^\top y)^2 \mathbb{1}(\widehat{v}^\top y \geqslant 0\right] \\
&\geqslant \inf_{u:\ \|u\|=1} \int (u^\top x)^2 \mathbb{1}(v^\top x \geqslant 0, z^\top x \geqslant 0, \|y\|_2 \leqslant \alpha) p_{0,v}(y) dy \\
&\geqslant \beta \inf_{u:\ \|u\|=1} \int (u^\top x)^2 \mathbb{1}(v^\top x \geqslant 0, z^\top x \geqslant 0, \|y\|_2 \leqslant \alpha) dy \\
&\geqslant \frac{\beta\alpha^4}{8\sqrt{2}}. \qquad (2.7.4)
\end{aligned}$$

The last line follows by using Lemma 2.7.1. For $\varepsilon < 1$, this is impossible by the definition of ζ. This shows that $\theta(w, v) \leqslant \pi/2$. We can therefore apply Lemma 2.7.1 to (2.7.3) to get

$$\begin{aligned}
\zeta\varepsilon &\geqslant \beta \|w - v\|_2^2 \inf_{u\in\mathbb{R}^2,\ \|u\|=1} \int (u^\top y)^2 \mathbb{1}(\widehat{w}^\top y \geqslant 0,\ \widehat{v}^\top y \geqslant 0,\ \|y\|_2 \leqslant \alpha) dy \\
&\geqslant \frac{\beta\alpha^4}{8\sqrt{2}} \|w - v\|_2^2 \\
&= \zeta B^2 \|w - v\|_2^2 .
\end{aligned}$$

This shows that $\|w - v\|_2^2 \leqslant B^{-2}\varepsilon$. Since σ is 1-Lipschitz, Hölder's inequality and $\mathbb{E}\|x\|_2^2 \leqslant B^2$ imply that $G(w) \leqslant \varepsilon$. □

2.8 Noisy teacher network proofs

As in the agnostic case, we have a key lemma that shows $\widehat{H}$ is small when we run gradient descent for a sufficiently large time. Note that one difference with the proof in the agnostic case is that we do not need to consider different auxiliary errors for the strictly increasing and ReLU cases; H alone suffices.

Lemma 2.8.1. Suppose that $\|x\|_2 \leqslant B_X$ a.s. under $\mathcal{D}_x$. Let σ be non-decreasing and L-Lipschitz. Suppose that the bound

$$\|(1/n)\sum_{i=1}^n \big(\sigma(v^\top x_i) - y_i\big)\,\alpha_i x_i\|_2 \leqslant K \leqslant 1. \tag{2.8.1}$$

holds for scalars satisfying $\alpha_i \in [0, L]$. Then gradient descent run with fixed step size $\eta \leqslant (1/4)L^{-2}B_X^{-2}$ from initialization $w_0 = 0$ finds weights w_t satisfying $\widehat{H}(w_t) \leqslant 4LK$ within $T = \lceil \eta^{-1}K^{-1} \rceil$ iterations, with $\|w_t - v\|_2 \leqslant 1$ for each $t = 0, \ldots, T-1$.

Proof. Just as in the proof of Lemma 2.3.6, the theorem can be shown by proving the following induction statement. We claim that for every $t \in \mathbb{N}$, either (a) $\widehat{H}(w_\tau) \leqslant 4LK$ for some $\tau < t$, or (b) $\|w_t - v\|_2^2 \leqslant \|w_{t-1} - v\|_2^2 - \eta K$. If the induction hypothesis holds, then until gradient descent finds a point where $\widehat{H}(w_t) \leqslant 4LK$, the squared distance $\|w_t - v\|_2^2$ decreases by ηK at every iteration. Since $\|w_0 - v\|_2^2 = 1$, this means there can be at most $\eta^{-1}K^{-1}$ iterations until we reach $\widehat{H}(w_t) \leqslant 4LK$. This shows the induction statement implies the theorem.

We begin with the proof by supposing the induction hypothesis holds for t, and considering the case $t+1$. If (a) holds, then we are done. So now consider the case that for every $\tau \leqslant t$, we have $\widehat{H}(w_\tau) > 4LK$. Since (a) does not hold, $\|w_\tau - v\|_2^2 \leqslant \|w_{\tau-1} - v\|_2^2 - \eta K$ holds for each $\tau = 1, \ldots, t$. Since $\|w_0 - v\|_2 = 1$, this implies

$$\|w_\tau - v\|_2 \leqslant 1 \;\forall \tau \leqslant t. \tag{2.8.2}$$

We can therefore bound

$$\begin{aligned}\left\langle \nabla \widehat{F}(w_t), w_t - v \right\rangle &= \left\langle \frac{1}{n} \sum_{1=1}^{n} \big(\sigma(w_t^\top x_i) - y_i\big)\, \sigma'(w_t^\top x_i) x_i, w_t - v \right\rangle \\ &= \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\big)\, \sigma'(w_t^\top x_i)(w_t^\top x_i - v^\top x_i) \\ &\quad + \left\langle \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(v^\top x_i) - y_i\big)\, \sigma'(w_t^\top x_i) x_i, w_t - v \right\rangle \\ &\geqslant \frac{L^{-1}}{n} \sum_{i=1}^{n} \big(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\big)^2 \sigma'(w_t^\top x_i) - K \left\| w_t - v \right\|_2 \\ &\geqslant 2L^{-1} \widehat{H}(w_t) - K. \end{aligned} \tag{2.8.3}$$

In the first inequality, we have used Fact 2.3.10 for the first term. For the second term, we use (2.8.1) and that $\alpha_i := \sigma'(w_t^\top x_i) \in [0, L]$. The last inequality uses (2.8.2).

For the gradient upper bound, we have

$$\begin{aligned}\left\| \nabla \widehat{F}(w_t) \right\|_2^2 &= \left\| \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\big)\, \sigma'(w_t^\top x_i) x_i + \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(v^\top x_i) - y_i\big)\, \sigma'(w_t^\top x_i) x_i \right\|_2^2 \\ &\leqslant 2 \left\| \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(w_t^\top x_i) - \sigma(v^\top x_i)\big)\, \sigma'(w_t^\top x_i) x_i \right\|_2^2 \\ &\quad + 2 \left\| \frac{1}{n} \sum_{i=1}^{n} \big(\sigma(v^\top x_i) - y_i\big)\, \sigma'(w_t^\top x_i) x_i \right\|_2^2 \\ &\leqslant \frac{2LB_X^2}{n} \sum_{i=1}^{n} \big(\sigma(w^\top x_i) - \sigma(v^\top x_i)\big)^2 \sigma'(w_t^\top x_i) + 2K^2 \\ &= 4LB_X^2 \widehat{H}(w_t) + 2K^2. \end{aligned} \tag{2.8.4}$$

The first inequality uses Young's inequality. The second uses that $\sigma'(z) \leqslant L$ and that $\|x\|_2 \leqslant B_X$ a.s. and (2.8.1).

Putting (2.8.3) and (2.8.4) together, the choice of $\eta \leqslant (1/4)L^{-2}B_X^{-2}$ gives us

$$\begin{aligned}\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 &= 2\eta \left\langle \nabla \widehat{F}(w_t), w_t - v \right\rangle - \eta^2 \left\| \nabla \widehat{F}(w_t) \right\|_2^2 \\ &\geqslant 2\eta(L^{-1}\widehat{H}(w_t) - K) - \eta^2 \left(4LB_X^2 \widehat{H}(w_t) + 2K^2 \right) \\ &\geqslant \eta L^{-1}\widehat{H}(w_t) - 3\eta K.\end{aligned}$$

In particular, this implies

$$\|w_{t+1} - v\|_2^2 \leqslant \|w_t - v\|_2^2 + 3\eta K - \eta L^{-1}\widehat{H}(w_t) \tag{2.8.5}$$

Since $\widehat{H}(w_t) > 4KL$, this completes the induction. The base case follows easily since $\|w_0 - v\|_2 = 1$ allows for us to deduce the desired bound on $\|w_1 - v\|_2^2$ using (2.8.5). □

To prove a concrete bound on the K term of Lemma 2.8.1, we will need the following definition of norm sub-Gaussian random vectors.

Definition 2.8.2. A random vector $z \in \mathbb{R}^d$ is said to be *norm sub-Gaussian with parameter* $s > 0$ if

$$\mathbb{P}(\|z - \mathbb{E}z\| \geqslant t) \leqslant 2\exp(-t^2/2s^2).$$

A Hoeffding-type inequality for norm sub-Gaussian vectors was recently shown by [JNG19].

Lemma 2.8.3 (Lemma 6, [JNG19])**.** Suppose $z_1, \ldots, z_n \in \mathbb{R}^d$ are random vectors with filtration $\mathcal{F}_t := \sigma(z_1, \ldots, z_t)$ such that $z_i | \mathcal{F}_{i-1}$ is a zero-mean norm sub-Gaussian vector with parameter $s_i \in \mathbb{R}$ for each i. Then, there exists an absolute constant $c > 0$ such that for any $\delta > 0$, with probability at least $1 - \delta$,

$$\left\| \sum_{i=1}^n z_i \right\| \leqslant c\sqrt{\log(2d/\delta) \sum_{i=1}^n s_i^2}.$$

Using this, we can show that if $\xi_i := \sigma(v^\top x_i) - y_i$ is s sub-Gaussian, then we can get a bound on K at rate $n^{-1/2}$. We note that if we make the stronger assumption that ξ_i is bounded a.s., we can get rid of the $\log(d)$ dependence by using concentration of bounded random variables in a Hilbert space (e.g. [PS86], Corollary 2).

Lemma 2.8.4. Suppose that $\|x\|_2 \leqslant B_X$ a.s. under $\mathcal{D}_x$, and let σ be any continuous function. Assume $\xi_i := \sigma(v^\top x_i) - y_i$ is s sub-Gaussian and satisfies $\mathbb{E}[\xi_i|x_i] = 0$. Then there exists an absolute constant $c_0 > 0$ such that for constants $\alpha_i \in [0, L]$, with probability at least $1-\delta$, we have

$$\|(1/n)\textstyle\sum_{i=1}^n \big(\sigma(v^\top x_i) - y_i\big)\,\alpha_i x_i\| \leqslant c_0 L B_X s\sqrt{n^{-1}\log(2d/\delta)}.$$

Proof of Lemma 2.8.4. Define $z_i := \big(\sigma(v^\top x_i) - y_i\big)\,\alpha_i x_i$. Using iterated expectations, we see that $\mathbb{E}[z_i] = 0$. Since $\sigma(v^\top x_i) - y_i$ is s-sub-Gaussian and $\|\alpha_i x_i\|_2 \leqslant L B_X$, it follows from the definition that z_i is norm sub-Gaussian with parameter $L B_X s$ for each i. By Lemma 2.8.3, we have with probability at least $1-\delta$,

$$\left\|\sum_{i=1}^n z_i\right\| \leqslant c\sqrt{\log(2d/\delta) L^2 B_X^2 n s^2}.$$

Dividing each side by n proves the lemma. □

Proof of Theorem 2.4.1. By Lemmas 2.8.1 and 2.8.4, there exists some w_t, $t < T$ and $\|w_t - v\|_2 \leqslant 1$, such that

$$\widehat{H}(w_t) \leqslant 4LK \leqslant 4c_0 L^2 B_X s\sqrt{\frac{\log(2d/\delta)}{n}}.$$

Consider σ satisfying Assumption 2.3.1 first, with γ corresponding to $\rho = 2B_X$. Since $\|w_t\|_2 \leqslant 2$, we can use Lemma 2.3.5 to transform the above bound for $\widehat{H}$ into one for $\widehat{G}$,

$$\widehat{G}(w_t) \leqslant 4c_0\gamma^{-1} L^2 B_X s\sqrt{\frac{\log(2d/\delta)}{n}}.$$

Since $\|w - v\|_2 \leqslant 1$ implies $G(w) \leqslant L^2 B_X^2/2$, standard results from Rademacher complexity imply (e.g. Theorem 26.5 of [SB14]) that with probability at least $1-\delta$,

$$G(w_t) \leqslant \widehat{G}(w_t) + \mathbb{E}_{S\sim\mathcal{D}^n}\mathfrak{R}_S(\ell \circ \sigma \circ \mathcal{G}) + 2L^2 B_X^2\sqrt{\frac{2\log(4/\delta)}{n}},$$

where $\ell(w; x) = (1/2)(\sigma(w^\top x) - \sigma(v^\top x))^2$ and $\mathcal{G}$ are from Lemma 2.3.9. For the second term above, Lemma 2.3.9 and rescaling δ yields that

$$G(w_t) \leqslant \frac{2L^3 B_X^2}{\sqrt{n}} + \frac{2L^2 B_X^2\sqrt{2\log(8/\delta)}}{\sqrt{n}} + \frac{4c_0\gamma^{-1} L^2 B_X s\sqrt{\log(4d/\delta)}}{\sqrt{n}}.$$

Then Claim 2.3.4 completes the proof for strictly increasing σ.

When σ is ReLU, the proof follows the same argument given in the proof of Theorem 2.3.3. Denoting the loss function $\tilde{\ell}(w;x) = (1/2)(\sigma(w^\top x) - \sigma(v^\top x))^2 \sigma'(w^\top x)$, we have

$$\mathbb{E}_{S\sim\mathcal{D}^n} \mathfrak{R}_S \left(\tilde{\ell} \circ \sigma \circ \mathcal{G}\right) \leqslant \frac{2B_X^2}{\sqrt{n}}. \tag{2.8.6}$$

By Lemmas 2.8.1 and 2.8.4, there exists some w_t, $t < T$ and $\|w_t - v\|_2 \leqslant 1$, such that

$$\widehat{H}(w_t) \leqslant 4LK \leqslant 4c_0 L^2 B_X s \sqrt{\frac{\log(2d/\delta)}{n}}. \tag{2.8.7}$$

Using standard results from Rademacher complexity,

$$H(w_t) \leqslant \widehat{H}(w_t) + \mathbb{E}_{S\sim\mathcal{D}^n} \mathfrak{R}_S(\tilde{\ell} \circ \sigma \circ \mathcal{G}) + 2B_X^2 \sqrt{\frac{2\log(4/\delta)}{n}}.$$

By (2.8.6), this means

$$H(w_t) \leqslant \frac{4c_0 B_X s \sqrt{\log(4d/\delta)}}{\sqrt{n}} + \frac{2B_X^2}{\sqrt{n}} + \frac{2B_X^2 \sqrt{2\log(8/\delta)}}{\sqrt{n}}.$$

Since $\mathcal{D}$ satisfies Assumption 2.3.2 and $\|w_t - v\|_2 \leqslant 1$, Lemma 2.3.5 shows that $G(w_t) \leqslant 8\sqrt{2}\alpha^{-4}\beta^{-1} H(w_t)$. Then Claim 2.3.4 translates the bound for $G(w_t)$ into one for $F(w_t)$. □

2.9 Realizable setting

In this section we assume $y = \sigma(v^\top x)$ a.s. for some $\|v\|_2 \leqslant 1$. As in the agnostic and noisy teacher network setting, we use the auxiliary loss

$$H(w) := (1/2)\mathbb{E}_{x\sim\mathcal{D}}[(\sigma(w^\top x) - \sigma(v^\top x))^2 \sigma'(w^\top x)].$$

Note that in the realizable setting, the previous auxiliary loss G defined in (2.3.4) coincides with the true objective F, i.e. we have

$$F(w) := (1/2)\mathbb{E}_{x\sim\mathcal{D}}[(\sigma(w^\top x) - \sigma(v^\top x))^2].$$

For purpose of comparison with [YS20], we provide analyses for two settings in the realizable case: in the first setting, we consider gradient descent on the population loss,

$$w_{t+1} = w_t - \eta \nabla F(w_t), \tag{2.9.1}$$

and return $w_{t*} := \mathrm{argmin}_{0 \leqslant t < T} F(w_t)$. The second setting is online SGD with samples $x_t \sim \mathcal{D}$. Here we compute unbiased estimates (conditional on w_t) of the population risk $F_t(w_t) := (1/2)(\sigma(w_t^\top x_t) - \sigma(v^\top x_t))^2$ and update the weights by

$$w_{t+1} = w_t - \eta \nabla F_t(w_t) \tag{2.9.2}$$

For SGD, we output $w_{t*} = \mathrm{argmin}_{0 \leqslant t < T} F_t(w_t)$.

We summarize our results in the realizable case in Theorem 2.9.1.

Theorem 2.9.1. Suppose $\|x\|_2 \leqslant B$ a.s. and σ is non-decreasing and L-Lipschitz. Let $\eta \leqslant L^{-2}B^{-2}$ be the step size.

(a) Let σ satisfy Assumption 2.3.1, and let γ be the constant corresponding to $\rho = 4B$. For any initialization satisfying $\|w_0\|_2 \leqslant 2$, if we run gradient descent on the population risk $T = \lceil 2\varepsilon^{-1} L \eta^{-1} \gamma^{-1} \|w_0 - v\|_2^2 \rceil$ iterations, then there exists $t < T$ such that $F(w_t) \leqslant \varepsilon$. For stochastic gradient descent, for any $\delta > 0$, running SGD for $\tilde{T} = 6T \log(1/\delta)$ guarantees there exists w_t, $t < T$, such that w.p. at least $1 - \delta$, $F(w_t) \leqslant \varepsilon$.

(b) Let σ be ReLU and further assume that $\mathcal{D}$ satisfies Assumption 2.3.2 for constants $\alpha, \beta > 0$ and that $w_0 = 0$. Let $\nu = \alpha^4 \beta / 8\sqrt{2}$. If we run gradient descent on the population risk $T = \lceil 2\varepsilon^{-1} L \eta^{-1} \nu^{-1} \|w_0 - v\|_2^2 \rceil$ iterations, then there exists $t < T$ such that $F(w_t) \leqslant \varepsilon$. For stochastic gradient descent, for any $\delta > 0$, running SGD for $\tilde{T} = 6T \log(1/\delta)$ guarantees there exists w_t, $t < T$, such that w.p. at least $1 - \delta$, $F(w_t) \leqslant \varepsilon$.

A few remarks on the above theorem: first, in comparison with our noisy neuron result in Theorem 2.4.1, we are able to achieve $\mathsf{OPT} + \varepsilon = \varepsilon$ population risk with sample complexity and runtime of order ε^{-1} rather than ε^{-2} using the same assumptions by invoking a martingale Bernstein inequality rather than Hoeffding. Second, although Theorem 2.9.1 requires the distribution to be bounded almost surely, we show in Section 2.9.1 below that for GD on the population loss, we can accomodate essentially any distribution with finite expected squared norm.

[YS20] used the marginal spread assumption to show that with probability 1/2, a single neuron in the realizable setting can be learned using gradient-based optimization with random initialization for Lipschitz activation functions satisfying $\inf_{0<z<\alpha} \sigma'(z) > 0$, where α is the same constant in Assumption 2.3.2, and thus includes essentially all neural network activation functions like softplus, sigmoid, tanh, and ReLU. Under the additional assumption of spherical symmetry, they showed that this can be improved to a high probability guarantee for the ReLU activation. For gradient descent on the population risk, they proved linear convergence, i.e. a runtime of order $O(\log(1/\varepsilon))$, while for SGD their runtime and sample complexity is of order $O(\varepsilon^{-2}\log(1/\varepsilon))$. In comparison, our result for the non-ReLU activations requires only boundedness of the distributions and holds with high probability over random initializations, with runtime and sample complexity of order $O(\varepsilon^{-1})$ for both gradient descent on the population risk and SGD. Our results for ReLU use the same marginal spread assumption as Yehudai and Shamir, but our proof technique differs in that we do not require the angle $\theta(w_t, v)$ between the weights in the GD trajectory and the target neuron be decreasing. As they pointed out, angle monotonicity fails to hold for the trajectory of gradient descent even when the distribution is a non-centered Gaussian, so that proofs based on angle monotonicity will not translate to more general distributions. Indeed, our proofs in the agnostic and noisy teacher network setting use essentially the same proof technique as the realizable case without relying on angle monotonicity. Instead, we show a type of inductive bias of gradient descent in the sense that if initialized at the origin, the angle between the target vector and the population risk minimizer cannot become larger than $\pi/2$, even in the agnostic setting.

2.9.1 Gradient descent on population loss

The key lemma for the proof is as follows.

Lemma 2.9.2. Consider gradient descent on the population risk given in (2.9.1). Let w_0 be the initial point of gradient descent and assume $\|w_0\|_2 \leqslant 2$. Suppose that $\mathcal{D}$ satisfies $\mathbb{E}_x[\|x\|_2^2] \leqslant B^2$. Let σ be non-decreasing and L-Lipschitz. Assume the step size satisfies $\eta \leqslant L^{-2}B^{-2}$. Then for any $T \in \mathbb{N}$, we have for all $t = 0, \ldots, T-1$, $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$, and

$$\|w_0 - v\|_2^2 - \|w_T - v\|_2^2 \geqslant \eta L^{-1} \sum_{t=0}^{T-1} \overline{F}(w_t).$$

Proof. We begin with the identity, for $t < T$,

$$\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 = 2\eta \langle \nabla F(w_t), w_t - v\rangle - \eta^2 \|\nabla F(w_t)\|_2^2. \tag{2.9.3}$$

First, we have

$$\begin{aligned}
\|\nabla F(w_t)\|_2 &\leqslant \mathbb{E}_x \left\|(\sigma(w_t^\top x) - \sigma(v^\top x))\sigma'(w_t^\top x)x\right\|_2 \\
&\leqslant \sqrt{\mathbb{E}_x \left[\sigma'(w_t^\top x)(\sigma(w_t^\top x) - \sigma(v^\top x))^2\right]}\sqrt{\mathbb{E}_x \sigma'(w_t^\top x)\|x\|_2^2} \\
&\leqslant B\sqrt{L}\sqrt{\mathbb{E}_x \left[\sigma'(w_t^\top x)(\sigma(w_t^\top x) - \sigma(v^\top x))^2\right]}.
\end{aligned}$$

The first inequality is by Jensen. The second inequality uses that $\sigma'(z) \geqslant 0$ and Hölder, and the third inequality uses that σ is L-Lipschitz and that $\mathbb{E}[\|x\|_2^2] \leqslant B^2$. We therefore have the gradient upper bound

$$\|\nabla F(w_t)\|_2^2 \leqslant 2B^2 L \overline{F}(w_t). \tag{2.9.4}$$

For the inner product term of (2.9.3), since $\sigma'(z) \geqslant 0$, we can use Fact 2.3.10 to get

$$\langle \nabla F(w_t), w_t - v\rangle \geqslant L^{-1}\mathbb{E}_x\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w_t^\top x)\right] = 2L^{-1}\overline{F}(w_t). \tag{2.9.5}$$

Putting (2.9.5) and (2.9.4) into (2.9.3), we get

$$\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 \geqslant 4\eta L^{-1}\overline{F}(w_t) - 2\eta^2 B^2 L \overline{F}(w_t) \geqslant 2\eta L^{-1}\overline{F}(w_t),$$

where we have used $\eta \leqslant L^{-2}B^{-2}$. Telescoping the above over $t < T$ gives

$$\|w_0 - v\|_2^2 - \|w_T - v\|_2^2 \geqslant 2\eta L^{-1} \sum_{t=0}^{T-1} \overline{F}(w_t).$$

Dividing each side by ηT shows the desired bound. □

We will show that if σ satisfies Assumption 2.3.1, then Lemma 2.9.2 allows for a population risk bound for essentially any distribution with $\mathbb{E}[\|x\|_2^2] \leqslant B^2$. In particular, we consider distributions with finite expected norm squared and the possible types of tail bounds for the norm.

Assumption 2.9.3. (a) Bounded distributions: there exists $B > 0$ such that $\|x\|_2 \leqslant B$ a.s.

(b) Exponential tails: there exist $a_0, C_e > 0$ such that $\mathbb{P}(\|x\|_2^2 \geqslant a) \leqslant C_e \exp(-a)$ holds for all $a \geqslant a_0$.

(c) Polynomial tails: there exist $a_0, C_p > 0$ and $\beta > 1$ such that $\mathbb{P}(\|x\|_2^2 \geqslant b) \leqslant C_p a^{-\beta}$ holds for all $a \geqslant a_0$.

If either (a), (b), or (c) holds, there exists $B > 0$ such that $\mathbb{E}\,\|x\|_2^2 \leqslant B^2$. One can verify that for (b), taking $B^2 = 2(a_0 \vee C_e)$ suffices, and for (c), $B^2 = 2(a_0 \vee C_p^{1/\beta}/(1-\beta))$ suffices. In fact, any distribution that satisfies $\mathbb{E}\,\|x\|_2^2 < \infty$ cannot have a tail bound of the form $\mathbb{P}(\|x\|_2^2 \geqslant a) = \Omega(a^{-1})$, since in this case we would have

$$\mathbb{E}\,\|x\|_2^2 = \int_0^\infty \mathbb{P}(\|x\|_2^2 > t)dt \geqslant C \int_{a_0}^\infty t^{-1} dt = \infty.$$

So the polynomial tail assumption (c) is tight up to logarithmic factors for distributions with finite $\mathbb{E}\,\|x\|_2^2$.

Theorem 2.9.4. Let $\mathbb{E}[\|x\|_2^2] \leqslant B^2$ and assume $\mathcal{D}$ satisfies one of the conditions in Assumption 2.9.3. Let σ satisfy Assumption 2.3.1.

(a) Under Assumption 2.9.3a, let γ be the constant corresponding to $\rho = 4B$ in Assumption 2.3.1. Running gradient descent for $T = \lceil 2\varepsilon^{-1} L \eta^{-1} \gamma^{-1} \|w_0 - v\|_2^2 \rceil$ guarantees there exists $t \in [T-1]$ such that $F(w_t) \leqslant \varepsilon$.

(b) Under Assumption 2.9.3b, let γ be the constant corresponding to $\rho = 4\sqrt{\log(18C_e/\varepsilon)}$. Running gradient descent for $T = \lceil 2\varepsilon^{-1}L\eta^{-1}\gamma^{-1}\left\|w_0 - v\right\|_2^2\rceil$ guarantees there exists $t \in [T-1]$ such that $F(w_t) \leqslant \varepsilon$.

(c) Under Assumption 2.9.3c, let γ be the constant corresponding to $\rho = 4(18C_p/\varepsilon(\beta - 1))^{(1-\beta)/2}$. Running gradient descent for $T = \lceil 2\varepsilon^{-1}L\eta^{-1}\gamma^{-1}\left\|w_0 - v\right\|_2^2\rceil$ guarantees there exists $t \in [T-1]$ such that $F(w_t) \leqslant \varepsilon$.

Proof. First, note that the conditions of Lemma 2.9.2 hold, so that we have for all $t = 0, \ldots, T-1$, $\left\|w_t\right\|_2 \leqslant 4$ and

$$\eta \sum_{t=0}^{T-1} \overline{F}(w_t) \leqslant L\left\|w_0 - v\right\|_2^2 - L\left\|w_T - v\right\|_2^2. \tag{2.9.6}$$

By taking $T = \zeta^{-1}L\varepsilon^{-1}\eta^{-1}\left\|w_0 - v\right\|_2^2$ for arbitrary $\zeta > 0$, (2.9.6) implies that there exists $t \in [T-1]$ such that

$$\overline{F}(w_t) = \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w_t^\top x)\right] \leqslant \frac{L\left\|w_0 - v\right\|_2^2}{\eta T} \leqslant \zeta\varepsilon. \tag{2.9.7}$$

It therefore suffices to bound $F(w_t)$ in terms of the left hand side of (2.9.7). We will do so by using the distributional assumptions given in Assumption 2.9.3 and by choosing ζ appropriately.

We begin by noting that (2.9.7) implies, for any $\rho > 0$,

$$\mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w_t^\top x)\mathbb{1}(|w_t^\top x| \leqslant \rho)\right] \leqslant \zeta\varepsilon. \tag{2.9.8}$$

For any $\rho > 0$, since $\left\|w_t\right\|_2 \leqslant 4$, the inclusion

$$\left\{\left\|x\right\|_2 \leqslant \rho/4\right\} \subset \left\{|w_t^\top x| \leqslant \rho\right\}, \tag{2.9.9}$$

holds. Under Assumption 2.9.3a, by taking $\rho = 4B$ and letting γ be the corresponding constant from Assumption 2.3.1, eqs. (2.9.8) and (2.9.9) imply

$$\gamma\mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2\right] \leqslant \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \sigma'(w_t^\top x)\mathbb{1}(\left\|x\right\|_2 \leqslant \rho/4)\right] \leqslant \zeta\varepsilon.$$

By taking $\zeta = \gamma/2$, this implies $F(w_t) \leqslant \varepsilon$.

Under Assumption 2.9.3b, by taking $\rho = 4\sqrt{a_0}$, we get

$$\begin{aligned}\mathbb{E}\left[\|x\|_2^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] &= \int_{a_0}^{\infty} \mathbb{P}(\|x\|_2^2 > t)dt \\ &\leqslant C_e \exp(-a_0). \qquad (2.9.10)\end{aligned}$$

Note that Assumption 2.9.3b holds if we take a_0 larger. We can therefore let a_0 be large enough so that $a_0 \geqslant \log(18C_e/\varepsilon)$, so that then

$$\mathbb{E}\left[\|x\|_2^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] \leqslant \varepsilon/18. \qquad (2.9.11)$$

Similarly, under Assumption 2.9.3c, we can let γ be the constant corresponding to $\rho = 4\sqrt{a_0}$ and take $a_0 \geqslant (\varepsilon(\beta-1)/18C_p)^{1/(1-\beta)}$ so that

$$\begin{aligned}\mathbb{E}\left[\|x\|_2^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] &= \int_{a_0}^{\infty} \mathbb{P}(\|x\|_2^2 > t)dt \\ &\leqslant C_p \frac{a_0^{1-\beta}}{\beta - 1} \\ &\leqslant \varepsilon/18.\end{aligned}$$

and so (2.9.11) holds as well under Assumption 2.9.3c. We can therefore bound

$$\begin{aligned}\mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x)\right)^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] &\leqslant \mathbb{E}\left[\|w_t - v\|_2^2 \|x\|_2^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] \\ &\leqslant \|w_0 - v\|_2^2 \mathbb{E}\left[\|x\|_2^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2)\right] \\ &\leqslant \|w_0 - v\|_2^2 \varepsilon/18 \\ &\leqslant \varepsilon/2. \qquad (2.9.12)\end{aligned}$$

The first inequality uses that σ is 1-Lipschitz and Cauchy–Schwarz. The second inequality uses (2.9.6). The third inequality uses (2.9.11). The final inequality uses that $\|w_0 - v\|_2 \leqslant \|w_0\|_2 + \|v\|_2 \leqslant 3$.

We can then guarantee

$$\begin{aligned}
2\gamma F(w_t) &= \gamma \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x) \right)^2 \right] \\
&= \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x) \right)^2 \gamma \mathbb{1}(|w_t^\top x| \leqslant \rho) \right] \\
&\quad + \gamma \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x) \right)^2 \mathbb{1}(|w_t^\top x| > \rho) \right] \\
&\leqslant \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x) \right)^2 \sigma'(w_t^\top x) \mathbb{1}(|w_t^\top x| \leqslant \rho) \right] \\
&\quad + \gamma \mathbb{E}\left[\left(\sigma(w_t^\top x) - \sigma(v^\top x) \right)^2 \mathbb{1}(\|x\|_2^2 > \rho^2/4^2) \right] \\
&\leqslant \zeta\varepsilon + \gamma\varepsilon/2 \\
&\leqslant \gamma\varepsilon.
\end{aligned}$$

The first inequality follows since Assumption 2.3.1 implies $\sigma'(z)\mathbb{1}(|z| \leqslant \rho) \geqslant \gamma \mathbb{1}(|z| \leqslant \rho)$ and by (2.9.9). The second inequality uses (2.9.8) and (2.9.12). The final inequality takes $\zeta = \gamma/2$. □

Remark 2.9.5. The precise runtime guarantee in Theorem 2.9.1 will depend upon the activation function and tail distribution. The worst-case activation functions (like the sigmoid) can have $\gamma \sim \exp(-\rho)$, and so if one only has polynomial tails, the runtime can be exponential in ε^{-1} in this case. If the distribution of $\|x\|_2^2$ has exponential tails, as is the case if the components of x are sub-Gaussian, runtime will be polynomial in ε^{-1}. On the other hand, if the γ in Assumption 2.3.1 is a fixed constant independent of ρ (as it is for the leaky ReLU), any of the tail bounds under consideration will have runtime of order ε^{-1}.

2.9.2 Stochastic gradient descent proofs

We consider the online version of stochastic gradient descent, where we sample independent samples $x_t \sim \mathcal{D}$ at each step and compute stochastic gradient updates g_t, such that

$$g_t = \left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t) \right) \sigma'(w_t^\top x_t) x_t, \quad w_{t+1} = w_t - \eta g_t.$$

As in the gradient descent case, we have a key lemma that relates the distance of the weights at iteration t from the optimal v with the distance from initialization and the cumulative loss.

Lemma 2.9.6. Assume that σ is non-decreasing and L-Lipschitz, and that $\mathcal{D}$ satisfies $\|x\|_2 \leqslant B$ a.s. Assume the initialization satisfies $\|w_0\|_2 \leqslant 2$. Let $T \in \mathbb{N}$ and run stochastic gradient descent for $T-1$ iterations at a fixed learning rate η satisfying $\eta \leqslant L^{-2}B^{-2}$. Then with probability one over $\mathcal{D}$, we have $\|w_{t+1} - v\|_2 \leqslant \|w_t - v\|_2$ for all $t < T$, and

$$\|w_0 - v\|_2^2 - \|w_T - v\|_2^2 \geqslant 2\eta L^{-1} \sum_{t=0}^{T-1} \overline{F}_t,$$

where $\overline{F}_t := \frac{1}{2}\left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t)\right)^2 \sigma'(w_t^\top x_t)$.

Proof. We begin with the decomposition

$$\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 = 2\eta \langle g_t, w_t - v\rangle - \eta^2 \|g_t\|_2^2 . \tag{2.9.13}$$

By Assumption 2.3.1, since $\|x\|_2 \leqslant B$ a.s. it holds with probability one that

$$\|g_t\|_2^2 = \left\|\left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t)\right)\sigma'(w_t^\top x_t) x_t\right\|_2^2 \leqslant 2LB^2\overline{F}_t. \tag{2.9.14}$$

By Fact 2.3.10, since $\sigma'(z) \geqslant 0$, we have with probability one,

$$\begin{aligned}\langle g_t, w_t - v\rangle &= \left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t)\right)\sigma'(w_t^\top x_t)(w_t^\top x_t - v^\top x_t)\\ &\geqslant L^{-1}\left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t)\right)^2 \sigma'(w_t^\top x_t)\\ &= 2L^{-1}\overline{F}_t. \end{aligned}\tag{2.9.15}$$

Putting (2.9.14) and (2.9.15) into (2.9.13), we get

$$\begin{aligned}\|w_t - v\|_2^2 - \|w_{t+1} - v\|_2^2 &\geqslant 4\eta L^{-1}\overline{F}_t - 2\eta^2 LB^2 \overline{F}_t\\ &\geqslant 2\eta L^{-1}\overline{F}_t,\end{aligned}$$

by taking $\eta \leqslant L^{-2}B^{-2}$. Telescoping over $t < T$ gives the desired bound.

□

We now want to translate the bound on the empirical error to that of the true error. For this we use a martingale Bernstein inequality of [BLL11]. A similar analysis of SGD was used by [JT20b] for a one-hidden-layer ReLU network.

Lemma 2.9.7 ([BLL11], Theorem 1)**.** Let $\{Y_t\}$ be a martingale adapted to the filtration $\mathcal{F}_t$, and let $Y_0 = 0$. Let $\{D_t\}$ be the corresponding martingale difference sequence. Define the sequence of conditional variance

$$V_t := \sum_{k=1}^{t} \mathbb{E}[D_k^2 | \mathcal{F}_{k-1}],$$

and assume that $D_t \leqslant R$ almost surely. Then for any $\delta \in (0,1)$, with probability greater than $1-\delta$,

$$Y_t \leqslant R \log(1/\delta) + (e-2)V_t/R.$$

Lemma 2.9.8. Suppose that $\|x\|_2 \leqslant B$ a.s., and let σ be non-decreasing and L-Lipschitz. Assume that the trajectory of SGD satisfies $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$ for all t a.s. We then have with probability at least $1-\delta$,

$$\frac{1}{T}\sum_{t=0}^{T-1} \overline{F}(w_t) \leqslant \frac{4}{T}\sum_{t=0}^{T-1} \overline{F}_t + \frac{2}{T} B^2 L^3 \|w_0 - v\|_2^2 \log(1/\delta).$$

Proof. Let $\mathcal{F}_t = \sigma(x_0, \ldots, x_t)$ be the σ-algebra generated by the first $t+1$ draws from $\mathcal{D}$. Then the random variable $G_t := \sum_{\tau=0}^{t}(\overline{F}(w_\tau) - \overline{F}_\tau)$ is a martingale with respect to the filtration $\mathcal{F}_t$ with martingale difference sequence $D_t := \overline{F}(w_t) - \overline{F}_t$. We need bounds on D_t and on $\mathbb{E}[D_t^2|\mathcal{F}_{t-1}]$ in order to apply Lemma 2.9.7.

Since σ is L-Lipschitz and $\|x\|_2 \leqslant B$ a.s., with probability one we have

$$D_t \leqslant \overline{F}(w_t) \leqslant \frac{1}{2} L^3 B^2 \|w_t - v\|_2^2 \leqslant \frac{1}{2} L^3 B^2 \|w_0 - v\|_2^2 . \tag{2.9.16}$$

The last inequality uses the assumption that $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$ a.s. Similarly,

$$\begin{aligned} \mathbb{E}[\overline{F}_t^2 | \mathcal{F}_{t-1}] &= \frac{1}{4}\mathbb{E}\left[\left(\sigma(w_t^\top x_t) - \sigma(v^\top x_t)\right)^4 \sigma'(w_t^\top x_t)^2 | \mathcal{F}_{t-1}\right] \\ &\leqslant \frac{1}{4} L^3 B^2 \|w_t - v\|_2^2 \, \mathbb{E}_x \left[\left(\sigma(w_t x_t) - \sigma(v^\top x_t)\right)^2 \sigma'(w_t^\top x_t) | \mathcal{F}_{t-1}\right] \\ &\leqslant \frac{1}{2} L^3 B^2 \|w_0 - v\|_2^2 \, \overline{F}(w_t). \end{aligned} \tag{2.9.17}$$

In the first inequality, we have used $\|x\|_2^2 \leqslant B^2$ a.s. and L-Lipschitzness of σ. For the second, we use the assumption that $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$ together with the fact that $\mathbb{E}_x[\overline{F}_t|\mathcal{F}_{t-1}] = \overline{F}(w_t)$. We then can use (2.9.17) to bound the squared increments,

$$\begin{aligned}\mathbb{E}[D_t^2|\mathcal{F}_{t-1}] &= \overline{F}(w_t)^2 - 2\overline{F}(w_t)\mathbb{E}[\overline{F}_t|\mathcal{F}_{t-1}] + \mathbb{E}[\overline{F}_t^2|\mathcal{F}_{t-1}] \\ &= -\overline{F}(w_t)^2 + \mathbb{E}[\overline{F}_t^2|\mathcal{F}_{t-1}] \\ &\leqslant \frac{1}{2}L^3B^2\|w_0 - v\|_2^2\,\overline{F}(w_t). \qquad (2.9.18)\end{aligned}$$

This allows for us to bound

$$V_T := \sum_{t=0}^{T-1} \mathbb{E}[D_t^2|\mathcal{F}_{t-1}] \leqslant \frac{1}{2}B^2L^3\|w_0 - v\|_2^2 \sum_{t=0}^{T-1}\overline{F}(w_t).$$

Since $D_t \leqslant \overline{F}(w_t) \leqslant (1/2)L^3B^2\|w_0 - v\|_2^2$ a.s. by (2.9.16), Lemma 2.9.7 implies that with probability at least $1-\delta$, we have

$$\sum_{t=0}^{T-1}(\overline{F}(w_t) - \overline{F}_t) \leqslant (\exp(1) - 2)\sum_{t=0}^{T-1}\overline{F}(w_t) + \frac{1}{2}L^3B^2\|w_0 - v\|_2^2\log(1/\delta),$$

and using that $(1 - \exp(1) + 2)^{-1} \leqslant 4$, we divide each side by T and get

$$\frac{1}{T}\sum_{t=0}^{T-1}\overline{F}(w_t) \leqslant \frac{4}{T}\sum_{t=0}^{T-1}\overline{F}_t + \frac{2}{T}L^3B^2\|w_0 - v\|_2^2\log(1/\delta). \qquad (2.9.19)$$

□

With the above in hand, we can prove Theorem 2.9.1 in the SGD setting.

Proof of Theorem 2.9.1, SGD. By the assumptions in the theorem, Lemma 2.9.6 holds, so that we have for any $t = 0, \ldots, T-1$, $\|w_t\|_2 \leqslant 4$ and

$$\|w_t - v\|_2^2 + 2\eta L^{-1}\sum_{\tau=0}^{t-1}\overline{F}_\tau \leqslant \|w_0 - v\|_2^2. \qquad (2.9.20)$$

This shows that $\|w_t - v\|_2 \leqslant \|w_0 - v\|_2$ holds for all $t = 0, \ldots, T-1$ a.s., allowing for the application of Lemma 2.9.8 to get

$$\frac{1}{T}\sum_{t=0}^{T-1}\overline{F}(w_t) \leqslant \frac{4}{T}\sum_{t=1}^{T}\overline{F}_t + \frac{2}{T}L^3B^2\|w_0 - v\|_2^2\log(1/\delta). \qquad (2.9.21)$$

Dividing both sides of (2.9.20) by $\eta T L^{-1}$ yields

$$\min_{t<T} \overline{F}(w_t) \leqslant \frac{1}{T}\sum_{t=0}^{T-1} \overline{F}(w_t) \leqslant \frac{L\left\|w_0 - v\right\|_2^2}{\eta T} + \frac{2}{T} L^3 B^2 \left\|w_0 - v\right\|_2^2 \log(1/\delta).$$

For arbitrary $\zeta > 0$, taking $T = \lceil 2\varepsilon^{-1}\zeta^{-1}\eta^{-1}L^3B^2\left\|w_0 - v\right\|_2^2 \log(1/\delta)\rceil$ shows there exists T such that $\overline{F}(w_t) \leqslant \zeta\varepsilon$. When σ satisfies Assumption 2.3.1, since $\left\|w_t\right\|_2 \leqslant 4$ for all t, it holds that $\overline{F}(w_t) \geqslant \gamma F(w_t)$, so that $\zeta = \gamma$ furnishes the desired bound.

When σ is ReLU and $\mathcal{D}$ satisfies Assumption 2.3.2, we note that Lemma 2.9.6 implies $\left\|w_t - v\right\|_2 \leqslant \left\|w_0 - v\right\|_2$ a.s. Thus taking $\zeta = \alpha^4\beta/8\sqrt{2}$ and using Lemma 2.3.5 completes the proof. □

2.10 Remaining Proofs

Proof of Lemma 2.3.8. Since σ is non-decreasing, $|\sigma(v^\top x) - y| \leqslant |\sigma(B_X)| + B_Y$. In particular, each summand defining $\widehat{F}(v)$ is a random variable with absolute value at most $a = (|\sigma(B_X)| + B_Y)^2$. As $\mathbb{E}[\widehat{F}(v)] = F(v) = \mathsf{OPT}$, Hoeffding's inequality implies the lemma. □

Proof of Lemma 2.3.9. The bound $\mathfrak{R}_S(\mathcal{G}) \leqslant 2\max_i \left\|x_i\right\|_2/\sqrt{n}$ follows since $\left\|w\right\|_2 \leqslant 2$ holds on $\mathcal{G}$ with standard results Rademacher complexity theory (e.g. Sec. 26.2 of [SB14]); this shows $\mathfrak{R}(\mathcal{G}) \leqslant 2B_X/\sqrt{n}$. Using the contraction property of the Rademacher complexity, this implies $\mathfrak{R}(\sigma \circ \mathcal{G}) \leqslant 2B_X L/\sqrt{n}$. Finally, note that if $\left\|w - v\right\|_2 \leqslant 1$ and $\left\|x\right\|_2 \leqslant B_X$, we have

$$\left\|\nabla \ell(w; x)\right\| = \left\|\left(\sigma(w^\top x) - \sigma(v^\top x)\right)\sigma'(w^\top x)x\right\| \leqslant L^2 \left\|w - v\right\| \left\|x\right\| \leqslant L^2 B_X. \tag{2.10.1}$$

In particular, ℓ is $L^2 B_X$ Lipschitz. The result follows. □

CHAPTER 3

Learning noisy halfspaces with logistic regression

3.1 Introduction

In this chapter, we take a closer look at learning single neurons $x \mapsto \sigma(\langle w, x\rangle)$ under the zero-one loss, which is the standard loss of interest for classification problems. The standard approach for minimizing the classification error in neural networks is to perform gradient descent on convex surrogates of the zero-one loss, such as the cross entropy loss $\ell(z) = \log(1 + \exp(-z))$, by considering the objective function

$$F(w) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(y\sigma(\langle w, x\rangle)).$$

Using similar ideas from Chapter 2, we can derive guarantees for the population risk under the cross-entropy loss when using a single neuron under the assumption that σ is strictly increasing and $\sigma(0) = 0$, with the final result showing that gradient descent can efficiently find approximate minimizers of the cross-entropy loss. However, our goal is not to find minimizers for the *surrogate* loss, but minimizers for the zero-one loss itself. It turns out that this is a much more intricate matter, even when the activation function σ is the identity function. Note also that if $\sigma(z)$ is such that $\sigma(z) > 0$ if $z > 0$ and $\sigma(z) < 0$ if $z < 0$, then the hypothesis space induced by $\sigma(\langle w, x\rangle)$ is the same as that induced by $\langle w, x\rangle$, i.e. the single neuron with the identity activation. (This condition is satisfied by many common activation functions in practice, such as the leaky ReLU, tanh, arctan, ELU, swish activation, etc.) For this reason, in this chapter we restrict ourselves to the case of $\sigma(z) = z$. This reduces our problem to the agnostic learning of halfspaces, a long-studied problem in learning theory.

By a *halfspace* we mean a function $x \mapsto \mathrm{sgn}(w^\top x) \in \{\pm 1\}$ for some $w \in \mathbb{R}^d$. Let $\mathcal{D}$ be a joint distribution over (x, y), where the inputs $x \in \mathbb{R}^d$ and the labels $y \in \{\pm 1\}$, and denote by $\mathcal{D}_x$ the marginal of $\mathcal{D}$ over x. We are interested in the performance of halfspaces found by gradient descent in comparison to the best-performing halfspace over $\mathcal{D}$, so let us define, for $w \in \mathbb{R}^d$,

$$\mathrm{err}^{0-1}(w) := \mathbb{P}_{(x,y)\sim\mathcal{D}}(\mathrm{sgn}(w^\top x) \neq y),$$
$$\mathsf{OPT}_{01} := \min_{\|w\|=1} \mathrm{err}^{0-1}(w).$$

Due to the non-convexity and discontinuity of the zero-one loss, the standard approach for minimizing the classification error is to consider a convex surrogate loss $\ell : \mathbb{R} \to \mathbb{R}$ for which $\mathbb{1}(z < 0) \leqslant O(\ell(z))$ and to instead minimize the surrogate risk

$$F(w) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\big[\ell(yw^\top x)\big]. \tag{3.1.1}$$

Without access to the population risk itself, one can take samples $\{(x_i, y_i)\}_{i=1}^n \overset{\text{i.i.d.}}{\sim} \mathcal{D}$ and optimize (3.1.1) by gradient descent on the empirical risk $\widehat{F}(w)$, defined by taking the expectation in (3.1.1) over the empirical distribution of the samples. By using standard tools from convex optimization and Rademacher complexity, such an approach is guaranteed to efficiently minimize the population surrogate risk up to optimization and statistical error. The question is then, given that we have found a halfspace $x \mapsto w^\top x$ that minimizes the *surrogate* risk, how does this halfspace compare to the *best* halfspace as measured by the zero-one loss? And how does the choice of the surrogate loss affect this behavior? To the best of our knowledge, no previous work has been able to demonstrate that gradient descent on convex surrogates can yield approximate minimizers for the classification error over halfspaces, even for the case of the standard logistic (binary cross-entropy) loss $\ell(z) = \log(1 + \exp(-z))$ or the hinge loss $\ell(z) = \max(1 - z, 0)$.

We show below that the answer to these questions depend upon what we refer to as the *soft margin function* of the distribution at a given minimizer for the zero-one loss. (We note that in general, there may be multiple minimizers for the zero-one loss, and so we can only refer to *a* given minimizer.) For $\bar{v} \in \mathbb{R}^d$ satisfying $\|\bar{v}\| = 1$, we say that the halfspace $\bar{v}$ satisfies the $\phi_{\bar{v}}$-soft-margin property if for some function $\phi_{\bar{v}} : [0, 1] \to \mathbb{R}$, for all $\gamma \in [0, 1]$,

$$\mathbb{P}_{\mathcal{D}_x}(|\bar{v}^\top x| \leqslant \gamma) \leqslant \phi_v(\gamma).$$

To get a flavor for how this soft margin can be used to show that gradient descent finds approximately optimal halfspaces, for bounded distributions $\mathcal{D}_x$, we show in Theorem 3.5.2 below that with high probability,

$$\mathrm{err}^{0-1}(w_T) \leqslant \inf_{\gamma\in(0,1)} \left\{ O(\gamma^{-1}\mathsf{OPT}_{01}) + \phi_{\bar v}(\gamma) + O(\gamma^{-1}n^{-1/2}) + \varepsilon \right\},$$

where $\phi_{\bar v}$ is a soft margin function corresponding to a unit norm minimizer $\bar v$ of the population zero-one loss. Thus, by analyzing the properties of $\phi_{\bar v}$, one can immediately derive approximate agnostic learning results for the output of gradient descent. In particular, we are able to show the following guarantees for the output of gradient descent:

1. **Hard margin distributions**. If $\|x\| \leqslant B_X$ almost surely and there is $\bar\gamma > 0$ such that $\bar v^\top x \geqslant \bar\gamma$ a.s., then $\mathrm{err}^{0-1}(w_t) \leqslant \tilde O(\bar\gamma^{-1}\mathsf{OPT}_{01}) + \varepsilon$.

2. **Sub-exponential distributions satisfying anti-concentration.** If random vectors from $\mathcal{D}_x$ are sub-exponential and satisfy an anti-concentration inequality for projections onto one dimensional subspaces, then $\mathrm{err}^{0-1}(w_t) \leqslant \tilde O(\mathsf{OPT}_{01}^{1/2}) + \varepsilon$. This covers any log-concave isotropic distribution.

For each of our guarantees, the runtime and sample complexity are $\mathrm{poly}(d, \varepsilon^{-1})$. The exact rates are given in Corollaries 3.5.3, 3.5.6 and 3.5.11. In Table 3.1 we compare our results with known lower bounds in the literature. To the best of our knowledge, our results are the first to show that gradient descent on convex surrogates for the zero-one loss can learn halfspaces in the presence of agnostic label noise, despite the ubiquity of this approach for classification problems.

The remainder of the chapter is organized as follows. In Section 3.2, we review the literature on learning halfspaces in the presence of noise. In Section 3.3, we discuss the notion of soft margins which will be essential to our proofs, and provide examples of soft margin behavior for different distributions. In Section 3.4 we show that gradient descent efficiently finds minimizers of convex surrogate risks and discuss how the tail behavior of the

Table 3.1: Comparison of halfspace results with other upper and lower bounds in the literature.

Algorithm	$\mathcal{D}_x$	Population Risk	Known Lower Bound
Non-convex G.D. [DKT20b]	Concentration, anti-concentration	$O(\mathsf{OPT}_{01})$	N/A
Convex G.D. (this work)	Sub-exponential, anti-concentration	$\tilde{O}(\mathsf{OPT}_{01}^{1/2})$	$\Omega(\mathsf{OPT}_{01}\log^{\alpha}({}^{1}\!/_{\mathsf{OPT}_{01}}))$ [DKT20b]
Convex G.D. (this work)	Hard margin	$\tilde{O}(\bar{\gamma}^{-1}\mathsf{OPT}_{01})$	$\Omega(\bar{\gamma}^{-1}\mathsf{OPT}_{01})$ [DGT19]

loss function can affect the time and sample complexities of gradient descent. In Section 3.5 we provide our main results, which relies upon using soft margins to convert minimizers for the convex surrogate risk to approximate minimizers for the classification error. We conclude in Section 3.6.

3.2 Related Work

The problem of learning halfspaces is a classical problem in machine learning with a history almost as long as the history of machine learning itself, starting from the perceptron [Ros58] and support vector machines [BGV92] to today. Much of the early works on this problem focused on the realizable setting, i.e. where $\mathsf{OPT}_{01} = 0$. In this setting, the Perceptron algorithm or methods from linear programming can be used to efficiently find the optimal halfspace. In the setting of agnostic PAC learning [KSS94] where $\mathsf{OPT}_{01} > 0$ in general, the question of which distributions can be learned up to classification error $\mathsf{OPT}_{01} + \varepsilon$, and whether it is possible to do so in $\mathrm{poly}(d, 1/\varepsilon)$ time (where d is the input dimension), is significantly more difficult and is still an active area of research. It is known that without distributional assumptions, learning up to even $O(\mathsf{OPT}_{01}) + \varepsilon$ is NP-hard, both for proper learning [GR09] and improper learning [Dan16]. Due to this difficulty, it is common to make a number of assumptions on either $\mathcal{D}_x$ or to impose some type of structure to the learning problem.

A common structure imposed is that of structured noise: one can assume that there exists some underlying halfspace $y = \mathrm{sgn}(v^\top x)$ that is corrupted with probability $\eta(x) \in [0, 1/2)$, possibly dependent on the features x. The simplest setting is that of random classification noise, where $\eta(x) \equiv \eta$, so that each label is flipped with the same probability [AL88]; polynomial time algorithms for learning under this noise condition were shown by [BFK98]. The Massart noise model introduced by [MN06] relaxes this assumption to $\eta(x) \leqslant \eta$ for some absolute constant $\eta < 1/2$. The Tsybakov noise model [Tsy04] is a generalization of the Massart noise model that instead requires a tail bound on $\mathbb{P}(\eta(x) \geqslant 1/2 - t)$ for $t > 0$. [ABH15] showed that optimally learning halfspaces under Massart noise is possible for the uniform distribution on the unit sphere, and [ABH16] showed this for log-concave isotropic distributions. The recent landmark result of [DGT19] provided the first distribution-independent result for optimally learning halfspaces under Massart noise, answering a long-standing [Slo88] open problem in computational learning.

By contrast, in the agnostic PAC learning setting, one makes no assumptions on $\eta(x)$, so one can equivalently view agnostic PAC learning as an adversarial noise model in which an adversary can corrupt the label of a sample x with any probability $\eta(x) \in [0, 1]$. Recent work suggests that even when $\mathcal{D}_x$ is the Gaussian, agnostically learning up to exactly $\mathsf{OPT}_{01} + \varepsilon$ likely requires $\exp(1/\varepsilon)$ time [GGK20, DKZ20]. In terms of positive results in the agnostic setting, [KKM08] showed that a variant of the Average algorithm [Ser99] can achieve risk $O(\mathsf{OPT}_{01}\sqrt{\log(1/\mathsf{OPT}_{01})})$ risk in $\mathrm{poly}(d, 1/\varepsilon)$ time when $\mathcal{D}_x$ is uniform over the unit sphere. [ABL17] demonstrated that a localization-based algorithm can achieve $O(\mathsf{OPT}_{01}) + \varepsilon$

under log-concave isotropic marginals. [DKT20b] showed that for a broad class of distributions, the output of projected SGD on a nonconvex surrogate for the zero-one loss produces a halfspace with risk $O(\mathsf{OPT}_{01}) + \varepsilon$ in $\mathrm{poly}(d, 1/\varepsilon)$ time. For more background on learning halfspaces in the presence of noise, we refer the reader to [BH21].

We note that [DKT20b] also showed that the minimizer of the surrogate risk of any *convex* surrogate for the zero-one loss is a halfspace with classification error $\omega(\mathsf{OPT}_{01})$. [BLS12] and [ABL17] showed similar lower bounds that together imply that empirical risk minimization procedures for convex surrogates yield halfspaces with classification error $\Omega(\mathsf{OPT}_{01})$. Given such lower bounds, we wish to emphasize that in this chapter we are *not* making a claim about the optimality of gradient descent (on convex surrogates) for learning halfspaces. Rather, our main interest is the characterization of what are the strongest learning guarantees possible with what is perhaps the simplest learning algorithm possible. Given the success of gradient descent for the learning of deep neural networks, and the numerous questions that this success has brought to the theory of statistics and machine learning, we think it is important to develop a thorough understanding of what are the possibilities of vanilla gradient descent, especially in the simplest setting possible.

Recent work has shown that gradient descent finds approximate minimizers for the population risk of single neurons $x \mapsto \sigma(w^\top x)$ under the squared loss [DGK20b, FCG20], despite the computational intractability of finding the optimal single neuron [GKK19]. The main contribution of this chapter is that despite the computational difficulties in *exact* agnostic learning, the standard gradient descent algorithm satisfies an *approximate* agnostic PAC learning guarantee, in line with the results found by [FCG20] for the single neuron.

3.2.1 Notation

We say that a differentiable loss function ℓ is L-Lipschitz if $|\ell'(z)| \leqslant L$ for all z in its domain, and we say the loss is H-smooth if its derivative ℓ' is H-Lipschitz. We use the word "decreasing" interchangeably with "non-increasing". We use the standard $O(\cdot), \Omega(\cdot)$ order notations to hide universal constants and $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$ to additionally suppress logarithmic factors. Throughout this chapter, $\|x\|$ refers to the standard Euclidean norm on $\mathbb{R}^d$ induced by the inner product $x^\top x$. We will emphasize that a vector v is of unit norm by writing $\bar{v}$. We assume $\mathcal{D}$ is a probability distribution over $\mathbb{R}^d \times \{\pm 1\}$ with marginal distribution $\mathcal{D}_x$ over $\mathbb{R}^d$.

3.3 Soft Margins

In this section we will formally introduce the soft margin function and describe some common distributions for which it takes a simple form.

Definition 3.3.1. Let $\bar{v} \in \mathbb{R}^d$ satisfy $\|\bar{v}\| = 1$. We say $\bar{v}$ satisfies the *soft margin condition with respect to a function* $\phi_{\bar{v}} : \mathbb{R} \to \mathbb{R}$ if for all $\gamma \in [0, 1]$, it holds that

$$\mathbb{E}_{x \sim \mathcal{D}_x} \left[\mathbb{1} \left(|\bar{v}^\top x| \leqslant \gamma \right) \right] \leqslant \phi_{\bar{v}}(\gamma).$$

We note that our definition of soft margin is essentially an unnormalized version of the soft margin function considered by [FSS18] in the context of learning GLMs, since they defined $\phi_{\bar{v}}(\gamma)$ as the probability that $|\bar{v}^\top x / \|x\| \, | \leqslant \gamma$. This concept was also considered by [BZ17] for s-concave isotropic distributions under the name ‘probability of a band’.

Below we will consider some examples of soft margin function behavior. We shall see later that our final generalization bounds will depend on the behavior of $\phi_{\bar{v}}(\gamma)$ for γ sufficiently small, and thus in the below examples we only care about the behavior of $\phi_{\bar{v}}(\cdot)$ in small neighborhoods of the origin. In our first example, we show that (hard) margin distributions have simple soft margin functions.

Example 3.3.2 (Hard margin distributions)**.** If $\mathcal{D}_x$ is a hard margin distribution in the sense that $\bar{v}^\top x \geqslant \gamma^* > 0$ for some $\gamma^* > 0$ almost surely, then $\phi_{\bar{v}}(\gamma) = 0$ for $\gamma < \gamma^*$.

Proof. This follows immediately: $\mathbb{P}(|\bar{v}^\top x| \leqslant \gamma) = 0$ when $\gamma < \gamma^*$. □

Note that the soft margin function in Example 3.3.2 is specific to the vector $\bar{v}$, and does not necessarily hold for arbitrary unit vectors in $\mathbb{R}^d$. By contrast, for many distributions it is possible to derive bounds on soft margin functions that hold for *any* vector $\bar{v}$, which we shall see below is a key step for deriving approximate agnostic learning guarantees for the output of gradient descent.

The next example shows that provided the projections of $\mathcal{D}_x$ onto one dimensional subspaces satisfy an anti-concentration property, then all soft margins function for that distribution take a simple form. To do so we first introduce the following definition.

Definition 3.3.3 (Anti-concentration)**.** For $\bar{v} \in \mathbb{R}^d$, denote by $p_{\bar{v}}(\cdot)$ the marginal distribution of $x \sim \mathcal{D}_x$ on the subspace spanned by $\bar{v}$. We say $\mathcal{D}_x$ satisfies *U-anti-concentration* if there is some $U > 0$ such that for all unit norm $\bar{v}$, $p_{\bar{v}}(z) \leqslant U$ for all $z \in \mathbb{R}$.

A similar assumption was used in [DKT20a, DKT20b, DKT21] for learning halfspaces; in their setup, the anti-concentration assumption was for the projections of $\mathcal{D}_x$ onto two dimensional subspaces rather than the one dimensional version we consider here.

Example 3.3.4 (Distributions satisfying anti-concentration)**.** If $\mathcal{D}_x$ satisfies U anti concentration, then for any unit norm $\bar{v}$, $\phi_{\bar{v}}(\gamma) \leqslant 2U\gamma$.

Proof. We can write $\mathbb{P}(|\bar{v}^\top x| \leqslant \gamma) = \int_{-\gamma}^{\gamma} p_{\bar{v}}(z)\mathrm{d}z \leqslant 2\gamma U$. □

We will show below that log-concave isotropic distributions satisfy U-anti-concentration for $U = 1$. We first remind the reader of the definition of log-concave isotropic distributions.

Definition 3.3.5. We say that a distribution $\mathcal{D}_x$ over $x \in \mathbb{R}^d$ is *log-concave* if it has a density function $p(\cdot)$ such that $\log p(\cdot)$ is concave. We call $\mathcal{D}_x$ *isotropic* if its mean is the zero vector and its covariance matrix is the identity matrix.

Typical examples of log-concave isotropic distributions include the standard Gaussian and the uniform distribution over a convex set.

Example 3.3.6 (Log-concave isotropic distributions)**.** If $\mathcal{D}_x$ is log-concave isotropic then it satisfies 1-anti-concentration, and thus for any $\bar{v}$ with $\|\bar{v}\| = 1$, $\phi_{\bar{v}}(\gamma) \leqslant 2\gamma$.

Proof. This was demonstrated in [BZ17, Proof of Theorem 11].[1] □

3.4 Gradient Descent Finds Minimizers of the Surrogate Risk

We begin by demonstrating that gradient descent finds weights that achieve the best population level surrogate risk. The following theorem is a standard result from stochastic optimization. For completeness, we present its proof in Section 3.11.

Theorem 3.4.1. Suppose $\|x\| \leqslant B_X$ a.s. Let ℓ be convex, L-Lipschitz, and H-smooth, with $\ell(0) \leqslant 1$. Let $v \in \mathbb{R}^d$ be arbitrary with $\|v\| \leqslant V$ for some $V > 1$, and suppose that the initialization w_0 satisfies $\|w_0\| \leqslant V$. For any $\varepsilon, \delta > 0$ and for any provided $\eta \leqslant (2/5)H^{-1}B_X^{-2}$, if gradient descent is run for $T = (4/3)\eta^{-1}\varepsilon^{-1}\|w_0 - v\|^2$, then with probability at least $1 - \delta$,

$$F(w_{T-1}) \leqslant F(v) + \frac{4B_X V L}{\sqrt{n}} + 8B_X V\sqrt{\frac{2\log(2/\delta)}{n}}.$$

This shows that gradient descent learns halfspaces that have a population surrogate risk competitive with that of the best predictor with bounded norm for any norm threshold V. For distributions that are linearly separable by some margin $\gamma > 0$, the above theorem allows us to derive upper bounds on the sample complexity that suggest that exponentially tailed losses are preferable to polynomially tailed losses from both time and sample complexity perspectives, touching on a recent problem posed by [JDS20].

Corollary 3.4.2 (Sample complexity for linearly separable data)**.** Assume $\|x\| \leqslant B_X$ a.s. Suppose that for some $\bar{v} \in \mathbb{R}^d$, $\|\bar{v}\| = 1$, there is $\gamma > 0$ such that $y\bar{v}^\top x \geqslant \gamma$ a.s. If ℓ is convex, decreasing, L-Lipschitz, and H-smooth, and if we fix a step size of $\eta \leqslant (2/5)H^{-1}B_X^{-2}$, then

[1] The cited theorem implies a similar bound of the form $O(\gamma)$ holds for the more general set of s-concave isotropic distributions. We focus here on log-concave isotropic distributions for simplicity.

- Assume ℓ has polynomial tails, so that for some $C_0, p > 0$ and $\ell(z) \leqslant C_0 z^{-p}$ holds for all $z \geqslant 1$. Provided $n = \Omega(\gamma^{-2}\varepsilon^{-2-2/p})$, then running gradient descent for $T = \Omega(\gamma^{-2}\varepsilon^{-1-2/p})$ iterations guarantees that $\mathrm{err}^{0-1}(w_T) \leqslant \varepsilon$.
- Assume ℓ has exponential tails, so that for some $C_0, C_1, p > 0$, $\ell(z) \leqslant C_0 \exp(-C_1 z^p)$ holds for all $z \geqslant 1$. Then $n = \tilde{\Omega}(\gamma^{-2}\varepsilon^{-2})$ and $T = \tilde{\Omega}(\gamma^{-2}\varepsilon^{-1})$ guarantees that $\mathrm{err}^{0-1}(w_T) \leqslant \varepsilon$.

The proof for the above Corollary can be found in Section 3.10. At a high level, the above result shows that if the tails of the loss function are heavier, one may need to run gradient descent for longer to drive the population surrogate risk, and hence the zero-one risk, to zero.[2] In the subsequent sections, we shall see that this phenomenon persists beyond the linearly separable case to the more general agnostic learning setting.

Remark 3.4.3. The sample complexity in Theorem 3.4.1 can be improved from $O(\varepsilon^{-2})$ to $O(\varepsilon^{-1})$ if we use online stochastic gradient descent rather than vanilla gradient descent. The proof of this is somewhat more involved as it requires a technical workaround to the unboundedness of the loss function, and may be of independent interest. We present the full analysis of this in Section 3.7.

3.5 Gradient Descent Finds Approximate Minimizers for the Zero One Loss

We now show how we can use the soft margin function to develop bounds for the zero-one loss of the output of gradient descent.

[2] We note that in Corollary 3.4.2, there is a gap for the sample complexity and runtime when using polynomially tailed vs. exponentially tailed losses. However, such a gap may be an artifact of our analysis. Deriving matching lower bounds for the sample complexity or runtime of gradient descent on polynomially tailed losses remains an open problem.

3.5.1 Bounded Distributions

We first focus on the case when the marginal distribution $\mathcal{D}_x$ is bounded almost surely.

By Theorem 3.4.1, since by Markov's inequality we have that $\mathrm{err}^{0-1}(w) \leqslant \ell(0)^{-1} F(w)$, if we want to show that the zero-one population risk for the output of gradient descent is competitive with that of the optimal zero-one loss achieved by some halfspace $v \in \mathbb{R}^d$, it suffices to bound $F(v)$ by some function of OPT_{01}. To do so we decompose the expectation for $F(v)$ into a sum of three terms which incorporate OPT_{01}, the soft margin function, and a term that drives the surrogate risk to zero by driving up the margin on those samples that are correctly classified.

Lemma 3.5.1. Let $\bar{v}$ be a unit norm population risk minimizer for the zero-one loss, and suppose $\bar{v}$ satisfies the soft margin condition with respect to some $\phi : [0,1] \to \mathbb{R}$. Assume that $\|x\| \leqslant B_X$ a.s. Let $v = V\bar{v}$ for $V > 0$ be a scaled version of $\bar{v}$. If ℓ is decreasing, L-Lipschitz and $\ell(0) \leqslant 1$, then

$$F(v) \leqslant \inf_{\gamma > 0} \Big\{ (1 + LVB_X)\mathsf{OPT}_{01} + \phi(\gamma) + \ell(V\gamma) \Big\}.$$

Proof. We begin by writing the expectation as a sum of three terms,

$$\begin{aligned} \mathbb{E}[\ell(yv^\top x)] = \mathbb{E}\left[\ell(yv^\top x)\mathbb{1}\left(y\bar{v}^\top x \leqslant 0\right)\right] \\ + \mathbb{E}\left[\ell(yv^\top x)\mathbb{1}\left(0 < y\bar{v}^\top x \leqslant \gamma\right)\right] \\ + \mathbb{E}\left[\ell(yv^\top x)\mathbb{1}\left(y\bar{v}^\top x > \gamma\right)\right]. \end{aligned} \tag{3.5.1}$$

For the first term, we use that ℓ is L-Lipschitz and decreasing as well as Cauchy–Schwarz to get

$$\begin{aligned} \mathbb{E}[\ell(yv^\top x)\mathbb{1}(y\bar{v}^\top x \leqslant 0)] &\leqslant \mathbb{E}[(1 + L|v^\top x|)\mathbb{1}(y\bar{v}^\top x \leqslant 0)] \\ &\leqslant (1 + LVB_X)\mathbb{E}[\mathbb{1}(y\bar{v}^\top x \leqslant 0)] \\ &= (1 + LVB_X)\mathsf{OPT}_{01}. \end{aligned}$$

In the last inequality we use that $\|x\| \leqslant B_X$ a.s. For the second term,

$$\mathbb{E}\left[\ell(yv^\top x)\mathbb{1}\left(0 < y\bar{v}^\top x \leqslant \gamma\right)\right] \leqslant \ell(0)\mathbb{E}\left[\mathbb{1}\left(0 < y\bar{v}^\top x \leqslant \gamma\right)\right] \leqslant \phi(\gamma), \tag{3.5.2}$$

where we have used that ℓ is decreasing in the first inequality and Definition 3.3.1 in the second. Finally, for the last term, we can use that ℓ is decreasing to get

$$\begin{aligned}&\mathbb{E}\left[\ell(yv^\top x)\mathbb{1}\left(y\bar{v}^\top x > \gamma\right)\right]\\ &\quad = \mathbb{E}\left[\ell(yV\bar{v}^\top x)\mathbb{1}\left(yV\bar{v}^\top x > V\gamma\right)\right] \leqslant \ell(V\gamma).\end{aligned} \tag{3.5.3}$$

□

In order to concretize this bound, we want to take V large enough so that the $\ell(V\gamma)$ term is driven to zero, but not so large so that the term in front of OPT_{01} grows too large. Theorem 3.4.1 is given in terms of an arbitrary $v \in \mathbb{R}^d$, and so in particular holds for $v = V\bar{v}$. We can view the results of Theorem 3.4.1 as stating an equivalence between running gradient descent for longer and for driving the norm $\|v\| = V$ to be larger.

We formalize the above intuition into Theorem 3.5.2 below. Before doing so, we introduce the following notation. For general decreasing function ℓ, for which an inverse function may or may not exist, we overload the notation ℓ^{-1} by denoting $\ell^{-1}(t) := \inf\{z : \ell(z) \leqslant t\}$.

Theorem 3.5.2. Suppose $\|x\| \leqslant B_X$ a.s. Let ℓ be convex, decreasing, L-Lipschitz, and H-smooth, with $0 < \ell(0) \leqslant 1$. Assume that a unit norm population risk minimizer of the zero-one loss, $\bar{v}$, satisfies the ϕ-soft-margin condition for some increasing $\phi : \mathbb{R} \to \mathbb{R}$. Fix a step size $\eta \leqslant (2/5)H^{-1}B_X^{-2}$. Let $\varepsilon_1, \gamma > 0$ and $\varepsilon_2 \geqslant 0$ be arbitrary. Denote by w_T the output of gradient descent run for $T = (4/3)\eta^{-1}\varepsilon_1^{-1}\gamma^{-2}[\ell^{-1}(\varepsilon_2)]^{-2}$ iterations after initialization at the origin. Then, with probability at least $1-\delta$,

$$\mathrm{err}^{0-1}(w_T) \leqslant \ell(0)^{-1}\big[(1 + LB_X\gamma^{-1}\ell^{-1}(\varepsilon_2))\mathsf{OPT}_{01} + \phi(\gamma) + O(\gamma^{-1}\ell^{-1}(\varepsilon_2)n^{-1/2}) + \varepsilon_1 + \varepsilon_2\big],$$

where $O(\cdot)$ hides absolute constants that depend on L, H, and $\log(1/\delta)$.

Proof. We take $v = V\bar{v}$ for a given unit-norm zero-one population risk minimizer $\bar{v}$ in Theorem 3.4.1 to get that for some universal constant $C > 0$ depending only on L and $\log(1/\delta)$, with probability at least $1 - \delta$,

$$F(w_T) \leqslant F(v) + \varepsilon_1/2 + CVB_X n^{-1/2}. \tag{3.5.4}$$

By Lemma 3.5.1, for any $\gamma > 0$ it holds that

$$F(v) \leqslant (1 + LVB_X)\mathsf{OPT}_{01} + \phi(\gamma) + \ell(V\gamma).$$

Let now $V = \gamma^{-1}\ell^{-1}(\varepsilon_2)$. Then $\ell(V\gamma) = \varepsilon_2$, and putting this together with (3.5.4), we get

$$F(w_T) \leqslant (1 + L\gamma^{-1})\mathsf{OPT}_{01} + \phi(\gamma) + O(\gamma^{-1}\ell^{-1}(\varepsilon_2)n^{-1/2}) + \varepsilon_1 + \varepsilon_2. \tag{3.5.5}$$

Finally, by Markov's inequality,

$$\mathbb{P}(yw_T^\top x < 0) \leqslant \frac{\mathbb{E}[\ell(yw_{T-1}^\top x)]}{\ell(0)} = \frac{F(w_T)}{\ell(0)}. \tag{3.5.6}$$

Putting (3.5.5) together with (3.5.6) completes the proof. □

A few comments on the proof of the above theorem are in order. Note that the only place we use smoothness of the loss function is in showing that gradient descent minimizes the population risk in (3.5.4), and it is not difficult to remove the H-smoothness assumption to accommodate e.g., the hinge loss. On the other hand, that ℓ is L-Lipschitz is key to the proof of Lemma 3.5.1. Non-Lipschitz losses such as the exponential loss or squared hinge loss would incur additional factors of γ^{-1} in front of OPT_{01} in the final bound for Theorem 3.5.2.[3] We shall see below in the proof of Proposition 3.5.5 that this would yield worse guarantees for $\mathrm{err}^{0-1}(w_T)$.

Additionally, in concordance with the result from Corollary 3.4.2, we see that if the tail of ℓ is fatter, then $\ell^{-1}(\varepsilon_2)$ will be larger and so our guarantees would be worse. In particular,

[3] This is because the first term in (3.5.1) would be bounded by $\mathsf{OPT}_{01} \cdot \sup_{|z| \leqslant VB_X} \ell(z)$. For Lipschitz losses this incurs a term of order $O(V)$ while (for example) the exponential loss would have a term of order $O(\exp(V))$, and our proof requires $V = \Omega(\gamma^{-1})$.

for losses with exponential tails, $\ell^{-1}(\varepsilon_2) = O(\log(1/\varepsilon_2))$, and so by using such losses we incur only additional logarithmic factors in $1/\varepsilon_2$. For this reason, we will restrict our attention in the below results to the logistic loss—which is convex, decreasing, 1-Lipschitz and $1/4$-smooth—although they apply equally to more general losses with different bounds that will depend on the tail behavior of the loss.

We now demonstrate how to convert the bounds given in Theorem 3.5.2 into bounds solely involving OPT_{01} by substituting the forms of the soft margin functions given in Section 3.3.

Corollary 3.5.3 (Hard margin distributions)**.** Suppose that $\|x\| \leqslant B_X$ a.s. and that a unit norm population risk minimizer $\bar{v}$ for the zero-one loss satisfies $|\bar{v}^\top x| \geqslant \bar{\gamma} > 0$ almost surely under $\mathcal{D}_x$ for some $\bar{\gamma} > 0$. For simplicity assume that $\ell(z) = \log(1+\exp(-z))$ is the logistic loss. Then for any $\varepsilon, \delta > 0$, with probability at least $1-\delta$, running gradient descent for $T = \tilde{O}(\eta^{-1}\varepsilon^{-1}\bar{\gamma}^{-2})$ is guaranteed to find a point w_T such that

$$\mathrm{err}^{0-1}(w_T) \leqslant \frac{1}{\log 2}\Big[\mathsf{OPT}_{01} + 2B_X\bar{\gamma}^{-1}\mathsf{OPT}_{01}\log(2/\mathsf{OPT}_{01})\Big] + \varepsilon,$$

provided $n = \tilde{\Omega}(\bar{\gamma}^{-2}B_X^2\log(1/\delta)\varepsilon^{-2})$.

Proof. Since $|\bar{v}^\top x| \geqslant \gamma^* > 0$, $\phi(\gamma^*) = 0$. Note that the logistic loss is $1/4$-smooth and satisfies $\ell^{-1}(\varepsilon) \in [\log(1/(2\varepsilon)), \log(2/\varepsilon)]$. By taking $\varepsilon_2 = \mathsf{OPT}_{01}$ the result follows by applying Theorem 3.5.2 with runtime $T = 4\eta^{-1}\ \varepsilon^{-1}\ \bar{\gamma}^{-2}\ \log^2(1/2\mathsf{OPT}_{01})$. □

Remark 3.5.4. The bound $\tilde{O}(\bar{\gamma}^{-1}\mathsf{OPT}_{01})$ in Corollary 3.5.3 is tight up to logarithmic factors[4] if one wishes to use gradient descent on a convex surrogate of the form $\ell(yw^\top x)$. [DGT19, Theorem 3.1] showed that for any convex and decreasing ℓ, there exists a distribution over the unit ball with margin $\bar{\gamma} > 0$ such that a population risk minimizer $w^* := \mathrm{argmin}_w \mathbb{E}[\ell(yw^\top x)]$ has zero-one population risk at least $\Omega(\bar{\gamma}^{-1}\kappa)$, where κ is the

[4] In fact, one can get rid of the logarithmic factors here and elsewhere in the chapter by using the hinge loss rather than the logistic loss. In this case one needs to modify Lemma 3.11.1 to accomodate non-smooth losses, which can be done with runtime $O(\varepsilon^{-2})$ rather than $O(\varepsilon^{-1})$ by e.g. [SB14, Lemma 14.1]. Then we use the fact that $\ell^{-1}(0) = 1$ for the hinge loss.

upper bound for the Massart noise probability. The Massart noise case is more restrictive than the agnostic setting and satisfies $\mathsf{OPT}_{01} \leqslant \kappa$. A similar matching lower bound was shown by [BLS12, Proposition 1].

In the below Proposition we demonstrate the utility of having *soft* margins. As we saw in the examples in Section 3.3, there any many distributions that satisfy $\phi(\gamma) = O(\gamma)$. We show below the types of bounds one can expect when $\phi(\gamma) = O(\gamma^p)$ for some $p > 0$.

Proposition 3.5.5 (Soft margin distributions)**.** Suppose $\|x\| \leqslant B_X$ a.s. and that the soft margin function for a population risk minimizer of the zero-one loss satisfies $\phi(\gamma) \leqslant C_0\gamma^p$ for some $p > 0$. For simplicity assume that ℓ is the logistic loss, and let $\eta \leqslant (2/5)B_X^{-2}$. Assuming $\mathsf{OPT}_{01} > 0$, then for any $\varepsilon, \delta > 0$, with probability at least $1 - \delta$, gradient descent run for $T = \tilde{O}(\eta^{-1}\varepsilon^{-1}\mathsf{OPT}_{01}^{-2/(1+p)})$ iterations with $n = \tilde{\Omega}(\mathsf{OPT}_{01}^{-2/(1+p)}\log(1/\delta)\varepsilon^{-2})$ samples satisfies

$$\mathrm{err}^{0-1}(w_T) \leqslant \tilde{O}\left((C_0 + B_X)\mathsf{OPT}_{01}^{\frac{p}{1+p}}\right) + \varepsilon,$$

Proof. By Theorem 3.5.2, we have $\mathrm{err}^{0-1}(w_T)$ is at most

$$\frac{1}{\log 2}\Big[\left(1 + LB_X\gamma^{-1}\ell^{-1}(\varepsilon_2)\right)\mathsf{OPT}_{01} + C_0\gamma^p + O(\gamma^{-1}B_X\ell^{-1}(\varepsilon_2)n^{-1/2}) + \varepsilon_1 + \varepsilon_2\Big].$$

For the logistic loss, $L = 1$ and $\ell^{-1}(\varepsilon) \in [\log(1/2\varepsilon), \log(2/\varepsilon)]$ and so we take $\varepsilon_2 = \mathsf{OPT}_{01}$. Choosing $\gamma^p = \gamma^{-1}\mathsf{OPT}_{01}$, we get $\gamma = \mathsf{OPT}_{01}^{1/(1+p)}$ and hence

$$\mathrm{err}^{0-1}(w_T) \leqslant 2\left(2 + B_X\mathsf{OPT}_{01}^{-\frac{1}{1+p}}\log(2/\mathsf{OPT}_{01})\right)\mathsf{OPT}_{01} + 2C_0\mathsf{OPT}_{01}^{\frac{1}{1+p}} + 2\varepsilon_1,$$

provided $n = \Omega(\mathsf{OPT}_{01}^{\frac{-2}{1+p}}\varepsilon_1^{-2}\log(1/\delta)\log^2(1/\mathsf{OPT}_{01}))$ and the number of iterations is $T = 4\eta^{-1}\varepsilon_1^{-1}\mathsf{OPT}_{01}^{-2/(1+p)}\log^2(1/2\mathsf{OPT}_{01})$. □

By applying Proposition 3.5.5 to Examples 3.3.4 and 3.3.6 we get the following approximate agnostic learning guarantees for the output of gradient descent for log-concave isotropic distributions and other distributions satisfying U-anti-concentration.

Corollary 3.5.6. Suppose that $\mathcal{D}_x$ satisfies U-anti-concentration and $\|x\| \leqslant B_X$ a.s. Then for any $\varepsilon, \delta > 0$, with probability at least $1 - \delta$, gradient descent on the logistic loss with step size $\eta \leqslant (2/5)B_X^{-2}$ and run for $T = \tilde{O}(\eta^{-1}\varepsilon^{-1}\mathsf{OPT}_{01}^{-1})$ iterations with $n = \tilde{\Omega}(\mathsf{OPT}_{01}^{-1}\log(1/\delta)\varepsilon^{-2})$ samples returns weights w_T satisfying $\mathrm{err}^{0-1}(w_T) \leqslant \tilde{O}(\mathsf{OPT}_{01}^{1/2}) + \varepsilon$, where $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$ hide universal constant depending on B_X, U, $\log(1/\delta)$ and $\log(1/\mathsf{OPT}_{01})$ only.

To conclude this section, we compare our result to the variant of the $\mathsf{Average}$ algorithm, which estimates the vector $w_{\mathsf{Avg}} = d^{-1}\mathbb{E}_{(x,y)}[xy]$. [KKM08] showed that when $\mathcal{D}_x$ is the uniform distribution over the unit sphere, w_{Avg} achieves risk $O(\mathsf{OPT}_{01}\sqrt{\log(1/\mathsf{OPT}_{01})})$. Estimation of w_{Avg} can be viewed as the output convex optimization procedure, since it is the minimum of the convex objective function $F_{\mathsf{Avg}}(w) = \mathbb{E}[(\langle w, x\rangle - y)^2]$.

Although $\ell(w) = (\langle w, x\rangle - y)^2$ is convex, it is not decreasing and thus is not covered by our analysis. On the other hand, this loss function is not typically used in practice for classification problems, and the aim of this work is to characterize the guarantees for the most typical loss functions used in practice, like the logistic loss. Finally, we wish to note that the approach of soft margins is not likely to yield good bounds for the classification error when $\mathcal{D}_x$ is the uniform distribution on the unit sphere. This is because the soft margin function behavior on this distribution has a necessary dimension dependence; we provide detailed calculations for this in Section 3.8.

3.5.2 Unbounded Distributions

We show in this section that we can achieve essentially the same results from Section 3.5.1 if we relax the assumption that $\mathcal{D}_x$ is bounded almost surely to being sub-exponential.

Definition 3.5.7 (Sub-exponential distributions)**.** We say $\mathcal{D}_x$ is C_m*-sub-exponential* if every $x \sim \mathcal{D}_x$ is a sub-exponential random vector with sub-exponential norm at most C_m. In particular, for any $\bar{v}$ with $\|\bar{v}\| = 1$, $\mathbb{P}_{\mathcal{D}_x}(|\bar{v}^\top x| \geqslant t) \leqslant \exp(-t/C_m)$.

We show in the following example that any log-concave isotropic distribution is C_m-sub-exponential for an absolute constant C_m independent of the dimension d.

Example 3.5.8. If $\mathcal{D}_x$ is log-concave isotropic, then $\mathcal{D}_x$ is $O(1)$-sub-exponential.

Proof. By Section 5.2.4 and Definition 5.22 of [Ver10], it suffices to show that for any unit norm $\bar{v}$, we have $(\mathbb{E}|\bar{v}^\top x|^p)^{1/p} \leqslant O(p)$. By [BZ17, Theorem 3], if we define a coordinate system in which $\bar{v}$ is an axis, then $\bar{v}^\top x$ is equal to the first marginal of $\mathcal{D}_x$ and is a one dimensional log-concave isotropic distribution. By [LV07, Theorem 5.22], this implies

$$(\mathbb{E}[|\bar{v}^\top x|^p])^{1/p} \leqslant 2p\mathbb{E}|\bar{v}^\top x| \leqslant 2p\sqrt{\mathbb{E}|\bar{v}^\top x|^2} \leqslant 2p.$$

In the second inequality we use Jensen's inequality and in the last inequality we have used that $\bar{v}^\top x = x_1$ is isotropic. □

As was the case for bounded distributions, the key to the proof for unbounded distributions comes from bounding the surrogate risk at a minimizer for the zero-one loss by some function of the zero-one loss.

Lemma 3.5.9. Suppose $\mathcal{D}_x$ is C_m-sub-exponential. Denote by $\bar{v}$ as a unit norm population risk minimizer for the zero-one loss, and let $v = V\bar{v}$ for $V > 0$ be a scaled version of $\bar{v}$. If ℓ is decreasing, L-Lipschitz and $\ell(0) \leqslant 1$, then

$$\mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(yv^\top x) \leqslant \inf_{\gamma>0}\Big\{\phi(\gamma) + \ell(V\gamma) + \big(1 + C_m + LVC_m\log(1/\mathsf{OPT}_{01})\big)\mathsf{OPT}_{01}\Big\}.$$

Proof. We again use the decomposition (3.5.1), with the only difference coming from the bound for the first term, which we show here. Fix $\xi > 0$ to be chosen later. We can write

$$\begin{aligned}
\mathbb{E}[\ell(yv^\top x)\mathbb{1}(y\bar{v}^\top x \leqslant 0)] &\leqslant \mathbb{E}[(1 + LV|\bar{v}^\top x|)\mathbb{1}(y\bar{v}^\top x < 0)] \\
&= \mathsf{OPT}_{01} + LV\mathbb{E}[|\bar{v}^\top x|\mathbb{1}(y\bar{v}^\top x \leqslant 0,\ |\bar{v}^\top x| \leqslant \xi)] \\
&\qquad + \mathbb{E}[|\bar{v}^\top x|\mathbb{1}(y\bar{v}^\top x \leqslant 0,\ |\bar{v}^\top x| > \xi)] \\
&\leqslant (1 + LV\xi)\mathsf{OPT}_{01} + \int_\xi^\infty \mathbb{P}(|\bar{v}^\top x| > t)\mathrm{d}t \\
&\leqslant (1 + LV\xi)\mathsf{OPT}_{01} + \int_\xi^\infty \exp(-t/C_m)\mathrm{d}t \\
&= (1 + LV\xi)\mathsf{OPT}_{01} + C_m \exp(-\xi/C_m).
\end{aligned}$$

The first inequality comes from Cauchy–Schwarz, the second from truncating, and the last from the definition of C_m-sub-exponential. Taking $\xi = C_m \log(1/\mathsf{OPT}_{01})$ results in

$$\mathbb{E}[\ell(yv^\top x)\mathbb{1}(y\bar{v}^\top x \leqslant 0)] \leqslant (1 + C_m + LVC_m \log(1/\mathsf{OPT}_{01}))\,\mathsf{OPT}_{01}.$$

□

To derive an analogue of Theorem 3.5.2 for unbounded distributions, we need to extend the analysis for the generalization bound for the output of gradient descent we presented in Theorem 3.4.1 to unbounded distributions. Rather than using (full-batch) vanilla gradient descent, we instead use online stochastic gradient descent. The reason for this is that dealing with unbounded distributions is significantly simpler with online SGD due to the ability to work with expectations rather than high-probability bounds. It is straightforward to extend our results to vanilla gradient descent at the expense of a more involved proof by using methods from e.g., [ZYW19].

Below we present our result for unbounded distributions. Its proof is similar to that of Theorem 3.5.2 and can be found in Section 3.9.

Theorem 3.5.10. Suppose $\mathcal{D}_x$ is C_m-sub-exponential, and let $\mathbb{E}[\|x\|^2] \leqslant B_X^2$. Let ℓ be convex, L-Lipschitz, and decreasing with $0 < \ell(0) \leqslant 1$. Let $\varepsilon_1, \gamma > 0$ and $\varepsilon_2 \geqslant 0$ be arbitrary, and fix a step size $\eta \leqslant L^{-2}B_X^{-2}\varepsilon_1/4$. By running online SGD for $T = 2\eta^{-1}\varepsilon_1^{-1}\gamma^{-2}[\ell^{-1}(\varepsilon_2)]^{-2}$ iterations after initialization at the origin, SGD finds a point such that in expectation over $(x_1, \ldots, x_T) \sim \mathcal{D}^T$,

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant 1/\ell(0)\Big[\phi(\gamma) + \varepsilon_1 + \varepsilon_2 + \big(1 + C_m + LC_m\ell^{-1}(\varepsilon_2)\gamma^{-1}\log(1/\mathsf{OPT}_{01})\big)\,\mathsf{OPT}_{01}\Big].$$

The above theorem yields the following bound for sub-exponential distributions satisfying U-anti-concentration. Recall from Examples 3.3.6 and 3.5.8 that log-concave isotropic distributions are $O(1)$-sub-exponential and satisfy anti-concentration with $U = 1$.

Corollary 3.5.11. Suppose $\mathcal{D}_x$ is C_m-sub-exponential with $\mathbb{E}[\|x\|^2] \leqslant B_X^2$ and assume U-anti-concentration holds. Let ℓ be the logistic loss and let $\varepsilon > 0$. Fix a step size $\eta \leqslant B_X^{-2}\varepsilon/16$. By running online SGD for $T = \tilde{O}(\eta^{-1}\varepsilon^{-1}C_mU^{-1}\mathsf{OPT}_{01}^{-1})$ iterations, there exists a point w_t, $t < T$, such that

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant \tilde{O}\left((C_m/U)^{1/2}\mathsf{OPT}_{01}^{1/2}\right) + \varepsilon.$$

Proof. By Example 3.3.4, $\phi(\gamma) \leqslant 2\gamma U$. Since $\ell^{-1}(\varepsilon) \in [\log(1/2\varepsilon), \log(2/\varepsilon)]$, we can take $\varepsilon_2 = \mathsf{OPT}_{01}$ in Theorem 3.5.10 to get

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant 1/\log(2)\Big[2\gamma U + \varepsilon + \big(2 + C_m + LC_m\gamma^{-1}\log^2(2/\mathsf{OPT}_{01})\big)\,\mathsf{OPT}_{01}\Big].$$

This bound is optimized when $U\gamma = C_m\gamma^{-1}\mathsf{OPT}_{01}$, i.e., $\gamma = U^{-1/2}C_m^{1/2}\mathsf{OPT}_{01}^{\frac{1}{2}}$. Substituting this value for γ we get the desired bound with $T = 2\log(2)\eta^{-1}\varepsilon^{-1}C_mU^{-1}\mathsf{OPT}_{01}^{-1}\log^2(1/2\mathsf{OPT}_{01})$.

□

Remark 3.5.12. [DKT20b, Theorem 1.4] recently showed that if the marginal of $\mathcal{D}$ over x is the standard Gaussian in d dimensions, for every convex, non-decreasing loss ℓ, the minimizer $v = \mathrm{argmin}_w F(w)$ satisfies $\mathrm{err}^{0-1}(v) = \Omega(\mathsf{OPT}_{01}\sqrt{\log(1/\mathsf{OPT}_{01})})$. Thus, there is a large gap between our upper bound of $\tilde{O}(\mathsf{OPT}^{1/2})$ and their corresponding lower bound. We think it is an interesting question if either the lower bound or the upper bound could be sharpened.

We also wish to note that [DKT20b] showed that by using gradient descent on a certain bounded and decreasing non-convex surrogate for the zero-one loss, it is possible to show that gradient descent finds a point with $\mathrm{err}^{0-1}(w_T) \leqslant O(\mathsf{OPT}_{01}) + \varepsilon$. In comparison with our result, this is perhaps not surprising: if one is able to show that gradient descent with a *bounded* and decreasing loss function can achieve population risk bounded by $O(\mathbb{E}[\ell(yv^\top x)])$ for arbitrary $v \in \mathbb{R}^d$, then the same proof technique that yields Theorem 3.5.10 from Lemma 3.5.9 would demonstrate that $\mathrm{err}^{0-1}(w_t) \leqslant O(\mathsf{OPT}_{01})$. Since the only globally bounded convex function is constant, this approach would require working with a non-convex loss.

3.6 Conclusion and Future Work

In this work we analyzed the problem of learning halfspaces in the presence of agnostic label noise. We showed that the simple approach of gradient descent on convex surrogates for the zero-one loss (such as the cross entropy or hinge losses) can yield approximate minimizers for the zero-one loss for both hard margin distributions and sub-exponential distributions satisfying an anti-concentration inequality enjoyed by log-concave isotropic distributions. Our results match (up to logarithmic factors) lower bounds shown for hard margin distributions. For future work, we are interested in exploring the utility of the soft margin for understanding other classification problems.

3.7 Fast Rates with Stochastic Gradient Descent

In Theorem 3.4.1, we showed that $F(w_T) \leqslant F(v) + O(1/\sqrt{n})$ given n samples from $\mathcal{D}$ by using vanilla (full-batch) gradient descent. In this section we demonstrate that by instead using stochastic gradient descent, one can achieve $F(w_T) \leqslant O(F(v)) + O(1/n)$ by appealing to a martingle Bernstein bound. We note that although the population risk guarantee degrades from $F(v)$ to $O(F(v))$, our bounds for the zero-one risk in vanilla gradient descent already have constant-factor errors and so the constant-factor error for $F(v)$ will not change the order of our final bounds.

The version of stochastic gradient descent that we study is the standard online SGD. Suppose we sample $z_t = (x_t, y_t) \overset{\text{i.i.d.}}{\sim} \mathcal{D}$ for $t = 1, \dots, T$, and let us denote the σ-algebra generated by the first t samples as $\mathcal{G}_t = \sigma(z_1, \dots, z_t)$. Define

$$\widehat{F}_t(w) := \ell(y_t w^\top x_t), \quad \mathbb{E}[\widehat{F}_t(w_t) | \mathcal{G}_{t-1}] = F(w_t) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \ell(y w_t^\top x).$$

The online stochastic gradient descent updates take the form

$$w_{t+1} := w_t - \eta \nabla \widehat{F}_t(w_t).$$

We are able to show an improved rate of $O(\varepsilon^{-1})$ when using online SGD.

Theorem 3.7.1 (Fast rate for online SGD)**.** Assume that $\ell(\cdot) \geqslant 0$ is convex, strictly decreasing, L-Lipschitz and H-smooth. Assume $\|x\| \leqslant B_X$ a.s. For simplicity assume that $w_0 = 0$. Let $v \in \mathbb{R}^d$ be arbitrary with $\|v\| \leqslant V$. Let $\eta \leqslant (32 H B_X^2)^{-1}$. Then for any $\varepsilon, \delta > 0$, by running online stochastic gradient descent for $T = O(\varepsilon^{-1} V^2 \log(1/\delta))$ iterations, with probability at least $1 - \delta$ there exists a point w_{t*}, with $t^* < T$, such that

$$\mathrm{err}^{0-1}(w_{t*}) \leqslant O(\mathbb{E}[\ell(y v^\top x)]) + \varepsilon,$$

where $O(\cdot)$ hides constant factors that depend on L, H and B_X only.

In this section we will sketch the proof for the above theorem. First, we note the following guarantee for the empirical risk. This result is a standard result in online convex optimization (see, e.g., Theorem 14.13 in [SB14]).

Lemma 3.7.2. Suppose that $\ell(\cdot) \geqslant 0$ is convex and H-smooth, and that $\|x\| \leqslant B_X$ a.s. Then for any $\alpha \in (0, 1)$, for fixed step size $\eta \leqslant \alpha/(8 H B_X^2)$, and for any $T \geqslant 1$, it holds that

$$\frac{1}{T} \sum_{t=0}^{T-1} \widehat{F}_t(w_t) \leqslant (1 + \alpha) \frac{1}{T} \sum_{t=0}^{T-1} \widehat{F}_t(v) + \frac{\|w_0 - v\|^2}{\eta T}.$$

From here, one could take expectations and show that in expectation over the randomness of SGD, the population risk found by gradient descent is at most $(1+\alpha)F(v) + O(1/T)$, but we are interested in developing a generalization bound that has the same fast rate but holds with high probability, which requires significantly more work. Much of the literature for fast rates in stochastic optimization require additional structure to achieve such results: [BJM06] showed that the empirical risk minimizer converges at a fast rate to its expectation under a low-noise assumption; [SSS09] achieved fast rates for the output of stochastic optimization by using explicit regularization by a strongly convex regularizer; [SST10] shows that projected online SGD achieves fast rates when $\min_v \mathbb{E}[\ell(yv^\top x)] = 0$. By contrast, we show below that the standard online SGD algorithm achieves a constant-factor approximation to the best population risk at a fast rate. We do so by appealing to the martingale Bernstein inequality provided in Lemma 2.9.7.

We would like to take $Y_t = \sum_{\tau<t}[F(w_t) - \widehat{F}_t(w_t)]$, which has martingale difference sequence $D_t = F(w_t) - \widehat{F}_t(w_t)$. The difficulty here is showing that $D_t \leqslant R$ almost surely for some absolute constant R. The obvious fix would be to show that the weights w_t stay within a bounded region throughout gradient descent via early stopping. In the case of full-batch gradient descent, this is indeed possible: in Lemma 3.11.1 we showed that $\|w_t - v\| \leqslant \|w_0 - v\|$ throughout gradient descent, which would imply that $\ell(yw_t^\top x)$ is uniformly bounded for all samples x throughout G.D., in which case $D_t \leqslant F(w_t)$ would hold almost surely. But for online stochastic gradient descent, since we must continue to take draws from the distribution in order to reduce the optimization error, there isn't a straightforward way to get a bound on $\|w_t\|$ to hold almost surely throughout the gradient descent trajectory.

Our way around this is to realize that in the end, our end goal is to show something of the form

$$\mathrm{err}^{0-1}(w_t) \leqslant O(\mathbb{E}[\ell(yv^\top x)]) + O(1/T),$$

since then we could use a decomposition similar to Lemma 3.5.1 to bound the right hand side by terms involving OPT_{01} and a soft margin function. Since for a non-negative H-smooth loss $[\ell'(z)]^2 \leqslant 4H\ell(z)$ holds, it actually suffices to show that the losses $\{[\ell'(y_t w^\top x_t)]^2\}_1^T$ concentrate around their expectation at a fast rate. Roughly, this is because one would have

$$\begin{aligned}\min_{t<T} \mathbb{E}_{\mathcal{D}}\left([\ell'(yw_t^\top x)]^2\right) &\leqslant \frac{1}{T}\sum_{t=0}^{T-1}[\ell'(y_t w_t^\top x_t)^2] + O(1/T)\\ &\leqslant \frac{4H}{T}\sum_{t=0}^{T-1}\ell(y_t w_t^\top x_t) + O(1/T)\\ &\leqslant \frac{4H}{T}\sum_{t=0}^{T-1}\ell(y_t v^\top x_t) + O(1/T). \qquad (3.7.1)\end{aligned}$$

To finish the proof we can then use the fact that v is a fixed vector of constant norm to show that the empirical risk on the last line of (3.7.1) concentrates around $O(\mathbb{E}[\ell(yv^\top x)])$ at rate $O(1/T)$. For decreasing and convex loss functions, $\ell'(z)^2$ is decreasing so the above provides a bound for $\mathrm{err}^{0-1}(w_t)$ by Markov's inequality.

This shows that the key to the proof is to show that $\{\ell'(y_t w_t x_t)^2\}$ concentrates at rate $O(1/T)$. The reason this is easier than showing concentration of $\{\ell(y_t w_t x_t)\}$ is because for Lipschitz losses, $\ell'(y_t w_t^\top x_t)^2$ is uniformly bounded regardless of the norm of w_t. This ensures that the almost sure condition needed for the martingale difference sequence in Lemma 2.9.7 holds trivially. We note that a similar technique has been utilized before for the analysis of SGD [JT20b, CG20, FCG19], although in these settings the authors used the concentration of $\{\ell'(z_t)\}$ rather than $\{\ell'(z_t)^2\}$ since they considered the logistic loss, for which $|\ell'(z)| \leqslant \ell(z)$. Since not all smooth loss functions satisfy this inequality, we instead use concentration of $\{\ell'(z_t)^2\}$.

Below we formalize the above proof sketch. We first show that $\{\ell'(y_t w_t^\top x_t)^2\}$ concentrates at rate $O(1/T)$ for any fixed sequence of gradient descent iterates $\{w_t\}$.

Lemma 3.7.3. Let ℓ be any differentiable L-Lipschitz function, and let $z_t = (x_t, y_t) \overset{\text{i.i.d.}}{\sim} \mathcal{D}$. Denote $\mathcal{G}_t = \sigma(z_1, \ldots, z_t)$ the σ-algebra generated by the first t draws from $\mathcal{D}$, and let $\{w_t\}$ be any sequence of random variables such that w_t is $\mathcal{G}_{t-1}$-measurable for each t. Then for any $\delta > 0$, with probability at least $1 - \delta$,

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{(x,y)\sim\mathcal{D}}\left(\left[\ell'(yw_t^\top x)\right]^2\right) \leqslant \frac{4}{T}\sum_{t=0}^{T-1}\left[\ell'(y_t w_t^\top x_t)\right]^2 + \frac{4L^2\log(1/\delta)}{T}. \tag{3.7.2}$$

Proof. For simplicity, let us denote

$$J(w) := \mathbb{E}_{(x,y)\sim\mathcal{D}}\left(\left[\ell'(yw^\top x)\right]^2\right), \quad \widehat{J}_t(w) := \left[\ell'(y_t w^\top x_t)\right]^2.$$

We begin by showing the second inequality in (3.7.2). Define the random variable

$$Y_t := \sum_{\tau<t}(J(w_\tau) - \widehat{J}_\tau(w_\tau)) \tag{3.7.3}$$

Then Y_t is a martingale with respect to the filtration $\mathcal{G}_{t-1}$ with martingale difference sequence $D_t := J(w_t) - \widehat{J}_t(w_t)$. We need bounds on D_t and on $\mathbb{E}[D_t^2|\mathcal{G}_{t-1}]$ in order to apply Lemma 2.9.7. Since ℓ is L-Lipschitz,

$$D_t \leqslant J(w_t) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\left([-\ell'(yv^\top x)]^2\right) \leqslant L^2.$$

Similarly,

$$\begin{aligned}\mathbb{E}[\widehat{J}_t(w_t)^2|\mathcal{G}_{t-1}] &= \mathbb{E}\left(\left[\ell'(y_t w_t^\top x_t)\right]^4|\mathcal{G}_{t-1}\right)\\ &\leqslant L^2\mathbb{E}\left(\left[\ell'(y_t w_t^\top x_t)\right]^2|\mathcal{G}_{t-1}\right)\\ &= L^2 J(w_t).\end{aligned} \tag{3.7.4}$$

In the inequality we use that ℓ is L-Lipschitz, so that $|\ell'(\alpha)| \leqslant L$. We then can use (3.7.4) to bound the squared increments,

$$\begin{aligned}\mathbb{E}[D_t^2|\mathcal{G}_{t-1}] &= J(w_t)^2 - 2J(w_t)\mathbb{E}[\widehat{J}_t(w_t)|\mathcal{G}_{t-1}] + \mathbb{E}[\widehat{J}_t(w_t)^2|\mathcal{G}_{t-1}]\\ &\leqslant \mathbb{E}[\widehat{J}_t(w_t)^2|\mathcal{G}_{t-1}]\\ &\leqslant L^2 J(w_t).\end{aligned}$$

This allows for us to bound

$$U_{T-1} = \sum_{t=0}^{T-1}\mathbb{E}[D_t^2|\mathcal{G}_{t-1}] \leqslant L^2\sum_{t=0}^{T-1} J(w_t).$$

Lemma 2.9.7 thus implies that with probability at least $1 - \delta$, we have

$$\sum_{t=0}^{T-1} (J(w_t) - \widehat{J}_t(w_t)) \leqslant L^2 \log(1/\delta) + (\exp(1) - 2) \sum_{t=0}^{T-1} J(w_t).$$

Using that $(1 - \exp(1) + 2)^{-1} \leqslant 4$, we divide each side by T and get

$$\frac{1}{T} \sum_{0=t}^{T-1} J(w_t) \leqslant \frac{4}{T} \sum_{t=0}^{T-1} \widehat{J}_t(w_t) + \frac{4L^2 \log(1/\delta)}{T}.$$

This completes the proof. □

Next, we show that the average of $\{\ell(y_t v^\top x_t)\}$ is at most twice its mean at rate $O(1/T)$.

Lemma 3.7.4. Let ℓ be any L-Lipschitz function, and suppose that $\ell(0) \leqslant 1$ and $\|x\|_2 \leqslant B$ a.s. Let $v \in \mathbb{R}^d$ be arbitrary with $\|v\| \leqslant V$. For any $\delta > 0$, with probability at least $1 - \delta$,

$$\frac{1}{T} \sum_{t=0}^{T-1} \widehat{F}_t(v) \leqslant 2F(v) + \frac{2(1 + LVB_X)\log(1/\delta)}{T}.$$

Proof. Let $\mathcal{G}_t = \sigma(z_1, \ldots, z_t)$ be the σ-algebra generated by the first t draws from $\mathcal{D}$. Then the random variable $Y_t := \sum_{\tau<t}(\widehat{F}_\tau(v) - F(v))$ is a martingale with respect to the filtration $\mathcal{G}_{t-1}$ with martingale difference sequence $D_t := \widehat{F}_t(v) - F(v)$. We need bounds on D_t and on $\mathbb{E}[D_t^2|\mathcal{G}_{t-1}]$ in order to apply Lemma 2.9.7. Since ℓ is L-Lipschitz and $\|x\| \leqslant B_X$ a.s., that $\|v\| \leqslant V$ implies that almost surely,

$$D_t \leqslant \widehat{F}_t(v) = \ell(y_t v^\top x_t) \leqslant (1 + LVB_X). \tag{3.7.5}$$

Similarly,

$$\begin{aligned} \mathbb{E}[\widehat{F}_t(v)^2|\mathcal{G}_{t-1}] &= \mathbb{E}\left[\ell(y_t v^\top x_t)^2|\mathcal{G}_{t-1}\right] \\ &\leqslant (1 + LVB_X)\mathbb{E}[\ell(y_t v^\top x_t)] \\ &= (1 + LVB_X)F(v). \end{aligned} \tag{3.7.6}$$

In the inequality, we have used that (x_t, y_t) is independent from $\mathcal{G}_{t-1}$ together with (3.7.5). We then can use (3.7.6) to bound the squared increments,

$$\begin{aligned}\mathbb{E}[D_t^2|\mathcal{G}_{t-1}] &= F(v)^2 - 2F(v)\mathbb{E}[\widehat{F}_t(v)|\mathcal{G}_{t-1}] + \mathbb{E}[\widehat{F}_t(v)^2|\mathcal{G}_{t-1}] \\ &\leqslant \mathbb{E}[\widehat{F}_t(v)^2|\mathcal{G}_{t-1}] \\ &\leqslant (1 + LVB_X)F(v).\end{aligned}$$

This allows for us to bound

$$U_{T-1} := \sum_{t=0}^{T-1} \mathbb{E}[D_t^2|\mathcal{G}_{t-1}] \leqslant (1 + LVB_X)TF(v).$$

Lemma 2.9.7 thus implies that with probability at least $1 - \delta$, we have

$$\sum_{t=0}^{T-1}(\widehat{F}_t(v) - F(v)) \leqslant (1 + LVB_X)\log(1/\delta) + (\exp(1) - 2)TF(v).$$

Using that $\exp(1) - 2 \leqslant 1$, we divide each side by T and get

$$\frac{1}{T}\sum_{t=0}^{T-1}\widehat{F}_t(v) \leqslant 2F(v) + \frac{2(1 + LVB_X)\log(1/\delta)}{T}.$$

□

Finally, we put these ingredients together for the proof of Theorem 3.7.1.

Proof. Since ℓ is convex and H-smooth, we can take $\alpha = 1/4$ in Lemma 3.7.2 to get

$$\frac{1}{T}\sum_{t=0}^{T-1}\widehat{F}_t(w_t) \leqslant \frac{5}{4T}\sum_{t=0}^{T-1}\widehat{F}_t(v) + \frac{V^2}{\eta T}. \tag{3.7.7}$$

We can therefore bound

$$\begin{aligned}\min_{t<T} \mathbb{E}\left([\ell'(yw_t^\top x)]^2\right) &\leqslant \frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}_{(x,y)\sim\mathcal{D}}\left([\ell'(yw_t^\top x)]^2\right)\\ &\leqslant \frac{4}{T}\sum_{t=0}^{T-1}[\ell'(y_t w_t^\top x_t)]^2 + \frac{4L^2\log(2/\delta)}{T}\\ &\leqslant \frac{16H}{T}\sum_{t=0}^{T-1}\widehat{F}_t(w_t) + \frac{4L^2\log(2/\delta)}{T}\\ &\leqslant \frac{20H}{T}\sum_{t=0}^{T-1}\widehat{F}_t(v) + \frac{5L^2\log(2/\delta)+V^2}{\eta T}\\ &\leqslant 40HF(v) + \frac{40H(1+LVB_X)\eta\log(2/\delta)+5L^2\eta\log(2/\delta)+V^2}{\eta T}.\end{aligned} \tag{3.7.8}$$

The second inequality holds since ℓ is L-Lipschitz so that we can apply Lemma 3.7.3. The third inequality uses that ℓ is non-negative and H-smooth, so that $[\ell'(z)]^2 \leqslant 4H\ell(z)$ (see [SST10, Lemma 2.1]). The fourth inequality uses (3.7.7), and the final inequality uses Lemma 3.7.4.

Since ℓ is convex and decreasing, $\frac{\mathrm{d}}{\mathrm{d}z}\ell'(z)^2 = 2\ell'(z)\ell''(z) \leqslant 0$, so $\ell'(z)^2$ is decreasing. By Markov's inequality, this implies

$$\mathbb{P}(yw_t^\top x < 0) = \mathbb{P}\left([\ell'(yw_t^\top x)]^2 \geqslant \ell'(0)^2\right) \leqslant [\ell'(0)]^{-2}\mathbb{E}\left([\ell'(yw_t^\top x)]^2\right).$$

Substituting this into (3.7.8), this implies that with probability at least $1-\delta$,

$$\mathrm{err}^{0-1}(w_t) \leqslant O(F(v)) + O(V^2\log(1/\delta)/T).$$

□

We note that the above proof works for an arbitrary initialization w_0 such that $\|w_0\|$ is bounded by an absolute constant with high probability, e.g. with the random initialization $w_0 \overset{\text{i.i.d.}}{\sim} N(0, I_d/d)$. The only difference is that we need to replace V^2 with $\|w_0 - v\|^2 \leqslant O(V^2)$ in (3.7.7) and the subsequent lines.

3.8 Soft Margin for Uniform Distribution

We show here that the soft margin function for the uniform distribution on the sphere has an unavoidable dimension dependence. Consider $x \sim \mathcal{D}$ is uniform on the sphere in d dimensions. Then x has the same distribution as $z/\|z\|$, where $z \sim N(0, I_d)$ is the d-dimensional Gaussian. The soft margin function on x thus satisfies, for $\|v\| = 1$,

$$\phi(\gamma) = \mathbb{P}_x(|v^\top x| \leqslant \gamma) = \mathbb{P}_z\left(|v^\top z|^2/\|z\|^2 \leqslant \gamma^2\right).$$

By symmetry, we can rotate the coordinate system so that $v = (1, 0, \dots)$, which results in $\phi(\gamma)$ taking the form

$$\begin{aligned}\mathbb{P}\left(\frac{z_1^2}{\sum_{i=1}^d z_i^2} \leqslant \gamma^2\right) &= \mathbb{P}\left((1-\gamma^2)z_1^2 \leqslant \gamma^2 \textstyle\sum_{i=2}^d z_i^2\right)\\ &= \mathbb{P}\left(z_1^2 \leqslant \frac{\gamma^2}{1-\gamma^2}\textstyle\sum_{i=2}^d z_i^2\right)\\ &\geqslant \mathbb{P}(z_1^2 \leqslant \gamma^2 \textstyle\sum_{i=2}^d z_i^2).\end{aligned}$$

Since $\gamma^2 \sum_{i=2}^d z_i^2 = \Theta(\gamma^2 d)$ with high probability by concentration of the χ^2 distribution, and since $\mathbb{P}(|z_1| \leqslant a) = \Theta(a)$ for the Gaussian, this shows that $\phi(\gamma) = \Omega(\gamma\sqrt{d})$ when $\mathcal{D}_x$ is uniform on the sphere. Thus our approach of using the soft margin in Theorem 3.5.2 to derive generalization bounds will result in multiplicative terms attached to OPT that will grow with d for such a distribution.

3.9 Proofs for Unbounded Distributions

In this section we prove Theorem 3.5.10.

3.9.1 Empirical Risk

First, we derive an analogue of Lemma 3.11.1 that holds for any distribution satisfying $\mathbb{E}[\|x\|^2] \leqslant B_X^2$ by appealing to online stochastic gradient descent. Note that any distribution over $\mathbb{R}^d$ with sub-Gaussian coordinates satisfies $\mathbb{E}[\|x\|^2] \leqslant B^2$ for some $B \in \mathbb{R}$.

We use the same notation from Section 3.7, where we assume samples $z_t = (x_t, y_t) \overset{\text{i.i.d.}}{\sim} \mathcal{D}$ for $t = 1, \ldots, T$, and $\mathcal{G}_t := \sigma(z_1, \ldots, z_t)$, and denote

$$\widehat{F}_t(w) := \ell(y_t w^\top x_t), \quad \mathbb{E}[\widehat{F}_t(w_t)|\mathcal{G}_{t-1}] = F(w_t) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(y w_t^\top x).$$

The online stochastic gradient descent updates take the form

$$w_{t+1} := w_t - \eta \nabla \widehat{F}_t(w_t).$$

Lemma 3.9.1. Suppose $\mathbb{E}_{\mathcal{D}_x}[\|x\|^2] \leqslant B_X^2$. Suppose that ℓ is convex and L-Lipschitz. Let $v \in \mathbb{R}^d$ and $\varepsilon, \alpha \in (0,1)$ be arbitrary, and consider any initialization $w_0 \in \mathbb{R}^d$. Provided $\eta \leqslant L^{-2} B_X^{-2} \varepsilon / 2$, then for any $T \in \mathbb{N}$,

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} F(w_t) \leqslant F(v) + \frac{\|w_0 - v\|^2}{\eta T} + \varepsilon.$$

Proof. The proof is very similar to that of the proof of Lemma 3.7.2 described in Section 3.9.1, so we describe here the main modifications. The key difference comes from the gradient upper bound: for $g_t = \ell'(y_t w_t^\top x_t)$, instead of getting an upper bound that holds a.s. in terms of the loss, we only show that its expectation is bounded by a constant:

$$\mathbb{E}[\|g_t\|^2 |\mathcal{G}_{t-1}] \leqslant \mathbb{E}[\ell'(y_t w_t x_t)^2 \|x_t\|^2 |\mathcal{G}_{t-1}] \leqslant L^2 \mathbb{E}[\|x_t\|^2 |\mathcal{G}_{t-1}] \leqslant L^2 B_X^2.$$

By convexity, $\langle g_t, w_t - v\rangle \geqslant \widehat{F}_t(w_t) - \widehat{F}_t(v)$. Thus taking $\eta = O(\varepsilon)$, we get

$$\begin{aligned} \|w_t - v\|^2 - \mathbb{E}[\|w_{t+1} - v\|^2 |\mathcal{G}_{t-1}] &\geqslant \mathbb{E}[2\eta(\widehat{F}_t(w_t) - \widehat{F}_t(v)) - \eta^2 \|g_t\|^2 |\mathcal{G}_{t-1}] \\ &\geqslant 2\eta(F(w_t) - F(v)) - \eta^2 L^2 B_X^2 \\ &\geqslant 2\eta(F(w_t) - F(v) - \varepsilon). \end{aligned}$$

Taking expectations with respect to the randomness of SGD and summing from 0 to $T-1$, we get

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} F(w_t) \leqslant F(v) + \frac{\|w_0 - v\|^2}{\eta T} + \varepsilon.$$

□

We note that the above analysis is quite loose and we are aware of a number of ways to achieve faster rates by introducing various assumptions on ℓ and $\mathcal{D}_x$; we chose the presentation above for simplicity.

With the above result in hand, we can prove Theorem 3.5.10.

Proof. Let $\varepsilon_1 > 0$. By taking $\eta \leqslant L^{-2}B_X^{-2}\varepsilon_1/8$ and $T = 2V^2\eta^{-1}\varepsilon_1^{-1}$, Lemma 3.9.1 and Markov's inequality, this implies that there exists some $t < T$ such that

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant \mathbb{E}[F(w_t)] \leqslant {}^1\!/\!_{\ell(0)}\Big[F(v) + \frac{V^2}{\eta T} + \varepsilon_1/2 \leqslant F(v) + \varepsilon_1\Big].$$

By Lemma 3.5.9, this implies that for any $\gamma > 0$,

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant {}^1\!/\!_{\ell(0)}\Big[\,(1 + C_m + LVC_m\log({}^1\!/\!_{\mathsf{OPT}_{01}}))\,\mathsf{OPT}_{01} + \phi(\gamma) + \ell(V\gamma) + \varepsilon_1\Big].$$

For $\varepsilon_2 \geqslant 0$, by taking $V = \gamma^{-1}\ell^{-1}(\varepsilon_2)$, this means that for any $\gamma > 0$, we have

$$\mathbb{E}[\mathrm{err}^{0-1}(w_t)] \leqslant {}^1\!/\!_{\ell(0)}\Big[\,\big(1 + C_m + LC_m\ell^{-1}(\varepsilon_2)\gamma^{-1}\log({}^1\!/\!_{\mathsf{OPT}_{01}})\big)\,\mathsf{OPT}_{01} + \phi(\gamma) + \varepsilon_1 + \varepsilon_2\Big].$$

For $V = \gamma^{-1}\ell^{-1}(\varepsilon_2)$, we need $T = 2\gamma^{-2}\eta^{-1}\varepsilon_1^{-1}[\ell^{-1}(\varepsilon_2)]^2$.

□

3.10 Loss Functions and Sample Complexity for Separable Data

We present here the proof of Corollary 3.4.2.

Proof. Let $v = V\bar{v}$. By Theorem 3.4.1, for any $\varepsilon, \delta > 0$ and $V > 0$, running gradient descent for $T = 4[\ell(0)]^{-1}\eta^{-1}V^2\varepsilon^{-1}$ iterations guarantees that $w = w_{T-1}$ satisfies

$$F(w) \leqslant F(v) + \ell(0)\cdot\varepsilon/3 + CVn^{-1/2},$$

for some absolute constant $C > 0$ depending only on L, B_X, and $\log(1/\delta)$. By Markov's inequality, this implies

$$\mathbb{P}(yw^\top x < 0) \leqslant \frac{1}{\ell(0)}F(w) \leqslant \frac{1}{\ell(0)}\left(F(v) + \frac{\ell(0)}{3}\varepsilon + CVn^{-1/2}\right). \tag{3.10.1}$$

Since $y\bar{v}^\top x \geqslant \gamma$ a.s., we have

$$F(v) = \mathbb{E}_{(x,y)\sim\mathcal{D}}\ell(yV\bar{v}^\top x) \leqslant \ell(V\gamma).$$

If ℓ has polynomial tails, then by taking $V \geqslant \gamma^{-1}(6C_0[\ell(0)]^{-1}\varepsilon^{-1})^{1/p}$ we get $F(v) \leqslant C_0(\gamma V)^{-p}$ which is at most $\frac{\ell(0)\varepsilon}{6}$. Substituting this into (3.10.1), this implies

$$\mathbb{P}(yw^\top x < 0) \leqslant \frac{\varepsilon}{2} + \frac{CV}{\ell(0)n^{1/2}}. \tag{3.10.2}$$

Thus, provided $n = \Omega(\gamma^{-2}\varepsilon^{-2-\frac{2}{p}})$, if we run gradient descent for $T = \tilde{\Omega}(\gamma^{-2}\varepsilon^{-1-\frac{2}{p}})$ iterations, we have that $\mathrm{err}^{0-1}(w) \leqslant \varepsilon$.

If ℓ has exponential tails, then by taking $V \geqslant \gamma^{-1}[C_1^{-1}\log(6C_0\ell(0)\varepsilon^{-1})]^{1/p}$ we get $F(v) \leqslant \frac{\ell(0)\varepsilon}{6}$, and so (3.10.2) holds in this case as well. This shows that for exponential tails, taking $n = \tilde{\Omega}(\gamma^{-2}\varepsilon^{-2})$ and $T = \tilde{\Omega}(\gamma^{-2}\varepsilon^{-1})$ suffices to achieve $\mathrm{err}^{0-1}(w) \leqslant \varepsilon$. □

3.11 Remaining Proofs

In this section we provide the proof of Theorem 3.4.1. We first will prove the following bound on the empirical risk.

Lemma 3.11.1. Suppose that ℓ is convex and H-smooth. Assume $\|x\| \leqslant B_X$ a.s. Fix a step size $\eta \leqslant (2/5)H^{-1}B_X^{-2}$, and let $v \in \mathbb{R}^d$ be arbitrary. Then for any initialization w_0, and for any $\varepsilon > 0$, running gradient descent for $T = (4/3)\varepsilon^{-1}\eta^{-1}\|w_0 - v\|^2$ ensures that for all $t < T$, $\|w_t - v\| \leqslant \|w_0 - v\|$, and

$$\widehat{F}(w_{T-1}) \leqslant \frac{1}{T}\sum_{t=0}^{T-1}\widehat{F}(w_t) \leqslant \widehat{F}(v) + \varepsilon.$$

To prove this, we first introduce the following upper bound for the norm of the gradient.

Lemma 3.11.2 ([Sha20], Proof of Lemma 3)**.** Suppose that ℓ is H-smooth. Then for any $\rho \in (0, 1)$, provided $\eta \leqslant 2\rho H^{-1}B_X^{-2}$, $\widehat{F}(w_t)$ is decreasing in t. Moreover, if $T \in \mathbb{N}$ is arbitrary and $u \in \mathbb{R}^d$ is such that $\widehat{F}(u) \leqslant \widehat{F}(w_T)$, then for any $t < T$, we have the following gradient upper bound,

$$\left\|\nabla \widehat{F}(w_t)\right\|^2 \leqslant \frac{1}{\eta(1-\rho)}\left(\widehat{F}(w_t) - \widehat{F}(u)\right). \tag{3.11.1}$$

With this gradient upper bound, we can prove Lemma 3.11.1.

Proof. Let $\varepsilon > 0$ be fixed and let $T = (4/3)\varepsilon^{-1}\eta^{-1}\left\|w_0 - v\right\|^2$ be as in the statement of the lemma. We are done if $\widehat{F}(w_T) < \widehat{F}(v)$, so let us assume that $\widehat{F}(v) \leqslant \widehat{F}(w_T)$. We proceed by providing the appropriate lower bounds for

$$\left\|w_t - v\right\|^2 - \left\|w_{t+1} - v\right\|^2 = 2\eta\left\langle \widehat{F}(w_t), w_t - v\right\rangle - \eta^2\left\|\widehat{F}(w_t)\right\|^2.$$

For any $v \in \mathbb{R}^d$, by convexity of ℓ,

$$\begin{aligned}\left\langle \nabla\widehat{F}(w), w - v\right\rangle &= \frac{1}{n}\sum_{i=1}^n \ell'(y_i w^\top x_i)(y_i w^\top x_i - y_i v^\top x_i)\\ &\geqslant \frac{1}{n}\sum_{i=1}^n [\ell(y_i w^\top x_i) - \ell(y_i v^\top x_i)]\\ &= \widehat{F}(w) - \widehat{F}(v),\end{aligned} \tag{3.11.2}$$

by convexity of ℓ. On the other hand, since $\widehat{F}(v) \leqslant \widehat{F}(w_T)$, by Lemma 3.11.2, for any $t < T$, (3.11.1) holds, i.e.

$$\left\|\nabla \widehat{F}(w_t)\right\|^2 \leqslant \frac{1}{\eta(1-\rho)}\left(\widehat{F}(w_t) - \widehat{F}(v)\right). \tag{3.11.3}$$

Thus, for $\eta \leqslant (2/5)H^{-1}B_X^{-2}$, putting eqs. (3.11.2) and (3.11.3) together yields

$$\begin{aligned}\left\|w_t - v\right\|^2 - \left\|w_{t+1} - v\right\|^2 &= 2\eta\left\langle \nabla\widehat{F}(w_t), w_t - v\right\rangle - \eta^2\left\|\nabla\widehat{F}(w_t)\right\|^2\\ &\geqslant 2\eta(\widehat{F}(w_t) - \widehat{F}(v)) - \eta^2 \cdot \frac{1}{\eta(1-1/5)}\left(\widehat{F}(w_t) - \widehat{F}(v)\right)\\ &= \frac{3}{4}\eta\left(\widehat{F}(w_t) - \widehat{F}(v)\right).\end{aligned}$$

Summing and teloscoping over $t < T$,

$$\frac{1}{T}\sum_{t=0}^{T-1}\widehat{F}(w_t) \leqslant \widehat{F}(v) + \frac{(4/3)\left\|w_0 - v\right\|^2}{\eta T} \leqslant \widehat{F}(v) + \varepsilon.$$

By Lemma 3.11.2, $\widehat{F}(w_t)$ is decreasing in t, and therefore

$$\widehat{F}(w_{T-1}) = \min_{t<T} \widehat{F}(w_t) \leqslant T^{-1} \sum_{t<T} \widehat{F}(w_t),$$

completing the proof. □

Lemma 3.11.1 shows that throughout the trajectory of gradient descent, $\|w_t\|$ stays bounded by the norm of the reference vector v. We can thus use Rademacher complexity bounds to prove Theorem 3.4.1.

Proof. By Lemma 3.11.1, it suffices to show that the gap between the empirical and population surrogate risk is small. To do so, we use a Rademacher complexity argument. Denote by $\mathcal{G}$ the function class

$$\mathcal{G}_V := \{x \mapsto w^\top x : \|w\| \leqslant 3V\}.$$

Since ℓ is L-Lipschitz and $\ell(0) \leqslant 1$, it holds that $\ell(yw^\top x) \leqslant 1 + 3LV \leqslant 4LV$. We therefore use standard results in Rademacher complexity (e.g. Theorem 26.12 of [SB14]) to get that with probability at least $1-\delta$, for any $w \in \mathcal{G}_V$,

$$F(w) \leqslant \widehat{F}(w) + \frac{2B_X V L}{\sqrt{n}} + 4B_X V\sqrt{\frac{2\log(2/\delta)}{n}}.$$

Since the output of gradient descent satisfies $\|w_{T-1} - v\| \leqslant \|w_0 - v\| \leqslant 2V$, we see that $w_{T-1} \in \mathcal{G}_V$. We can thus apply the Rademacher complexity bound to both $w_{T-1} \in \mathcal{G}_V$ and $v \in \mathcal{G}_V$, proving the theorem. □

CHAPTER 4

Learning noisy halfspaces using one-hidden-layer neural networks trained by stochastic gradient descent

4.1 Introduction

In this chapter we show that SGD-trained neural networks are capable of learning halfspaces with agnostic label noise, despite the capacity of such networks to overfit to random labels. For a distribution $\mathcal{D}$ over features $(x, y) \in \mathbb{R}^d \times \{\pm 1\}$, let us define

$$\mathsf{OPT}_{\mathsf{lin}} := \min_{v \in \mathbb{R}^d,\ \|v\|=1} \mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\left(\langle v, x\rangle\right)\Big) \tag{4.1.1}$$

as the optimal classification error achieved by a halfspace $\langle v, \cdot\rangle$. We prove that for a broad class of distributions, SGD-trained one-hidden-layer neural networks achieve classification error at most $\tilde{O}(\sqrt{\mathsf{OPT}_{\mathsf{lin}}})$ in polynomial time. Equivalently, one-hidden-layer neural networks can learn halfspaces up to risk $\tilde{O}(\sqrt{\mathsf{OPT}_{\mathsf{lin}}})$ in the distribution-specific agnostic PAC learning setting. Our result holds for neural networks with leaky-ReLU activations trained on the cross-entropy loss and, importantly, hold for any initialization, and for networks of arbitrary width.

By comparing the generalization of the neural network with that of the *best* linear classifier over the distribution, we can make two different but equally important claims about the training of overparameterized neural networks. The first view is that SGD produces neural networks with classification error that is competitive with that of the best linear classifier over the distribution, and that this behavior can occur for neural networks of any width and any initialization. In this view, our work provides theoretical support for the hypothesis put forward by [NKK19] that the performance of SGD-trained networks in the early epochs of training can be explained by that of a linear classifier.

The second view is that of the problem of learning halfspaces in the presence of adversarial label noise. (Note that adversarial *label noise* is distinct from the notions of adversarial examples or adversarial training [GSS14, MMS18], where the features x are perturbed rather than the labels y.) In this setting, one views the (clean) data as initially coming from a linearly separable distribution but for which each sample $(x, y) \sim \mathcal{D}$ has its label flipped $y \mapsto -y$ with some sample-dependent probability $\eta(x) \in [0, 1]$. Then the best error achieved by a halfspace is $\mathbb{E}_{x\sim\mathcal{D}_x}[\eta(x)] = \mathsf{OPT}_{\mathsf{lin}}$. Viewed from this perspective, our result shows that despite the clear capacity of an overparameterized neural network to overfit to corrupted labels, when trained by SGD, such networks can still generalize (albeit achieving the suboptimal risk $\sqrt{\mathsf{OPT}_{\mathsf{lin}}}$). We note that the optimization algorithm we consider is vanilla online SGD without any explicit regularization methods such as weight decay or dropout. This suggests that the ability of neural networks to generalize in the presence of noise is not solely due to explicit regularization, but that some forms of *implicit* regularization induced by gradient-based optimization play an important role.

4.2 Related Work

We discuss here a number of works related to the questions of optimization and generalization in deep learning. An approach that has attracted significant attention recently is the neural tangent kernel (NTK) approximation [JGH18]. This approximation relies upon the fact that for a specific initialization scheme, extremely wide neural networks are well-approximated by the behavior of the neural network at initialization, which in the infinite width limit produces a kernel (the NTK) [DZP19, DLL18, ALS19, ZCZ19, CG20, ADH19a, ADH19b, CG19a, FCG19, ZG19, JT20b, CCZ21]. Using an assumption on separability of the training data, it is commonly shown that SGD-trained neural networks in the NTK regime can perfectly fit any training data. Under certain conditions, one can also derive generalization bounds for the performance of SGD-trained networks for distributions that can be perfectly classified by functions related to the NTK.

Although significant insights into the training dynamics of SGD-trained networks have come from this approach, it is known that neural networks deployed in practice can traverse far enough from their initialization such that the NTK approximation no longer holds [FDP20]. A line of work known as the mean field approximation allows for ultra-wide networks to be far from their initialization by connecting the trajectory of the weights of the neural network to the solution of an associated partial differential equation [MMM19, COB19, CCG20]. A separate line of work has sought to demonstrate that the concept classes that can be learned by neural networks trained by gradient descent are a strict superset of those that can be learned by the NTK [AL19, WLL19, LWM19, WGL20, LMZ20].

More relevant to our work is understanding the generalization of neural network classifiers when the data distribution has some form of label noise. Works that explicitly derive generalization bounds for SGD-trained neural networks in the presence of label noise are scarce. Even for the simple concept class of halfspaces $x \mapsto \operatorname{sgn}(\langle v, x\rangle)$, there are often tremendous difficulties in determining whether or not *any* algorithm can efficiently learn in the presence of noise. For this reason let us take a small detour to detail some of the difficulties in learning halfspaces in the presence of noise, to emphasize the difficulty of learning more complicated function classes in the presence of noise.

The most general (and most difficult) noise class is that of adversarial label noise, which is equivalent to the agnostic PAC learning framework [KSS94]. In this setting, one makes no assumption on the relationship between the features and the labels, and so continuing with the notation from (4.1.1), the optimal risk $\mathsf{OPT}_{\mathsf{lin}}$ achieved by a halfspace is strictly positive in general. It is known that learning up to classification error $O(\mathsf{OPT}_{\mathsf{lin}}) + \varepsilon$ cannot be done in $\operatorname{poly}(d, \varepsilon^{-1})$ time without assumptions on the marginal distribution of $\mathcal{D}$ [Dan16]. For this reason it is common to assume some type of structure on the noise or the distribution to get tractable guarantees.

One relaxation of the noise condition is known as the Massart noise [MN06] where one assumes that each sample has its label flipped with some instance-dependent probability $\eta(x) \leqslant \eta < 1/2$. Under this noise model, it was recently shown that there are efficient algorithms that can learn up to risk $\eta + \varepsilon$ [DGT19]. A more simple noise setting is that of random classification noise (RCN) [AL88], where the labels of each sample are flipped with probability η. Polynomial time algorithms for learning under this model were first shown by [BFK98]. Previous theoretical works on the ability of neural network classifiers to generalize in the presence of label noise were restricted to the RCN setting [HLY20] or Massart noise setting [LSO19]. In this paper, we consider the most general setting of adversarial label noise.

In terms of distribution-specific learning guarantees in the presence of noise, polynomial time algorithms for learning halfspaces under Massart noise for the uniform distribution on the sphere were first shown by [ABH15], and for log-concave isotropic distributions by [ABH16]. [ABL17] constructed a localization-based algorithm that efficiently learns halfspaces up to risk $O(\mathsf{OPT}_{\mathsf{lin}})$ when the marginal is log-concave isotropic. For more background on learning halfspaces in the presence of noise, we refer the reader to [BH21].

Returning to the neural network literature, in light of the above it should not be surprising that computational tractability issues arise even for the case of neural networks consisting of a single neuron. [GKK19] showed that learning a single ReLU neuron up to the best-possible risk $\mathsf{OPT}_{\text{ReLU}}$ (under the squared loss) is computationally intractable, even when the marginal is a standard Gaussian. By contrast, [FCG20] showed that gradient descent on the empirical risk can learn single ReLUs up to risk $O(\sqrt{\mathsf{OPT}_{\text{ReLU}}})$ efficiently for many distributions. Two recent works have shown that even in the realizable setting—i.e., when the labels are generated by a neural network without noise—it is computationally hard to learn one-hidden-layer neural networks with (non-stochastic) gradient descent when the marginal distribution is Gaussian [GGJ20, DKK20].

In terms of results that show neural networks can generalize in the presence of noise, [LSO19] considered clustered distributions with real-valued labels (using the squared loss) and analyzed the performance of GD-trained one-hidden-layer neural networks when a fraction of the labels are switched. They derived guarantees for the empirical risk but did not derive a generalization bound for the resulting classifier. [HLY20] analyzed the performance of regularized neural networks in the NTK regime when trained on data with labels corrupted by RCN, and argued that regularization was helpful for generalization. By contrast, our work shows that neural networks can generalize for linearly separable distributions corrupted by adversarial label noise without any explicit regularization, suggesting that certain forms of implicit regularization in the choice of the algorithm plays an important role. We note that a number of researchers have sought to understand the implicit bias of gradient descent [SHN18, JT19, LL20, JT20a, MGW20, LXX20]. Such works assume that the distribution is linearly separable by a large margin, and characterize the solutions found by gradient descent (or gradient flow) in terms of the maximum margin solution.

Finally, we note some recent works that connected the training dynamics of SGD-trained neural networks with linear models. [BGM18] showed that SGD-trained one-hidden-layer leaky ReLU networks can generalize on linearly separable data. [Sha18] compared the performance of residual networks with those of linear predictors in the regression setting. They showed that there exist weights for residual networks with generalization performance competitive with linear predictors, and they proved that SGD is able to find those weights when there is a residual connection from the input layer to the output layer. [NKK19] provided experimental evidence for the hypothesis that much of the performance of SGD-trained neural networks in the early epochs of training can be explained by linear classifiers. [HXA20] provided theoretical evidence for this hypothesis by showing that overparameterized neural networks with the NTK initialization and scaling have similar dynamics to a linear predictor defined in terms of the network's NTK. [STR20] showed that neural networks are biased towards simple classifiers even when more complex classifiers are capable of improving generalization.

4.3 Problem Description and Results

In this section we study the problem we consider and our main results.

4.3.1 Notation

For a vector v, we denote $\|v\|$ as its Euclidean norm. For a matrix W, we use $\|W\|_F$ to denote its Frobenius norm. We use the standard $O(\cdot)$ and $\Omega(\cdot)$ notations to ignore universal constants when describing growth rates of functions. The notation $\tilde{O}(\cdot)$ and $\tilde{\Omega}(\cdot)$ further ignores logarithmic factors. We use $a \vee b$ to denote the maximum of $a, b \in \mathbb{R}$, and $a \wedge b$ their minimum. The notation $\mathbb{1}(E)$ denotes the indicator function of the set E, which is one on the set and zero outside of it.

4.3.2 Problem Setup

Consider a distribution $\mathcal{D}$ over $(x, y) \in \mathbb{R}^d \times \{\pm 1\}$ with marginal distribution $\mathcal{D}_x$ over x. Let $m \in \mathbb{N}$, and consider a one-hidden-layer leaky ReLU network with m neurons,

$$f_x(W) := \sum_{j=1}^{m} a_j \sigma(\langle w_j, x \rangle), \tag{4.3.1}$$

where $\sigma(z) = \max(\alpha z, z)$ is the leaky-ReLU activation with $\alpha \in (0, 1]$. Assume that $a_j \overset{\text{i.i.d.}}{\sim} \mathrm{Unif}(\pm a)$ for some $a > 0$ and that the $\{a_j\}$ are randomly initialized and not updated throughout training, as is commonly assumed in theoretical analyses of SGD-trained neural networks [DZP19, ADH19b, JT20b].[1] We are interested in the classification error for the neural network,

$$\mathrm{err}(W) := \mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \mathrm{sgn}\left(f_x(W)\right)\Big),$$

where $\mathrm{sgn}(z) = 1$ if $z > 0$, $\mathrm{sgn}(0) = 0$, and $\mathrm{sgn}(z) = -1$ otherwise. We will seek to minimize $\mathrm{err}(W)$ by minimizing,

$$L(W) := \mathbb{E}_{(x,y)\sim\mathcal{D}} \ell(y f_x(W)),$$

[1] The specific choice of the initialization of the second layer is immaterial; our analysis holds for any second-layer weights that are fixed at a random initialization. The only difference that may arise is in the sample complexity: if with high probability $\|a\| = \Theta(1)$ then the sample complexity requirement will be the same within constant factors, while for initializations satisfying $\|a\| = \omega(1)$ or $\|a\| = o(1)$ our upper bound for the sample complexity will become worse as the network becomes larger.

where ℓ is a convex loss function. We will use the fact that for any convex, twice differentiable and decreasing function ℓ, the function $-\ell'$ is non-negative and decreasing, and thus $-\ell'$ can also serve as a loss function. In particular, by Markov's inequality, these properties allow us to bound the classification error by the population risk under $-\ell'$:

$$\begin{aligned}\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\big(f_x(W)\big)\Big) &= \mathbb{P}\Big(y \cdot f_x(W) \leqslant 0\Big)\\ &= \mathbb{P}\Big(-\ell'\big(yf_x(W)\big) \geqslant 0\Big)\\ &\leqslant \frac{\mathbb{E}_{(x,y)\sim\mathcal{D}} - \ell'\big(yf_x(W)\big)}{-\ell'(0)} \end{aligned} \tag{4.3.2}$$

Thus, provided $-\ell'(0) > 0$, upper bounds for the population risk under $-\ell'$ yield guarantees for the classification error. This property has previously been used to derive generalization bounds for deep neural networks trained by gradient descent [CG20, FCG19, JT20b, CCZ21]. To this end, we make the following assumptions on the loss throughout this paper.

Assumption 4.3.1. The loss $\ell(\cdot) : \mathbb{R} \to \mathbb{R}$ is convex, twice differentiable, decreasing, 1-Lipschitz, and satisfies $-\ell'(0) > 0$. Moreover, for $z \geqslant 1$, ℓ satisfies $-\ell'(z) \leqslant 1/z$.

The assumption that $-\ell'(z) \leqslant 1/z$ for $z \geqslant 1$ is to ensure that the surrogate loss $-\ell'$ is not too large on samples that are classified correctly. Note that the standard loss used for training neural networks in binary classification tasks—the binary cross-entropy loss $\ell(z) = \log(1 + \exp(-z))$—satisfies all of the conditions in Assumption 4.3.1. We denote the population risk under the surrogate loss $-\ell'$ as follows,

$$\mathcal{E}(W) := \mathbb{E}_{(x,y)\sim\mathcal{D}} - \ell'(yf_x(W)).$$

We seek to minimize the population risk by minimizing the empirical risk induced by a set of i.i.d. examples $\{(x_t, y_t)\}_{t\geqslant 1}$ using the online stochastic gradient descent algorithm. Denote $f_t(W) = f_{x_t}(W)$ as the neural network output for sample x_t, and denote the loss under ℓ and $-\ell'$ for sample x_t by

$$\widehat{L}_t(W) := \ell(y_t f_t(W)), \quad \widehat{\mathcal{E}}_t(W) := -\ell'(y_t f_t(W)). \tag{4.3.3}$$

The updates of online stochastic gradient descent are given by

$$W^{(t+1)} := W^{(t)} - \eta \nabla \widehat{L}_t(W^{(t)}) = W^{(t)} + \eta \widehat{\mathcal{E}}_t(W^{(t)}) y_t \nabla f_t(W^{(t)}). \tag{4.3.4}$$

Before proceeding with our main theorem we will introduce some of the definitions and assumptions which will be used in our analysis. The first is that of sub-exponential distributions.

Definition 4.3.2 (Sub-exponential distributions)**.** We say $\mathcal{D}_x$ is C_m*-sub-exponential* if every $x \sim \mathcal{D}_x$ is a sub-exponential random vector with sub-exponential norm at most C_m. In particular, for any $\bar{v} \in \mathbb{R}^d$ with $\|\bar{v}\| = 1$, $\mathbb{P}_{\mathcal{D}_x}(|\bar{v}^\top x| \geqslant t) \leqslant \exp(-t/C_m)$.

We note that every sub-Gaussian distribution is sub-exponential. The next property we shall use is that of a *soft margin*, which we introduced in Definition 3.3.1. The soft margin can be seen as a probabilistic analogue of the standard hard margin, where we relax the typical requirement for a margin-based condition from holding almost surely to holding with some controlled probability. As written above, the soft margin condition can hold for a specific vector $\bar{v} \in \mathbb{R}^d$, and our final generalization bound below will only care about the soft margin function for a halfspace $\bar{v}$ that achieves population risk $\mathsf{OPT}_{\mathsf{lin}}$. However, for many distributions, one can show that *all* unit norm vectors $\bar{v}$ satisfy a soft margin of the form $\phi_{\bar{v}}(\gamma) = O(\gamma)$. One important class of such distributions are those satisfying a type of anti-concentration property.

Definition 4.3.3 (Anti-concentration)**.** For $\bar{v} \in \mathbb{R}^d$, denote by $p_{\bar{v}}(\cdot)$ the marginal distribution of $x \sim \mathcal{D}_x$ on the subspace spanned by $\bar{v}$. We say $\mathcal{D}_x$ satisfies U*-anti-concentration* if there is some $U > 0$ such that for all unit norm $\bar{v}$, $p_{\bar{v}}(z) \leqslant U$ for all $z \in \mathbb{R}$.

Anti-concentration is a typical assumption used for deriving distribution-specific agnostic PAC learning guarantees [KLS09, DKT20a, DKT20b, FCG21] as it allows for one to ignore pathological distributions where arbitrarily large probability mass can be concentrated in tiny regions of the domain. We previously used this property to derive guarantees for the single neuron classifier in Chapter 3. Below, we collect some examples of soft margin function behavior for different distributions, including those satisfying the above anti-concentration property. We shall see in Theorem 4.3.5 that the behavior of $\phi(\gamma)$ for $\gamma \ll 1$ will be the determining factor in our generalization bound, and thus in the below examples one only needs to pay attention to the behavior of $\phi(\gamma)$ for γ sufficiently small.

Example 4.3.4. 1. If $|\bar{v}^\top x| > \gamma^*$ a.s., then $\phi_{\bar{v}}(\gamma) = 0$ for $\gamma < \gamma^*$.

2. If $\mathcal{D}_x$ satisfies U-anti-concentration, then for any $\bar{v}$ with $\|\bar{v}\| = 1$, $\phi_{\bar{v}}(\gamma) \leqslant 2U\gamma$ holds.

3. If $\mathcal{D}_x$ is isotropic and log-concave (i.e. its probability density function is log-concave), then $\mathcal{D}_x$ satisfies 1-anti-concentration and hence $\phi_{\bar{v}}(\gamma) \leqslant 2\gamma$ for all $\bar{v}$.

The proofs for the properties described in Example 4.3.4 can be found in [FCG21, Section 3].

4.3.3 Main Results

With the above in place, we can provide our main result.

Theorem 4.3.5. Assume $\mathcal{D}_x$ is C_m-subexponential and there exists $B_X > 0$ such that $\mathbb{E}[\|x\|^2] \leqslant B_X^2 < \infty$. Denote $\mathsf{OPT}_{\mathsf{lin}} := \min_{\|w\|=1} \mathbb{P}_{(x,y)\sim\mathcal{D}}(y\langle w, x\rangle < 0)$ as the best classification error achieved by a unit norm halfspace v^*. Let $m \in \mathbb{N}$ be arbitrary, and consider a leaky-ReLU network of the form (4.3.1) where $a = 1/\sqrt{m}$. Let $W^{(0)}$ be an arbitrary initialization and denote $G_0 := \|W^{(0)}\|_F$. Let the step size satisfy $\eta \leqslant B_X^{-2}$. Then for any $\gamma > 0$, by running online SGD for $T = O(\eta^{-1}\gamma^{-2}[\phi_{v^*}(\gamma) + \mathsf{OPT}_{\mathsf{lin}}]^{-2}[1 \vee G_0])$ iterations, there exists a point $t^* < T$ such that in expectation over $(x_1, \ldots, x_T) \sim \mathcal{D}^T$,

$$\begin{aligned}
&\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\left(f_x(W^{(t^*)})\right)\Big) \\
&\quad\leqslant 2|\ell'(0)|^{-1}\alpha^{-1}\bigg[\Big(1 + \gamma^{-1}C_m + \gamma^{-1}C_m\log(1/\mathsf{OPT}_{\mathsf{lin}})\Big)\mathsf{OPT}_{\mathsf{lin}} + \phi_{v^*}(\gamma)\bigg].
\end{aligned}$$

To concretize the generalization bound in Theorem 4.3.5 we need to analyze the properties of the soft margin function ϕ_{v^*} at the best halfspace and then optimize over the choice of γ. But before doing so, let us make a few remarks on Theorem 4.3.5 that hold in general. The sample complexity (number of SGD iterations) T, and the resulting generalization bound, are independent of the number of neurons m, showing that the neural network can generalize despite the capacity to overfit.[2] If $\|x\| \leqslant B_X$ a.s. for some absolute constant B_X, then the sample complexity is dimension-independent, while if $\mathcal{D}_x$ is isotropic, $\mathbb{E}[\|x\|^2] = d$ and so the sample complexity is linear in d. Finally, we note that large learning rates and arbitrary initializations are allowed.

In the remainder of the section, we will discuss the implications of Theorem 4.3.5 for common distributions. The first distribution we consider is a hard margin distribution.

Corollary 4.3.6 (Hard margin distributions)**.** Suppose there exists some $v^* \in \mathbb{R}^d$, $\|v^*\| = 1$, and $\gamma_0 > 0$ such that $\mathbb{P}\big(y \neq \mathrm{sgn}(\langle v^*, x\rangle)\big) = \mathsf{OPT}_{\mathsf{lin}}$ and $|\langle v^*, x\rangle| \geqslant \gamma_0 > 0$ almost surely over $\mathcal{D}_x$. Assume for simplicity that ℓ is the binary cross-entropy loss, $\ell(z) = \log(1 + \exp(-z))$. Then under the settings of Theorem 4.3.5, there exists some $t^* < T = O(\eta^{-1}\gamma_0^{-2}\mathsf{OPT}_{\mathsf{lin}}^{-2}[1 \vee G_0])$ such that in expectation over $(x_1, \ldots, x_T) \sim \mathcal{D}^T$,

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \mathrm{sgn}\left(f_x(W^{(t^*)})\right)\Big) \leqslant \tilde{O}(\gamma_0^{-1}\mathsf{OPT}_{\mathsf{lin}}).$$

Proof. Since $|\langle v^*, x\rangle| \geqslant \gamma_0 > 0$, the soft margin at v^* satisfies $\phi_{v^*}(\gamma_0) = 0$. Since $-\ell'(0) = 1/2$, by Theorem 4.3.5,

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \mathrm{sgn}\left(f_x(W^{(t^*)})\right)\Big) \leqslant 4\alpha^{-1}\big(1 + \gamma_0^{-1}C_m + \gamma_0^{-1}C_m\log(1/\mathsf{OPT}_{\mathsf{lin}})\big)\mathsf{OPT}_{\mathsf{lin}}.$$

□

[2] [BGM18, Theorem 7] showed that if there are T samples and $m = \Omega(T/d)$, then for any set of labels $(y_1, \ldots, y_T) \in \{\pm 1\}^T$ and for almost every $(x_1, \ldots, x_T) \sim \mathcal{D}_x^T$, there exist hidden layer weights W^* and outer layer weights $\vec{a} \in \mathbb{R}^m$ such that $f_t(W^*) = y_t$ for all $t \in [T]$. In contrast, Theorem 4.3.5 shows that when m is sufficiently large there exist neural networks that can fit random labels of the data but SGD training avoids these networks.

The above result shows that if the data comes from a linearly separable data distribution with margin γ_0 but is then corrupted by adversarial label noise, then SGD-trained networks will still find weights that can generalize with classification error at most $\tilde{O}(\gamma_0^{-1}\mathsf{OPT}_{\mathsf{lin}})$. In the next corollary we show that for distributions satisfying U-anti-concentration we get a generalization bound of the form $\tilde{O}(\sqrt{\mathsf{OPT}_{\mathsf{lin}}})$.

Corollary 4.3.7 (Distributions satisfying anti-concentration)**.** Assume $\mathcal{D}_x$ satisfies U anti concentration. Assume for simplicity that ℓ is the binary cross-entropy loss, $\ell(z) = \log(1 + \exp(-z))$. Then under the settings of Theorem 4.3.5, with a number of iterations satisfying $T = O(\eta^{-1}\mathsf{OPT}_{\mathsf{lin}}^{-3}[1 \vee G_0])$ there exists $t^* < T$ such that in expectation over $(x_1, \ldots, x_T) \sim \mathcal{D}^T$,

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\left(f_x(W^{(t^*)})\right)\Big) \leqslant \tilde{O}(\sqrt{\mathsf{OPT}_{\mathsf{lin}}}).$$

Proof. By Example 4.3.4, $\phi_{v^*}(\gamma) \leqslant 2U\gamma$. Substituting this into Theorem 4.3.5 and using that $-\ell'(0) = 1/2$, we get

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\left(f_x(W^{(t^*)})\right)\Big) \leqslant 4\alpha^{-1}\big[2U\gamma + 3\gamma^{-1}C_m\mathsf{OPT}\log(1/\mathsf{OPT})\big].$$

This bound is optimized when $\gamma = \mathsf{OPT}^{1/2}$, and results in a bound for the classification error that is at most $O(\mathsf{OPT}^{1/2}\log(1/\mathsf{OPT}))$. □

The above corollary covers, for instance, log-concave isotropic distributions like the Gaussian or the uniform distribution over a convex set by Example 4.3.4.

Taken together, Corollaries 4.3.6 and 4.3.7 demonstrate that despite the capacity for overparameterized neural networks to overfit to the data, SGD-trained neural networks are fairly robust to adversarial label noise. We emphasize that our results hold for SGD-trained neural networks of arbitrary width and following an arbitrary initialization, and that the resulting generalization and sample complexity do not depend on the number of neurons m. In particular, the above phenomenon cannot be explained by the neural tangent kernel approximation, which is highly dependent on assumptions about the initialization, learning rate, and number of neurons.

4.3.4 Comparisons with Related Work

We now discuss how our result relates to others appearing in the literature. First, [BGM18] showed that by running multiple-pass SGD on the hinge loss one can learn linearly separable data. They assume a noiseless ($\mathsf{OPT}_{\mathsf{lin}} = 0$) model over a norm-bounded domain and assume a hard margin distribution, so that $y\langle v^*, x\rangle > \gamma_0$ for some $\gamma_0 > 0$. In the noiseless setting, Corollaries 4.3.6 and 4.3.7 generalize their result to include unbounded, linearly separable (marginal) distributions without a hard margin like log-concave isotropic distributions. More significantly, our results hold in the adversarial label noise setting (a.k.a., agnostic PAC learning). This allows for us to compare the generalization of an SGD-trained neural network with that of the *best* linear classifier over the distribution, and make a much more general claim about the dynamics of SGD-trained neural networks.

[HXA20] showed that for sufficiently wide neural networks with the NTK initialization scheme, and under the assumption that the components of the input distribution are independent, the dynamics in the early stages of SGD-training are closely related to that of a linear predictor defined in terms of the NTK of the neural network. By contrast, our result holds for any initialization and neural networks of any width and covers a larger class of distributions. Their result was for the squared loss, while ours holds for the standard losses used for classification problems. Our results can be understood as a claim about the 'early training dynamics' of SGD, since we show that there exists *some* iterate of SGD that performs almost as well as the best linear classifier over the distribution, and we provide an upper bound on the number of iterations required to reach this point. One might expect that under more stringent assumptions (on, say, the initialization, learning rate schedule, and/or network architecture), stronger guarantees for the classification error could hold in the later stages of training; we will revisit this question with experimental results in Section 4.5.

[LSO19] considered a handcrafted distribution consisting of noisy clusters and showed that sufficiently wide one-hidden-layer neural networks trained by GD on the squared loss with the NTK initialization have favorable properties in the early training dynamics. A direct comparison of our results is difficult as they do not provide a guarantee for the generalization error of the resulting neural network. But at a high level, their analysis focused on a noise model akin to Massart noise (a more restrictive setting than the agnostic noise considered in this paper), and they made a number of assumptions—a particular (large) initialization, sufficiently wide network, and the use of the squared loss for classification—that were not used in this work. The results of [LSO19] covered general, smooth activation functions (but not leaky-ReLU).

[HLY20] showed that ultra-wide networks with NTK scaling and initialization trained by SGD with various forms of regularization can generalize when the labels are corrupted with random classification noise. Their generalization bound was given in terms of the classification error on the 'clean' data distribution (without any noise) and allowed for general activation functions (including leaky-ReLU). In comparison, we assume that the training data and the test data come from the same distribution, and our generalization bound is given in terms of the performance of the best linear classifier over the distribution. Our generalization guarantee holds without any explicit forms of regularization, suggesting that the mechanism responsible for the lack of overfitting is not explicit regularization, but forms of regularization that are *implicit* to the SGD algorithm.

4.4 Proof of the Main Results

We will show that stochastic gradient descent achieves small classification error by using a proof technique similar to that of [BGM18], who showed the convergence and generalization of gradient descent on the hinge loss for one-hidden-layer leaky ReLU networks on linearly separable data.[3] Their proof relies upon the fact that both the classification error and the hinge loss for the best halfspace are zero. In our setting—without the assumption of linear separability, and with more general loss functions—their strategy for showing that the empirical risk can be driven to zero will not work. (We remind the reader that our goal is to show that the neural network will generalize when it is of *arbitrary* width, and when significant noise is present, and thus we cannot guarantee the smallest empirical or population loss is arbitrarily close to zero.) Instead, we need to compare the performance of the neural network with that of the best linear classifier over the data, which will in general have error (both classification and loss value) bounded away from zero. To do so, we use some of the ideas used in [FCG21] to derive generalization bounds for the classification error when the surrogate loss is bounded away from zero.

To begin, let us introduce some notation. Let $v^* \in \mathbb{R}^d$ be a unit norm halfspace that minimizes the halfspace error, so that

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\Big(y \neq \operatorname{sgn}\big(\langle v^*, x\rangle\big)\Big) = \mathsf{OPT}_{\mathsf{lin}}.$$

Denote the matrix $V \in \mathbb{R}^{m\times d}$ as having rows $v_j^\top \in \mathbb{R}^d$ defined by

$$v_j = \frac{1}{\sqrt{m}} \operatorname{sgn}(a_j) v^*. \tag{4.4.1}$$

The scaling of each row of the matrix V ensures that $\|V\|_F = 1$. For $\gamma > 0$, denote

$$\widehat{\xi}_t(\gamma) := \mathbb{1}(y_t\langle v^*, x_t\rangle \in [0,\gamma)) + (1 + \gamma^{-1}|\langle v^*, x_t\rangle|)\mathbb{1}(y_t\langle v^*, x_t\rangle < 0).$$

The expected value of the above quantity will be an important quantity in our proof. To give some idea of how this quantity will fit in to our analysis, assume for the moment that $\|x\| \leqslant 1$ a.s. Then taking expectations of the above and using Cauchy–Schwarz, we get

$$\mathbb{E}\widehat{\xi}_t(\gamma) \leqslant \phi_{v^*}(\gamma) + (1+\gamma^{-1})\mathbb{E}[|\langle v^*, x_t\rangle|\mathbb{1}(y_t\langle v^*, x_t\rangle < 0)] \leqslant \phi_{v^*}(\gamma) + (1+\gamma^{-1})\mathsf{OPT}_{\mathsf{lin}}. \tag{4.4.2}$$

[3] This proof technique can be viewed as an extension of the Perceptron proof presented in [SB14, Theorem 9.1].

The above appears (in a more general form) in the bound for the classification error presented in Theorem 4.3.5. In particular, the goal below will be to show that the classification error can be bounded by a constant multiple of $\mathbb{E}[\widehat{\xi}_t(\gamma)]$.

Continuing, let us denote

$$\widehat{H}_t := \langle W^{(t)}, V\rangle, \quad \widehat{G}_t^2 = \left\| W^{(t)} \right\|_F^2. \tag{4.4.3}$$

The quantity $\widehat{H}_t$ measures the correlation between the weights found by SGD and those of the best linear classifier over the distribution. We define the population-level versions of each of the random variables above by replacing the $\widehat{\cdot}$ with their expectation $\mathbb{E}_{\mathrm{sgd}}(\cdot)$ over the randomness of the draws $(x_1, \dots, x_t)$ of the distribution used for SGD. That is,

$$\begin{aligned}
L_t &:= \mathbb{E}_{\mathrm{sgd}} \widehat{L}_t(W^{(t)}),\\
\mathcal{E}_t &:= \mathbb{E}_{\mathrm{sgd}} \widehat{\mathcal{E}}_t(W^{(t)}),\\
H_t &:= \mathbb{E}_{\mathrm{sgd}} \widehat{H}_t,\\
G_t^2 &:= \mathbb{E}_{\mathrm{sgd}}[\widehat{G}_t^2],\\
\xi(\gamma) &:= \mathbb{E}_{(x_t, y_t)\sim\mathcal{D}} \widehat{\xi}_t(\gamma).
\end{aligned} \tag{4.4.4}$$

Our proof strategy will be to show that until gradient descent finds weights with small risk, the correlation H_T between the weights found by SGD and those of the best linear predictor will grow at least as fast as $\Omega(T)$, while G_T always grows at a rate of at most $O(\sqrt{T})$. Since $\|V\|_F = 1$, by Cauchy–Schwarz we have the bound $H_T \leqslant G_T$, and so the growth rates $H_T = \Omega(T)$ and $G_T = O(\sqrt{T})$ can only be satisfied for a small number of iterations. In particular, there can only be a small number of iterations until SGD finds weights with small risk.

To see how we might be able to show that the correlation H_T is increasing, note that we have the identity

$$\widehat{H}_{t+1} - \widehat{H}_t = -\eta \langle \nabla \widehat{L}_t(W^{(t)}), V\rangle = -\eta \ell'(y_t f_t(W^{(t)})) y_t \langle \nabla f_t(W^{(t)}), V\rangle.$$

Since $-\ell' \geqslant 0$, the inequality $\widehat{H}_{t+1} > \widehat{H}_t$ holds if we can show $y_t\langle \nabla f_t(W^{(t)}), V\rangle > 0$, i.e. if we can show that the gradient of the neural network is correlated with the weights of the best linear predictor. For this reason, the following technical lemma is a key ingredient in our proof.

Lemma 4.4.1. For V defined in (4.4.1), for any $(x_t, y_t) \in \mathbb{R}^d \times \{\pm 1\}$, for any $W \in \mathbb{R}^{m\times d}$, and any $\gamma \in (0,1)$,

$$y_t\langle \nabla f_t(W), V\rangle \geqslant a\gamma\sqrt{m}\Big[\alpha - \widehat{\xi}_t(\gamma)\Big]. \tag{4.4.5}$$

The proof of the above lemma is in Section 4.7. As alluded to above, with this technical lemma we can show that until the surrogate risk is as small as a constant factor of $\xi(\gamma)$, the correlation of the weights found by SGD and those of the best linear predictor is increasing.

Lemma 4.4.2. For any $t \in \mathbb{N} \cup \{0\}$, for any $\gamma > 0$, it holds that

$$H_{t+1} \geqslant H_t + \eta a\gamma\sqrt{m}\big[\alpha\mathcal{E}_t - \xi(\gamma)\big].$$

Proof. Since $\widehat{H}_{t+1} = \langle W^{(t+1)}, V\rangle = \langle W^{(t)}, V\rangle - \eta\langle \widehat{L}_t(W^{(t)}), V\rangle$, we can write

$$\begin{aligned}
\widehat{H}_{t+1} &= \widehat{H}_t - \eta\langle \nabla\widehat{L}_t(W^{(t)}), V\rangle \\
&= \widehat{H}_t - \eta\ell'(y_t f_t(W^{(t)}))y_t\langle \nabla f_t(W^{(t)}), V\rangle \\
&\geqslant \widehat{H}_t - \eta\ell'(y_t f_t(W^{(t)}))a\gamma\sqrt{m}[\alpha - \widehat{\xi}_t(\gamma)] \\
&\geqslant \widehat{H}_t + \eta a\gamma\sqrt{m}\left[\alpha\widehat{\mathcal{E}}_t(W^{(t)}) - \widehat{\xi}_t(\gamma)\right].
\end{aligned}$$

In the first inequality we have used Lemma 4.4.1 and that $-\ell' \geqslant 0$, and in the second inequality we have used that $-\ell' \leqslant 1$. Taking expectations over the draws of the distribution on both sides completes the proof. □

Notice that if $\alpha\mathcal{E}_t > \xi(\gamma)$, Lemma 4.4.2 shows that $H_{t+1} - H_t > 0$. We will later repeat this argument for T iterations to show that until we find a point with $\alpha\mathcal{E}_t \leqslant 2\xi(\gamma)$, H_T will grow at least as fast as $\Omega(T)$.

All that remains is to show that $G_T = O(\sqrt{T})$. We will accomplish this by first demonstrating a bound on $G_{t+1}^2 - G_t^2$.

Lemma 4.4.3. For any $t \in \mathbb{N} \cup \{0\}$, $\eta > 0$, and if $\mathbb{E}[\|x\|^2] \leqslant B_X^2$,

$$G_{t+1}^2 \leqslant G_t^2 + 2\eta + \eta^2 m a^2 B_X^2.$$

Proof. We begin with the identity

$$\widehat{G}_{t+1}^2 = \left\| W^{(t)} - \eta \nabla \widehat{L}_t(W^{(t)}) \right\|_F^2 = \left\| W^{(t)} \right\|_F^2 - 2\eta \left\langle W^{(t)}, \nabla \widehat{L}_t(W^{(t)}) \right\rangle + \eta^2 \left\| \nabla \widehat{L}_t(W^{(t)}) \right\|_F^2 . \tag{4.4.6}$$

We proceed by analyzing the last two terms. We have

$$\begin{aligned} \langle W^{(t)}, \nabla \widehat{L}_t(W^{(t)}) \rangle &= \ell'(y_t f_t(W^{(t)})) y_t \langle W^{(t)}, \nabla f_{x_t}(W^{(t)}) \rangle \\ &= \ell'(y_t f_t(W^{(t)})) y_t \sum_{j=1}^m a_j \sigma'(\langle w_j^{(t)}, x_t \rangle) \langle w_j^{(t)}, x_t \rangle \\ &= \ell'(y_t f_t(W^{(t)})) y_t \sum_{j=1}^m a_j \sigma(\langle w_j^{(t)}, x_t \rangle) \\ &= \ell'(y_t f_t(W^{(t)})) y_t f_t(W^{(t)})). \end{aligned}$$

The third equality uses that σ is homogeneous, so $\sigma'(z)z = \sigma(z)$. We can therefore bound

$$-2\eta \left\langle W^{(t)}, \nabla \widehat{L}_t(W^{(t)}) \right\rangle = -2\eta \ell'(y_t f_t(W^{(t)})) y_t f_t(W^{(t)}) \leqslant 2\eta. \tag{4.4.7}$$

To see that the inequality holds, note that $-\ell'(z) \cdot z \leqslant 1$ if $z \leqslant 1$ since $-\ell'(z) \in [0,1]$, and if $z \geqslant 1$ then $-\ell'(z) \leqslant 1/z$ by Assumption 4.3.1. For the gradient norm term, if we denote $\vec{a} \in \mathbb{R}^m$ as the vector with j-th entry a_j and $\Sigma_t^W \in \mathbb{R}^{m \times m}$ as the diagonal matrix with j-th diagonal entry $\sigma'(\langle w_j, x_t \rangle)$, then

$$\begin{aligned} \left\| \nabla \widehat{L}_t(W) \right\|_F^2 &= \left\| \ell'(y_t f_t(W)) \Sigma_t^W \vec{a} x_t^\top \right\|_F^2 \\ &= \ell'(y_t f_t(W))^2 \left\| \Sigma_t^W \vec{a} \right\|_2^2 \|x_t\|^2 \\ &\leqslant m a^2 \|x_t\|^2 . \end{aligned} \tag{4.4.8}$$

The second equation uses that $\left\|bd^\top\right\|_F = \|b\|_2 \|d\|_2$ for vectors b, d. The inequality uses that $|\ell'| \in [0,1]$.

Substituting (4.4.7) and (4.4.8) into (4.4.6), we get

$$\widehat{G}_{t+1}^2 \leqslant \widehat{G}_t^2 + 2\eta + ma^2\eta^2 \|x_t\|^2 .$$

Taking expectations of both sides over the draws of the distribution we get

$$G_{t+1}^2 \leqslant G_t^2 + 2\eta + ma^2\eta^2 B_X^2,$$

where we have used that $\mathbb{E}[\|x\|^2] \leqslant B_X^2$. □

We now have all of the ingredients needed to prove Theorem 4.3.5.

Proof of Theorem 4.3.5. First, let us note that for V defined as (4.4.1) (satisfying $\|V\|_F = 1$), we have by Cauchy–Schwarz,

$$H_t^2 = (\mathbb{E}[\langle W^{(t)}, V\rangle])^2 \leqslant \mathbb{E}\|W^{(t)}\|_F^2 \mathbb{E}\,\|V\|_F^2 = G_t^2 \iff |H_t| \leqslant G_t. \tag{4.4.9}$$

For $a = 1/\sqrt{m}$, and for $\eta \leqslant (ma^2B_X^2)^{-1} = B_X^{-2}$, Lemma 4.4.3 becomes

$$G_{t+1}^2 \leqslant G_t^2 + 2\eta + \eta^2 ma^2 B_X^2 \leqslant G_t^2 + 3\eta.$$

Summing the above from $t = 0, \ldots, T-1$, we get

$$G_T^2 \leqslant G_0^2 + 3\eta T. \tag{4.4.10}$$

Similarly, Lemma 4.4.2 becomes

$$H_{t+1} \geqslant H_t + \eta\gamma[\alpha\mathcal{E}_t - \xi].$$

(Note that $\xi = \xi(\gamma)$ depends on γ, but we have dropped the notation for simplicity.) Summing the above, we get

$$H_T \geqslant H_0 + \eta\gamma \sum_{t=0}^{T-1} [\alpha\mathcal{E}_t - \xi]. \tag{4.4.11}$$

We can therefore bound

$$\begin{aligned}-G_0 + \eta\gamma \sum_{t=0}^{T-1} [\alpha \mathcal{E}_t - \xi] &\leqslant H_0 + \eta\gamma \sum_{t=0}^{T-1} [\alpha \mathcal{E}_t - \xi] \\ &\leqslant H_T \\ &\leqslant G_T \\ &\leqslant G_0 + \sqrt{T} \cdot 2\sqrt{\eta}. \end{aligned} \tag{4.4.12}$$

The first inequality uses (4.4.9). The second inequality uses (4.4.11). The third inequality again uses (4.4.9). The final inequality uses (4.4.10) together with $\sqrt{a+b} \leqslant \sqrt{a} + \sqrt{b}$.

We claim now that this implies that within a polynomial number of samples, SGD finds weights satisfying $\mathcal{E}_t \leqslant 2\alpha^{-1}\xi$. Suppose that for every iteration $t = 1, \ldots, T$, we have $\mathcal{E}_t > 2\alpha^{-1}\xi$. Then (4.4.12) gives

$$\eta\alpha\gamma\xi T \leqslant 2G_0 + 2\sqrt{\eta} \cdot \sqrt{T} \iff \eta\alpha\gamma\xi \cdot T - 2\sqrt{\eta} \cdot \sqrt{T} - 2G_0 \leqslant 0.$$

This is an equation of the form $\beta_2 x^2 - \beta_1 x - \beta_0 \leqslant 0$, and thus using the quadratic formula, this implies $\sqrt{T} \leqslant (2\beta_2)^{-1}(-\beta_1 + \sqrt{\beta_1^2 - 4\beta_0\beta_2})$. Squaring both sides and using a bit of algebra, this implies

$$T \leqslant \beta_2^{-2}\beta_1^2 + \beta_2^{-3/2}\beta_1\beta_0^{1/2} + \beta_2^{-1}\beta_0.$$

In particular, we have

$$\begin{aligned} T &\leqslant \eta^{-2}\alpha^{-2}\gamma^{-2}\xi^{-2} \cdot 4\eta + \eta^{-3/2}\alpha^{-3/2}\gamma^{-3/2}\xi^{-3/2} \cdot 2\eta^{1/2} \cdot G_0^{1/2} + \eta^{-1}\alpha^{-1}\xi^{-1} \cdot 2G_0 \\ &\leqslant 4\eta^{-1}\alpha^{-2}\gamma^{-2}\xi^{-2}(G_0 \vee 1). \end{aligned}$$

That is, within $T = O(\eta^{-1}\gamma^{-2}\xi^{-2}[G_0 \vee 1])$ iterations, gradient descent finds a point satisfying

$$\mathcal{E}_t = \mathbb{E}_{\mathrm{sgd}}\big[- \ell'\big(y f_x(W^{(t)})\big)\big] \leqslant 2\alpha^{-1}\xi. \tag{4.4.13}$$

By Markov's inequality (see (4.3.2)) this implies

$$\mathbb{P}(y f_x(W^{(t)}) < 0) \leqslant 2|\ell'(0)|^{-1}\alpha^{-1}\xi.$$

To complete the proof, we want to bound ξ. Recall from the calculation (4.4.2) that

$$\xi = \xi(\gamma) = \phi_{v^*}(\gamma) + \mathsf{OPT}_{\mathsf{lin}} + \gamma^{-1}\mathbb{E}\Big[|\langle v^*, x\rangle| \mathbb{1}(y\langle v^*, x\rangle < 0)\Big].$$

Fix $\rho > 0$ to be chosen later. We can write

$$\begin{aligned} \mathbb{E}[|\langle v^*, x\rangle| \mathbb{1}(y\langle v^*, x\rangle < 0)] &= \mathbb{E}[|\bar{v}^\top x| \mathbb{1}(y\bar{v}^\top x \leqslant 0,\ |\bar{v}^\top x| > \rho)] \\ &\quad + \mathbb{E}[|\bar{v}^\top x| \mathbb{1}(y\bar{v}^\top x \leqslant 0,\ |\bar{v}^\top x| \leqslant \rho)] \\ &\leqslant \rho \mathsf{OPT}_{\mathsf{lin}} + \int_\rho^\infty \mathbb{P}(|\bar{v}^\top x| > t) \mathrm{d}t \\ &\leqslant \rho \mathsf{OPT}_{\mathsf{lin}} + \int_\rho^\infty \exp(-t/C_m) \mathrm{d}t \\ &= \rho \mathsf{OPT}_{\mathsf{lin}} + C_m \exp(-\rho/C_m). \end{aligned} \tag{4.4.14}$$

The first inequality comes from Cauchy–Schwarz, the second from truncating, and the last from the definition of C_m-sub-exponential. Taking $\rho = C_m \log(1/\mathsf{OPT})$ results in

$$\mathbb{E}[|\langle v^*, x\rangle| \mathbb{1}(y\bar{v}^\top x \leqslant 0)] \leqslant C_m \mathsf{OPT}_{\mathsf{lin}} \log(1/\mathsf{OPT}_{\mathsf{lin}}) + C_m \mathsf{OPT}_{\mathsf{lin}}.$$

Substituting the above into (4.4.13), we get

$$\mathbb{P}(y f_x(W^{(t)}) < 0) \leqslant 2|\ell'(0)|^{-1}\alpha^{-1}\left[\phi_{v^*}(\gamma) + (1+\gamma^{-1}C_m)\mathsf{OPT}_{\mathsf{lin}} + \gamma^{-1}C_m \mathsf{OPT}_{\mathsf{lin}} \log(1/\mathsf{OPT}_{\mathsf{lin}})\right].$$

□

4.5 Experiments

In this section, we provide some experimental verification of our theoretical results. We consider a distribution $\mathcal{D}_{b,\gamma_0}$ that is a mixture of two 2D Gaussians perturbed by both random classification noise and deterministic (adversarial) label noise. The distribution is constructed as follows. We first take two independent Gaussians with independent components of unit variance and means $(-3, 0)$ and $(3, 0)$, and assign the label -1 to the left

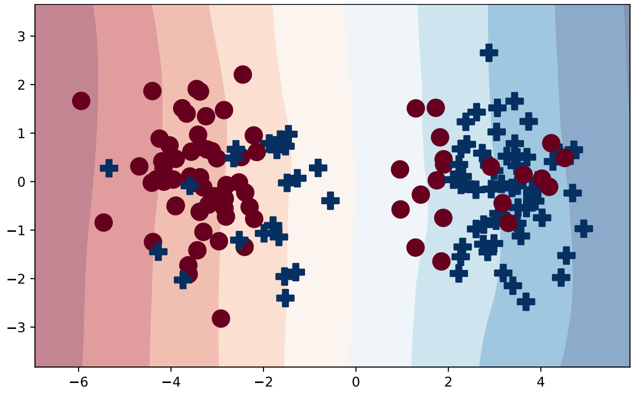

Figure 4.1: Samples from $\mathcal{D}_{2.04,0.5}$ with random classification noise of 10% on $\{|x_1| > 2.04\}$ with the boundary term $b = 2.04$ chosen so that $\mathsf{OPT}_{\mathsf{lin}} = 0.25$. Blue plus signs correspond to $y = +1$ and red circles to $y = -1$. The contour plot displays the class probability for the output of a leaky ReLU network trained by online SGD and has dark hues when the neural network is more confident in its predictions.

cluster and +1 to the right cluster. We remove all samples with first component x_1 satisfying $|x_1| \leqslant \gamma_0 = 0.5$, so that we have a hard margin distribution with margin γ_0. We then introduce a boundary factor $b > \gamma_0$, and for samples with first component satisfying $|x_1| \leqslant b$ we deterministically flip the label to the opposite sign. Finally, for samples with $|x_1| > b$, we introduce random classification noise at level 10%, flipping the labels in those regions with probability 0.1 each. The symmetry of the distribution implies that an optimal halfspace is the vector $v^* = (1, 0)$.

The boundary factor b can be tweaked to incorporate more deterministic label noise which will affect the best linear classifier: if b is larger, $\mathsf{OPT}_{\mathsf{lin}}$ is larger as well. We give details on the precise relationship of b and $\mathsf{OPT}_{\mathsf{lin}}$ in Section 4.8. But because this 'noise' is deterministic, the best classifier over $\mathcal{D}_{b,\gamma_0}$ (the Bayes optimal classifier) can always achieve accuracy of at least 90% by using the decision rule

$$y_{\text{Bayes}} = \begin{cases} +1, & x_1 \in (-b, 0) \cup (b, \infty), \\ -1, & x_1 \in (-\infty, b] \cup [0, b]. \end{cases} \tag{4.5.1}$$

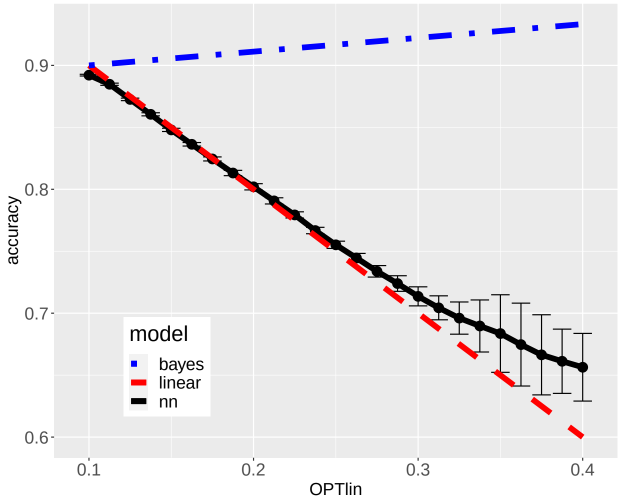

Figure 4.2: Test classification accuracy for data coming $\mathcal{D}_{b,0.5}$. The red dashed line is the accuracy of the best linear classifier, and the black solid line is the average accuracy of the neural network with error bars over ten random initializations of the first layer weights (experimental details can be found in Section 4.8). The blue dash-dotted line is the Bayes optimal classifier accuracy.

Since the error for the Bayes decision rule corresponds to the region $\{|x_1| > b\}$ with random classification noise, we can exactly calculate the error for the Bayes classifier as well as $\mathsf{OPT}_{\mathsf{lin}}$. As b increases, the region with random classification noise becomes smaller, and thus the Bayes classifier gets better as the linear classifier becomes worse on $\mathcal{D}_{b,\gamma_0}$. This makes $\mathcal{D}_{b,\gamma_0}$ a good candidate for understanding the performance of SGD-trained one-hidden-layer networks in comparison to linear classifiers. Further, to our knowledge no previous work has been able to show that neural networks can provably generalize if the data distribution is $\mathcal{D}_{b,\gamma_0}$.[4]

Since $\mathcal{D}_{b,\gamma_0}$ is a subexponential hard margin distribution, Corollary 4.3.6 shows that we can expect an SGD-trained leaky ReLU network on $\mathcal{D}_{b,0.5}$ to achieve a test set accuracy of at least $1 - C \cdot \mathsf{OPT}_{\mathsf{lin}} \log(1/\mathsf{OPT}_{\mathsf{lin}})$ for some constant $C \geqslant 1$. We ran experiments on such a neural network with $m = 1000$ neurons and learning rate $\eta = 0.01$ and first layer weights initialized as independent normal random variables with variance $1/m$ (see Section 4.8 for more details on the experiment setup). In Figure 4.1 we plot the decision boundary for the SGD-trained neural network on the distribution $\mathcal{D}_{2.04,0.5}$, where $b = 2.04$ is chosen so that $\mathsf{OPT}_{\mathsf{lin}} = 0.25$. We notice that the decision boundary is almost exactly linear and is essentially the same as that of the best linear classifier $(x_1, x_2) \mapsto \operatorname{sgn}(x_1)$. And in Figure 4.2, we see that the neural network accuracy is almost exactly equal to $1 - \mathsf{OPT}_{\mathsf{lin}}$ when $\mathsf{OPT}_{\mathsf{lin}} \leqslant 0.30$ and that the network slightly outperforms the best linear classifier when $\mathsf{OPT}_{\mathsf{lin}} > 0.30$.

In Section 4.8 we conduct additional experiments to better understand whether this behavior is consistent across hyperparameter and architectural modifications to the network. When using the bias-free networks of the form (4.3.1) we consider in this paper, we found that one-hidden-layer SGD-trained networks failed to generalize better than a linear classifier when using tanh activations (Figures 4.4 and 4.5), using different learning rates (Figures 4.6,

[4] There are two reasons that no other work can show generalization bounds in the settings we consider. The first is the presence of adversarial label noise. The second is that our generalization bound holds for neural networks with finite width and any initialization. All previous works fail to allow at least one of these conditions.

4.7, 4.8), different initialization variances (Figures 4.9, 4.10, 4.11), and using multiple-pass SGD rather than online SGD (Figures 4.12, 4.13). On the other hand, we found that introducing bias terms can lead to decision boundaries closer to that of the Bayes-optimal classifier (Figures 4.14, 4.15, 4.16). Interestingly, this behavior was strongly dependent on the initialization scheme used: when using an initialization variance of $1/m^4$, a linear decision boundary was consistently learned, while using an initialization variance of $1/m$ lead to approximately Bayes-optimal decision boundaries. By contrast, the result we present in Theorem 4.3.5 holds for *arbitrary* initialization schemes. This suggests that a new analytical approach would be needed in order to guarantee neural network generalization performance better than that of a linear classifier on $\mathcal{D}_{\gamma_0,b}$.

4.6 Discussion

We have shown that overparameterized one-hidden-layer networks can generalize almost as well as the best linear classifier over the distribution for a broad class of distributions. Our results imply two related but distinct insights on SGD-trained neural networks. First, regardless of the initialization scheme and number of neurons, SGD training will produce neural networks that are competitive with the best linear predictor over the data, providing theoretical support for the hypothesis presented by [NKK19] that the performance of SGD-trained networks in the early stages of training can be explained by that of a linear classifier. Second, a linearly separable dataset can be corrupted by adversarial label noise and overparameterized neural networks will still be able to generalize, despite the capacity to overfit to the label noise.

A number of extensions and open questions remain. First, our analysis was specific to one-hidden-layer networks with the leaky-ReLU activation. We are interested in extending our results to more general neural network architectures. Second, a natural question is whether or not there are concept classes that are more expressive than halfspaces for which overparameterized neural networks can generalize for noisy data. We are particularly keen on understanding this question for finite width neural networks that are not well-approximated by the NTK.

4.7 Proof of Lemma 4.4.1

In this section we will prove a stronger version of Lemma 4.4.1 that holds for any increasing activation.

Lemma 4.7.1. Suppose that σ is non-decreasing. For V defined in (4.4.1), for any $(x,y) \in \mathbb{R}^d \times \{\pm 1\}$, for any $W \in \mathbb{R}^{m\times d}$, and any $\gamma \in (0,1)$:

$$\begin{aligned} &y\langle \nabla f_x(W), V\rangle \\ &\geqslant a\gamma m^{-1/2}\Big[1 - \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - (1+\gamma^{-1})|\langle v^*, x\rangle|\mathbb{1}(y\langle v^*, x\rangle < 0)\Big]\sum_{j=1}^m \sigma'(\langle w_j, x\rangle). \end{aligned} \tag{4.7.1}$$

For $\sigma(z) = \max(\alpha z, z)$, we have $\sum_{j=1}^m \sigma'(\langle w_j, x\rangle) \in [\alpha m, m]$, and hence the above implies Lemma 4.4.1:

$$\begin{aligned} &y\langle \nabla f_x(W), V\rangle \\ &\geqslant a\gamma m^{-1/2}\Big[\alpha m - m\mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - m(1+\gamma^{-1})|\langle v^*, x\rangle|\mathbb{1}(y\langle v^*, x\rangle < 0)\Big] \\ &= a\gamma\sqrt{m}\Big[\alpha - \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - (1+\gamma^{-1})|\langle v^*, x\rangle|\mathbb{1}(y\langle v^*, x\rangle < 0)\Big]. \end{aligned}$$

Proof of Lemma 4.7.1. By the definition of V (see (4.4.1)), we have

$$\begin{aligned} &y\langle \nabla f_x(W), V\rangle = \sum_{j=1}^m a_j\sigma'(\langle w_j, x\rangle)\langle y v_j, x\rangle \\ &= am^{-1/2}\sum_{j=1}^m \sigma'(\langle w_j, x\rangle)\langle yv^*, x\rangle \\ &= am^{-1/2}\sum_{j=1}^m \sigma'(\langle w_j, x\rangle)\langle yv^*, x\rangle\Big[\mathbb{1}(y\langle v^*, x\rangle \geqslant \gamma) + \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) + \mathbb{1}(y\langle v^*, x\rangle < 0)\Big]. \end{aligned}$$

The second line uses that $a_j v_j = |a_j| v^* = a v^*$. Continuing, we have

$$\begin{aligned}
&y\langle \nabla f_x(W), V\rangle \\
&\geqslant a\gamma m^{-1/2} \mathbb{1}(y\langle v^*, x\rangle \geqslant \gamma) \cdot \sum_{j=1}^m \sigma'(\langle w_j, x\rangle) \\
&\quad + a m^{-1/2} \sum_{j=1}^m \sigma'(\langle w_j, x\rangle)\langle y v^*, x\rangle \Big[\mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) + \mathbb{1}(y\langle v^*, x\rangle < 0)\Big] \\
&\geqslant a\gamma m^{-1/2} \mathbb{1}(y\langle v^*, x\rangle \geqslant \gamma) \cdot \sum_{j=1}^m \sigma'(\langle w_j, x\rangle) + a m^{-1/2} \sum_{j=1}^m \sigma'(\langle w_j, x\rangle)\langle y v^*, x\rangle \mathbb{1}(y\langle v^*, x\rangle < 0) \\
&= a\gamma m^{-1/2} [1 - \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - \mathbb{1}(y\langle v^*, x\rangle < 0)] \sum_{j=1}^m \sigma'(\langle w_j, x\rangle) \\
&\quad + a m^{-1/2} \sum_{j=1}^m \sigma'(\langle w_j, x\rangle)\langle y v^*, x\rangle \mathbb{1}(y\langle v^*, x\rangle < 0) \\
&\geqslant a\gamma m^{-1/2} [1 - \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - \mathbb{1}(y\langle v^*, x\rangle < 0)] \sum_{j=1}^m \sigma'(\langle w_j, x\rangle) \\
&\quad - a m^{-1/2} |\langle v^*, x\rangle| \mathbb{1}(y\langle v^*, x\rangle < 0) \sum_{j=1}^m \sigma'(\langle w_j, x\rangle) \\
&= a m^{-1/2} \Big[\gamma - \gamma \mathbb{1}(y\langle v^*, x\rangle \in [0,\gamma)) - (\gamma+1)|\langle v^*, x\rangle| \mathbb{1}(y\langle v^*, x\rangle < 0)\Big] \sum_{j=1}^m \sigma'(\langle w_j, x\rangle).
\end{aligned}$$

The first and second inequalities use that $\sigma'(z) \geqslant 0$ and that $a > 0$. The third inequality uses that $x \geqslant -|x|$. This proves (4.7.1). □

4.8 Additional Experiments and Experiment Details

In this section, we give details on the experiments given in Section 4.5. Let us first describe how we calculate $\mathsf{OPT}_{\mathsf{lin}}$ for $\mathcal{D}_{b,\gamma_0}$. To remind the reader, we begin by constructing $\mathcal{D}_{b,\gamma_0}$ with a mixture of two independent Gaussians centered at $(-3,0)$ and $(3,0)$ with independent unit variance components and then remove all data that has x_1 component in the interval $[-\gamma_0, \gamma_0]$. We assign initial labels to be -1 if $x_1 < 0$ and 1 if $x_1 > 0$. For boundary factor

$b > \gamma_0$, the deterministic adversarial label noise then assigns the label 1 if $-b < x_1 < -\gamma_0$, and assigns the label -1 if $\gamma_0 < x_1 < b$. The final labels are determined by flipping labels for samples with $|x_1| > b$ with probability p each.

By construction, an optimal unit-norm halfspace classifier is given by the vector $(1, 0)$, and this classifier is a hard-margin classifier with margin $\gamma_0 > 0$. The optimal halfspace classification error is given as the sum of two terms: (1) the random classification noise for the region $|x_1| > b$, and (2) the deterministic noise in the region $|x_1| < b$. The error introduced from the deterministic, adversarial noise is the proportion of 2D Gaussian that has x_1 coordinate lying between $3 - \gamma_0$ and $3 - b$, conditioned on the fact that x_1 is at most $3 - \gamma_0$. We can directly calculate this as

$$\mathrm{err}_{\mathrm{det}} = \frac{\mathbb{P}(3 - b < N(0,1) \leqslant 3 - \gamma_0)}{\mathbb{P}(N(0,1) \leqslant 3 - \gamma_0)} = \frac{\Phi(3 - \gamma_0) - \Phi(3 - b)}{\Phi(3 - \gamma_0)},$$

where Φ is the standard normal cumulative distribution function. Similarly, the error for the best linear classifier introduced by the random classification noise at rate p is given by p times the proportion of a 2D Gaussian that has x_1 coordinate smaller than $3-b$, conditioned on the x_1 coordinate being at most $3 - \gamma_0$. That is,

$$\mathrm{err}_{\mathrm{rcn}} = p\frac{\mathbb{P}(N(0,1) \leqslant 3 - b)}{\mathbb{P}(N(0,1) \leqslant 3 - \gamma_0)} = p\frac{\Phi(3 - b)}{\Phi(3 - \gamma_0)}. \tag{4.8.1}$$

The total error for the optimal linear classifier is then given by

$$\begin{aligned}\mathsf{OPT}_{\mathsf{lin}} &= \mathrm{err}_{\mathrm{det}} + \mathrm{err}_{\mathrm{rcn}}\\ &= \frac{1}{\Phi(3 - \gamma_0)}\Big(\Phi(3 - \gamma_0) - \Phi(3 - b) + p\Phi(3 - b)\Big)\\ &= \frac{1}{\Phi(3 - \gamma_0)}\Big(\Phi(3 - \gamma_0) - (1 - p)\Phi(3 - b)\Big).\end{aligned}$$

Solving for the boundary term in terms of $\mathsf{OPT}_{\mathsf{lin}}$ results in

$$b = 3 - \Phi^{-1}\left(\frac{1 - \mathsf{OPT}_{\mathsf{lin}}}{1 - p}\Phi(3 - \gamma_0)\Big)\right).$$

We then consider $\mathsf{OPT}_{\mathsf{lin}}$ in a grid and take the corresponding values of the boundary term b to produce a distribution with hard margin $\gamma_0 = 0.5$ where the best population risk achieved by a linear classifier is $\mathsf{OPT}_{\mathsf{lin}}$. We note that the Bayes-optimal classifier has decision rule given by (4.5.1) with the Bayes risk equal to err_{rcn}.

The baseline neural network model we use, and the neural network used for Figure 4.2, is as follows. We use a bias-free one-hidden-layer network (4.3.1) with leaky ReLU activations (with $\alpha = 0.1$) and $m = 1000$ neurons with outer layer fixed at initialization with half of the a_j equal to $+1/\sqrt{m}$ and the other half equal to $-1/\sqrt{m}$. We initialize the hidden layer weights independently with normal random variables with variance $1/m$, so that $G_0^2 = \|W^{(0)}\|_F^2 = O(1)$ with high probability (ignoring $d = 2$ as a small constant). We use online SGD (i.e. batch size one with a new sample used at each iteration) with $T =$ 20,000 samples[5] trained on the cross-entropy loss with fixed learning rate $\eta = 0.01$. We use a validation set of size 10,000 and evaluate performance on the validation set every 100 SGD iterations, and we take the model with the smallest validation error over the T samples and evaluate its performance on a fresh test set (sampled independently from the training and validation sets) of 100,000 samples to produce the final test set accuracy. We then repeat this experiment ten times for each level of $\mathsf{OPT}_{\mathsf{lin}}$ considered with the ten trials using different seeds for both the initialization of the first layer weights and for the sequence of data observed in online SGD (i.e. for fixed data $\{x_t\}_1^T$, we use a permutation $\pi : [T] \to [T]$ to permute the data $\{x_t\}_1^T \mapsto \{x_{\pi(t)}\}_1^T$). We plot the average across the ten trials with error bars corresponding to one standard deviation in Figure 4.2; in all subsequent modifications to this baseline neural network model, we will always plot the mean and error bars over the ten trials considered. We calculate the Bayes-optimal classification error by using the boundary term corresponding to each value of $\mathsf{OPT}_{\mathsf{lin}}$ and plotting $\mathrm{err}_{\mathrm{rcn}}$ as the blue dash-dotted line in Figure 4.2. Code for our experiments is available on Github.[6]

In Figure 4.3, we show the decision boundary of the baseline neural network for $\mathsf{OPT}_{\mathsf{lin}} \in$

[5] In ablation studies with $T = 100{,}000$ samples, we observed no discernible difference in the classification accuracy, unless otherwise stated.

[6] https://github.com/spencerfrei/nn_generalization_agnostic_noise

$\{0.1, 0.25, 0.40\}$ for four independent initializations of the first layer weights. For each level of $\mathsf{OPT}_{\mathsf{lin}}$, the neural network classifier has a nearly linear decision boundary.

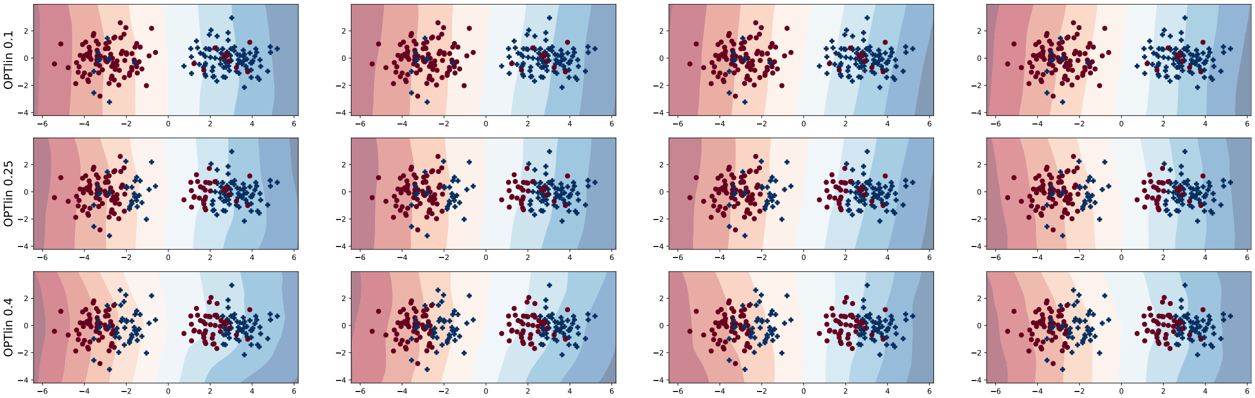

Figure 4.3: Decision boundary of an SGD-trained neural network on $\mathcal{D}_{b,\gamma_0}$, where b is chosen so that $\mathsf{OPT}_{\mathsf{lin}} \in \{0.1, 0.25, 0.40\}$, across four different random initializations. The decision boundary is the line where the region changes from light red to light blue, and the dark regions are areas where the neural network classifier has the highest confidence. Even in the presence of substantial, adversarial noise, the decision boundary is close to linear.

In Figures 4.4 and 4.5, we modify the baseline neural network by having tanh activations instead of leaky ReLU. Although tanh is highly nonlinear, the performance of tanh networks is essentially the same as the leaky ReLU network, and the decision boundaries are approximately linear even for large $\mathsf{OPT}_{\mathsf{lin}}$.

In Figures 4.6, 4.7, and 4.8, we consider variations of the learning rate from the baseline $\eta = 0.01$ to $\eta \in \{0.1, 0.001\}$. Overall, the test accuracy is essentially the same, albeit of smaller variance across initializations when the learning rate is smaller. When the learning rate is smaller, the decision boundary is almost perfectly linear, even when $\mathsf{OPT}_{\mathsf{lin}} = 0.4$. When $\eta = 0.1$, the decision boundary changes significantly for different initializations of the first layer weights, resulting in a higher variance for the test accuracy, but the decision boundary is still a rough perturbation of the best linear classifier decision boundary.

In Figures 4.9, 4.10, and 4.11, we examine the effect of modifying the initialization

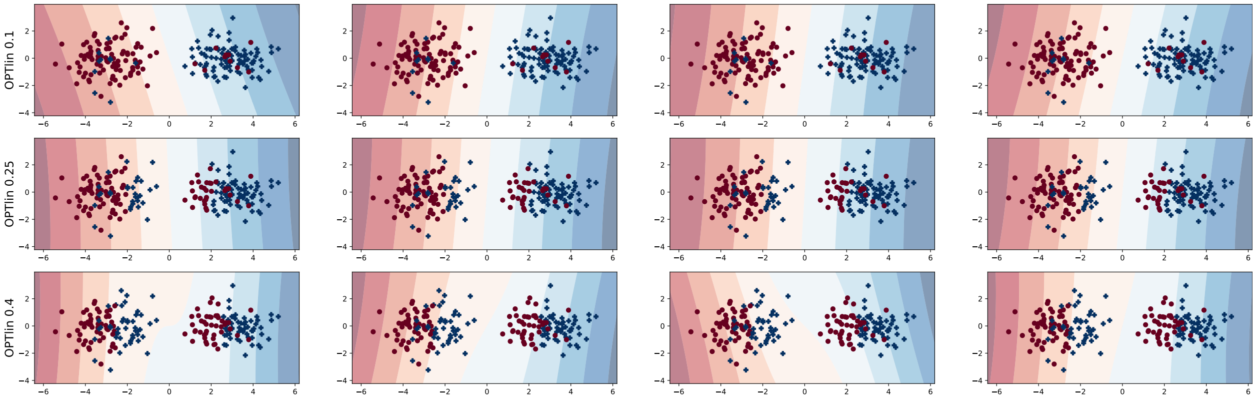

Figure 4.4: Decision boundary for the same setup as the baseline neural network except with tanh activations. Columns correspond to different random initializations. Compare with Figure 4.3. Even for nonlinear activations we still see an almost perfectly linear decision boundary for $\mathsf{OPT}_{\mathsf{lin}} = 0.25$.

of the first layer weights from the baseline variance of $1/m$ to $\mathrm{Var}(w_{i,j}^{(0)}) \in \{m^{-2}, 1\}$. The overall accuracy is essentially the same across initialization variances. The decision boundary becomes more smooth and linear when the variance is smaller. When the variance is larger, the decision boundary is more disjointed and nonsmooth, but is still roughly a perturbation of the best linear classifier decision boundary.

In Figures 4.12 and 4.13, we consider the modification of using 100 epochs of multiple-pass SGD with batch size 32. All other architectural and optimization hyperparameters from the baseline case are the same. We see that the decision boundary and test accuracy has less variance across random initializations, which we interpret as being due to the averaging effect of increasing the batch size from 1 to 32. The test classification accuracy is virtually indistinguishable from the online SGD case.

In Figures 4.14, 4.15, and 4.16, we consider two modifications to the neural network: (1) increasing the width from the baseline of $m = 10^3$ to $m = 10^5$, and (2) introducing trainable bias terms and training the second layer weights. The difference in (1) is imperceptible and so we do not plot the decision boundary in this case. On the other hand, we observed that with

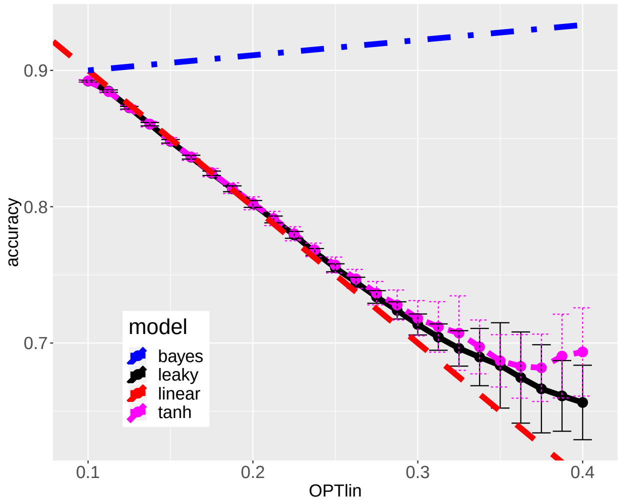

Figure 4.5: Test classification accuracy. The performance of leaky ReLU and tanh networks are almost exactly the same and match the performance of the best linear predictor until extreme levels of noise.

trainable biases and second layer weights, the neural network can come close to Bayes-optimal classifier accuracy provided the initialization variance is chosen appropriately. In particular, with an initialization variance of $1/m$, the network is able to learn a nonlinear decision boundary, but with an initialization variance of $1/m^4$, the network only learns a linear decision boundary.[7] We note that our result in Theorem 4.3.5 holds for *any* initialization, and thus these experiments suggest that we would need to introduce new analyses in order to get generalization performance much better than a linear classifier. Additionally, these experiments suggest that the ability of an SGD-trained network to generalize better than a linear classifier on $\mathcal{D}_{\gamma_0,b}$ is strongly dependent upon the initialization scheme used and the usage of bias terms.

As a final study on $\mathcal{D}_{b,\gamma_0}$, we consider a three-hidden-layer fully connected network of the

[7] For the experiments involving trainable biases and second layer weights, we increased the sample size from $T = 20{,}000$ to $T = 100{,}000$ since the validation accuracy was still continuing to increase with $T = 20{,}000$ for the initialization variance of $1/m$. This was the only set of experiments where we noticed such behavior.

form

$$x^{(1)} = \sigma(W^{(1)}x), \quad x^{(l)} = \sigma(W^{(l)}x^{(l-1)}),\ l = 2, 3, \quad f_x(\vec{W}, \vec{b}) = a^\top x^{(3)}, \tag{4.8.2}$$

where $W^{(1)} \in \mathbb{R}^{m \times d}$, $W^{(l)} \in \mathbb{R}^{m \times m}$ for $l = 2, 3$, $a \in \mathbb{R}^{m \times 1}$ are all trainable weights, and σ is again the leaky ReLU with $\alpha = 0.1$. In Figures 4.17 and 4.18, we plot the decision boundary and accuracy for this four layer network (with $m = 100$) with each layer's weights initialized with variance $1/m$ and the final layer weights initialized at $\pm 1/\sqrt{m}$ and the same learning rate of 0.01. This network is able to learn a better partition of the input space and is able to generalize almost as well as the Bayes optimal classifier, enjoying the same trend of increase in performance as $\mathsf{OPT}_{\mathsf{lin}}$ increases that holds for the Bayes optimal classifier. This experiment suggests that although there is evidence that bias-free one-hidden-layer networks fail to learn $\mathcal{D}_{b,\gamma_0}$ up to an accuracy better than a linear classifier, bias-free networks with multiple hidden layers can.

We also conducted a series of experiments to emphasize that although it seems that bias-free SGD-trained one-hidden-layer networks cannot learn $\mathcal{D}_{b,\gamma_0}$ to an accuracy better than a linear classifier, there are simple distributions for which such networks easily outperform linear predictors. We construct a distribution $\tilde{\mathcal{D}}_b$ as follows. We introduce a boundary factor $b > 0$ and sample an isotropic 2D Gaussian, and then assign the label $+1$ if $x_2 < b|x_1|$, and the label -1 otherwise. Every (bias-free) halfspace for the marginal distribution of a 2D Gaussian partitions any circle centered at the origin into two equal-sized halves. By symmetry of the isotropic Gaussian, this means the best halfspace will have error exactly equal to the proportion of $+1$ lying in the region with $0 < x_2 \leqslant b|x_1|$. If we denote the angle corresponding to the region $\{x_2 \geqslant b|x_1|\}$ where $y = -1$ as 2θ, then this means the error of the best linear classifier is given by $\mathsf{OPT}_{\mathsf{lin}} = \frac{\pi - 2\theta}{2\pi} = {}^1\!/\!_2 - {}^\theta\!/\!_\pi$ (see Figure 4.19). The angle $\theta \in [0, {}^\pi\!/\!_2]$ is given by $\theta = \arctan(1/b)$, and thus we can solve for $\mathsf{OPT}_{\mathsf{lin}}$ in terms of b. When $b \to 0$, the error for the best halfspace converges to 0, while as $b \to \infty$ we have $\mathsf{OPT}_{\mathsf{lin}} \to {}^1\!/\!_2$. The Bayes classifier achieves accuracy 100% with the decision rule $y_{\text{Bayes}} = 1$ if $x_2 < b|x_1|$ and -1 otherwise.

The 2D Gaussian satisfies 1-anti-concentration and Corollary 4.3.7 guarantees that an SGD-trained neural network will achieve a test set accuracy of at least $1 - \tilde{\Omega}(\sqrt{\mathsf{OPT}_{\mathsf{lin}}})$. We see in Figures 4.21 and 4.20 that the neural network performs quite a bit better than the best linear classifier (and significantly better than $1 - \sqrt{\mathsf{OPT}_{\mathsf{lin}}}$), with the decision boundary notably nonlinear and attuned to the distribution of the data. In summary, one-hidden-layer bias-free leaky ReLU networks trained by SGD can learn nonlinear decision boundaries, but apparently not the type of decision boundary necessary to outperform linear classifiers on $\mathcal{D}_{b,\gamma_0}$.

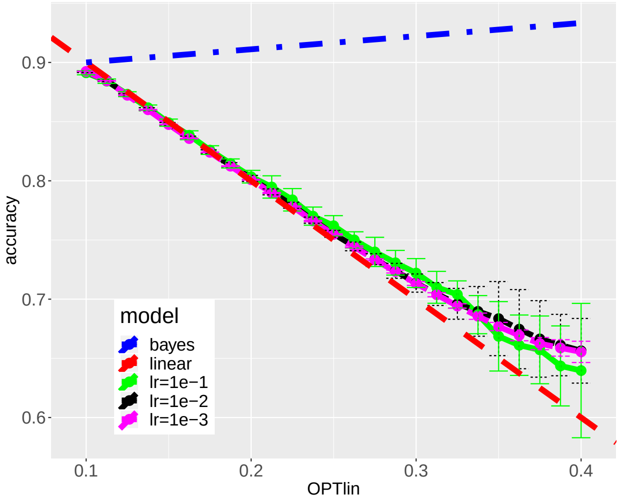

Figure 4.6: Test classification accuracy for learning rates $\eta = 0.1$ and $\eta = 0.001$ compared to baseline $\eta = 0.01$. Large learning rates lead to a larger variance in performance.

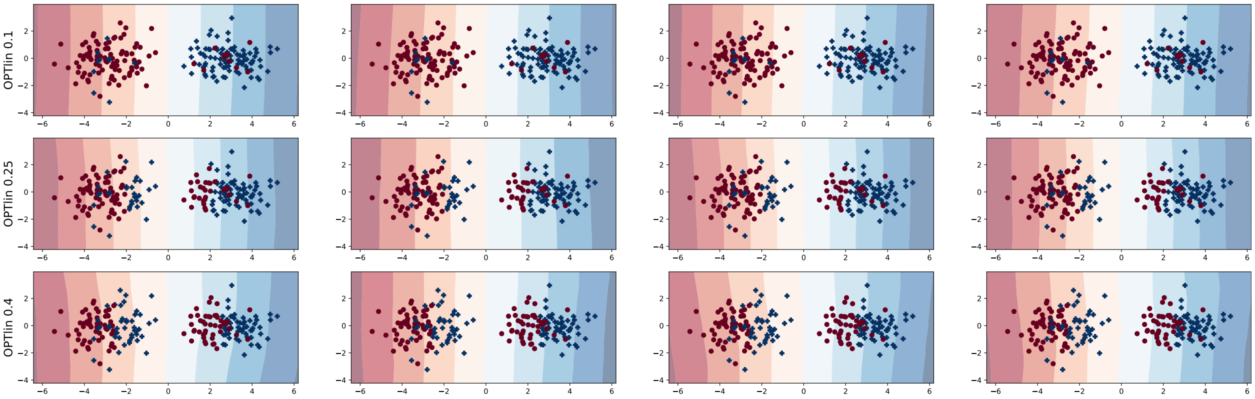

Figure 4.7: Decision boundary for $\eta = 0.001$ is consistently linear.

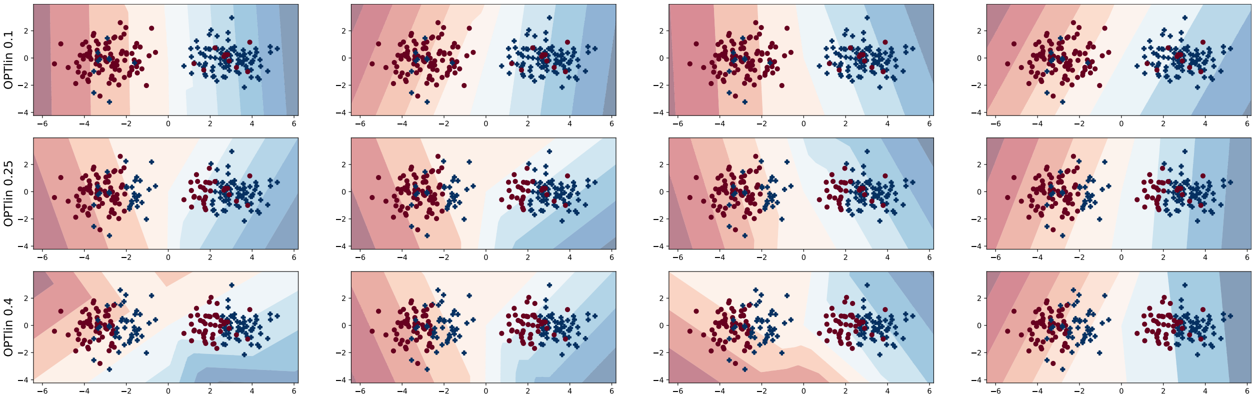

Figure 4.8: Decision boundary for $\eta = 0.1$ varies over initializations but is roughly a perturbation of the linear classifier decision boundary.

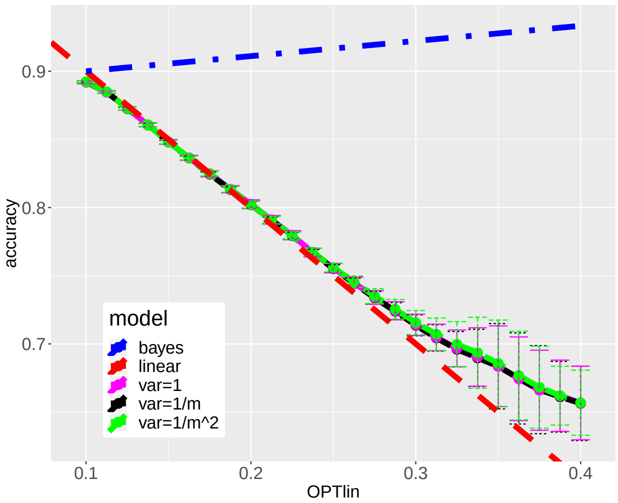

Figure 4.9: Test classification accuracy for different values of the variance of the first layer weight initialization. The baseline neural network has variance $1/m$.

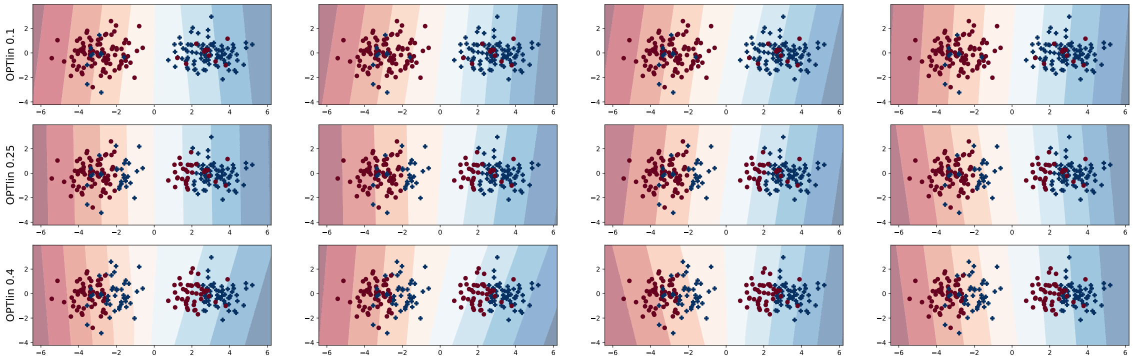

Figure 4.10: Decision boundary for the smaller variance $1/m^2$ is more consistently linear.

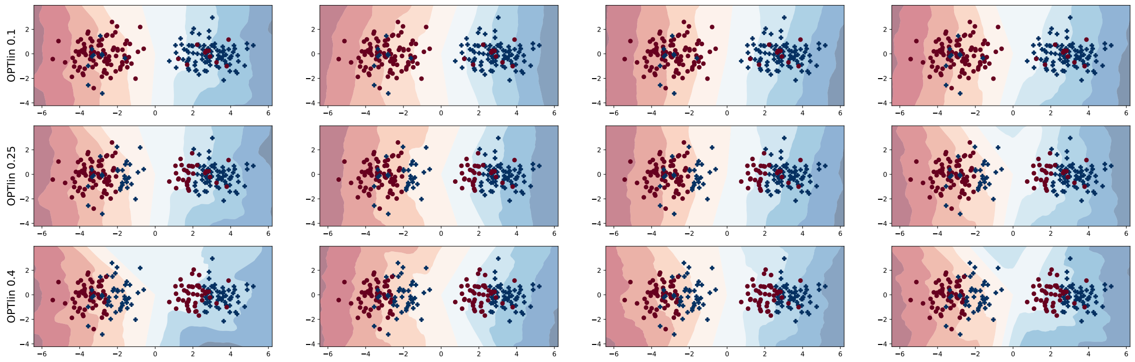

Figure 4.11: Decision boundary for variance 1 has more variation across random initializations, but are roughly perturbations of the linear classifier decision boundary.

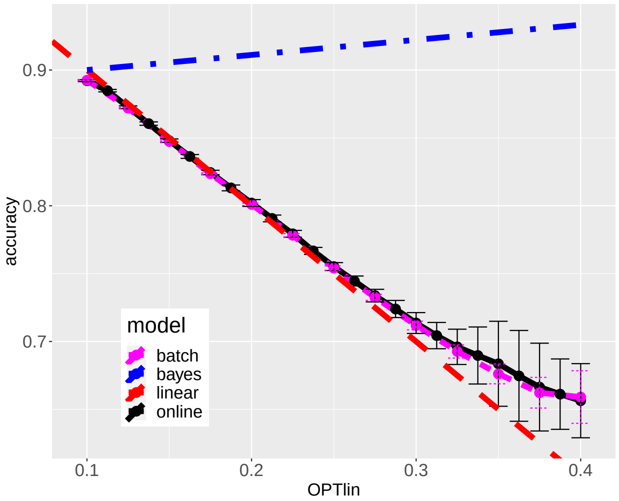

Figure 4.12: Test classification accuracy for multiple-pass batch SGD. The differences with online SGD are essentially indistinguishable.

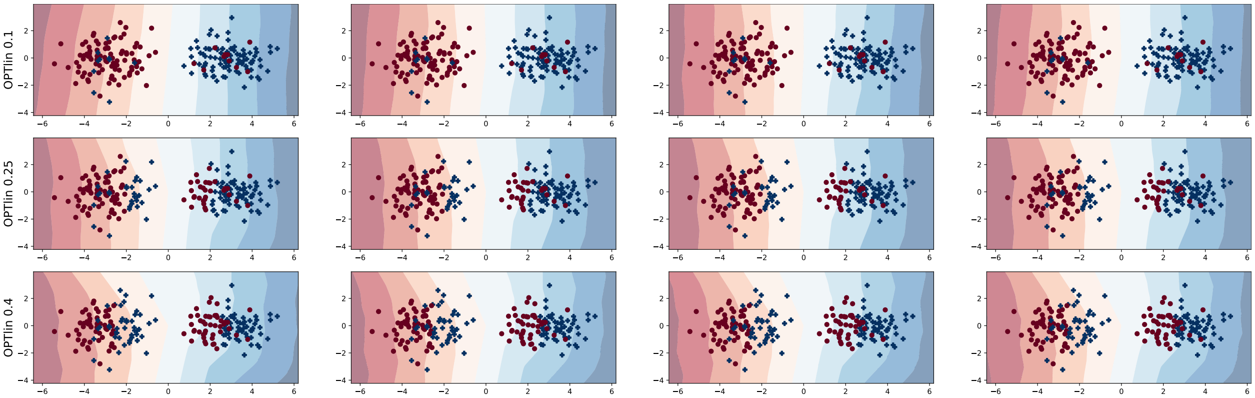

Figure 4.13: Decision boundary when using 100 epochs multiple-pass SGD of batch size 32. Columns correspond to different random initializations. The decision boundary is more consistent across randomizations than the baseline online SGD algorithm.

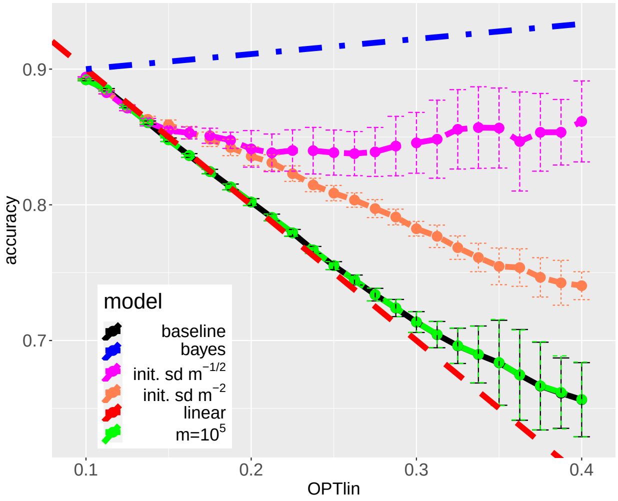

Figure 4.14: Test classification accuracy when introducing bias terms and trainable second layer weights (pink and coral dashed lines) as well as when increasing the width from $m =$ 1,000 to $m = 100{,}000$ (green line). The pink dashed line uses an initialization variance of $1/m$ while the coral dashed line uses an initialization variance of $1/m^4$. Note that the performance of a neural network with width $m = 1{,}000$ and width $m = 100{,}000$ is imperceptible. With trainable bias and second layer weights, the accuracy of the network varies significantly based on the initialization scheme. Note that our result (Theorem 4.3.5) holds for an arbitrary initialization.

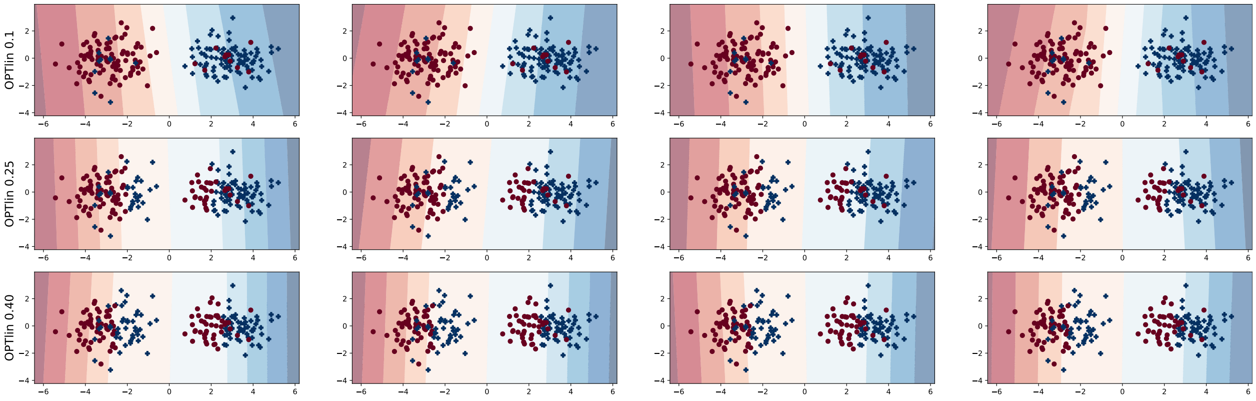

Figure 4.15: Decision boundary when using trainable biases and second layer weights with an initialization variance of $1/m^4$. The boundary is almost exactly linear.

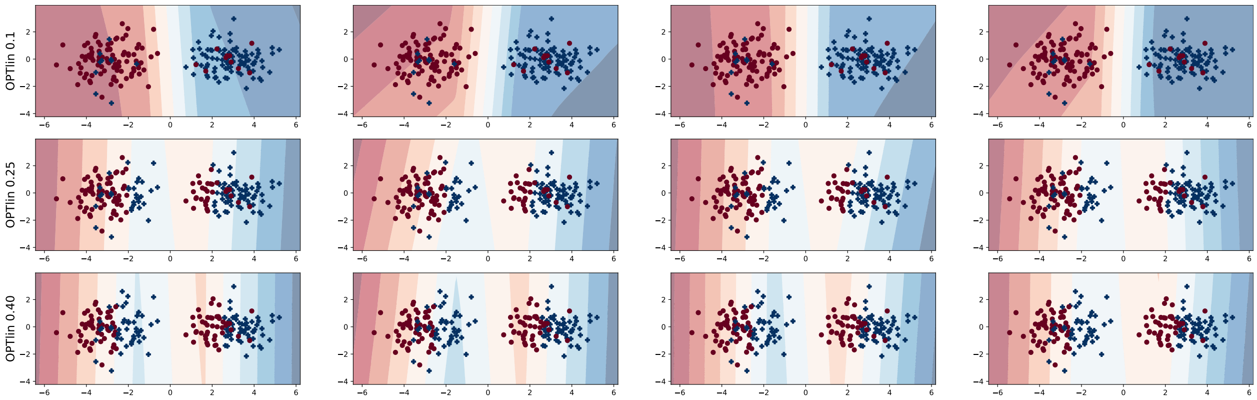

Figure 4.16: Same as Figure 4.15 but using an initialization variance of $1/m$. Here, the network can learn the appropriate nonlinear decision boundary.

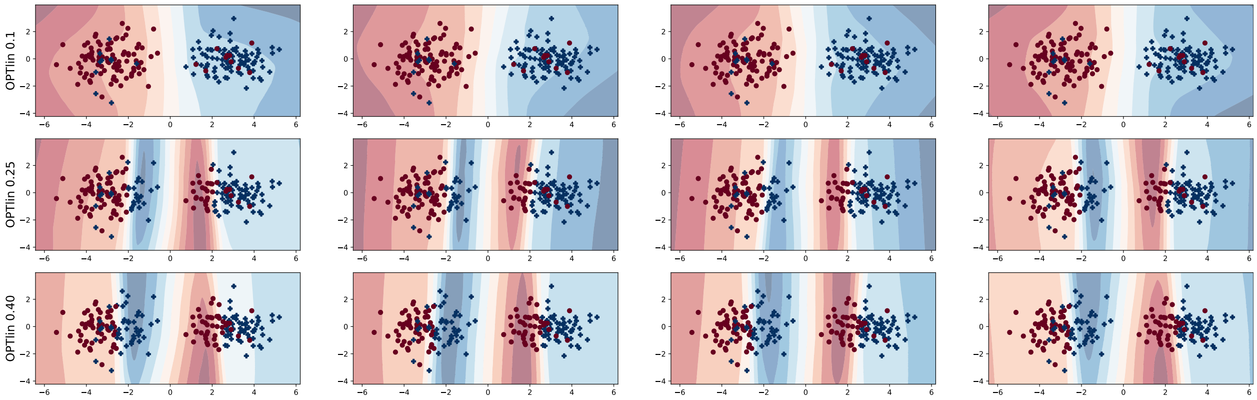

Figure 4.17: Decision boundary for four layer network given in (4.8.2). Columns correspond to different random initializations. Compare with Figure 4.3. With four layers, the network is able to appropriately partition the input space and generalize well.

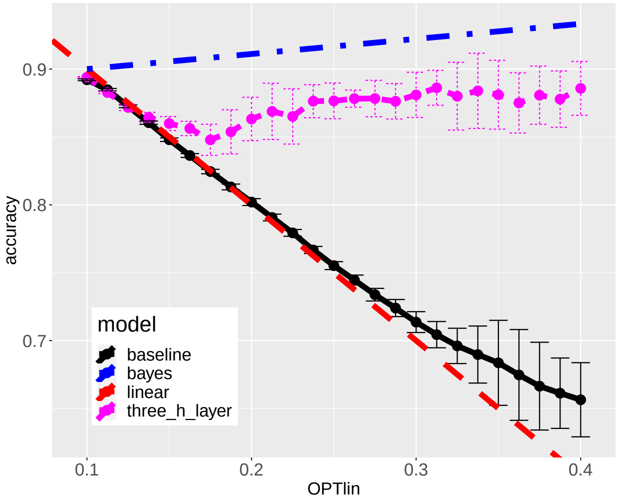

Figure 4.18: Test classification accuracy using the four layer network. The four layer network accuracy is larger for $\mathsf{OPT}_{\mathsf{lin}} = 0.4$ than it is for $\mathsf{OPT}_{\mathsf{lin}} = 0.15$, a behavior closer to that of the Bayes classifier.

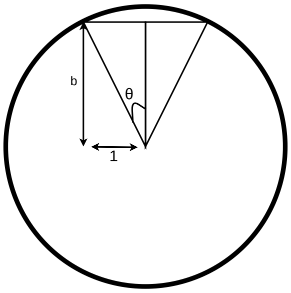

Figure 4.19: Calculation of the angle 2θ for the distribution $\tilde{\mathcal{D}}_b$.

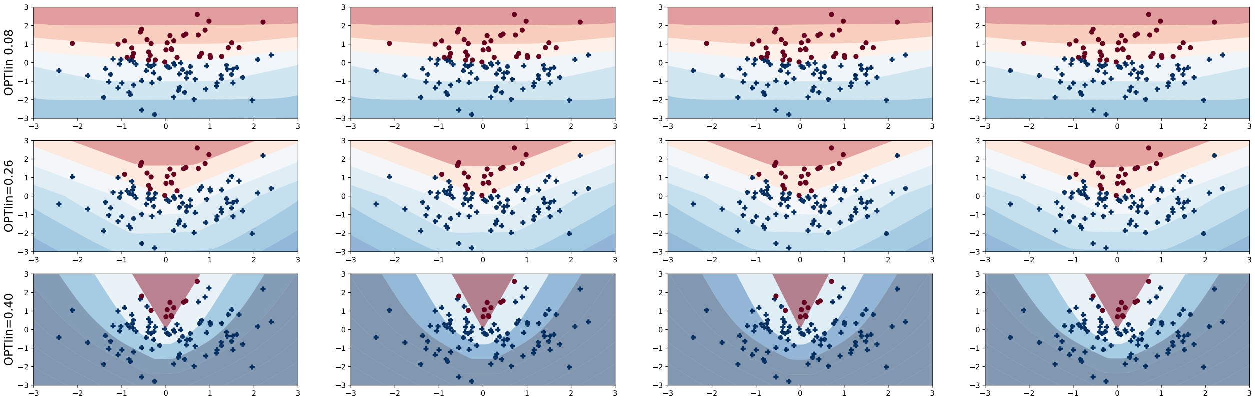

Figure 4.20: Decision boundary for the same setup as the baseline neural network for data coming from $\tilde{\mathcal{D}}_b$ for four random initializations (across columns) and for $\mathsf{OPT}_{\mathsf{lin}} \in \{0.08, 0.26, 0.40\}$ (across rows). Compare with Figure 4.3. The decision boundaries are noticeably nonlinear.

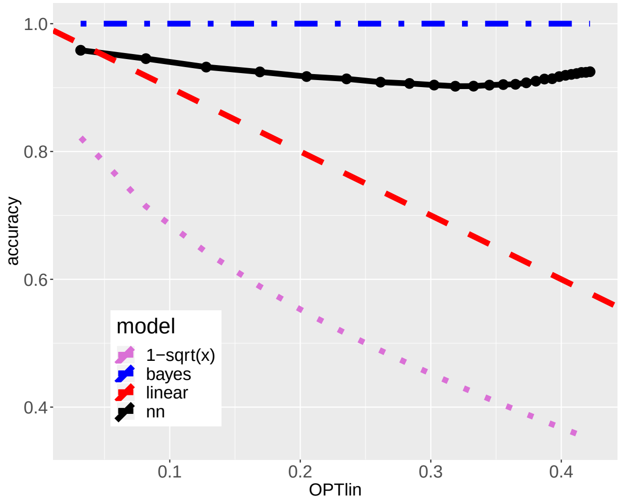

Figure 4.21: Test classification accuracy for data coming $\tilde{\mathcal{D}}_b$. Corollary 4.3.7 guarantees performance of at least $1 - \Omega(\sqrt{\mathsf{OPT}_{\mathsf{lin}}})$, but the neural network performs significantly better due to the ability to produce a nonlinear decision boundary. Note that the variance over ten initializations of the first layer weights are so small that the error bars are not visible.

CHAPTER 5

Learning with deep residual networks trained by gradient descent

5.1 Introduction

An important recent development in the practical deployment of neural networks has been the introduction of skip connections between layers, leading to a class of architectures known as residual networks. Residual networks were first introduced by [HZR16] to much fanfare, quickly becoming a standard architecture choice for state-of-the-art neural network classifiers. The motivation for residual networks came from the poor behavior of very deep traditional fully connected networks: although deeper fully connected networks can clearly express any function that a shallower one can, in practice (i.e. using gradient descent) it can be difficult to choose hyperparameters that result in small training error. Deep residual networks, on the other hand, are remarkably stable in practice, in the sense that they avoid getting stuck at initialization or having unpredictable oscillations in training and validation error, two common occurrences when training deep non-residual networks. Moreover, deep residual networks have been shown to generalize with better performance and far fewer parameters than non-residual networks [TL18, CSS19, IMA16]. We note that much of the recent neural network generalization literature has focused on non-residual architectures [BFT17, NBS18, AGN18, GRS18, CG20] with bounds for the generalization gap that grow exponentially as the depth of the network increases. [LLW18] recently studied a class of residual networks and proved algorithm-independent bounds for the generalization gap that become larger as the depth of the network increases, with a dependence on the depth that is somewhere between sublinear and exponential (a precise characterization requires further assumptions and/or analysis). We note that verifying the non-vacuousness of algorithm-independent generalization bounds relies on empirical arguments about what values the quantities that appear in the bounds generally take in practical networks (i.e. norms of weight matrices and interlayer activations), while algorithm-dependent generalization bounds such as the ones we provide in this paper can be understood without relying on experiments.

5.1.1 Our Contributions

In this work, we consider fully connected deep ReLU residual networks and study optimization and generalization properties of such networks that are trained with discrete time gradient descent following Gaussian initialization.

We consider binary classification under the cross-entropy loss and focus on data that come from distributions $\mathcal{D}$ for which there exists a function f for which $y \cdot f(x) \geqslant \gamma > 0$ for all $(x, y) \in \operatorname{supp} \mathcal{D}$ from a large function class $\mathcal{F}$ (see Assumption 5.3.2). By analyzing the trajectory of the parameters of the network during gradient descent, for any error threshold $\varepsilon > 0$, we are able to show:

1. Under the cross-entropy loss, we can study an analogous surrogate error and bound the true classification error by the true surrogate error. This method was introduced by [CG20].

2. If $m^* = \tilde{O}(\mathrm{poly}(\gamma^{-1})) \cdot \max(d, \varepsilon^{-2})$, then provided every layer of the network has at least $m \geqslant m^*$ units, gradient descent with small enough step size finds a point with empirical surrogate error at most ε in at most $\tilde{O}(\mathrm{poly}(\gamma^{-1}) \cdot \varepsilon^{-1})$ steps with high probability. Here, $\tilde{O}(\cdot)$ hides logarithmic factors that may depend on the depth L of the network, the margin γ, number of samples n, error threshold ε, and probability level δ.

3. Provided $m^* = \tilde{O}(\mathrm{poly}(\gamma^{-1}, \varepsilon^{-1}))$ and $n = \tilde{O}(\mathrm{poly}(\gamma^{-1}, \varepsilon^{-1}))$, the difference between the empirical surrogate error and the true surrogate error is at most ε with high probability, and therefore the above provide a bound on the true classification error of the learned network.

We emphasize that our guarantees above come with at most logarithmic dependence on the depth of the network. Our methods are adapted from those used in the fully connected architecture by [CG20] to the residual network architecture, and rely upon the neural tangent kernel approximation [JGH18]. This approximation relies upon the fact that for a particular initialization of the weights of the network, gradient descent-trained networks can be closely approximated by their tangent kernel at initialization. The tangent kernel at initialization is expressive enough to be guaranteed to find small surrogate training error, but has sufficiently small complexity to guarantee a small generalization gap between the training and test errors. By showing that these competing phenomena occur simultaneously, we are able to derive the test error guarantees of Corollary 5.3.7. The key insight of our analysis is that the Lipschitz constant of the network output for deep residual networks as well as the semismoothness property (Lemma 5.4.2) have at most logarithmic dependence on the depth, while the known analogues for non-residual architectures all have polynomial dependence on the depth.

5.1.2 Additional Related Work

In the last year there has been a variety of works developing algorithm-dependent guarantees for neural network optimization and generalization [LL18, ALS19, ZCZ19, DZP19, ADH19b, CG20, ZG19, CG19a]. [LL18] were among the first to theoretically analyze the properties of overparameterized fully connected neural networks trained with Gaussian random initialization, focusing on a two layer (one hidden layer) model under a data separability assumption. Their work provided two significant insights into the training process of overparameterized ReLU neural networks: (1) the weights stay close to their initial values throughout the optimization trajectory, and (2) the ReLU activation patterns for a given example do not change much throughout the optimization trajectory. These insights were the backbone of the authors' strong generalization result for stochastic gradient descent (SGD) in the two layer case. The insights of [LL18] provided a basis to various subsequent studies. [DZP19] analyzed a two layer model using a method based on the Gram matrix using inspiration from kernel methods, showing that gradient descent following Gaussian initialization finds zero training loss solutions at a linear rate. [ZCZ19] and [ALS19] extended the results of Li and Liang to the arbitrary L hidden layer fully connected case, again considering (stochastic) gradient descent trained from random initialization. Both authors showed that, provided the networks were sufficiently wide, arbitrarily deep networks would converge to a zero training loss solution at a linear rate, using an assumption about separability of the data. Recently, [ZG19] provided an improved analysis of the global convergence of gradient descent and SGD for training deep neural networks, which enjoys a milder over-parameterization condition and better iteration complexity than previous work. Under the same data separability assumption, [ZYC19] showed that deep residual networks can achieve zero training loss for the squared loss at a linear rate with overparameterization essentially independent of the depth of the network. We note that [ZYC19] studied optimization for the regression problem rather than classification, and their results do not distinguish the case with random labels from that with true labels; hence, it is not immediately clear how to translate their analysis to a generalization bound for classification under the cross-entropy loss as we are able to do in this paper.

The above results provide a concrete answer to the question of why overparameterized deep neural networks can achieve zero training loss using gradient descent. However, the theoretical tools of [DZP19, ALS19, ZCZ19, ZG19] apply to data with random labels as well as true labels, and thus do not explain the generalization to unseen data observed experimentally. [DR17] optimized PAC-Bayes bounds for the generalization error of a class of stochastic neural networks that are perturbations of standard neural networks trained by SGD. [CG20] proved a guarantee for arbitrarily small generalization error for classification in deep fully connected neural networks trained with gradient descent using random initialization. The same authors recently provided an improved result for deep fully connected networks trained by stochastic gradient descent using a different approach that relied on the neural tangent kernel and online-to-batch conversion [CG19a]. [EMW19] recently developed algorithm-dependent generalization bounds for a special residual network architecture with many different kinds of skip connections by using kernel methods.

5.2 Network Architecture and Optimization Problem

We begin with the notation of the paper. We denote vectors by lowercase letters and matrices by uppercase letters, with the assumption that a vector v is a column vector and its transpose $v^\top$ is a row vector. We use the standard $O(\cdot), \Omega(\cdot), \Theta(\cdot)$ complexity notations to ignore universal constants, with $\tilde{O}(\cdot), \tilde{\Omega}(\cdot)$ additionally ignoring logarithmic factors. For $n \in \mathbb{N}$, we write $[n] = \{1, 2, \ldots, n\}$. Denote the number of hidden units at layer l as m_l, $l = 1, \ldots, L+1$. Let the l-th layer weights be $W_l \in \mathbb{R}^{m_{l-1} \times m_l}$, and concatenate all of the layer weights into a vector $W = (W_1, \ldots, W_{L+1})$. Denote by $w_{l,j}$ the j-th column of W_l. Let $\sigma(x) = \max(0, x)$ be the ReLU nonlinearity, and let θ be a constant scaling parameter. We consider a class of residual networks defined by the following architecture:

$$\begin{aligned} x_1 &= \sigma(W_1^\top x), \qquad x_l = x_{l-1} + \theta\sigma\left(W_l^\top x_{l-1}\right), \;\; l = 2, \ldots, L, \\ x_{L+1} &= \sigma(W_{L+1}^\top x_L). \end{aligned}$$

Above, we denote x_l as the l-th hidden layer activations of input $x \in \mathbb{R}^d$, with $x_0 := x$. In order for this network to be defined, it is necessary that $m_1 = m_2 = \cdots = m_L$. We are free to choose m_{L+1}, as long as $m_{L+1} = \Theta(m_1)$ (see Assumption 5.3.4). We define a constant, non-trainable vector $v = (1, 1, \ldots, 1, -1, -1, \ldots, -1)^\top \in \mathbb{R}^{m_{L+1}}$ with equal parts $+1$ and -1's that determines the network output,

$$f_W(x) = v^\top x_{L+1}.$$

We note that our methods can be extended to the case of a trainable top layer weights v by choosing the appropriate scale of initialization for v. We choose to fix the top layer weights in this paper for simplicity of exposition.

We will find it useful to consider the matrix multiplication form of the ReLU activations, which we describe below. Let $\mathbb{1}(A)$ denote the indicator function of a set A, and define diagonal matrices $\Sigma_l(x) \in \mathbb{R}^{m_l \times m_l}$ by $[\Sigma_l(x)]_{j,j} = \mathbb{1}(w_{l,j}^\top x_{l-1} > 0)$, $l = 1, \ldots, L+1$. By convention we denote products of matrices $\prod_{i=a}^b M_i$ by $M_b \cdot M_{b-1} \cdot \ldots \cdot M_a$ when $a \leqslant b$, and by the identity matrix when $a > b$. With this convention, we can introduce notation for the l-to-l' interlayer activations $H_l^{l'}(x)$ of the network. For $2 \leqslant l \leqslant l' \leqslant L$ and input $x \in \mathbb{R}^d$ we denote

$$H_l^{l'}(x) := \prod_{r=l}^{l'} \left(I + \theta \Sigma_r(x) W_r^\top \right). \qquad (2 \leqslant l \leqslant l' \leqslant L) \tag{5.2.1}$$

If $l = 1 < l'$, we denote $H_1^{l'}(x) = H_2^{l'}(x) \Sigma_1(x) W_1^\top$, and if $l' = L+1 > l$, we denote $H_l^{L+1}(x) = \Sigma_{L+1}(x) W_{L+1}^\top H_l^L(x)$. Using this notation, we can write the output of the neural network as $f_W(x) = v^\top H_{l+1}^{L+1}(x) x_l$ for any $l \in \{0\} \cup [L+1]$ and $x \in \mathbb{R}^d$. For notational simplicity, we will denote $\Sigma_l(x)$ by Σ_l and $H_l^{l'}(x)$ by $H_l^{l'}$ when the dependence on the input is clear.

We assume we have i.i.d. samples $(x_i, y_i)_{i=1}^n \sim \mathcal{D}$ from a distribution $\mathcal{D}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{\pm 1\}$. We note the abuse of notation in the above, where $x_l \in \mathbb{R}^{m_l}$ refers to the l-th hidden layer activations of an arbitrary input $x \in \mathbb{R}^d$ while x_i refers to the i-th sample $x_i \in \mathbb{R}^d$. We shall use $x_{l,i} \in \mathbb{R}^{m_l}$ when referring to the l-th hidden layer activations of a sample $x_i \in \mathbb{R}^d$ (where $i \in [n]$ and $l \in [L+1]$), while $x_l \in \mathbb{R}^{m_l}$ shall refer to the l-th hidden layer activation of arbitrary input $x \in \mathbb{R}^d$.

Let $\ell(x) = \log(1 + \exp(-x))$ be the cross-entropy loss. We consider the empirical risk minimization problem optimized by constant step size gradient descent,

$$\min_W L_S(W) := \frac{1}{n} \sum_{i=1}^n \ell(y_i \cdot f_W(x_i)), \qquad W_l^{(k+1)} = W_l^{(k)} - \eta \cdot \nabla_{W_l} L_S(W^{(k)}) \quad (l \in [L+1]).$$

We shall see below that a key quantity for studying the trajectory of the weights in the above optimization regime is a surrogate loss defined by the derivative of the cross-entropy loss. We denote the empirical and true surrogate loss by

$$\mathcal{E}_S(W) := -\frac{1}{n} \sum_{i=1}^n \ell'(y_i \cdot f_W(x_i)), \quad \mathcal{E}_D(W) := \mathbb{E}_{(x,y) \sim \mathcal{D}}[-\ell'(y \cdot f_W(x))],$$

respectively. The empirical surrogate loss was first introduced by [CG20] for the study of deep non-residual networks. Finally, we note here a formula for the gradient of the output of the network with respect to different layer weights:

$$\nabla_{W_l} f_W(x) = \theta^{\mathbb{1}(2 \leqslant l \leqslant L)} x_{l-1} v^\top H_{l+1}^{L+1} \Sigma_l(x), \qquad (1 \leqslant l \leqslant L+1). \tag{5.2.2}$$

5.3 Main Theory

We first go over the assumptions necessary for our proof and then shall discuss our main results. Our assumptions align with those made by [CG20] in the fully connected case. The first main assumption is that the input data is normalized.

Assumption 5.3.1. Input data are normalized: $\mathrm{supp}(\mathcal{D}_x) \subset S^{d-1} = \{x \in \mathbb{R}^d : \|x\|_2 = 1\}$.

Data normalization is common in statistical learning theory literature, from linear models up to and including recent work in neural networks [LL18, ZCZ19, DZP19, ALS19, ADH19b, CG20], and can easily be satisfied for arbitrary training data by mapping samples $x \mapsto x/\|x\|_2$.

The next assumption is on the data generating distribution. Because overparameterized networks can memorize data, any hope of demonstrating that neural networks have a small generalization gap must restrict the class of data distribution processes to one where some type of learning is possible.

Assumption 5.3.2. Let $p(u)$ denote the density of a standard d-dimensional Gaussian vector. Define

$$\mathcal{F} = \left\{ \int_{\mathbb{R}^d} c(u)\sigma(u^\top x)p(u)\mathrm{d}u : \ \|c(\cdot)\|_\infty \leqslant 1 \right\}.$$

Assume there exists $f(\cdot) \in \mathcal{F}$ and constant $\gamma > 0$ such that $y \cdot f(x) \geqslant \gamma$ for all $(x, y) \in \mathrm{supp}(\mathcal{D})$.

Assumption 5.3.2 was introduced by [CG20] for the analysis of fully connected networks and is applicable for distributions where samples can be perfectly classified by the random kitchen sinks model of [RR08]. One can view a function from this class as the infinite width limit of a one-hidden-layer neural network with regularizer given by a function $c(\cdot)$ with bounded ℓ^∞-norm. As pointed out by [CG20], this assumption includes the linearly separable case.

Our next assumption concerns the scaling of the weights at initialization.

Assumption 5.3.3 (Gaussian initialization)**.** We say that the weight matrices $W_l \in \mathbb{R}^{m_{l-1}\times m_l}$ are generated via Gaussian initialization if each of the entries of W_l are generated independently from $N(0, 2/m_l)$.

This assumption is common to much of the recent theoretical analyses of neural networks [LL18, ZCZ19, ALS19, DZP19, ADH19b, CG20] and is known as the He initialization due to its usage in the first ResNet paper by [HZR16]. This assumption guarantees that the spectral norms of the weights are controlled at initialization.

Our last assumption concerns the widths of the networks we consider and allows us to exclude pathological dependencies between the width and other parameters that define the architecture and optimization problem.

Assumption 5.3.4 (Widths are of the same order)**.** We assume $m_{L+1} = \Theta(m_L)$. We call $m = m_L \wedge m_{L+1}$ the width of the network.

Our first theorem shows that provided we have sufficient overparameterization and sufficiently small step size, the iterates $W^{(k)}$ of gradient descent stay within a small neighborhood of their initialization. Additionally, the empirical surrogate error can be bounded by a term that decreases as we increase the width m of the network.

Theorem 5.3.5. Suppose $W^{(0)}$ are generated via Gaussian initialization and that the residual scaling parameter satisfies $\theta = 1/\Omega(L)$. For $\tau > 0$, denote a τ-neighborhood of the weights $W^{(0)} = (W_1^{(0)}, \ldots, W_{L+1}^{(0)})$ at initialization by

$$\mathcal{W}(W^{(0)}, \tau) := \left\{ W = (W_1, \ldots, W_{L+1}) : \left\| W_l - W_l^{(0)} \right\|_F \leqslant \tau \ \forall l \in [L+1] \right\}.$$

There exist absolute constants $\nu, \nu', \nu'', C, C' > 0$ such that for any $\delta > 0$, provided $\tau \leqslant \nu\gamma^{12}(\log m)^{-\frac{3}{2}}$, $\eta \leqslant \nu'(\tau m^{-\frac{1}{2}} \wedge \gamma^4 m^{-1})$, and $K\eta \leqslant \nu''\tau^2\gamma^4(\log(n/\delta))^{-\frac{1}{2}}$, then if the width of the network is such that,

$$m \geqslant C'\left(\tau^{-\frac{4}{3}} d\log\frac{m}{\tau\delta} \vee d\log\frac{mL}{\delta} \vee \tau^{-\frac{2}{3}}(\log m)^{-1}\log\frac{L}{\delta} \vee \gamma^{-2}\left(d\log\frac{1}{\gamma} \vee \log\frac{L}{\delta}\right) \vee \log\frac{n}{\delta}\right)$$

then with probability at least $1-\delta$, gradient descent starting at $W^{(0)}$ with step size η generates K iterates $W^{(1)}, \ldots, W^{(K)}$ that satisfy:

(i) $W^{(k)} \in \mathcal{W}(W^{(0)}, \tau)$ for all $k \in [K]$.

(ii) There exists $k \in \{0, \ldots, K-1\}$ with $\mathcal{E}_S(W^{(k)}) \leqslant C \cdot m^{-\frac{1}{2}} \cdot (K\eta)^{-\frac{1}{2}} \left(\log\frac{n}{\delta}\right)^{\frac{1}{4}} \cdot \gamma^{-2}$.

This theorem allows us to restrict our attention from the large class of all deep residual neural networks to the reduced complexity class of those with weights that satisfy $W \in \mathcal{W}(W^{(0)}, \tau)$. Our analysis provides a characterization of the radius of this reduced complexity class in terms of parameters that define the network architecture and optimization problem. Additionally, this theorem allows us to translate the optimization problem over the empirical loss $L_S(W)$ into one for the empirical surrogate loss $\mathcal{E}_S(W^{(k)})$, a quantity that is simply related to the classification error (its expectation is bounded by a constant multiple of the classification error under 0-1 loss; see Appendix 5.6.2).

Our next theorem characterizes the Rademacher complexity of the class of residual networks with weights in a τ-neighborhood of the initialization. Additionally, it connects the test accuracy with the empirical surrogate loss and the Rademacher complexity.

Theorem 5.3.6. Let $W^{(0)}$ denote the weights at Gaussian initialization and suppose the residual scaling parameter satisfies $\theta = 1/\Omega(L)$. Suppose $\tau \leqslant 1$. Then there exist absolute constants $C_1, C_2, C_3 > 0$ such that for any $\delta > 0$, provided

$$m \geqslant C_1\left(\tau^{-\frac{2}{3}}(\log m)^{-1}\log(L/\delta) \vee \tau^{-\frac{4}{3}} d\log(m/(\tau\delta)) \vee d\log(mL/\delta)\right),$$

then with probability at least $1-\delta$, we have the following bound on the Rademacher complexity,

$$\mathfrak{R}_n\left(\{f_W : W \in \mathcal{W}(W^{(0)}, \tau)\}\right) \leqslant C_2\left(\tau^{\frac{4}{3}}\sqrt{m\log m} + \frac{\tau\sqrt{m}}{\sqrt{n}}\right),$$

so that for all $W \in \mathcal{W}(W^{(0)}, \tau)$,

$$\mathbb{P}_{(x,y)\sim\mathcal{D}}\left(y\cdot f_W(x) < 0\right) \leqslant 2\mathcal{E}_S(W) + C_2\left(\tau^{\frac{4}{3}}\sqrt{m\log m} + \frac{\tau\sqrt{m}}{\sqrt{n}}\right) + C_3\sqrt{\frac{\log(1/\delta)}{n}}. \quad (5.3.1)$$

We shall see in Section 5.4 that we are able to derive the above bound on the Rademacher complexity by using a semi-smoothness property of the neural network output and an upper bound on the gradient of the network output. Standard arguments from statistical learning theory provide the first and third terms in (5.3.1).

The missing ingredients needed to realize the result of Theorem 5.3.6 for networks trained by gradient descent are supplied by Theorem 5.3.5, which gives (i) control of the growth of the empirical surrogate error $\mathcal{E}_S$ along the gradient descent trajectory, and (ii) the distance τ from initialization before which we are guaranteed to find small empirical surrogate error. Putting these together yields Corollary 5.3.7.

Corollary 5.3.7. Suppose that the residual scaling parameter satisfies $\theta = 1/\Omega(L)$. Let $\varepsilon, \delta > 0$ be fixed. Suppose that $m^* = \tilde{O}(\mathrm{poly}(\gamma^{-1}))\cdot\max(d, \varepsilon^{-14})\cdot\log(1/\delta)$ and $n = \tilde{O}(\mathrm{poly}(\gamma^{-1}))\cdot\varepsilon^{-4}$. Then for any $m \geqslant m^*$, with probability at least $1-\delta$ over the initialization and training sample, there is an iterate $k \in \{0, \ldots, K-1\}$ with $K = \tilde{O}(\mathrm{poly}(\gamma^{-1}))\cdot\varepsilon^{-2}$ such that gradient descent with Gaussian initialization and step size $\eta = O(\gamma^4\cdot m^{-1})$ satisfies

$$\mathbb{P}_{(x,y)\sim D}[y\cdot f_{W^{(k)}}(x) < 0] \leqslant \varepsilon.$$

This corollary shows that for deep residual networks, provided we have sufficient overparameterization, gradient descent is guaranteed to find networks that have arbitrarily high classification accuracy. In comparison with the results of [CG20], the width m, number of samples n, step size η, and number of iterates K required for the guarantees for residual networks given in Theorem 5.3.5 and Corollary 5.3.7 all have (at most) logarithmic dependence on L as opposed to the exponential dependence in the corresponding results for the non-residual architecture. Additionally, we note that the step size and number of iterations required for our guarantees are independent of the depth, and this is due to the advantage of the residual architecture. Our analysis shows that the presence of skip connections in the network architecture removes the complications relating to the depth that traditionally arise in the analysis of non-residual architectures for a variety of reasons. The first is a technical one from the proof, in which we show that the Lipschitz constant of the network output and the semismoothness of the network depend at most logarithmically on the depth, so that the network width does not blow up as the depth increases (see Lemmas 5.4.1 and 5.4.2 below). Second, the presence of skip-connections allows for representations that are learned in the first layer to be directly passed to later layers without needing to use a wider network to relearn those representations. This property was key to our proof of the gradient lower bound of Lemma 5.4.3 and has been used in previous approximation results for deep residual networks, e.g., [Yar17].

5.4 Proof Sketch of the Main Theory

In this section we will provide a proof sketch of Theorems 5.3.5 and 5.3.6 and Corollary 5.3.7, following the proof technique of [CG20]. We will first collect the key lemmas needed for their proofs, leaving the proofs of these lemmas for Appendix 5.7. We shall assume throughout this section that the residual scaling parameter satisfies $\theta = 1/\Omega(L)$, which we note is a common assumption in the literature of residual network analysis [DLL18, ALS19, ZYC19].

Our first key lemma shows that the interlayer activations defined in (5.2.1) are uniformly bounded in x and l provided the network is sufficiently wide.

Lemma 5.4.1 (Hidden layer and interlayer activations are bounded)**.** Suppose that $W_1, \ldots, W_{L+1}$

are generated via Gaussian initialization. Then there exist absolute constants $C_0, C_1, C_2 > 0$ such that if $m \geqslant C_0 d \log(mL/\delta)$, then with probability at least $1-\delta$, for any $l, l' = 1, \ldots, L+1$ with $l \leqslant l'$ and $x \in S^{d-1}$, we have $C_1 \leqslant \|x_l\|_2 \leqslant C_2$ and $\left\|H_l^{l'}\right\|_2 \leqslant C_2$.

Due to the scaling of θ, we are able to get bounds on the interlayer and hidden layer activations that do not grow with L. As we shall see, this will be key for the sublinear dependence on L for the results of Theorems 5.3.5 and 5.3.6. The fully connected architecture studied by [CG20] had additional polynomial terms in L for both upper bounds for $\|x_l\|_2$ and $\left\|H_l^{l'}\right\|_2$.

Our next lemma describes a semi-smoothness property of the neural network output f_W and the empirical loss L_S.

Lemma 5.4.2 (Semismoothness of network output and objective loss)**.** Let $W_1, \ldots, W_{L+1}$ be generated via Gaussian initialization, and let $\tau \leqslant 1$. Define

$$h(\widehat{W}, \tilde{W}) := \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta \sum_{l=2}^{L} \left\|\widehat{W}_l - \tilde{W}_l\right\|_2 + \left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 .$$

There exist absolute constants $C, \overline{C} > 0$ such that if

$$m \geqslant C \left(\tau^{-\frac{2}{3}} (\log m)^{-1} \log(L/\delta) \vee \tau^{-\frac{4}{3}} d \log(m/(\tau\delta)) \vee d \log(mL/\delta)\right),$$

then with probability at least $1-\delta$, we have for all $x \in S^{d-1}$ and $\widehat{W}, \tilde{W} \in \mathcal{W}(W, \tau)$,

$$\begin{aligned} f_{\widehat{W}}(x) - f_{\tilde{W}}(x) \leqslant{}& \overline{C}\tau^{\frac{1}{3}}\sqrt{m \log m} \cdot h(\widehat{W}, \tilde{W}) + \overline{C}\sqrt{m} \cdot h(\widehat{W}, \tilde{W})^2 \\ &+ \sum_{l=1}^{L+1} \operatorname{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)^\top \nabla_{W_l} f_{\tilde{W}}(x)\right]. \end{aligned}$$

and

$$\begin{aligned} L_S(\widehat{W}) - L_S(\tilde{W}) \leqslant{}& \overline{C}\tau^{\frac{1}{3}}\sqrt{m \log m} \cdot h(\widehat{W}, \tilde{W}) \cdot \mathcal{E}_S(\tilde{W}) + \overline{C} m \cdot h(\widehat{W}, \tilde{W})^2 \\ &+ \sum_{l=1}^{L+1} \operatorname{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)^\top \nabla_{W_l} L_S(\tilde{W})\right]. \end{aligned}$$

The semismoothness of the neural network output function f_W will be used in the analysis of generalization by Rademacher complexity arguments. For the objective loss L_S, we apply this lemma for weights along the trajectory of gradient descent. Since the difference in the weights of two consecutive steps of gradient descent satisfy $W_l^{(k+1)} - W_l^{(k)} = -\eta \nabla_{W_l} L_S(W^{(k)})$, the last term in the bound for the objective loss L_S will take the form $-\eta \sum_{l=1}^{L+1} \|\nabla_{W_l} L_S(W^{(k)})\|_F^2$. Thus by simultaneously demonstrating (i) a lower bound for the gradient for at least one of the layers and (ii) an upper bound for the gradient at all layers (and hence an upper bound for $h(W^{(k+1)}, W^{(k)})$), we can connect the empirical surrogate loss $\mathcal{E}_S(W^{(k)})$ at iteration k with that of the objective loss $L_S(W^{(k)})$ that will lead us to Theorem 5.3.5. Compared with the fully connected architecture of [CG20], our bounds do not have any polynomial terms in L.

Thus the only remaining key items needed for our proof are upper bounds and lower bounds for the gradient of the objective loss, described in the following two lemmas.

Lemma 5.4.3. Let $W = (W_1, \ldots, W_{L+1})$ be weights at Gaussian initialization. There exist absolute constants $C, \underline{C}, \nu$ such that for any $\delta > 0$, provided $\tau \leqslant \nu\gamma^3$ and $m \geqslant C\gamma^{-2}\left(d \log \gamma^{-1} + \log(L/\delta)\right) \vee C \log(n/\delta)$, then with probability at least $1 - \delta$, for all $\tilde{W} \in \mathcal{W}(W, \tau)$, we have

$$\left\|\nabla_{W_{L+1}} L_S(\tilde{W})\right\|_F^2 \geqslant \underline{C} \cdot m_{L+1} \cdot \gamma^4 \cdot \mathcal{E}_S(\tilde{W})^2.$$

Lemma 5.4.4. Let $W = (W_1, \ldots, W_{L+1})$ be weights at Gaussian initialization. There exists an absolute constant $C > 0$ such that for any $\delta > 0$, provided $m \geqslant C\left(d \vee \log(L/\delta)\right)$ and $\tau \leqslant 1$, we have for all $\tilde{W} \in \mathcal{W}(W, \tau)$ and all l,

$$\left\|\nabla_{W_l} L_S(\tilde{W})\right\|_F \leqslant \theta^{\mathbb{1}(2 \leqslant l \leqslant L)} \cdot C\sqrt{m} \cdot \mathcal{E}_S(\tilde{W}).$$

Note that we provide only a lower bound for the gradient at the last layer. It may be possible to improve the degrees of the polynomial terms of the results in Theorems 5.3.5 and 5.3.6 by deriving lower bounds for the other layers as well.

With all of the key lemmas in place, we can proceed with a proof sketch of Theorems 5.3.5 and 5.3.6. The complete proofs can be found in Appendix 5.6.

Proof of Theorem 5.3.5. Consider $h_k = h(W^{(k+1)}, W^{(k)})$, a quantity that measures the distance of the weights between gradient descent iterations. It takes the form

$$h_k = \eta \left[\left\| \nabla_{W_1} L_S(W^{(k)}) \right\|_2 + \theta \sum_{l=2}^{L} \left\| \nabla_{W_l} L_S(W^{(k)}) \right\|_2 + \left\| \nabla_{W_{L+1}} L_S(W^{(k)}) \right\|_2 \right].$$

By Lemma 5.4.4 we can show that $h_k \leqslant C\eta\sqrt{m}\mathcal{E}_S(W^{(k)})$. The gradient lower bound in Lemma 5.4.3 substituted into Lemma 5.4.2 shows that the dominating term in the semismoothness comes from the gradient lower bound, so that we have for any k,

$$L_S(W^{(k+1)}) - L_S(W^{(k)}) \leqslant -C \cdot \eta \cdot m_{L+1} \cdot \gamma^4 \cdot \mathcal{E}_S(W^{(k)})^2.$$

We can telescope the above over k to get a bound on the loss at iteration k in terms of the bound on the r.h.s. and the loss at initialization. A simple concentration argument shows that the loss at initialization is small with mild overparameterization. By letting $k^* = \operatorname{argmin}_{[K-1]} \mathcal{E}_S(W^{(k)})^2$, we can thus show

$$\mathcal{E}_S(W^{(k^*)}) \leqslant C_3 \left(K\eta \cdot m\right)^{-\frac{1}{2}} \left(L_S(W^{(0)})\right)^{\frac{1}{2}} \cdot \gamma^{-2} \leqslant C_3 \left(K\eta \cdot m\right)^{-\frac{1}{2}} \left(\log \frac{n}{\delta}\right)^{\frac{1}{4}} \cdot \gamma^{-2}.$$

□

We provide below a proof sketch of the bound for the Rademacher complexity given in Theorem 5.3.6, leaving the rest for Appendix 5.6.2.

Proof of Theorem 5.3.6. Let ξ_i be independent Rademacher random variables. We consider a first-order approximation to the network output at initialization,

$$F_{W^{(0)},W}(x) := f_{W^{(0)}}(x) + \sum_{l=1}^{L+1} \operatorname{tr} \left[\left(W_l - W_l^{(0)} \right)^\top \nabla_{W_l} f_{W^{(0)}}(x) \right],$$

and bound the Rademacher complexity by two terms,

$$\begin{aligned}\widehat{\mathfrak{R}}_S[\mathcal{F}(W^{(0)},\tau)] \leqslant{}& \mathbb{E}_\xi\left[\sup_{W\in\mathcal{W}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i[f(x_i) - F_{W^{(0)},W}(x_i)]\right] \\ &+ \mathbb{E}_\xi\left[\sup_{W\in\mathcal{W}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i \sum_{l=1}^{L+1} \mathrm{tr}\left[\left(W_l - W_l^{(0)}\right)^\top \nabla_{W_l} f_{W^{(0)}}(x)\right]\right]\end{aligned}$$

For the first term, taking $\tilde{W} = W^{(0)}$ in Lemma 5.4.2 results in $|f_W(x) - F_{W^{(0)},W}(x)| \leqslant C_3\tau^{\frac{4}{3}}\sqrt{m\log m}$. For the second term, since $\|AB\|_F \leqslant \|A\|_F\,\|B\|_2$, we reduce this term to a product of two terms. The first involves the norm of the distance of the weights from initialization, which is τ. The second is the norm of the gradient at initialization, which can be taken care of by using Cauchy–Schwarz and the gradient formula (5.2.2) to get $\|\nabla_{W_l} f_{W^{(0)}}\|_F \leqslant C_2\theta^{\mathbb{1}(2\leqslant\ell\leqslant L)}\sqrt{m}$. A standard application of Jensen inequality gives the $1/\sqrt{n}$ term. □

Finally, we can put together Theorems 5.3.5 and 5.3.6 by appropriately choosing the scale of τ, η, and K to get Corollary 5.3.7. We leave the detailed algebraic calculations for Appendix 5.6.3.

Proof of Corollary 5.3.7. We need only specify conditions on $\tau, \eta, K\eta$, and m such that the results of Theorems 5.3.5 and 5.3.6 will hold, and making sure that each of the four terms in (5.3.1) are of the same scale. This can be satisfied by imposing the condition $K\eta = \nu''\gamma^4\tau^2\left(\log(n/\delta)\right)^{-\frac{1}{2}}$ and

$$C_3\left(K\eta m\right)^{-\frac{1}{2}}\left(\log(n/\delta)\right)^{\frac{1}{4}}\cdot\gamma^{-2} = C_2\tau^{\frac{4}{3}}\sqrt{m\log m} = C_2\tau\sqrt{m/n} = C_3\sqrt{\log(1/\delta)/n} = \varepsilon/4.$$

□

5.5 Conclusions

In this paper, we derived algorithm-dependent optimization and generalization results for overparameterized deep residual networks trained with random initialization using gradient descent. We showed that this class of networks is both small enough to ensure a small generalization gap and also large enough to achieve a small training loss. Important to our analysis is the insight that the introduction of skip connections allows for us to essentially ignore the depth as a complicating factor in the analysis, in contrast with the well-known difficulty of achieving nonvacuous generalization bounds for deep non-residual networks. This provides a theoretical understanding for the increased stability and generalization of deep residual networks over non-residual ones observed in practice.

5.6 Proofs of Main Theorems and Corollaries

5.6.1 Proof of Theorem 5.3.5

We first show that $W^{(k)} \in \mathcal{W}(W^{(0)}, \tau/2)$ for all $k \leqslant K$ satisfying $K\eta \leqslant \nu'' \tau^2 \gamma^4 (\log(n/\delta))^{-1/2}$. Suppose $W^{(k')} \in \mathcal{W}(W^{(0)}, \tau/2)$ for all $k' = 1, \dots, k-1$. By Lemma 5.4.4, we have

$$\left\| \nabla_{W_l} L_S(W^{(k')}) \right\|_F \leqslant C_1 \theta^{\mathbb{1}(2 \leqslant l \leqslant L)} \sqrt{m} \cdot \mathcal{E}_S(W^{(k')}).$$

Since $\eta\sqrt{m} \leqslant \nu'\tau$ and $\mathcal{E}_S(\cdot) \leqslant 1$, we can make ν' small enough so that we have by the triangle inequality

$$\left\| W_l^{(k)} - W_l^{(0)} \right\|_F \leqslant \eta \left\| \nabla_{W_l} L_S(W^{(k-1)}) \right\|_F + \frac{\tau}{2} \leqslant \tau. \tag{5.6.1}$$

Therefore we are in the τ-neighborhood that allows us to apply the bounds described in the main section. Define

$$h_k := \eta \left[\left\| \nabla_{W_1} L_S(W^{(k)}) \right\|_2 + \theta \sum_{l=2}^{L} \left\| \nabla_{W_l} L_S(W^{(k)}) \right\|_2 + \left\| \nabla_{W_{L+1}} L_S(W^{(k)}) \right\|_2 \right].$$

Then using the upper bounds for the gradient given in Lemma 5.4.4, we have

$$h_k \leqslant \eta \left[C\sqrt{m}\mathcal{E}_S(W^{(k)}) + \theta \sum_{l=2}^{L} \left(\theta \sqrt{m} \mathcal{E}_S(W^{(k)}) \right) + C\sqrt{m}\mathcal{E}_S(W^{(k)}) \right] \leqslant C'\eta\sqrt{m}\mathcal{E}_S(W^{(k)}). \tag{5.6.2}$$

Notice that $h_k = h(W^{(k+1)}, W^{(k)})$ where h is from Lemma 5.4.2. Hence, we have

$$\begin{aligned}
&L_S(W^{(k+1)}) - L_S(W^{(k)}) \\
&\leqslant C\tau^{\frac{1}{3}}\sqrt{m\log m}\cdot h_k \cdot \mathcal{E}_S(W^{(k)}) + Cmh_k^2 - \eta\sum_{l=1}^{L+1}\left\|\nabla_{W_l}L_S(W^{(k)})\right\|_F^2 \\
&\leqslant C\eta\tau^{\frac{1}{3}}\sqrt{m\log m}\cdot\sqrt{m}\cdot\mathcal{E}_S(W^{(k)})^2 + Cm^2\eta^2\cdot\mathcal{E}_S(W^{(k)})^2 - C\eta\cdot m_{L+1}\cdot\gamma^4\cdot\mathcal{E}_S(W^{(k)})^2 \\
&\leqslant \mathcal{E}_S(W^{(k)})^2\cdot\left(C_1\eta\tau^{\frac{1}{3}}m\sqrt{\log m} + C_2m^2\cdot\eta^2 - C_3\eta\cdot m_{L+1}\cdot\gamma^4\right)
\end{aligned}$$

The first inequality follows by Lemma 5.4.2 and since $\mathrm{tr}(A^\top A) = \|A\|_F^2$. The second inequality uses the lower bound for the gradient given in Lemma 5.4.3 and (5.6.2). Therefore, if we take $\tau^{\frac{1}{3}}\sqrt{\log m} \leqslant \nu^{\frac{1}{3}}\gamma^4$, i.e. $\tau \leqslant \nu\cdot\gamma^{12}\left(\log m\right)^{-\frac{3}{2}}$ for some small enough constant ν, and if we take $\eta \leqslant \nu'\cdot\gamma^4 m^{-1}$, then there is a constant $C > 0$ such that

$$L_S(W^{(k+1)}) - L_S(W^{(k)}) \leqslant -C\cdot\eta\cdot m_{L+1}\cdot\gamma^4\cdot\mathcal{E}_S(W^{(k)})^2. \tag{5.6.3}$$

Re-writing this we have

$$\mathcal{E}_S(W^{(k)})^2 \leqslant C\gamma^{-4}\left(\eta m_{L+1}\right)^{-1}\left(L_S(W^{(k)}) - L_S(W^{(k+1)})\right). \tag{5.6.4}$$

Before completing this part of the proof, we will need the following bound on the loss at initialization:

$$L_S(W^{(0)}) \leqslant C\sqrt{\log\frac{n}{\delta}}. \tag{5.6.5}$$

To see this, we notice that $f_W(x_i)$ is a sum of $m_{L+1}/2$ independent random variables (conditional on $x_{L,i}$),

$$f_W(x_i) = \sum_{j=1}^{m_{L+1}/2}\left[\sigma(w_{L+1,j}^\top x_{L,i}) - \sigma(w_{L+1,j+m_{L+1}/2}^\top x_{L,i})\right].$$

Applying the upper bound for $\|x_{L+1}\|_2$ given by Lemma 5.4.1 and Hoeffding inequality gives a constant $C_1 > 0$ such that with probability at least $1-\delta$, $|f_{W^{(0)}}(x_i)| \leqslant C_1\sqrt{\log(n/\delta)}$ for all $i \in [n]$. Since $\ell(z) = \log(1+\exp(-z)) \leqslant |z| + 1$ for all $z \in \mathbb{R}$, we have

$$L_S(W^{(0)}) = \frac{1}{n}\sum_{i=1}^n \ell(y_i\cdot f_{W^{(0)}}(x_i)) \leqslant 1 + C_1\sqrt{\log\frac{n}{\delta}} \leqslant C\sqrt{\log(n/\delta)}.$$

We can thus bound the distance from initialization by

$$\begin{aligned}\left\| W_l^{(k)} - W_l^{(0)} \right\|_F &\leqslant \eta \sum_{k'=0}^{k-1} \left\| \nabla_{W_l} L_S(W^{(k')}) \right\|_F \\ &\leqslant C\eta\sqrt{m} \sum_{k'=0}^{k-1} \mathcal{E}_S(W^{(k')}) \\ &\leqslant C\eta\sqrt{m}\sqrt{k}\sqrt{\gamma^{-4}\left(\eta m_{L+1}\right)^{-1} \sum_{k'=0}^{k-1} \left(L_S(W^{(k)}) - L_S(W^{(k+1)})\right)} \\ &\leqslant C\sqrt{k\eta} \cdot \gamma^{-2} \left(\log \frac{n}{\delta}\right)^{\frac{1}{4}} \\ &\leqslant \frac{\tau}{2}.\end{aligned}$$

The first line comes from the definition of gradient descent and the triangle inequality. For the second line, (5.6.1) allows us to apply Lemma 5.4.4. The third line follows by Cauchy–Schwarz and (5.6.4). The next line follows by (5.6.5), and the last since $k\eta \leqslant \nu'' \tau^2 \gamma^4 (\log(n/\delta))^{-\frac{1}{2}}$. This completes the induction and shows that $W^{(k)} \in \mathcal{W}(W^{(0)}, \tau)$ for all $k \leqslant K$.

For the second part of the proof, we want to derive an upper bound on the lowest empirical surrogate error over the trajectory of gradient descent. Since we have shown that $W^{(k)} \in \mathcal{W}(W^{(0)}, \tau/2)$ for $k \leqslant K$, (5.6.3) and (5.6.5) both hold. Let $k^* = \mathrm{argmin}_{k \in \{0,\dots,K-1\}} \mathcal{E}_S(W^{(k)})^2$. Then telescoping (5.6.3) over k yields

$$\begin{aligned}L_S(W^{(K)}) - L_S(W^{(0)}) &\leqslant -C \cdot \eta \cdot m_{L+1} \cdot \gamma^4 \cdot \sum_{k=1}^{K} \mathcal{E}_S(W^{(k)})^2 \\ &\leqslant -C \cdot K\eta \cdot m_{L+1} \cdot \gamma^4 \cdot \mathcal{E}_S(W^{(k^*)})^2.\end{aligned}$$

Rearranging the above gives

$$\mathcal{E}_S(W^{(k^*)}) \leqslant C_3 \left(K\eta \cdot m\right)^{-\frac{1}{2}} \left(L_S(W^{(0)})\right)^{\frac{1}{2}} \cdot \gamma^{-2} \leqslant C_3 \left(K\eta \cdot m\right)^{-\frac{1}{2}} \left(\log \frac{n}{\delta}\right)^{\frac{1}{4}} \cdot \gamma^{-2},$$

where we have used that $L_S(\cdot)$ is always nonnegative in the first inequality and (5.6.5) in the second.

5.6.2 Proof of Theorem 5.3.6

Denote $\mathcal{F}(W^{(0)},\tau) = \{f_W(x) : W \in \mathcal{W}(W^{(0)},\tau)\}$, and recall the definition of the empirical Rademacher complexity,

$$\widehat{\mathfrak{R}}_S[\mathcal{F}(W^{(0)},\tau)] = \mathbb{E}_\xi\left[\sup_{f\in\mathcal{F}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i f(x_i)\right] = \mathbb{E}_\xi\left[\sup_{W\in\mathcal{W}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i f(x_i)\right], \tag{5.6.6}$$

where $\xi = (\xi_1,\ldots,\xi_n)^\top$ is an n-dimensional vector of i.i.d. $\xi_i \sim \mathrm{Unif}(\{\pm 1\})$. Since $y \in \{\pm 1\}$, $|\ell'(z)| \leqslant 1$ and $\ell'(\cdot)$ is 1-Lipschitz, standard uniform convergence arguments (see, e.g., [SB14]) yield that with probability at least $1-\delta$,

$$\sup_{W\in\mathcal{W}(W^{(0)},\tau)} |\mathcal{E}_S(W) - \mathcal{E}_\mathcal{D}(W)| \leqslant 2\mathbb{E}_S\widehat{\mathfrak{R}}_S\left[\mathcal{F}(W^{(0)},\tau)\right] + C_1\sqrt{\frac{\log(1/\delta)}{n}}.$$

Since $-\ell'(x) = (1+\exp(-x))^{-1}$ satisfies $-\ell'(x) < \frac{1}{2}$ if and only if $x<0$, Markov's inequality gives

$$\mathbb{P}_{(x,y)\sim D}\left(y\cdot f_W(x) < 0\right) \leqslant 2\mathbb{E}_{(x,y)\sim\mathcal{D}}\left(-\ell'(y\cdot f_W(x))\right) = 2\mathcal{E}_D(W),$$

so that it suffices to get a bound for the empirical Rademacher complexity (5.6.6). If we define

$$F_{W^{(0)},W}(x) := f_{W^{(0)}}(x) + \sum_{l=1}^{L+1} \mathrm{tr}\left[\left(W_l - W_l^{(0)}\right)^\top \nabla_{W_l} f_{W^{(0)}}(x)\right],$$

then since $\sup_{a+b\in A+B}(a+b) \leqslant \sup_{a\in A} a + \sup_{b\in B} b$, we have

$$\widehat{\mathfrak{R}}_S[\mathcal{F}(W^{(0)},\tau)] \leqslant \underbrace{\mathbb{E}_\xi\left[\sup_{W\in\mathcal{W}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i[f(x_i) - F_{W^{(0)},W}(x_i)]\right]}_{I_1}$$
$$+ \underbrace{\mathbb{E}_\xi\left[\sup_{W\in\mathcal{W}(W^{(0)},\tau)} \frac{1}{n}\sum_{i=1}^n \xi_i \sum_{l=1}^{L+1} \mathrm{tr}\left[\left(W_l - W_l^{(0)}\right)^\top \nabla_{W_l} f_{W^{(0)}}(x)\right]\right]}_{I_2}$$

For the I_1 term, we take $\tilde{W} = W^{(0)}$ in Lemma 5.4.2 to get

$$\begin{aligned} |f_W(x) - F_{W^{(0)},W}(x)| &\leqslant C\left[\tau^{\frac{4}{3}}\sqrt{m\log m}(2+L\theta)\right] + C\tau^2\sqrt{m}\,(2+L\theta) \\ &\leqslant C\tau^{\frac{4}{3}}\sqrt{m\log m}. \end{aligned}$$

For I_2, since $\|AB\|_F \leqslant \|A\|_F \|B\|_2$, Lemma 5.4.1 yields for all l and any matrix ξ,

$$\left\|x_l v^\top \cdot \xi\right\|_F \leqslant \left\|x_l v^\top\right\|_F \|\xi\|_2 \leqslant C\sqrt{m}\,\|\xi\|_2\,.$$

Applying this to the gradient of f at initialization given by (5.2.2) and using Lemma 5.4.1, there is a constant C_2 such that

$$\|\nabla_{W_l} f_{W^{(0)}}\|_F \leqslant C_2\theta^{\mathbb{1}(2\leqslant l\leqslant L)}\sqrt{m}. \tag{5.6.7}$$

We can therefore bound I_2 as follows:

$$\begin{aligned} I_2 &\leqslant \frac{\tau}{n}\sum_{l=1}^{L+1}\mathbb{E}_\xi\left\|\sum_{i=1}^n \xi_i \nabla_{W_l} f_{W^{(0)}}(x_i)\right\|_F \\ &\leqslant \frac{\tau}{n}\sum_{l=1}^{L+1}\sqrt{\mathbb{E}\left\|\sum_{i=1}^n \xi_i \nabla_{W_l} f_{W^{(0)}}(x_i)\right\|_F^2} \\ &= \frac{\tau}{n}\sum_{l=1}^{L+1}\sqrt{\sum_{i=1}^n \|\nabla_{W_l} f_{W^{(0)}}(x_i)\|_F^2} \\ &\leqslant C\frac{\tau}{n}\left(\sqrt{nm} + \sum_{l=2}^{L}\sqrt{nm\theta^2} + \sqrt{nm}\right) \\ &\leqslant C\sqrt{\frac{m}{n}}\tau. \end{aligned}$$

The first line above follows since $\operatorname{tr}(A^\top B) \leqslant \|A\|_F \|B\|_F$ and $W \in \mathcal{W}(W^{(0)}, \tau)$. The second comes from Jensen inequality, with the third since $\xi_i^2 = 1$. The fourth line comes from (5.6.7), with the final inequality by the scale of θ. This completes the proof.

5.6.3 Proof of Corollary 5.3.7

We need only specify conditions on $\tau, \eta, K\eta$, and m such that the results of Theorems 5.3.5 and 5.3.6 will hold, and such that each of the four terms in (5.3.1) are of the same scale

ε. To get the two theorems to hold, we need $\tau \leqslant \nu\gamma^{12} (\log m)^{-\frac{3}{2}}$, $\eta \leqslant \nu'(\gamma^4 m^{-1} \wedge \tau m^{-\frac{1}{2}})$, $K\eta \leqslant \nu''\tau^2\gamma^4 (\log(n/\delta))^{-\frac{1}{2}}$, and

$$m \geqslant C\left(\gamma^{-2} d \log\frac{1}{\gamma} \vee \gamma^{-2}\log\frac{L}{\delta} \vee d\log\frac{L}{\delta} \vee \tau^{-\frac{4}{3}} d\log\frac{L}{\tau\delta} \vee \tau^{-\frac{2}{3}}(\log m)^{-1}\log\frac{L}{\delta} \vee \log\frac{n}{\delta}\right).$$

We now find the appropriate scaling by first setting the upper bound for the surrogate loss given in Theorem 5.3.5 to ε and then ensuring τ is such that the inequality required for $K\eta$ is satisfied:

$$C_3 (K\eta m)^{-\frac{1}{2}} (\log(n/\delta))^{\frac{1}{4}} \cdot \gamma^{-2} = \varepsilon, \qquad K\eta = \nu''\gamma^4\tau^2 (\log(n/\delta))^{-\frac{1}{2}}.$$

Substituting the values for $K\eta$ above, we get $C_4 m^{-\frac{1}{2}}\gamma^{-2}\tau^{-1}\sqrt{\log(n/\delta)} = \varepsilon$, so that

$$\tau = C_6\gamma^{-4}\varepsilon^{-1}m^{-\frac{1}{2}}\sqrt{\log(n/\delta)}. \tag{5.6.8}$$

Let $\widehat{m}$ be such that $\nu\gamma^{12} (\log m)^{-\frac{3}{2}} = \tau$, so that $m(\log m)^{-3} = C\nu^{-2}\gamma^{-32} (\log(n/\delta))\, \varepsilon^{-2}$. It is clear that such a $\widehat{m}$ can be written $\widehat{m} = \tilde{\Omega}(\mathrm{poly}(\gamma^{-1})) \cdot \varepsilon^{-2}$. Finally we set

$$m^* = \max\left(\widehat{m}, d\log\frac{mL}{\delta}, \tau^{-\frac{4}{3}}\log\frac{m}{\tau\delta}\right).$$

By (5.6.8) we can write $\tau^{-\frac{4}{3}}\log(m/(\tau\delta)) = \gamma^{\frac{16}{3}} (\log(n/\delta))^{-\frac{2}{3}} \varepsilon^{\frac{4}{3}} m^{\frac{2}{3}} \log\left(m^{3/2}\gamma^4\varepsilon(\log(n/\delta))^{-\frac{1}{2}}/\delta\right)$. Thus we can take

$$m^* = \tilde{\Omega}(\mathrm{poly}(\gamma^{-1})) \cdot \max(d, \varepsilon^{-2}) \cdot \log\frac{1}{\delta}.$$

Using (5.6.8) we see that $K = C\gamma^{-4} (\log(n/\delta))^{\frac{1}{2}} \varepsilon^{-2}$ and $\eta \leqslant \nu'\gamma^4 m^{-1}$ gives the desired forms of K and η as well as the first term of (5.3.1). For the second term of (5.3.1), we again use (5.6.8) to get $\tau^{\frac{4}{3}}\sqrt{m\log m} \leqslant C\gamma^{-\frac{16}{3}} (\log(n/\delta))^{\frac{2}{3}} \varepsilon^{-\frac{4}{3}} m^{-\frac{1}{6}} = R\varepsilon^{-\frac{4}{3}}m^{-\frac{1}{6}}$ where $R = \tilde{O}(\mathrm{poly}(\gamma^{-1}))$. Since $\varepsilon^{-\frac{4}{3}}m^{-\frac{1}{6}} \leqslant \varepsilon$ iff $m \geqslant \varepsilon^{-14}$, this takes care of the second term in (5.3.1). For the third term, we again use (5.6.8) to write $\tau\sqrt{m/n} = C\gamma^{-4}\sqrt{\log(n/\delta)}n^{-\frac{1}{2}}\varepsilon^{-1} \leqslant \varepsilon$, which happens if $\sqrt{n/\log(n/\delta)} \geqslant C\varepsilon^{-2}\gamma^{-4}$, i.e., $n = \tilde{O}(\mathrm{poly}(\gamma^{-1}))\varepsilon^{-4}$. For the final term of (5.3.1), it's clear that $\sqrt{\log(1/\delta)/n} \leqslant \varepsilon$ is satisfied when $n \geqslant C\varepsilon^{-2}\log(1/\delta)$, which is less stringent than the $\tilde{O}(\mathrm{poly}(\gamma^{-1}))\varepsilon^{-4}$ requirement.

5.7 Proofs of Key Lemmas

In this section we provide proofs to the key lemmas discussed in Section 5.4. We shall first provide the technical lemmas needed for their proof, and leave the proofs of the technical lemmas for Appendix 5.8. Throughout this section, we assume that $\theta = 1/\Omega(L)$.

5.7.1 Proof of Lemma 5.4.1: hidden and interlayer activations are bounded

We first recall a standard result from random matrix theory; see, e.g. [Ver10], Corollary 5.35.

Lemma 5.7.1. Suppose $W_1, \ldots, W_{L+1}$ are generated by Gaussian initialization. Then there exist constants $C, C' > 0$ such that for any $\delta > 0$, if $m \geqslant d \vee C \log(L/\delta)$, then with probability at least $1 - \delta$, $\|W_l\|_2 \leqslant C'$ for all $l \in [L+1]$.

The next lemma bounds the spectral norm of the maps that the residual layers define. This is a key result that allows for the simplification of many of the arguments that are needed in non-residual architectures. Its proof is in Appendix 5.8.1.

Lemma 5.7.2. Suppose $W_1, \ldots, W_L$ are generated by Gaussian initialization. Then for any $\delta > 0$, there exist constants C_0, C_0', C such that if $m \geqslant C_0 \log(L/\delta)$, then with probability at least $1 - \delta$, for any $L \geqslant b \geqslant a \geqslant 2$, and for any tuple of diagonal matrices $\tilde{\Sigma}_a, \ldots, \tilde{\Sigma}_b$ satisfying $\left\|\tilde{\Sigma}_i\right\|_2 \leqslant 1$ for each $i = a, \ldots, b$, we have

$$\left\|(I + \theta\tilde{\Sigma}_b W_b^\top)(I + \theta\tilde{\Sigma}_{b-1}W_{b-1}^\top) \cdot \ldots \cdot (I + \theta\tilde{\Sigma}_a W_a^\top)\right\|_2 \leqslant \exp(C_0'\theta L) \leqslant 1.01. \tag{5.7.1}$$

In particular, if we consider $\tilde{\Sigma}_i = \Sigma_i(x)$ for any $x \in S^{d-1}$, we have with probability at least $1 - \delta$, for all $2 \leqslant a \leqslant b \leqslant L$ and for all $x \in S^{d-1}$,

$$\left\|(I + \theta\Sigma_b(x)W_b^\top)(I + \theta\Sigma_{b-1}(x)W_{b-1}^\top) \cdot \ldots \cdot (I + \theta\Sigma_a(x)W_a^\top)\right\|_2 \leqslant \exp(C_0'\theta L) \leqslant 1.01.$$

The next lemma we show concerns a Lipschitz property of the map $x \mapsto x_l$. Compared with the fully connected case, our Lipschitz constant does not involve any terms growing with L, which allows for the width dependence of our result to be only logarithmic in L. Its proof is in Appendix 5.8.2.

Lemma 5.7.3. Suppose $W_1, \dots, W_L$ are generated by Gaussian initialization. There are constants $C, C' > 0$ such that for any $\delta > 0$, if $m \geqslant C d \log(mL/\delta)$, then with probability at least $1 - \delta$, $\|x_l - x_l'\|_2 \leqslant C' \|x - x'\|_2$ for all $x, x' \in S^{d-1}$ and $l \in [L+1]$.

With the above technical lemmas in place, we can proceed with the proof of Lemma 5.4.1.

Proof of Lemma 5.4.1. We first show that a bound of the form $\underline{C} \leqslant \|\widehat{x}_l\|_2 \leqslant \overline{C}$ holds for all $\widehat{x}$ in an ε-net of S^{d-1} and then use the Lipschitz property from Lemma 5.7.3 to lift this result to all of S^{d-1}.

Let $\mathcal{N}^*$ be a τ_0-net of S^{d-1}. By applying Lemma A.6 of [CG20] to the first layer of our network, there exists a constant C_1 such that with probability at least $1 - \delta/3$, we can take $m = \Omega\left(d \log\left(m/(\tau_0\delta)\right)\right)$ large enough so that

$$\|\widehat{x}_1\|_2 \leqslant 1 + C_1\sqrt{\frac{d \log\left(m/(\tau_0\delta)\right)}{m}} \leqslant 1.004.$$

If $2 \leqslant l \leqslant L$, by an application of Lemma 5.7.2, by taking m larger we have with probability at least $1 - \delta/3$, for all $2 \leqslant l \leqslant L, \widehat{x} \in \mathcal{N}^*$,

$$\begin{aligned}\|\widehat{x}_l\|_2 &= \left\|(I + \theta\Sigma_l(\widehat{x})W_l^\top)\cdots(I + \theta\Sigma_2(\widehat{x})W_2^\top)\Sigma_1(\widehat{x})W_1^\top\widehat{x}\right\|_2 \\ &\leqslant \left\|(I + \theta\Sigma_l(\widehat{x})W_l^\top)\cdots(I + \theta\Sigma_2(\widehat{x})W_2^\top)\right\|_2 \|\widehat{x}_1\|_2 \\ &\leqslant 1.01 \cdot \left(1 + C_1\sqrt{\frac{d \log\left(m/(\tau_0\delta)\right)}{m}}\right) \leqslant 1.015.\end{aligned}$$

For the last fully connected layer, we can use a proof similar to that of Lemma A.6 in [CG20] using the above upper bound on $\|\widehat{x}_L\|_2$ to get that with probability at least $1 - \delta$, for any $l \in [L+1]$ and $\widehat{x} \in \mathcal{N}^*$,

$$\|\widehat{x}_l\|_2 \leqslant 1.02. \tag{5.7.2}$$

For any $x \in S^{d-1}$, there exists $\widehat{x} \in \mathcal{N}^*$ such that $\|x - \widehat{x}\|_2 \leqslant \tau_0$. By Lemma 5.7.3, this means that with probability at least $1 - \delta/2$, $\|x_l - \widehat{x}_l\|_2 \leqslant C_1 \tau_0$ for some $C_1 > 0$, and this holds over all $\widehat{x} \in \mathcal{N}^*$. Let $\tau_0 = 1/m$, so that $d \log\left(mL/(\tau_0 \delta)\right) \leqslant 2d \log(mL/\delta)$. Then (5.7.2) yields that with probability at least $1 - \delta$, for all $x \in S^{d-1}$ and all $l \in [L+1]$,

$$\|x_l\|_2 \leqslant \|\widehat{x}_l\|_2 + \|x_l - \widehat{x}_l\|_2 \leqslant 1.02 + C_1/m \leqslant 1.024.$$

As for the lower bound, we again let $\mathcal{N}^*$ be an arbitrary τ_0-net of S^{d-1}. For $l = 1$, we use Lemma A.6 in [CG20] to get constants C, C' such that provided $m \geqslant Cd \log\left(m/(\tau_0 \delta)\right)$, then we have with probability at least $1 - \delta/3$, for all $\widehat{x} \in \mathcal{N}^*$,

$$\|\widehat{x}_l\|_2 \geqslant 1 - C' \sqrt{dm^{-1} \log\left(3m/(\tau_0 \delta)\right)} \qquad (l = 1, 2, \ldots, L). \tag{5.7.3}$$

To see that the above holds for layers $2 \leqslant l \leqslant L$, we note that it deterministically holds that $\widehat{x}_{l,j} \geqslant \widehat{x}_{1,j}$ for such l and all j. For the final layer, we follow a proof similar to Lemma A.6 of [CG20] with an application of (5.7.2) to get that with probability at least $1 - \delta/3$,

$$\|\widehat{x}_{L+1}\|_2^2 \geqslant \|\widehat{x}_L\|_2^2 - C_3 \sqrt{dm^{-1} \log\left(3/(\tau_0 \delta)\right)}.$$

Thus $m = \Omega(d \log(m/(\tau_0 \delta))$ implies there is a constant C_4 such that with probability at least $1 - \delta$, for all $l \in [L+1]$ and $\widehat{x} \in \mathcal{N}^*$,

$$\|\widehat{x}_l\|_2 \geqslant C_4 > 0. \tag{5.7.4}$$

By Lemma 5.7.3, we have with probability at least $1 - \delta$, for all $x \in S^{d-1}$,

$$\|x_l\|_2 \geqslant \|\widehat{x}_l\|_2 - \|x_l - \widehat{x}_l\|_2 \geqslant C_4 - C_1 \tau_0.$$

Thus by taking τ_0 to be a sufficiently small universal constant, we get the desired lower bound.

We now demonstrate the upper bound for $\left\|H_l^{l'}\right\|_2$. Since $H_l^{l'} = x_{l'}$ when $l = 1$, we need only consider the case $l > 1$. If $l' \leqslant L$, then $H_l^{l'}$ appears in the bound for Lemma 5.7.2 and so we are done. For $l' = L+1$, by Lemmas 5.7.1 and 5.7.2 we have

$$\begin{aligned}
\left\|H_l^{L+1}\right\|_2 &= \left\|\Sigma_{L+1}(x)W_{L+1}^\top \prod_{r=l}^{L}\left(I + \theta\Sigma_r(x)W_r^\top\right)\right\|_2 \\
&\leqslant \left\|\Sigma_{L+1}(x)\right\|_2 \left\|W_{L+1}\right\|_2 \left\|\prod_{r=l}^{L}\left(I + \theta\Sigma_r(x)W_r^\top\right)\right\|_2 \leqslant C.
\end{aligned}$$

□

5.7.2 Proof of Lemma 5.4.2: semismoothness

To prove the semismoothness result, we need two technical lemmas. The first lemma concerns a Lipschitz-type property with respect to the weights, along with a characterization of the changing sparsity patterns of the rectifier activations at each layer. The second lemma characterizes how the neural network output behaves if we know that one of the initial layers has a given sparsity pattern. This allows us to develop the desired semi-smoothness even though ReLU is non-differentiable. The proof for Lemmas 5.7.4 and 5.7.5 can be found in Appendix 5.8.3 and 5.8.4, respectively.

Lemma 5.7.4. Let $W = (W_1, \ldots, W_{L+1})$ be generated by Gaussian initialization, and let $\widehat{W} = (\widehat{W}_1, \ldots, \widehat{W}_{L+1}), \tilde{W} = (\tilde{W}_1, \ldots, \tilde{W}_{L+1})$ be weight matrices such that $\widehat{W}, \tilde{W} \in \mathcal{W}(W, \tau)$. For $x \in S^{d-1}$, let $\Sigma_l(x), \widehat{\Sigma}_l(x), \tilde{\Sigma}_l(x)$ and $x_l, \widehat{x}_l, \tilde{x}_l$ be the binary matrices and hidden layer outputs of the l-th layers with parameters $W, \widehat{W}, \tilde{W}$ respectively. There exist absolute constants C_1, C_2, C_3 such that for any $\delta > 0$, if $m \geqslant C_1\tau^{-\frac{4}{3}} \cdot d\log(m/(\tau\delta)) \vee C_1 d\log(mL/\delta)$, then with probability at least $1-\delta$, for any $x \in S^{d-1}$ and any $l \in [L+1]$, we have

$$\left\|\widehat{x}_l - \tilde{x}_l\right\|_2 \leqslant \begin{cases} C_2\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2, & l = 1, \\ C_2\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta C_2\sum_{r=2}^{l}\left\|\widehat{W}_r - \tilde{W}_r\right\|_2, & 2 \leqslant l \leqslant L, \\ C_2\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta C_2\sum_{r=2}^{L}\left\|\widehat{W}_r - \tilde{W}_r\right\|_2 + C_2\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2, & l = L+1. \end{cases}$$

and

$$\left\|\widehat{\Sigma}_l(x) - \tilde{\Sigma}_l(x)\right\|_0 \leqslant C_3 m\tau^{\frac{2}{3}}.$$

Lemma 5.7.5. Let $W_1, \ldots, W_{L+1}$ be generated by Gaussian initialization. Let $\tilde{W}_l$ be such that $\left\|W_l - \tilde{W}_l\right\|_2 \leqslant \tau$ for all l, and let $\tilde{\Sigma}_l(x)$ be the diagonal activation matrices corresponding to $\tilde{W}_l$, and $\tilde{H}_l^{l'}(x)$ the corresponding interlayer activations defined in (5.2.1). Suppose that $\left\|\tilde{\Sigma}_l(x) - \Sigma_l(x)\right\|_0 \leqslant s$ for all $x \in S^{d-1}$ and all l. Define, for $l \geqslant 2$ and $a \in \mathbb{R}^{m_{l-1}}$,

$$g_l(a, x) := v^\top \tilde{H}_l^{L+1}(x) a.$$

Then there exists a constant $C > 0$ such that for any $\delta > 0$, provided $m \geqslant C\tau^{-\frac{2}{3}}(\log m)^{-1}\log(L/\delta)$, we have with probability at least $1 - \delta$ and all $2 \leqslant l \leqslant L+1$,

$$\sup_{\|x\|_2 = \|a\|_2 = 1,\ \|a\|_0 \leqslant s} |g_l(a, x)| \leqslant C_1 \left[\tau\sqrt{m} + \sqrt{s \log m}\right].$$

In comparison with the fully connected case of [CG20], our bounds in Lemmas 5.7.4 and 5.7.5 do not involve polynomial terms in L, and the residual scaling θ further scales the dependence of the hidden layer activations on the intermediate layers.

With the above two technical lemmas, we can proceed with the proof of Lemma 5.4.2.

Proof of semismoothness, Lemma 5.4.2. Recalling the notation of interlayer activations $H_l^{l'}$ from (5.2.1), we have for any $l \in [L+1]$ $f_{\widehat{W}}(x) = v^\top \widehat{H}_{l+1}^{L+1} \widehat{x}_l$, where we have denoted $H_l^{l'}(x) = H_l^{l'}$ for notational simplicity. Similarly, in what follows we denote $\Sigma(x)$ by Σ with the understanding that each diagonal matrix Σ still depends on x. We have the decomposition

$$\widehat{H}_2^{L+1}\widehat{\Sigma}_1\widehat{W}_1 x = \left(\widehat{H}_2^{L+1} - \tilde{H}_2^{L+1}\right)\widehat{\Sigma}_1\widehat{W}_1^\top x + \tilde{H}_2^{L+1}\widehat{\Sigma}_1\widehat{W}_1^\top x,$$

and for $2 \leqslant l \leqslant L$,

$$\widehat{H}_l^{L+1} - \tilde{H}_l^{L+1} = \left(\widehat{H}_{l+1}^{L+1} - \tilde{H}_{l+1}^{L+1}\right)\left(I + \theta\widehat{\Sigma}_l\widehat{W}_l^\top\right) + \theta\tilde{H}_{l+1}^{L+1}\left(\widehat{\Sigma}_l\widehat{W}_l^\top - \tilde{\Sigma}_l\tilde{W}_l^\top\right).$$

Thus we can write

$$\begin{aligned}\widehat{H}_1^{L+1}(x) - \tilde{H}_1^{L+1}(x) &= \left(\widehat{H}_2^{L+1} - \tilde{H}_2^{L+1}\right)\widehat{\Sigma}_1\widehat{W}_1^\top x + \tilde{H}_2^{L+1}\left(\widehat{\Sigma}_1\widehat{W}_1^\top - \tilde{\Sigma}_1\tilde{W}_1^\top\right)x \\ &= \left(\widehat{\Sigma}_{L+1}\widehat{W}_{L+1}^\top - \tilde{\Sigma}_{L+1}\tilde{W}_{L+1}^\top\right)\widehat{x}_L \\ &\quad + \theta\sum_{l=2}^{L}\tilde{H}_{l+1}^{L+1}\left(\widehat{\Sigma}_l\widehat{W}_l^\top - \tilde{\Sigma}_l\tilde{W}_l^\top\right)\widehat{x}_{l-1} + \tilde{H}_2^{L+1}\left(\widehat{\Sigma}_1\widehat{W}_1 - \tilde{\Sigma}_1\tilde{W}_1\right)x.\end{aligned}$$

We thus want to bound the quantity

$$\begin{aligned}f_{\widehat{W}}(x) - f_{\tilde{W}}(x) = v^\top &\left(\widehat{\Sigma}_{L+1}\widehat{W}_{L+1}^\top - \tilde{\Sigma}_{L+1}\tilde{W}_{L+1}^\top\right)\widehat{x}_L && (T_1) \\ &+ \theta v^\top\left[\sum_{l=2}^{L}\tilde{H}_{l+1}^{L+1}\left(\widehat{\Sigma}_l\widehat{W}_l^\top - \tilde{\Sigma}_l\tilde{W}_l^\top\right)\widehat{x}_{l-1}\right] && (T_2) \\ &+ v^\top\left[\tilde{H}_2^{L+1}\left(\widehat{\Sigma}_1\widehat{W}_1 - \tilde{\Sigma}_1\tilde{W}_1\right)x\right]. && (T_3) \qquad (5.7.5)\end{aligned}$$

We deal with the three terms separately. The idea in each is the same.

<u>First term, T_1.</u> We write this as the sum of three terms $v^\top(I_1 + I_2 + I_3)$, where

$$\begin{aligned}&\left(\widehat{\Sigma}_{L+1}\widehat{W}_{L+1}^\top - \tilde{\Sigma}_{L+1}\tilde{W}_{L+1}^\top\right)\widehat{x}_L \\ &= \underbrace{\left(\widehat{\Sigma}_{L+1} - \tilde{\Sigma}_{L+1}\right)\widehat{W}_{L+1}^\top\widehat{x}_L}_{I_1} + \underbrace{\tilde{\Sigma}_{L+1}\left(\widehat{W}_{L+1}^\top - \tilde{W}_{L+1}^\top\right)(\widehat{x}_L - \tilde{x}_L)}_{I_2} + \underbrace{\tilde{\Sigma}_{L+1}\left(\widehat{W}_{L+1}^\top - \tilde{W}_{L+1}^\top\right)\tilde{x}_L}_{I_3}.\end{aligned} \tag{5.7.6}$$

By directly checking the signs of the diagonal matrices, we can see that for any $l = 1, \dots, L+1$,

$$\left\|\left(\widehat{\Sigma}_l - \tilde{\Sigma}_l\right)\widehat{W}_l^\top\widehat{x}_{l-1}\right\|_2 \leqslant C_1\left\|\widehat{W}_l - \tilde{W}_l\right\|_2 + C_1\left\|\widehat{x}_{l-1} - \tilde{x}_{l-1}\right\|_2. \tag{5.7.7}$$

We will use Lemma 5.7.4 to get specific bounds for each l. Denote $|\Sigma|$ as the entrywise absolute values of a diagonal matrix Σ, so that $|\Sigma|\Sigma = \Sigma$ provided the diagonal entries are all in $\{0, \pm 1\}$. Then we can write

$$\begin{aligned}|v^\top I_1| &= \left\|v^\top\left|\widehat{\Sigma}_{L+1} - \tilde{\Sigma}_{L+1}\right|\left(\widehat{\Sigma}_{L+1} - \tilde{\Sigma}_{L+1}\right)\widehat{W}_{L+1}^\top\widehat{x}_L\right\|_2 \\ &\leqslant C_3\tau^{\frac{1}{3}}\sqrt{m}\left\|\left(\widehat{\Sigma}_{L+1} - \tilde{\Sigma}_{L+1}\right)\widehat{W}_{L+1}^\top\widehat{x}_L\right\|_2 \\ &\leqslant C_3\tau^{\frac{1}{3}}\sqrt{m}\cdot\left(C_1\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 + C_1\left\|\widehat{x}_L - \tilde{x}_L\right\|_2\right)\end{aligned} \tag{5.7.8}$$

The first inequality follows by first noting that for any vector a with $|a_i| \leqslant 1$ it holds that $\left\|v^\top a\right\|_2 \leqslant \|a\|_0^{\frac{1}{2}}$, and then applying Lemma 5.7.4 to get $\left\|\widehat{\Sigma}_{L+1} - \tilde{\Sigma}_{L+1}\right\|_0 \leqslant s = O\left(m\tau^{\frac{2}{3}}\right)$. The last line is by (5.7.7).

The I_2 term in (5.7.6) follows from a simple application of Cauchy–Schwarz:

$$|v^\top I_2| \leqslant \sqrt{m}\cdot C\cdot \left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 \|\widehat{x}_L - \tilde{x}_L\|_2 . \tag{5.7.9}$$

Putting together (5.7.8) and (5.7.9) shows that we can bound T_1 in (5.7.5) by

$$\begin{aligned}
T_1 &\leqslant C_3\tau^{\frac{1}{3}}\sqrt{m}\cdot\left(C_1\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 + C_1\|\widehat{x}_L - \tilde{x}_L\|_2\right) + \sqrt{m}\cdot C\cdot\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 \|\widehat{x}_L - \tilde{x}_L\|_2 \\
&\qquad + v^\top\tilde{\Sigma}_{L+1}\left(\widehat{W}_{L+1} - \tilde{W}_{L+1}\right)^\top \tilde{x}_L \\
&\leqslant C_3\tau^{\frac{1}{3}}\sqrt{m}\left(C_1\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 + C_1'\left[\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta\sum_{r=2}^{L}\left\|\widehat{W}_r - \tilde{W}_r\right\|_2\right]\right) \\
&\qquad + C\sqrt{m}\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2\left(\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta\sum_{r=2}^{L}\left\|\tilde{W}_r - \widehat{W}_r\right\|_2\right) \\
&\qquad + v^\top\tilde{\Sigma}_{L+1}\left(\widehat{W}_{L+1} - \tilde{W}_{L+1}\right)^\top \tilde{x}_L .
\end{aligned} \tag{5.7.10}$$

<u>**Second term, T_2.**</u> We again use a decomposition like (5.7.6):

$$\begin{aligned}
&\tilde{H}_{l+1}^{L+1}\left(\widehat{\Sigma}_l\widehat{W}_l^\top - \tilde{\Sigma}_l\tilde{W}_l^\top\right)\widehat{x}_{l-1} \\
&= \underbrace{\tilde{H}_{l+1}^{L+1}\left(\widehat{\Sigma}_l - \tilde{\Sigma}_l\right)\widehat{W}_l^\top\widehat{x}_{l-1}}_{I_1} + \underbrace{\tilde{H}_{l+1}^{L+1}\tilde{\Sigma}_l\left(\widehat{W}_l^\top - \tilde{W}_l^\top\right)(\widehat{x}_{l-1} - \tilde{x}_{l-1})}_{I_2} + \underbrace{\tilde{H}_{l+1}^{L+1}\tilde{\Sigma}_l\left(\widehat{W}_l^\top - \tilde{W}_l^\top\right)\tilde{x}_{l-1}}_{I_3} .
\end{aligned} \tag{5.7.11}$$

For I_1, we note that Lemma 5.7.4 gives sparsity level $s = O(m\tau^{\frac{2}{3}})$ for $\widehat{\Sigma}_l - \tilde{\Sigma}_l$. We thus proceed similarly as for the term T_1 to get

$$\begin{aligned}
|v^\top I_1| &\leqslant \left\|v^\top\tilde{\Sigma}_{L+1}\tilde{W}_{L+1}^\top\tilde{H}_{l+1}^{L}\left|\widehat{\Sigma}_l - \tilde{\Sigma}_l\right|\left(\widehat{\Sigma}_l - \tilde{\Sigma}_l\right)\widehat{W}_l^\top\widehat{x}_{l-1}\right\|_2 \\
&\leqslant C\tau^{\frac{1}{3}}\sqrt{m\log m}\cdot\left(C_1\left\|\widehat{W}_l - \tilde{W}_l\right\|_2 + C_2\|\widehat{x}_{l-1} - \tilde{x}_{l-1}\|_2\right).
\end{aligned}$$

The above follows since $s \log m \geqslant C \log(L/\delta)$ holds for $s = m\tau^{\frac{2}{3}}$, and we can hence apply Lemma 5.7.5 and (5.7.7). The bound for the I_2 term again follows by Cauchy–Schwarz,

$$|v^\top I_2| \leqslant \sqrt{m} \cdot C \cdot \left\| \widehat{W}_l - \tilde{W}_l \right\|_2 \left\| \widehat{x}_{l-1} - \tilde{x}_{l-1} \right\|_2 .$$

Thus, for the term T_2 in (5.7.5) we have

$$\begin{aligned}
T_2 \leqslant \theta \sum_{l=2}^{L} & \left(C_6 \tau^{\frac{1}{3}} \sqrt{m \log m} \left\| \widehat{W}_l - \tilde{W}_l \right\|_2 + C\tau^{\frac{1}{3}} \sqrt{m \log m} \left\| \widehat{W}_1 - \tilde{W}_1 \right\|_2 \right) \\
&+ \theta^2 \sum_{l=2}^{L} \left(\tau^{\frac{1}{3}} \sqrt{m \log m} \sum_{r=2}^{l} \left\| \tilde{W}_r - \widehat{W}_r \right\|_2 \right) \\
&+ \theta \sum_{r=2}^{L} \sqrt{m} C \left\| \widehat{W}_l - \tilde{W}_l \right\|_2 \left(\left\| \widehat{W}_1 - \tilde{W}_1 \right\|_2 + \theta \sum_{r=l}^{2} \left\| \widehat{W}_r - \tilde{W}_r \right\|_2 \right) \\
&+ \theta \sum_{l=2}^{L} v^\top \tilde{H}_{l+1}^{L+1} \tilde{\Sigma}_l \left(\widehat{W}_l^\top - \tilde{W}_l^\top \right) \tilde{x}_{l-1}.
\end{aligned} \tag{5.7.12}$$

Third term, T_3. For T_3, we work on the quantity

$$\tilde{H}_2^{L+1} \left(\widehat{\Sigma}_1 \widehat{W}_1^\top - \tilde{\Sigma}_1 \tilde{W}_1^\top \right) x = \tilde{H}_2^{L+1} \left(\widehat{\Sigma}_1 - \tilde{\Sigma}_1 \right) \widehat{W}_1^\top x + \tilde{H}_2^{L+1} \tilde{\Sigma}_1 \left(\widehat{W}_1 - \tilde{W}_1 \right) x.$$

Thus, we again have by Lemma 5.7.5,

$$\begin{aligned}
T_3 &\leqslant \left\| v^\top \tilde{H}_2^{L+1} \left| \widehat{\Sigma}_1 - \tilde{\Sigma}_1 \right| \right\|_2 \left\| \left(\widehat{\Sigma}_1 - \tilde{\Sigma}_1 \right) \widehat{W}_1 x \right\|_2 + v^\top \tilde{H}_2^{L+1} \tilde{\Sigma}_1 \left(\widehat{W}_1 - \tilde{W}_1 \right) x \\
&\leqslant \tau^{\frac{1}{3}} \sqrt{m \log m} \left\| \widehat{W}_1 - \tilde{W}_1 \right\|_2 + v^\top \tilde{H}_2^{L+1} \tilde{\Sigma}_1 \left(\widehat{W}_1 - \tilde{W}_1 \right) x.
\end{aligned} \tag{5.7.13}$$

Using the linearity of the trace operator and that $\operatorname{tr}(ABC) = \operatorname{tr}(CAB) = \operatorname{tr}(BCA)$ for any matrices A, B, C for which those products are defined, we can use the gradient formula (5.2.2) to calculate for any $l \in [L+1]$,

$$\theta^{\mathbb{1}(2 \leqslant l \leqslant L)} v^\top \tilde{H}_l^{L+1} \tilde{\Sigma}_l \left(\widehat{W}_l - \tilde{W}_l \right)^\top \tilde{x}_{l-1} = \operatorname{tr} \left[\left(\widehat{W}_l - \tilde{W}_l \right)^\top \nabla_{W_l} f_{\tilde{W}}(x) \right]. \tag{5.7.14}$$

Let now

$$h(\widehat{W}, \tilde{W}) := \left\| \widehat{W}_1 - \tilde{W}_1 \right\|_2 + \theta \sum_{l=2}^{L} \left\| \widehat{W}_l - \tilde{W}_l \right\|_2 + \left\| \widehat{W}_{L+1} - \tilde{W}_{L+1} \right\|_2 .$$

Substituting the bounds from (5.7.10), (5.7.12), (5.7.13) and (5.7.14) thus yield for some constant $\overline{C}$,

$$
\begin{aligned}
f_{\widehat{W}}(x) - f_{\tilde{W}}(x) \leqslant C\tau^{\frac{1}{3}}\sqrt{m\log m}&\left[\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta C\sum_{l=2}^{L}\left\|\widehat{W}_l - \tilde{W}_l\right\|_2 + C\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2\right] \\
&+ C\tau^{\frac{1}{3}}\sqrt{m\log m}\left[\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + C\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta C\sum_{l=2}^{l}\left\|\widehat{W}_l - \tilde{W}_l\right\|_2\right] \\
&+ C\sqrt{m}\Bigg[\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 \cdot \left\|\widehat{W}_1 - \tilde{W}_{L+1}\right\|_2 + \theta\left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2\sum_{r=2}^{L}\left\|\widehat{W}_r - \tilde{W}_r\right\|_2 \\
&+ \theta\sum_{l=2}^{L}\left\|\widehat{W}_l - \tilde{W}_l\right\|_2\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + \theta\sum_{l=2}^{L}\left\|\widehat{W}_l - \tilde{W}_l\right\|_2 \cdot \left(\theta\sum_{r=2}^{l}\left\|\widehat{W}_r - \tilde{W}_r\right\|_2\right)\Bigg] \\
&+ \sum_{l=1}^{L+1}\mathrm{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)\nabla_{W_l} f_{\tilde{W}}(x)\right] \\
\leqslant \overline{C}\tau^{\frac{1}{3}}\sqrt{m\log m}\cdot h(\widehat{W},\tilde{W}) &+ \overline{C}\sqrt{m}\cdot h(\widehat{W},\tilde{W})^2 + \sum_{l=1}^{L+1}\mathrm{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)\nabla_{W_l} f_{\tilde{W}}(x)\right] \quad (5.7.15)
\end{aligned}
$$

This completes the proof of semi-smoothness of f_W. For L_S, denote $\widehat{y}_i, \tilde{y}_i$ as the outputs of the network for input x_i under weights $\widehat{W}, \tilde{W}$ respectively. Since $\ell''(z) \leqslant 0.5$ for all $z \in \mathbb{R}$, if we denote $\Delta_i = \widehat{y}_i - \tilde{y}_i = f_{\widehat{W}}(x_i) - f_{\tilde{W}}(x_i)$, we have

$$
L_S(\widehat{W}) - L_S(\tilde{W}) \leqslant \frac{1}{n}\sum_{i=1}^{n}\left[\ell'(y_i\tilde{y}_i)\cdot y_i\cdot\Delta_i + \frac{1}{4}\Delta_i^2\right].
$$

Applying (5.7.15) and using that $-n^{-1}\sum_{i=1}^{n}\ell'(z_i) \leqslant 1$ for any $z_i \in \mathbb{R}$,

$$
\begin{aligned}
\frac{1}{n}\sum_{i=1}^{n}\ell'(y_i\tilde{y}_i)y_i\cdot\Delta_i \leqslant C\tau^{\frac{1}{3}}\sqrt{m\log m}\cdot h(\widehat{W},\tilde{W})\cdot\mathcal{E}_S(\tilde{W}) + C\sqrt{m}\cdot h(\widehat{W},\tilde{W})^2\cdot\mathcal{E}_S(\tilde{W}) \\
+ \sum_{l=1}^{L+1}\frac{1}{n}\sum_{i=1}^{n}\ell'(y_i\tilde{y}_i)\cdot y_i\cdot\mathrm{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)\nabla_{W_l} f_{\tilde{W}}(x_i)\right].
\end{aligned}
$$

Linearity of the trace operator allows the last term in the above display to be written as

$$
\sum_{l=1}^{L+1}\mathrm{tr}\left[\left(\widehat{W}_l - \tilde{W}_l\right)\nabla_{W_l} L_S(\tilde{W})\right].
$$

Moreover, using Lemma 5.7.4,

$$
\Delta_i^2 = \left[v^\top(\widehat{x}_{L+1,i} - \tilde{x}_{L+1,i})\right]^2 \leqslant \|v\|_2^2\,\|\widehat{x}_{L+1,i} - \tilde{x}_{L+1,i}\|_2^2 \leqslant C_2\cdot m\cdot h(\widehat{W},\tilde{W})^2.
$$

This term dominates the corresponding h^2 term coming from Δ_i and so completes the proof.

□

5.7.3 Proof of Lemma 5.4.3: gradient lower bound

This is the part of the proof that makes use of the assumption on the data distribution given in Assumption 5.3.2, and is key to the mild overparameterization required for our generalization result. The key technical lemma needed for the proof of the gradient lower bound is given below. The proof of Lemma 5.7.6 can be found in Appendix 5.8.5.

Lemma 5.7.6. Let $a(x,y) : S^{d-1} \times \{\pm 1\} \to [0,1]$. For any $\delta > 0$, there is a constant $C > 0$ such that if $m \geqslant C\gamma^{-2}\left(d\log(1/\gamma) + \log(L/\delta)\right)$ and $m \geqslant C\log(n/\delta)$ then for any such function a, we have with probability at least $1-\delta$,

$$\sum_{j=1}^{m_{L+1}} \left\| \frac{1}{n}\sum_{i=1}^n \left[a(x_i,y_i)\cdot y_i \cdot \sigma'\left(w_{L+1,j}^\top x_{L,i}\right)\cdot x_{L,i}\right]\right\|_2^2 \geqslant \frac{1}{67} m_{L+1}\gamma^2 \left(\frac{1}{n}\sum_{i=1}^n a(x_i,y_i)\right)^2.$$

Proof of Lemma 5.4.3. Let $\tilde y_i := f_{\tilde W}(x_i)$, and define $g_j := \frac{1}{n}\sum_{i=1}^n \left[\ell'(y_i\tilde y_i)\cdot v_j \cdot y_i \cdot \sigma'(w_{L+1,j}^\top x_{L,i})\cdot x_{L,i}\right]$ so that

$$\sum_{j=1}^{m_{L+1}} \|g_j\|_2^2 = \sum_{j=1}^{m_{L+1}} \left\| \frac{1}{n}\sum_{i=1}^n \left[\ell'(y_i\tilde y_i)\cdot y_i \cdot \sigma'(w_{L+1,j}^\top x_{L,i})\cdot x_{L,i}\right]\right\|_2^2 .$$

Recall that $\mathcal{E}_S(\tilde W) = -n^{-1}\sum_{i=1}^n \ell'(y_i\tilde y_i)$. Applying Lemma 5.7.6 gives

$$\sum_{j=1}^{m_{L+1}} \|g_j\|_2^2 \geqslant \frac{1}{67} m_{L+1}\gamma^2 [\mathcal{E}_S(\tilde W)]^2. \tag{5.7.16}$$

By Lemma 5.4.1, for any $j \in [m_{L+1}]$, we have

$$\|g_j\|_2 \leqslant \frac{1}{n}\sum_{i=1}^n \left\|\ell'(y_i\tilde y_i)\cdot v_j \cdot y_i \cdot \sigma'(w_{L+1,j}^\top x_{L,i})\cdot x_{L,i}\right\|_2 \leqslant 1.02\mathcal{E}_S(\tilde W). \tag{5.7.17}$$

Define

$$A := \left\{ j \in [m_{L+1}] : \|g_j\|_2^2 \geqslant \frac{1}{2\cdot 67}\gamma^2 \left(\mathcal{E}_S(\tilde W))\right)^2 \right\}.$$

We can get the following lower bound on $|A|$:

$$\begin{aligned}
|A|\mathcal{E}_S(\tilde{W})^2 &\geqslant \frac{1}{1.02^2}\sum_{j\in A}\|g_j\|_2^2 \\
&\geqslant \frac{1}{1.05}\left(\frac{1}{67}m_{L+1}\gamma^2[\mathcal{E}_S(\tilde{W})]^2 - \frac{1}{2\cdot 67}|A^c|\gamma^2[\mathcal{E}_S(\tilde{W})]^2\right) \\
&\geqslant \frac{1}{1.05\cdot 2\cdot 67}m_{L+1}\gamma^2[\mathcal{E}_S(\tilde{W})]^2.
\end{aligned}$$

The first line follows by (5.7.17), and the second by writing the sum over $[m_{L+1}]$ as a sum over A and A^c and then (5.7.16) and the definition of A. The last line holds since $|A^c| \leqslant m_{L+1}$, and all of the above allows for the bound

$$|A| \geqslant \frac{1}{141}m_{L+1}\gamma^2. \tag{5.7.18}$$

Let now $A' = \{j \in [m_{L+1}] : \sigma'(\tilde{w}_{L+1,j}^\top \tilde{x}_{L,i}) \neq \sigma'(w_{L+1,j}^\top x_{L,i})\}$. By Lemma 5.7.4, we have

$$|A'| = \left\|\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right\|_0 \leqslant C_1\tau^{\frac{2}{3}}m_{L+1}. \tag{5.7.19}$$

Since $\tau \leqslant \nu\gamma^3$, we can make ν small enough so that $C_1\tau^{\frac{2}{3}} < \gamma^2\cdot(1/141 - 1/150)$. Thus (5.7.18) and (5.7.19) imply

$$|A\backslash A'| \geqslant |A| - |A'| \geqslant \frac{1}{141}m_{L+1}\gamma^2 - C_1\tau^{\frac{2}{3}}m_{L+1} \geqslant \frac{1}{150}m_{L+1}\gamma^2. \tag{5.7.20}$$

By definition, $\nabla_{W_{L+1,j}}L_S(\tilde{W}) = \frac{1}{n}\sum_{i=1}^n \ell'(y_i\tilde{y}_i)\cdot v_j\cdot y_i\cdot\sigma'(\tilde{w}_{L+1,j}^\top\tilde{x}_{L,i})\cdot\tilde{x}_{L,i}$. For indices $j \in A\backslash A'$, we can therefore write

$$\begin{aligned}
\|g_j\|_2 - \left\|\nabla_{W_{L+1,j}}L_S(\tilde{W})\right\|_2 &\leqslant \left\|\frac{1}{n}\sum_{i=1}^n \ell'(y_i\tilde{y}_i)\cdot v_j\cdot y_i\cdot\sigma'(w_{L+1,j}^\top x_{L,i})\cdot(x_{L,i}-\tilde{x}_{L,i})\right\|_2 \\
&\leqslant \frac{1}{n}\sum_{i=1}^n \left\|\ell'(y_i\tilde{y}_i)\cdot v_j\cdot y_i\cdot\sigma'(w_{L+1,j}^\top x_{L,i})\cdot(x_{L,i}-\tilde{x}_{L,i})\right\|_2 \\
&\leqslant C_3\tau\mathcal{E}_S(\tilde{W}). \qquad (5.7.21)
\end{aligned}$$

The first inequality follows by the triangle inequality and since indices $j \in A\backslash A'$ satisfy $\sigma'(\tilde{w}_{L+1,j}^\top\tilde{x}_{L,i}) = \sigma(w_{L+1,j}^\top x_{L,i})$. The second inequality is an application of Jensen inequality.

The last inequality follows by Lemma 5.7.4 and since $v_j, y_i \in \{\pm 1\}$. Now take ν small enough so that $C_3 \tau < \big((2 \cdot 67)^{-1/2} - 1/16 \big)$. Then we can use (5.7.21) together with the definition of A to get for any index $j \in A \backslash A'$,

$$\left\| \nabla_{W_{L+1,j}} L_S(\tilde{W}) \right\|_2 \geqslant \frac{1}{\sqrt{2 \cdot 67}} \gamma \mathcal{E}_S(\tilde{W}) - C_3 \tau \mathcal{E}_S(\tilde{W}) \geqslant \frac{1}{16} \gamma \mathcal{E}_S(\tilde{W}). \tag{5.7.22}$$

Thus we can derive the lower bound for the gradient of the loss at the last layer:

$$\begin{aligned} \left\| \nabla_{W_{L+1}} L_S(\tilde{W}) \right\|_F^2 &= \sum_{j=1}^{m_{L+1}} \left\| \nabla_{W_{L+1,j}} L_S(\tilde{W}) \right\|_F^2 \\ &\geqslant \sum_{j \in A \backslash A'} \left\| \nabla_{W_{L+1,j}} L_S(\tilde{W}) \right\|_2^2 \\ &\geqslant \frac{1}{16^2} |A \backslash A'| \gamma^2 [\mathcal{E}_S(\tilde{W})]^2 \\ &\geqslant \frac{1}{150 \cdot 16^2} \gamma^4 m_{L+1} [\mathcal{E}_S(\tilde{W})]^2. \end{aligned}$$

The first line is by definition, and the second is since the spectral norm is at most the Frobenius norm. The third line uses (5.7.22), and the final inequality comes from (5.7.20).

□

5.7.4 Proof of Lemma 5.4.4: gradient upper bound

Proof. Using the gradient formula (5.2.2) and the $H_l^{l'}$ notation from (5.2.1), we can write

$$\nabla_{W_l} L_S(\tilde{W}) = \theta^{\mathbb{1}(2 \leqslant l \leqslant L)} \frac{1}{n} \sum_{i=1}^n \ell'(y_i \tilde{y}_i) \cdot y_i \cdot \tilde{x}_{l-1,i} v^\top \tilde{H}_{l+1}^{L+1} \tilde{\Sigma}_l(x_i), \quad (1 \leqslant l \leqslant L+1). \tag{5.7.23}$$

Since $\tau \leqslant 1$, there is a constant C such that w.h.p. $\left\| \tilde{W}_l \right\|_2 \leqslant C$ for all l. Thus, it is easy to see that an analogous version of Lemma 5.7.2 can be applied with Lemma 5.7.4 to get that with probability at least $1 - \delta$, for all $i \in [n]$ and for all l,

$$\|\tilde{x}_{l-1,i}\|_2 \leqslant C_1 \qquad \text{and} \qquad \left\| \tilde{H}_{l+1}^{L+1} \right\|_2 \leqslant C_2. \tag{5.7.24}$$

Therefore, we can bound

$$\begin{aligned}\left\|\nabla_{W_l} L_S(\tilde{W})\right\|_F &\leqslant \frac{1}{n}\sum_{i=1}^{n}\left\|\ell'(y_i\tilde{y}_i)\cdot y_i\cdot \tilde{x}_{l-1,i} v^\top \tilde{H}_{l+1}^{L+1}\tilde{\Sigma}_{l+1}(x_i)\right\|_F \\ &= \frac{1}{n}\sum_{i=1}^{n}\left\|\ell'(y_i\tilde{y}_i)\cdot y_i\cdot \tilde{x}_{l-1,i}\right\|_2 \left\|v^\top \tilde{H}_{l+1}^{L+1}\tilde{\Sigma}_{l+1}(x_i)\right\|_2 \\ &\leqslant C_3\sqrt{m}\mathcal{E}_S(\tilde{W}).\end{aligned}$$

The first line follows by the triangle inequality, and the second since for vectors a, b, we have $\left\|ab^\top\right\|_F = \|a\|_2\|b\|_2$. The last line is by Cauchy–Schwarz, (5.7.24), and the definition of $\mathcal{E}_S$, finishing the case $l = 1$. By substituting the definition of the gradient of the loss using the formula (5.7.23) we may similarly demonstrate the corresponding bounds for $l \geqslant 2$ with an application of Cauchy–Schwartz. □

5.8 Proofs of Technical Lemmas

In this section we go over the proofs of the technical lemmas that were introduced in Appendix 5.7. In the course of proving these technical lemmas, we will need to introduce a handful of auxiliary lemmas, whose proofs we leave for Appendix 5.9. Throughout this section, we continue to assume that $\theta = 1/\Omega(L)$.

5.8.1 Proof of Lemma 5.7.2: intermediate layers are bounded

By Lemma 5.7.1, there is a constant C_1 such that with probability at least $1-\delta$, $\|W_l\|_2 \leqslant C_1$ for all $l = a, \ldots, b$. Therefore for each $r \geqslant 2$, we have

$$\left\|I + \theta\tilde{\Sigma}_r W_r\right\|_2 \leqslant \|I\|_2 + \theta\left\|\tilde{\Sigma}_r\right\|_2\|W_r\|_2 \leqslant 1 + \theta C_1.$$

The submultiplicative property of the spectral norm gives

$$
\begin{aligned}
&\left\|(I+\theta\tilde{\Sigma}_b W_b^\top)(I+\theta\tilde{\Sigma}_{b-1}W_{b-1}^\top)\cdot \ldots \cdot (I+\theta\tilde{\Sigma}_a W_a^\top)\right\|_2 \\
&\leqslant \prod_{r=a}^{b}\left\|I+\theta\tilde{\Sigma}_r W_r^\top\right\|_2 \\
&\leqslant (1+\theta C_1)^L \\
&\leqslant \exp\left(C_1\theta L\right).
\end{aligned}
$$

The result follows by the choice of scale $\theta = 1/\Omega(L)$ and taking θ small.

5.8.2 Proof of Lemma 5.7.3: Lipschitz property with respect to input space at each layer

Before beginning with the proof, we introduce the following claim that will allow us to develop a Lipschitz property with respect to the weights. This was used in [CG20] and [ALS19].

Claim 5.8.1. For arbitrary $u, y \in \mathbb{R}^{m_l}$, let $D(u)$ be the diagonal matrix with diagonal entries $[D(u)]_{j,j} = \mathbb{1}(u_j \geqslant 0)$. Then there exists another diagonal matrix $\breve{D}(u)$ such that $\left\|D(u)+\breve{D}(u)\right\|_2 \vee \left\|\breve{D}(u)\right\|_2 \leqslant 1$ and $\sigma(u)-\sigma(y) = \big(D(u)+\breve{D}(u)\big)(u-y)$.

Proof of Claim 5.8.1. Simply define

$$
[\breve{D}(u)]_{j,j} = \begin{cases} [D(u)-D(y)]\frac{y_j}{u_j-y_j} & u_j \neq y_j, \\ 0 & u_j = y_j. \end{cases}
$$

□

Proof of Lemma 5.7.3. We note that for any x, y, the matrix $|\Sigma_l(x)-\Sigma_l(y)|$ is zero everywhere except possibly the diagonal where it is either zero or one. Therefore its spectral norm is uniformly bounded by 1 for all x, y. Using this, Lemma 5.7.1 gives with probability at least $1-\delta/3$, for all $x, x' \in S^{d-1}$,

$$\begin{aligned}\|x_1 - x_1'\|_2 &= \left\|(\Sigma_1(x_1) - \Sigma_1(x_1'))W_1^\top(x - x')\right\|_2\\ &\leqslant \|\Sigma_1(x_1) - \Sigma_1(x_1')\|_2 \|W_1\|_2 \|x - x'\|_2\\ &\leqslant 1 \cdot C \cdot \|x - x'\|_2 .\end{aligned}$$

For the case $L \geqslant l \geqslant 2$, we have residual links to analyze. Using Claim 5.8.1 we can write

$$\sigma(W_l^\top x_{l-1}) - \sigma(W_l^\top \widehat{x}_{l-1}) = (\Sigma_l(x) + \check{\Sigma}_l(x))W_l^\top(x_{l-1} - \widehat{x}_{l-1})$$

for diagonal matrix $\check{\Sigma}_l$ satisfying $\left\|\check{\Sigma}_l(x)\right\|_2 \leqslant 1$ and $\left\|\Sigma_l(x) + \check{\Sigma}_l(x)\right\|_2 \leqslant 1$. By Lemma 5.7.2, we have with probability at least $1-\delta/3$, for all $2 \leqslant l \leqslant L$ and all $x, x' \in S^{d-1}$,

$$\begin{aligned}\|x_l - x_l'\|_2 &\leqslant \left\|I + \theta(\Sigma_l(x) + \check{\Sigma}_l(x))W_l^\top\right\|_2 \left\|x_{l-1} - x_{l-1}'\right\|_2\\ &\leqslant (1 + \theta C_0)\left\|x_{l-1} - x_{l-1}'\right\|_2\\ &\leqslant \left(1 + \frac{C_0\theta L}{L}\right)^L \cdot \|x - x'\|_2\\ &\leqslant C_1 \|x - x'\|_2 ,\end{aligned}$$

since θL is uniformly bounded from above.

The case $l = L+1$ follows as in the case $l = 1$ by an application of Lemma 5.7.1, so that with probability at least $1-\delta/3$, $\left\|x_{L+1}' - x_{L+1}\right\|_2 \leqslant C_2 \|x - x'\|_2$. Putting the above three claims together, we get a constant C_3 such that with probability at least $1-\delta$, $\|x_l - x_l'\|_2 \leqslant C_3 \|x - x'\|_2$ for all $x, x' \in \mathcal{S}^{d-1}$ and for all $l \in [L+1]$.

□

5.8.3 Proof of Lemma 5.7.4: local Lipschitz property with respect to weights and sparsity bound

For this lemma, we need to introduce an auxiliary lemma that allows us to get control over the sparsity levels of the ReLU activation patterns. Its proof can be found in Appendix 5.9.1.

Lemma 5.8.2. There are absolute constants C, C' such that for any $\delta > 0$, if

$$m \geqslant C\left(\beta^{-1}\sqrt{d\log\frac{1}{\beta\delta}} \vee d\log\frac{mL}{\delta}\right),$$

then with probability at least $1-\delta$, the sets

$$\mathcal{S}_l(x,\beta) = \{j \in [m_l] : |w_{l,j}^\top x_{l-1}| \leqslant \beta\},\ x \in S^{d-1},\ l \in [L+1],$$

satisfy $|\mathcal{S}_l(\beta)| \leqslant C' m_l^{\frac{3}{2}}\beta$ for all $x \in S^{d-1}$ and $l \in [L+1]$.

Proof of Lemma 5.7.4. We begin with the Lipschitz property, and afterwards will show the sparsity bound. Consider $l = 1$. Since $\widehat{x}_1 = \sigma\left(\widehat{W}_1^\top x\right)$ and $\tilde{x}_1 = \sigma\left(\tilde{W}_1^\top x\right)$, by Claim 5.8.1, for every l there is a diagonal matrix $\check{\Sigma}_l(x)$ with $\left\|\check{\Sigma}_l(x)\right\|_2 \leqslant 1$ and $\left\|\widehat{\Sigma}_l(x) + \check{\Sigma}_l(x)\right\|_2 \leqslant 1$ such that

$$\begin{aligned}
\|\widehat{x}_1 - \tilde{x}_1\|_2 &= \left\|\left(\widehat{\Sigma}_1(x) + \check{\Sigma}_1(x)\right)\left(\widehat{W}_1^\top x - \tilde{W}_1^\top x\right)\right\|_2 \\
&\leqslant \left\|\widehat{\Sigma}_1(x) + \check{\Sigma}_1(x)\right\|_2 \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 \|x\|_2 \\
&\leqslant \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2. \qquad (5.8.1)
\end{aligned}$$

For $l = 2, \ldots, L$, we can write

$$\begin{aligned}
\widehat{x}_l - \tilde{x}_l &= \widehat{x}_{l-1} + \theta\sigma\left(\widehat{W}_l^\top \widehat{x}_{l-1}\right) - \tilde{x}_{l-1} - \theta\sigma\left(\tilde{W}_l^\top \tilde{x}_{l-1}\right) \\
&= \left[I + \theta\left(\widehat{\Sigma}_l(x) + \check{\Sigma}_l(x)\right)\tilde{W}_l^\top\right](\widehat{x}_{l-1} - \tilde{x}_{l-1}) + \theta\left[\widehat{\Sigma}_l(x) + \check{\Sigma}_l(x)\right]\left(\widehat{W}_l - \tilde{W}_l\right)^\top \widehat{x}_{l-1}.
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\|\widehat{x}_l - \tilde{x}_l\|_2 &\leqslant \left\|I + \theta(\widehat{\Sigma}_l(x) + \check{\Sigma}_l(x))\tilde{W}_l^\top\right\|_2 \|\widehat{x}_{l-1} - \tilde{x}_{l-1}\|_2 + \theta\left\|\widehat{\Sigma}_l(x) + \check{\Sigma}_l(x)\right\|_2 \left\|\widehat{W}_l - \tilde{W}_l\right\|_2 \|\widehat{x}_{l-1}\|_2 \\
&\leqslant (1 + C\theta)\|\widehat{x}_{l-1} - \tilde{x}_{l-1}\|_2 + \theta\left\|\widehat{W}_l - \tilde{W}_l\right\|_2 \|\widehat{x}_{l-1}\|_2. \qquad (5.8.2)
\end{aligned}$$

We notice an easy induction will complete the proof. For the base case $l = 2$, notice that $\|\widehat{x}_1\|_2 \leqslant \|x_1\|_2 + \|\widehat{x}_1 - x_1\|_2 \leqslant C + \tau \leqslant C'$, so that (5.8.1) and (5.8.2) give

$$\|\widehat{x}_2 - x_2\|_2 \leqslant (1 + C\theta)\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + C'\theta\left\|\widehat{W}_2 - \tilde{W}_2\right\|_2 \leqslant C_4\left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + C_4\theta\left\|\widehat{W}_2 - \tilde{W}_2\right\|_2.$$

Suppose by induction that there exists a constant C such that $\|\widehat{x}_{l-1} - x_{l-1}\|_2 \leqslant C_5 \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 +$ $C_5\theta \sum_{r=1}^{l-1} \left\|\widehat{W}_r - \tilde{W}_r\right\|_2$. Then as in the base case, $\|\widehat{x}_{l-1}\|_2 \leqslant C'$, so that (5.8.2) gives for all $l = 2, \ldots, L$,

$$\begin{aligned}
\|\widehat{x}_l - \tilde{x}_l\|_2 &\leqslant (1 + C\theta)\, C \left[C_5 \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + C_5\theta \sum_{r=1}^{l-1} \left\|\widehat{W}_r - \tilde{W}_r\right\|_2 \right] + C'\theta \left\|\widehat{W}_l - \tilde{W}_l\right\|_2 \\
&\leqslant C_6 \left\|\widehat{W}_1 - \tilde{W}_1\right\|_2 + C_6\theta \sum_{r=1}^{l} \left\|\widehat{W}_r - \tilde{W}_r\right\|_2 .
\end{aligned}$$

Finally, the case $l = L + 1$ follows similarly to the case $l \leqslant L$, as

$$\begin{aligned}
\|\widehat{x}_{L+1} - \tilde{x}_{L+1}\|_2 &= \left\| \left(\widehat{\Sigma}_{L+1}(x) + \check{\Sigma}_{L+1}(x) \right) \left(\widehat{W}_{L+1}^\top \widehat{x}_L - \tilde{W}_{L+1}^\top \tilde{x}_L \right) \right\|_2 \\
&\leqslant C \left\|\widehat{W}_{L+1} - \tilde{W}_{L+1}\right\|_2 + C' \left\|\widehat{x}_L - \tilde{x}_L\right\|_2 .
\end{aligned}$$

The bound for the sparsity levels of $\tilde{\Sigma}_l(x) - \widehat{\Sigma}_l(x)$ follows the same proof as Lemma B.5 in [CG20] with an application of our Lemma 5.8.2. Sketching this proof, we note that it suffices to prove a bound for $\left\|\widehat{\Sigma}_l(x) - \Sigma_l(x)\right\|_0$, use the same proof for $\left\|\tilde{\Sigma}_l(x) - \Sigma_l(x)\right\|_0$ and then use triangle inequality to get the final result. We write

$$\left\|\widehat{\Sigma}_l(x) - \Sigma_l(x)\right\|_0 = s_l^{(1)}(\beta) + s_l^{(2)}(\beta),$$

where

$$\begin{aligned}
s_l^{(1)}(\beta) &= |\{ j \in \mathcal{S}_l(x, \beta) : (\widehat{w}_{l,j}^\top \widehat{x}_{l-1}) \cdot (w_{l,j}^\top x_{l-1}) < 0 \}|, \\
s_l^{(2)}(\beta) &= |\{ j \in \mathcal{S}_l^c(x, \beta) : (\widehat{w}_{l,j}^\top \widehat{x}_{l-1}) \cdot (w_{l,j}^\top x_{l-1}) < 0 \}|,
\end{aligned}$$

which leads to

$$\left\|\widehat{\Sigma}_l(x) - \Sigma_l(x)\right\|_0 \leqslant C m^{\frac{3}{2}} \beta + C_5 \tau^2 \beta^{-2}.$$

The choice of $\beta = m_l^{-\frac{1}{2}} \tau^{\frac{2}{3}}$ completes the proof. □

5.8.4 Proof of Lemma 5.7.5: behavior of network output in $\mathcal{W}(W^{(0)}, \tau)$ when acting on sparse vectors

This technical lemma will require two auxiliary lemmas before we may begin the proof. Their proofs are left for Appendix 5.9.2 and 5.9.3.

Lemma 5.8.3. Consider the function $g_l : \mathbb{R}^{m_l} \times \mathbb{R}^{m_{L+1}} \to \mathbb{R}$ defined by

$$g_l(a, b) := b^\top W_{L+1}^\top \xi_l a, .$$

where $\xi_l \in \mathbb{R}^{m_L \times m_l}$, and $l \geqslant 2$. Suppose that with probability at least $1 - \delta/2$, $\|\xi_l\|_2 \leqslant C$ holds for all ξ_l, $l = 2, \ldots, L$. If $s \log m = \Omega\left(C \log(L/\delta)\right)$, then there is a constant $C_0 > 0$ such that probability at least $1 - \delta$, for all l,

$$\sup_{\|a\|_2 = \|b\|_2 = 1,\ \|a\|_0, \|b\|_0 \leqslant s} |g_l(a, b)| \leqslant C_0 \sqrt{\frac{1}{m} s \log m}.$$

Lemma 5.8.4. Consider the function $g_l : \mathbb{R}^{m_l} \to \mathbb{R}$ defined by

$$g_l(a) := v^\top \Sigma_{L+1}(x)^\top W_{L+1}^\top \xi_l a,$$

where $\xi_l \in \mathbb{R}^{m_L \times m_l}$ and $l \geqslant 2$. Assume that with probability at least $1 - \delta$, $\|\xi_l\|_2 \leqslant C_0$ for all l. Then provided $s \log m = \Omega\left(\log(L/\delta)\right)$, we have with probability at least $1 - \delta$, for all l,

$$\sup_{\|a\|_2 = 1,\ \|a\|_0 \leqslant s} |g_l(a)| \leqslant C_1 \sqrt{s \log m}.$$

With these lemmas in place, we can prove Lemma 5.7.5.

Proof of Lemma 5.7.5. By definition, $g_l(a, x) = v^\top \tilde{H}_l^{L+1} a$. First: since $\left\|\tilde{W}_l - W_l\right\|_2 \leqslant \tau$, there is an absolute constant $C_2 > 0$ such that with high probability, $\left\|\tilde{W}_l\right\|_2 \leqslant C_2$ for all l. Therefore, we have with high probability for all $x \in S^{d-1}$, all l, and all a considered,

$$\left\|\tilde{H}_l^L\right\|_2 \leqslant \left[\prod_{r=l}^{L} \left\|I + \theta \tilde{\Sigma}_r(x) \tilde{W}_r^\top\right\|_2\right] \|a\|_2 \leqslant (1 + \theta \cdot 1 \cdot C_2)^L \cdot 1 \leqslant C_3, \tag{5.8.3}$$

by our choice of θ. We proceed by bounding g_l by a sum of four terms:

$$\begin{aligned}|g_l(a,x)| \leqslant a &\leqslant \left|v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) \tilde{W}_{L+1}^\top \tilde{H}_l^L a\right| + \left|v^\top \Sigma_{L+1}(x) \tilde{W}_{L+1}^\top \tilde{H}_l^L a\right| \\ &\leqslant \left|v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) \left(\tilde{W}_{L+1}^\top - W_{L+1}^\top\right) \tilde{H}_l^L a\right| + \left|v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) W_{L+1}^\top \tilde{H}_l^L a\right| \\ &\qquad + \left|v^\top \Sigma_{L+1}(x) \left(\tilde{W}_{L+1}^\top - W_{L+1}^\top\right) \tilde{H}_l^L a\right| + \left|v^\top \Sigma_{L+1}(x) W_{L+1}^\top \tilde{H}_l^L a\right|.\end{aligned}$$

For the first term, we can write

$$\begin{aligned}&\left|v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) \left(\tilde{W}_{L+1}^\top - W_{L+1}^\top\right) \tilde{H}_l^L\right| \\ &\qquad \leqslant \|v\|_2 \left\|\left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) \left(\tilde{W}_{L+1}^\top - W_{L+1}^\top\right) H_l^L a\right\|_2 \\ &\qquad \leqslant C\sqrt{m} \left\|\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right\|_2 \left\|\tilde{W}_{L+1} - W_{L+1}\right\|_2 \left\|\tilde{H}_l^L a\right\|_2 \\ &\qquad \leqslant C'\tau\sqrt{m},\end{aligned}$$

where we have used Cauchy–Schwarz in the first line, properties of the spectral norm in the second, and (5.8.3) in the third. A similar calculation shows

$$\begin{aligned}\left|v^\top \Sigma_{L+1} \left(\tilde{W}_{l+1}^\top - W_{L+1}^\top\right) \tilde{H}_l^L\right| &\leqslant \|v\|_2 \left\|\Sigma_{L+1} \left(\tilde{W}_{L+1}^\top - W_{L+1}^\top\right) \tilde{H}_l^L\right\|_2 \\ &\leqslant C\tau\sqrt{m}.\end{aligned}$$

For the second and fourth terms, we use Lemmas 5.8.3 and 5.8.4. Let $\check{b}^\top = v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right)$. Then it is clear that $\left\|\check{b}\right\|_0 \leqslant s$ and $\left\|\check{b}\right\|_2 \leqslant \sqrt{m}$ (in fact, $\left\|\check{b}\right\|_2 \leqslant \sqrt{s}$, but this doesn't matter since the fourth term dominates the second term). Thus applying Lemma 5.8.3 to $b = \check{b}/\left\|\check{b}\right\|_2$,

$$\begin{aligned}|v^\top \left(\tilde{\Sigma}_{L+1}(x) - \Sigma_{L+1}(x)\right) W_{L+1}^\top \tilde{H}_l^L a| &\leqslant C\sqrt{m} \cdot \sqrt{\frac{s}{m} \log m} \\ &\leqslant C\sqrt{s \log m}.\end{aligned}$$

For the fourth term, we can directly apply Lemma 5.8.4 to get another term $\propto \sqrt{s \log m}$. □

5.8.5 Proof of Lemma 5.7.6

This lemma is the key to the sublinear dependence on L for the required width for the generalization result. Essential to its proof is the following proposition which states that there is a linear separability condition at each layer due to Assumption 5.3.2 with only a logarithmic dependence on the depth L. In fact, we only need linear separability at the second-to-last layer for the proof of Lemma 5.7.6.

Proposition 5.8.5. Suppose $m \geqslant C\gamma^{-2}\left(d\log\frac{1}{\gamma}+\log\frac{L}{\delta}\right)$ for some large constant C. Then there exists $\alpha \in S^{m_L-1}$ such that with probability at least $1-\delta$, for all $l = 1, \dots, L$, we have

$$y\left\langle \alpha, x_l\right\rangle \geqslant \gamma/2.$$

Proof of Proposition 5.8.5. We recall that Assumption 5.3.2 implies that there exists $c(\overline{\mathbf{u}})$ with $\|c(u)\|_\infty \leqslant 1$ such that $f(x) = \int_{\mathbb{R}^d} c(u)\sigma(u^\top x)p(u)du$ satisfies $y \cdot f(x) \geqslant \gamma$ for all $(x, y) \in \operatorname{supp}(\mathcal{D})$. Following Lemma C.1 in [CG20], if we define

$$\alpha := \sqrt{\frac{1}{m_1}} \cdot \left(c\left(\sqrt{\frac{m_1}{2}}w_{1,1}\right), \dots, c\left(\sqrt{\frac{m_1}{2}}w_{1,m_1}\right)\right),$$

then $\alpha = \alpha'/\left\|\alpha'\right\|_2 \in S^{m_1-1}$ satisfies $y \cdot \alpha^\top x_1 \geqslant \frac{\gamma}{2}$ for all $(x, y) \in \operatorname{supp}\mathcal{D}$.

We now show that the l-th layer activations x_l are linearly separable using α. We can write, for $l = 2, \dots, L$,

$$\begin{aligned}\left\langle \alpha, x_l\right\rangle &= \left\langle \alpha, (I+\theta\Sigma_l(x)W_l^\top)x_{l-1}\right\rangle \\ &= \left\langle \alpha, x_1\right\rangle + \theta\sum_{l'=2}^{l}\left\langle \alpha, \Sigma_{l'}(x)W_{l'}^\top x_{l'-1}\right\rangle. \qquad (5.8.4)\end{aligned}$$

Since $\left\langle \alpha, \Sigma_l(x)W_l^\top x_{l-1}\right\rangle = \sum_{k=1}^{m_l}\sqrt{\frac{1}{m_1}}c\left(\sqrt{\frac{m_1}{2}}w_{1,k}\right)\cdot\sigma(w_{l,k}^\top x_{l-1})$ and $\|c(\cdot)\|_\infty \leqslant 1$, we have for every $l \geqslant 2$,

$$-\sum_{k=1}^{m_l}\sqrt{\frac{1}{m_1}}\left|w_{l,k}^\top x_{l-1}\right| \leqslant \left\langle \alpha, \Sigma_l(x)W_l^\top x_{l-1}\right\rangle \leqslant \sum_{k=1}^{m_l}\sqrt{\frac{1}{m_1}}\left|w_{l,k}^\top x_{l-1}\right|. \qquad (5.8.5)$$

Thus it suffices to find an upper bound for the term on the r.h.s. of (5.8.5). Since we have

$$\mathbb{E}\left|w_{l,k}^{\top} x_{l-1}\right| = \sqrt{\frac{2}{\pi}}\sqrt{\frac{2}{m_1}}\left\|x_{l-1}\right\|_2 \leqslant C_2 m^{-\frac{1}{2}},$$

we can apply Hoeffding inequality to get absolute constants $C_4, C_5 > 0$ such that for fixed x and l, we have with probability at least $1-\delta$,

$$\begin{aligned}\sum_{k=1}^{m_l}\sqrt{\frac{1}{m_1}}\left|w_{l,k}^{\top} x_{l-1}\right| &\leqslant \sum_{k=1}^{m_l}\sqrt{\frac{1}{m}} C_2 m^{-\frac{1}{2}} + C_4\sqrt{\frac{1}{m}\log\frac{1}{\delta}} \\ &\leqslant C_5 + C_4\sqrt{\frac{1}{m}\log\frac{1}{\delta}}.\end{aligned}$$

Take a $\frac{1}{2}$-net $\mathcal{N}$ of S^{d-1} so that $|\mathcal{N}| \leqslant 5^d$ and every $x \in S^{d-1}$ has $\widehat{x} \in \mathcal{N}$ with $\|x-\widehat{x}\|_2 \leqslant \frac{1}{2}$. Then, provided $m \geqslant Cd\log\frac{L}{\delta}$, there is a constant $C_6 > 0$ such that we have with probability at least $1-\delta$, for all $\widehat{x} \in \mathcal{N}$ and all $l \leqslant L$,

$$\sum_{k=1}^{m_l}\sqrt{\frac{1}{m_1}}\left|w_{l,k}^{\top}\widehat{x}_{l-1}\right| \leqslant C_6.$$

By (5.8.5), this means for all $\widehat{x} \in \mathcal{N}$ and l, $-C_6 \leqslant \left\langle \alpha, \Sigma_l(\widehat{x}) W_l^{\top}\widehat{x}_{l-1}\right\rangle \leqslant C_6$. We can lift this to hold over S^{d-1} by using Lemma 5.7.3: for arbitrary $x \in S^{d-1}$ we have

$$\begin{aligned}\left|\left\langle \alpha, \Sigma_l(x) W_l^{\top} x_l\right\rangle\right| &\leqslant \left|\left\langle \alpha, \Sigma_l(x) W_l^{\top}(x_l - \widehat{x}_l)\right\rangle\right| + \left|\left\langle \alpha, \Sigma_l(x) W_l^{\top}\widehat{x}_l\right\rangle\right| \\ &\leqslant \left\|\tilde{\alpha}_l\right\|_2 \left\|\Sigma_l(x)\right\|_2 \left\|W_l\right\|_2 \left\|x_l - \widehat{x}_l\right\|_2 + C_6 \\ &\leqslant C_7,\end{aligned}$$

so that with probability at least $1-\delta$, for all $l \leqslant L$ and all $x \in S^{d-1}$, we have

$$-C_7 \leqslant \left\langle \alpha, \Sigma_l(x) W_l^{\top}\widehat{x}_{l-1}\right\rangle \leqslant C_7.$$

Substituting the above into (5.8.4), we get

$$\begin{cases}\langle \alpha, x_l\rangle \geqslant \langle \alpha, x_1\rangle - \theta L C_7, \\ -\langle \alpha, x_l\rangle \geqslant -\langle \alpha, x_1\rangle - \theta L C_7.\end{cases}$$

Considering the cases $y = \pm 1$ we thus get with probability at least $1 - \delta$ for all l and $(x, y) \in \operatorname{supp} \mathcal{D}$,

$$\begin{cases} y \langle \alpha, x_l \rangle \geqslant y \langle \alpha, x_1 \rangle - \theta L C_7 \geqslant \frac{\gamma}{2} - \theta L C_7, & y = 1, \\ y \langle \alpha, x_l \rangle \geqslant y \langle \alpha, x_1 \rangle - \theta L C_7 \geqslant \frac{\gamma}{2} - \theta L C_7, & y = -1. \end{cases}$$

Thus taking θ small enough so that $\theta L \leqslant \gamma C_7^{-1}/4$ completes the proof. □

With Proposition 5.8.5 in hand, we can prove Lemma 5.7.6.

Proof of Lemma 5.7.6. By Proposition 5.8.5, there exists $\alpha_L \in S^{m_L - 1}$ such that with probability at least $1 - \delta$, $y \langle \alpha_L, x_L \rangle \geqslant \gamma/4$ for all $(x, y) \in \operatorname{supp}(\mathcal{D})$. In particular, since a is non-negative, this implies for all i,

$$\langle a(x_i, y_i) \cdot y_i \cdot x_{L,i}, \alpha_L \rangle = a(x_i, y_i) \cdot y_i \langle x_{L,i}, \alpha_L \rangle \geqslant a(x_i, y_i) y_i \gamma/4. \tag{5.8.6}$$

Since $\mathbb{E}[\sigma'(w_{L+1,j}^\top x_{L,i}) | x_{L,i}] = \frac{1}{2}$, by Hoeffding inequality, with probability at least $1 - \delta/2$, for all $i = 1, \ldots, n$, we have

$$\frac{1}{m_{L+1}} \sum_{j=1}^{m_{L+1}} \sigma'(w_{L+1,j}^\top x_{L,i}) \geqslant \frac{1}{2} - C_1 \sqrt{\frac{1}{m_{L+1}} \log(n/\delta)} \geqslant \frac{49}{100}. \tag{5.8.7}$$

Therefore, we can bound

$$\begin{aligned}
&\sum_{j=1}^{m_{L+1}} \left\| \frac{1}{n} \sum_{i=1}^{n} \big[a(x_i, y_i) \cdot y_i \cdot \sigma'(w_{L+1,j}^\top x_{L,i}) \cdot x_{L,i} \big] \right\|_2^2 \\
&\geqslant m_{L+1} \left\| \frac{1}{m_{L+1}} \sum_{j=1}^{m_{L+1}} \frac{1}{n} \sum_{i=1}^{n} \big[a(x_i, y_i) \cdot y_i \cdot \sigma'(w_{L+1,j}^\top x_{L,i}) \cdot x_{L,i} \big] \right\|_2^2 \\
&= m_{L+1} \left\| \frac{1}{n} \sum_{i=1}^{n} \left[a(x_i, y_i) \cdot y_i \cdot x_{L,i} \frac{1}{m_{L+1}} \sum_{j=1}^{m_{L+1}} \sigma'(w_{L+1,j}^\top x_{L,i}) \right] \right\|_2^2 \\
&\geqslant m_{L+1} \left\langle \frac{1}{n} \sum_{i=1}^{n} a(x_i, y_i) \cdot y_i \cdot x_{L,i} \cdot \frac{1}{m_{L+1}} \sum_{j=1}^{m_{L+1}} \sigma'(w_{L+1,j}^\top x_{L,i}), \alpha_L \right\rangle^2 \\
&= m_{L+1} \left(\frac{1}{n} \sum_{i=1}^{n} a(x_i, y_i) \cdot y_i \cdot \frac{1}{m_{L+1}} \sum_{j=1}^{m_{L+1}} \sigma'(w_{L+1,j}^\top x_{L,i}) \cdot \langle x_{L,i}, \alpha_L \rangle \right)^2 \\
&\geqslant \left(\frac{49}{100} \right)^2 m_{L+1} \left(\frac{1}{n} \sum_{i=1}^{n} a(x_i, y_i) \right)^2 \cdot \frac{\gamma^2}{4^2} \\
&\geqslant \frac{1}{67} m_{L+1} \cdot \gamma^2 \left(\frac{1}{n} \sum_{i=1}^{n} a(x_i, y_i) \right)^2 .
\end{aligned}$$

The first inequality above follows by Jensen inequality. The second inequality follows by Cauchy–Schwarz and since $\|\alpha_L\|_2 = 1$. The third inequality follows with an application of (5.8.6) and (5.8.7), and the final inequality by arithmetic. □

5.9 Proofs of Auxiliary Lemmas

5.9.1 Proof of Lemma 5.8.2

Proof. By following a proof similar to that of Lemma A.8 in [CG20], one can easily prove the following claim:

Claim 5.9.1. For $v \in \mathbb{R}^{m_{l-1}}$, $\beta > 0$, and $l \in [L+1]$ define

$$\mathcal{S}_l(v, \beta) := \{ j \in [m_l] : |w_{l,j}^\top v| \leqslant \beta \}. \tag{5.9.1}$$

Suppose that there is an absolute constant $\xi \in (0,1)$ such that for any $\delta > 0$ we have with probability at least $1-\delta/2$, $\|v\|_2 \geqslant \xi$ for all $v \in \mathcal{V}$ for some finite set $\mathcal{V} \subset \mathbb{R}^{m_{l-1}}$. Then there exist absolute constants $C, C' > 0$ such that if $m \geqslant C\beta^{-1}\sqrt{\log(4|\mathcal{V}|/\delta)}$, then with probability at least $1-\delta$, we have $|\mathcal{S}_l(v,\beta)| \leqslant C' m_l^{3/2}\beta$ for all $v \in \mathcal{V}$.

By Lemmas 5.4.1 and 5.7.1, with probability at least $1-\delta/3$, we have $\|x_{l-1}\|_2 \geqslant C$ and $\|w_{l,j}\|_2 \leqslant C_1$ for all $x \in S^{d-1}$, $l \in [L+1]$, and $j \in [m_l]$. By Lemma 5.7.3, with probability at least $1-\delta/3$, we have $\|x_l - x_l'\|_2 \leqslant C_2 \|x - x'\|_2$ for all $x, x' \in S^{d-1}$. By taking $\mathcal{V}$ to be the $\beta/(C_1C_2)$-net $\mathcal{N}(S^{d-1}, \beta/(C_1C_2))$, since $|\mathcal{N}| \leqslant (4C_1C_2/\beta)^d$, the assumption that $m \geqslant C\beta^{-1}\sqrt{d\log(1/(\beta\delta))}$ allows us to apply Lemma 5.9.1 to get that with probability at least $1-\delta/3$, we have $|\mathcal{S}_l(\widehat{x}, 2\beta)| \leqslant 2C' m_l^{\frac{3}{2}}\beta$ for all l and $\widehat{x} \in \mathcal{N}$. For arbitrary $x \in S^{d-1}$, there exists $\widehat{x} \in \mathcal{N}$ with $\|x - \widehat{x}\|_2 \leqslant \beta/(C_1C_2)$. Thus, we have

$$\begin{aligned}
|w_{l,j}^\top x_{l-1}| &\leqslant |w_{l,j}^\top \widehat{x}_{l-1}| + |w_{l,j}^\top (x_{l-1} - \widehat{x}_{l-1})| \\
&\leqslant \beta + \|w_{l,j}\|_2 \|x_{l-1} - \widehat{x}_{l-1}\|_2 \\
&\leqslant \beta + C_1 \cdot C_2 \|x - \widehat{x}\|_2 \\
&\leqslant 2\beta,
\end{aligned}$$

i.e. $\mathcal{S}_l(x,\beta) \subset \mathcal{S}_l(\widehat{x}, 2\beta)$. Therefore $|\mathcal{S}_l(x,\beta)| \leqslant |\mathcal{S}_l(\widehat{x}, 2\beta)| \leqslant 2C' m_l^{\frac{3}{2}}\beta$, as desired. □

5.9.2 Proof of Lemma 5.8.3

Proof. The j-th row of $W_{L+1}^\top \xi_l a$ has distribution $w_{L+1,j}^\top \xi_l a \sim N\left(0, \frac{2}{m_{L+1}} \|\xi_l a\|_2^2\right)$, and hence $g_l(a,b) \sim N\left(0, \frac{2}{m_l} \|\xi_l a\|_2^2\right)$. Since $\|\xi_l\|_2 \leqslant C_0$ for all l with high probability, it is clear that $\|\xi_l a\|_2^2 \leqslant C_0^2$. Thus applying Hoeffding inequality gives a constant $C_3 > 0$ such that we have for fixed a and b, with probability at least $1-\delta$,

$$|b^\top W_{L+1}^\top \xi_l a| \leqslant C_3 \sqrt{\frac{1}{m_{L+1}} \log \frac{1}{\delta}}. \tag{5.9.2}$$

Let $\mathcal{M}_a$ be a fixed subspace of $\mathbb{R}^{m_l}$ with sparsity s, and let $\mathcal{N}_a(\mathcal{M}, 1/4)$ be a 1/4-net covering $\mathcal{M}_a$. There are $\binom{m_l}{s}$ choices of such $\mathcal{M}_a$. Let $\mathcal{N}_a = \cup_{\mathcal{M}_a} \mathcal{N}_a(\mathcal{M}_a, 1/4)$ be the union of such spaces. By Lemma 5.2 in [Ver10], for s larger than e.g. 15, we have

$$|\mathcal{N}_a| \leqslant \binom{m_l}{s} 9^s \leqslant m_l^s.$$

Similarly consider subspace $\mathcal{M}_b \subset \mathbb{R}^{m_{L+1}}$ with sparsity level s and let $\mathcal{N}_b(\mathcal{M}_b, 1/4)$ be a 1/4-net of $\mathbb{R}^{m_{L+1}}$ with sparsity level s and define $\mathcal{N}_b = \cup_{\mathcal{M}_b} \mathcal{N}_b(\mathcal{M}_b, 1/4)$, so that $|\mathcal{N}_b| \leqslant m_{L+1}^s$. We apply (5.9.2) to every $\widehat{a} \in \mathcal{N}_a$ and $\widehat{b} \in \mathcal{N}_b$ and use a union bound to get a constant $C_4 > 0$ such that with probability at least $1 - \delta$, for all $\widehat{a} \in \mathcal{N}_a, \widehat{b} \in \mathcal{N}_b$, and all l,

$$\begin{aligned}
|\widehat{b}^\top W_{L+1}^\top \xi_l \widehat{a}| &\leqslant C_3 \sqrt{\frac{1}{m_{L+1}} \log \frac{|\mathcal{N}_a| \cdot |\mathcal{N}_b| \cdot L}{\delta}} \\
&\leqslant C_3 \sqrt{\frac{1}{m_{L+1}} \log \frac{m_{L+1}^s \cdot m_l^s \cdot L}{\delta}} \\
&= C_3 \sqrt{\frac{1}{m_{L+1}} \left(s \log(m_{L+1} m_l) + \log \frac{L}{\delta} \right)} \\
&\leqslant C_4 \sqrt{\frac{s}{m_{L+1}} \log m}. && \left(s \log m = \Omega \left(\log \frac{L}{\delta} \right) \right)
\end{aligned}$$

For arbitrary $a \in S^{m_l - 1}$ and $b \in S^{m_{L+1} - 1}$ with $\|a\|_0, \|b\|_0 \leqslant s$, there are $\widehat{a} \in \mathcal{N}_a$ and $\widehat{b} \in \mathcal{N}_b$ with $\|a - \widehat{a}\|_2, \left\| b - \widehat{b} \right\|_2 \leqslant 1/4$. Note that g is linear in a and b. Triangle inequality gives

$$\begin{aligned}
|g_l(a, b)| &\leqslant |g_l(\widehat{a}, \widehat{b})| + |g_l(a, b) - g_l(\widehat{a}, \widehat{b})| \\
&\leqslant C_3 \sqrt{\frac{s}{m_{L+1}} \log m_{L+1}} + |g_l(a, b) - g_l(\widehat{a}, b)| + |g_l(\widehat{a}, \widehat{b}) - g_l(\widehat{a}, b)| \qquad (5.9.3)
\end{aligned}$$

We have for any $\widehat{a}$,

$$\begin{aligned}
|g_l(\widehat{a}, \widehat{b}) - g_l(\widehat{a}, b)| &= \left\| b - \widehat{b} \right\|_2 \left| g_l \left(\widehat{a}, \frac{b - \widehat{b}}{\left\| b - \widehat{b} \right\|_2} \right) \right| \\
&\leqslant \frac{1}{4} \sup_{\|b'\|_2 = \|a\|_2 = 1, \ \|a\|_0, \|b'\|_0 \leqslant s} |g_l(a, b')| . \qquad (5.9.4)
\end{aligned}$$

Similarly,

$$|g_l(a, b) - g_l(\widehat{a}, b)| \leqslant \frac{1}{4} \sup_{\|b\|_2 = \|a\|_2 = 1, \ \|a\|_0, \|b\|_0 \leqslant s} |g_l(a, b)| . \qquad (5.9.5)$$

Taking supremum over the left hand side of (5.9.3) and using the bounds in (5.9.4) and (5.9.5) completes the proof. □

5.9.3 Proof of Lemma 5.8.4

Proof. We notice that since $v = (1, \dots, 1, -1, \dots, -1)^\top$, we can write $g_l(a)$ as a sum of independent random variables in the following form:

$$g_l(a) = \sqrt{m_{L+1}} \sum_{j=1}^{m_{L+1}/2} \frac{1}{\sqrt{m_{L+1}}} \left[\sigma(w_{L+1,j}^\top \xi_{l+1} a) - \sigma(w_{L+1,j+m_{L+1}/2}^\top \xi_{l+1} a) \right].$$

Since $\|\xi_{l+1} a\|_2$ is uniformly bounded by a constant, Hoeffding inequality yields a constant $C_3 > 0$ such that for fixed a, with probability at least $1 - \delta$, we have

$$g_l(a) \leqslant C_3 \sqrt{m} \sqrt{\frac{1}{m} \log \frac{1}{\delta}}.$$

Let $\mathcal{M}$ be a fixed subspace of $\mathbb{R}^{m_l}$ with sparsity s, and let $\mathcal{N} = \cup_{\mathcal{M}} \mathcal{N}(\mathcal{M}, 1/2)$ be the union of all 1/2-nets covering each $\mathcal{M}$ so that $|\mathcal{N}| \leqslant m_l^s$. Using a union bound over all $\widehat{a} \in \mathcal{N}$ and l, we get that with probability at least $1 - \delta$, for all $\widehat{a} \in \mathcal{N}$ and all $l \leqslant L$,

$$g_l(\widehat{a}) \leqslant C_3 \sqrt{m} \cdot \sqrt{\frac{1}{m} \log \frac{|\mathcal{N}| \cdot L}{\delta}} \leqslant C_5 \sqrt{s \log m}.$$

For arbitrary $a \in S^{m_l - 1}$ satisfying $\|a\|_0 \leqslant s$, there is $\widehat{a} \in \mathcal{N}$ with $\|a - \widehat{a}\|_2 \leqslant 1/2$. Since g is linear,

$$|g_l(a)| \leqslant |g_l(\widehat{a})| + |g_l(a - \widehat{a})| \leqslant C_5 \sqrt{s \log m} + |g_l(a - \widehat{a})|. \tag{5.9.6}$$

For the second term, we have

$$|g_l(a - \widehat{a})| = \|a - \widehat{a}\|_2 \left| g_l \left(\frac{a - \widehat{a}}{\|a - \widehat{a}\|_2} \right) \right| \leqslant \frac{1}{2} \sup_{\|a\|_2 = 1,\ \|a\|_0 \leqslant s} |g_l(a)|.$$

Substituting this into (5.9.6) and taking supremums completes the proof. □

CHAPTER 6

Conclusion

In this thesis, we provided a number of analyses which explore the optimization and generalization questions for the SGD-training of neural networks. Our first set of results consisted of the simplest neural network possible, namely a single neuron neural network. We showed that for learning single neuron neural networks in the regression setting, there exists a surrogate loss for which the original optimization problem exhibits a type of proto-convexity with respect to the surrogate loss. For the classification setting, we connected the minimizers of convex surrogates for the zero-one loss to the minimizers for the zero-one loss itself, and showed that under benign distributional assumptions the two are quite closely related. This resulted in the first positive guarantee for the agnostic learning of halfspaces using gradient descent on convex losses.

We continued our analysis of the agnostic learning of halfspaces by showing that SGD-trained one-hidden-layer networks can also agnostically learn halfspaces under benign distributional assumptions. Our result here utilized that the gradients of the neural network were always partially correlated with those of the best linear predictor over the dataset, and that this correlation would increase until we reach a point with small loss. This approach avoided the non-convexity of the underlying optimization problem. Moreover, our guarantees were independent of the width of the neural network, in stark contrast to standard uniform convergence-based bounds on the VC dimension of the network which grow as the network becomes larger.

We finished the thesis with an analysis of a complicated deep residual network, showing that if the data can be classified under an infinitely-wide one-hidden-layer neural network, then deep residual networks trained by gradient descent can generalize as well.

There are numerous natural next steps from here. In this thesis, we were able to develop guarantees for constant width networks on noisy linear data, and for networks of unbounded width on nonlinear data without noise. But as of the time of writing, there exist no provable guarantees for the generalization of SGD-trained constant width neural networks on noisy nonlinear data distributions. Such work is a natural progression of the ideas considered in this dissertation.

In the course of developing the optimization guarantees for learning a single neuron and for learning halfspaces with noise, we utilized novel analyses to avoid the nonconvexity of the underlying optimization problem. Understanding the extent to which these techniques can be generalized into abstract principles for nonconvex optimization is another task we are interested in exploring further.

More broadly, there is an ever-expanding universe of problems in the theory of deep learning as new methods and techniques are developed which exhibit ever-more surprising behaviors of deep neural network models trained by gradient descent. Recent questions that have come to the fore include the ability of semi-supervised and self-supervised learning methods to improve the generalization performance of models in the supervised learning setting; the brittleness of deep neural network classifiers to adversarial perturbations of the input data; and the ubiquity of transformer-based architectures for problems across a variety of domain settings. We are excited about the possibility of theoretically understanding these perplexing empirical phenomena in the future.

REFERENCES

[ABH15] Pranjal Awasthi, Maria-Florina Balcan, Nika Haghtalab, and Ruth Urner. "Efficient Learning of Linear Separators under Bounded Noise." In *Conference on Learning Theory (COLT)*, 2015.

[ABH16] Pranjal Awasthi, Maria-Florina Balcan, Nika Haghtalab, and Hongyang Zhang. "Learning and 1-bit Compressed Sensing under Asymmetric Noise." In *Conference on Learning Theory (COLT)*, 2016.

[ABL17] Pranjal Awasthi, Maria Florina Balcan, and Philip M. Long. "The Power of Localization for Efficiently Learning Linear Separators with Noise." *J. ACM*, **63**(6), January 2017.

[ADH19a] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Ruslan Salakhutdinov, and Ruosong Wang. "On exact computation with an infinitely wide neural net." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[ADH19b] Sanjeev Arora, Simon S. Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. "Fine-Grained Analysis of Optimization and Generalization for Overparameterized Two-Layer Neural Networks." In *International Conference on Machine Learning (ICML)*, 2019.

[AGN18] Sanjeev Arora, Rong Ge, Behnam Neyshabur, and Yi Zhang. "Stronger Generalization Bounds for Deep Nets via a Compression Approach." In *International Conference on Machine Learning (ICML)*, 2018.

[AHW95] Peter Auer, Mark Herbster, and Manfred K. Warmuth. "Exponentially Many Local Minima for Single Neurons." In *Advances in Neural Information Processing Systems (NeurIPS)*, 1995.

[AL88] Dana Angluin and Philip Laird. "Learning from noisy examples." *Machine Learning*, **2**(4):343–370, 1988.

[AL19] Zeyuan Allen-Zhu and Yuanzhi Li. "What Can ResNet Learn Efficiently, Going Beyond Kernels?" In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[ALL19] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. "Learning and Generalization in Overparameterized Neural Networks, Going Beyond Two Layers." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[ALS19] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. "A Convergence Theory for Deep Learning via Over-Parameterization." In *International Conference on Machine Learning (ICML)*, 2019.

[BFK98] Avrim Blum, Alan Frieze, Ravi Kannan, and Santosh Vempala. “A polynomial-time algorithm for learning noisy linear threshold functions.” *Algorithmica*, **22**(1-2):35–52, 1998.

[BFT17] Peter L. Bartlett, Dylan J. Foster, and Matus J. Telgarsky. “Spectrally-normalized margin bounds for neural networks.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[BGM18] Alon Brutzkus, Amir Globerson, Eran Malach, and Shai Shalev-Shwartz. “SGD Learns Over-parameterized Networks that Provably Generalize on Linearly Separable Data.” In *International Conference on Learning Representations (ICLR)*, 2018.

[BGV92] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. “A training algorithm for optimal margin classifiers.” In *Conference on Learning Theory (COLT)*, 1992.

[BH21] Maria-Florina Balcan and Nika Haghtalab. “Noise in Classification.” In Tim Roughgarden, editor, *Beyond Worst Case Analysis of Algorithms*, chapter 16. Cambridge University Press, 2021.

[BJM06] Peter L. Bartlett, Michael I. Jordan, and Jon D. McAuliffe. “Convexity, classification, and risk bounds.” *Journal of the American Statistical Association*, **101**(473):138–156, 2006. (Was Department of Statistics, U.C. Berkeley Technical Report number 638, 2003).

[BLL11] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert E. Schapire. “Contextual Bandit Algorithms with Supervised Learning Guarantees.” In *Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

[BLS12] Shai Ben-David, David Loker, Nathan Srebro, and Karthik Sridharan. “Minimizing the misclassification error rate using a surrogate convex loss.” In *International Conference on Machine Learning (ICML)*, 2012.

[BRS89] M. L. Brady, R. Raghavan, and J. Slawny. “Back propagation fails to separate where perceptrons succeed.” *IEEE Transactions on Circuits and Systems*, **36**(5):665–674, 1989.

[BZ17] Maria-Florina F Balcan and Hongyang Zhang. “Sample and computationally efficient learning algorithms under s-concave distributions.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[CCG20] Zixiang Chen, Yuan Cao, Quanquan Gu, and Tong Zhang. “A Generalized Neural Tangent Kernel Analysis for Two-layer Neural Networks.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[CCZ21] Zixiang Chen, Yuan Cao, Difan Zou, and Quanquan Gu. "How Much Over-parameterization Is Sufficient to Learn Deep ReLU Networks?" In *International Conference on Learning Representations (ICLR)*, 2021.

[CG19a] Yuan Cao and Quanquan Gu. "Generalization Bounds of Stochastic Gradient Descent for Wide and Deep Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[CG19b] Yuan Cao and Quanquan Gu. "Tight Sample Complexity of Learning One-hidden-layer Convolutional Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[CG20] Yuan Cao and Quanquan Gu. "Generalization Error Bounds of Gradient Descent for Learning Over-parameterized Deep ReLU Networks." In *AAAI Conference on Artificial Intelligence*, 2020.

[COB19] Lenaic Chizat, Edouard Oyallon, and Francis Bach. "On Lazy Training in Differentiable Programming." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[CSS19] Seungwoo Choi, Seokjun Seo, Beomjun Shin, Hyeongmin Byun, Martin Kersner, Beomsu Kim, Dongyoung Kim, and Sungjoo Ha. "Temporal Convolution for Real-time Keyword Spotting on Mobile Devices." In *INTERSPEECH*, 2019.

[Cyb89] George Cybenko. "Approximation by superpositions of a sigmoidal function." *Mathematics of Control, Signals and Systems*, **2**(4):303–314, 1989.

[Dan16] Amit Daniely. "Complexity theoretic limitations on learning halfspaces." In *ACM Symposium on Theory of Computing (STOC)*, pp. 105–117, 2016.

[DDS09] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. "ImageNet: A Large-Scale Hierarchical Image Database." In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

[DGK20a] Ilias Diakonikolas, Surbhi Goel, Sushrut Karmalkar, Adam R Klivans, and Mahdi Soltanolkotabi. "Approximation Schemes for ReLU Regression." In *Conference on Learning Theory (COLT)*, 2020.

[DGK20b] Ilias Diakonikolas, Surbhi Goel, Sushrut Karmalkar, Adam R Klivans, and Mahdi Soltanolkotabi. "Approximation Schemes for ReLU Regression." In *Conference on Learning Theory (COLT)*, 2020.

[DGT19] Ilias Diakonikolas, Themis Gouleakis, and Christos Tzamos. "Distribution-independent pac learning of halfspaces with massart noise." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[DKK20] Ilias Diakonikolas, Daniel M. Kane, Vasilis Kontonis, and Nikos Zarifis. “Algorithms and SQ Lower Bounds for PAC Learning One-Hidden-Layer ReLU Networks.” In *Conference on Learning Theory (COLT)*, 2020.

[DKT20a] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. “Learning Halfspaces with Massart Noise Under Structured Distributions.” In *Conference on Learning Theory (COLT)*, 2020.

[DKT20b] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. “Non-Convex SGD Learns Halfspaces with Adversarial Label Noise.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[DKT21] Ilias Diakonikolas, Vasilis Kontonis, Christos Tzamos, and Nikos Zarifis. “Learning Halfspaces with Tsybakov Noise.” In *ACM Symposium on Theory of Computing (STOC)*, 2021.

[DKZ20] Ilias Diakonikolas, Daniel M Kane, and Nikos Zarifis. “Near-optimal sq lower bounds for agnostically learning halfspaces and relus under gaussian marginals.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[DLL18] Simon S. Du, Jason D. Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. “Gradient Descent Finds Global Minima of Deep Neural Networks.” In *International Conference on Machine Learning (ICML)*, 2018.

[DLT18] Simon S. Du, Jason D. Lee, and Yuandong Tian. “When is a Convolutional Filter Easy to Learn?” In *International Conference on Learning Representations (ICLR)*, 2018.

[DR17] Gintare Karolina Dziugaite and Daniel M. Roy. “Computing Nonvacuous Generalization Bounds for Deep (Stochastic) Neural Networks with Many More Parameters than Training Data.” In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

[DZP19] Simon S. Du, Xiyu Zhai, Barnabás Póczos, and Aarti Singh. “Gradient Descent Provably Optimizes Over-parameterized Neural Networks.” In *International Conference on Learning Representations (ICLR)*, 2019.

[EMW19] Weinan E, Chao Ma, Qingcan Wang, and Lei Wu. “Analysis of the Gradient Descent Algorithm for a Deep Neural Network Model with Skip-connections.” *Preprint, arXiv:1904.05263*, 2019.

[FCG19] Spencer Frei, Yuan Cao, and Quanquan Gu. “Algorithm-Dependent Generalization Bounds for Overparameterized Deep Residual Networks.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[FCG20] Spencer Frei, Yuan Cao, and Quanquan Gu. "Agnostic Learning of a Single Neuron with Gradient Descent." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[FCG21] Spencer Frei, Yuan Cao, and Quanquan Gu. "Agnostic Learning of Halfspaces with Gradient Descent via Soft Margins." In *International Conference on Machine Learning (ICML)*, 2021.

[FDP20] Stanislav Fort, Gintare Karolina Dziugaite, Mansheej Paul, Sepideh Kharaghani, Daniel M. Roy, and Surya Ganguli. "Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the Neural Tangent Kernel." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[FSS18] Dylan J. Foster, Ayush Sekhari, and Karthik Sridharan. "Uniform Convergence of Gradients for Non-Convex Learning and Optimization." In *Advances in Neural Information Processing Systems*, 2018.

[GGJ20] Surbhi Goel, Aravind Gollakota, Zhihan Jin, Sushrut Karmalkar, and Adam Klivans. "Superpolynomial Lower Bounds for Learning One-Layer Neural Networks using Gradient Descent." In *International Conference on Machine Learning (ICML)*, 2020.

[GGK20] Surbhi Goel, Aravind Gollakota, and Adam Klivans. "Statistical-query lower bounds via functional gradients." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[GKK17] Surbhi Goel, Varun Kanade, Adam Klivans, and Justin Thaler. "Reliably Learning the ReLU in Polynomial Time." In *Conference on Learning Theory (COLT)*, 2017.

[GKK19] Surbhi Goel, Sushrut Karmalkar, and Adam R. Klivans. "Time/Accuracy Tradeoffs for Learning a ReLU with respect to Gaussian Marginals." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[GKM18] Surbhi Goel, Adam R. Klivans, and Raghu Meka. "Learning One Convolutional Layer with Overlapping Patches." In Jennifer G. Dy and Andreas Krause, editors, *International Conference on Machine Learning*, 2018.

[GR09] Venkatesan Guruswami and Prasad Raghavendra. "Hardness of learning halfspaces with noise." *SIAM Journal on Computing*, **39**(2):742–765, 2009.

[GRS18] Noah Golowich, Alexander Rakhlin, and Ohad Shamir. "Size-Independent Sample Complexity of Neural Networks." In *Conference on Learning Theory (COLT)*. PMLR, 2018.

[GSS14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *Preprint, arXiv:1412.6572*, 2014.

[HKW95] David P. Helmbold, Jyrki Kivinen, and Manfred K. Warmuth. "Worst-Case Loss Bounds for Single Neurons." In *Advances in Neural Information Processing Systems (NeurIPS)*, 1995.

[HKW99] David P. Helmbold, Jyrki Kivinen, and Manfred K. Warmuth. "Relative loss bounds for single neurons." *IEEE Transactions on Neural Networks*, 1999.

[HLY20] Wei Hu, Zhiyuan Li, and Dingli Yu. "Simple and Effective Regularization Methods for Training on Noisily Labeled Data with Generalization Guarantee." In *International Conference on Learning Representations (ICLR)*, 2020.

[HXA20] Wei Hu, Lechao Xiao, Ben Adlam, and Jeffrey Pennington. "The Surprising Simplicity of the Early-Time Learning Dynamics of Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[HZR16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition." In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

[IMA16] Forrest N. Iandola, Matthew W. Moskewicz, Khalid Ashraf, Song Han, William J. Dally, and Kurt Keutzer. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 1MB model size." *Preprint, arXiv:1602.07360*, 2016.

[JDS20] Ziwei Ji, Miroslav Dudík, Robert E. Schapire, and Matus Telgarsky. "Gradient descent follows the regularization path for general losses." In *Conference on Learning Theory (COLT)*, 2020.

[JGH18] Arthur Jacot, Franck Gabriel, and Clément Hongler. "Neural Tangent Kernel: Convergence and Generalization in Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[JNG19] Chi Jin, Praneeth Netrapalli, Rong Ge, Sham M. Kakade, and Michael I. Jordan. "A Short Note on Concentration Inequalities for Random Vectors with SubGaussian Norm." *Preprint, arXiv:1902.03736*, 2019.

[JT19] Ziwei Ji and Matus Telgarsky. "The implicit bias of gradient descent on nonseparable data." In *Conference on Learning Theory (COLT)*, 2019.

[JT20a] Ziwei Ji and Matus Telgarsky. "Directional convergence and alignment in deep learning." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[JT20b] Ziwei Ji and Matus Telgarsky. "Polylogarithmic width suffices for gradient descent to achieve arbitrarily small test error with shallow ReLU networks." In *International Conference on Learning Representations (ICLR)*, 2020.

[KKK11] Sham M. Kakade, Adam Kalai, Varun Kanade, and Ohad Shamir. "Efficient Learning of Generalized Linear and Single Index Models with Isotonic Regression." In *Advances in Neural Information Processing Systems*, 2011.

[KKM08] Adam Tauman Kalai, Adam R. Klivans, Yishay Mansour, and Rocco A. Servedio. "Agnostically Learning Halfspaces." *SIAM J. Comput.*, **37**(6):1777–1805, 2008.

[KLS09] Adam R. Klivans, Philip M. Long, and Rocco A. Servedio. "Learning Halfspaces with Malicious Noise." *Journal of Machine Learning Research (JMLR)*, **10**(94):2715–2740, 2009.

[KS94] Michael J. Kearns and Robert E. Schapire. "Efficient distribution-free learning of probabilistic concepts." *Journal of Computer and System Sciences*, **48**(3):464 – 497, 1994.

[KS09] Adam Tauman Kalai and Ravi Sastry. "The Isotron Algorithm: High-Dimensional Isotonic Regression." In *Conference on Learning Theory (COLT)*, 2009.

[KSH12] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "ImageNet Classification with Deep Convolutional Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2012.

[KSS94] Michael J Kearns, Robert E Schapire, and Linda M Sellie. "Toward efficient agnostic learning." *Machine Learning*, **17**(2-3):115–141, 1994.

[LL18] Yuanzhi Li and Yingyu Liang. "Learning Overparameterized Neural Networks via Stochastic Gradient Descent on Structured Data." In *Conference on Neural Information Processing Systems*, pp. 8168–8177, 2018.

[LL20] Kaifeng Lyu and Jian Li. "Gradient Descent Maximizes the Margin of Homogeneous Neural Networks." In *International Conference on Learning Representations (ICLR)*, 2020.

[LLW18] Xingguo Li, Junwei Lu, Zhaoran Wang, Jarvis D. Haupt, and Tuo Zhao. "On Tighter Generalization Bound for Deep Neural Networks: CNNs, ResNets, and Beyond." *Preprint, arXiv:1806.05159*, 2018.

[LMZ20] Yuanzhi Li, Tengyu Ma, and Hongyang R. Zhang. "Learning Over-Parametrized Two-Layer ReLU Neural Networks beyond NTK." In *Conference on Learning Theory (COLT)*, 2020.

[LSO19] Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. “Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks.” In *Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.

[LV07] László Lovász and Santosh Vempala. “The Geometry of Logconcave Functions and Sampling Algorithms.” *Random Struct. Algorithms*, **30**(3):307–358, 2007.

[LWM19] Yuanzhi Li, Colin Wei, and Tengyu Ma. “Towards Explaining the Regularization Effect of Initial Large Learning Rate in Training Neural Networks.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[LXX20] Yan Li, Ethan X.Fang, Huan Xu, and Tuo Zhao. “Implicit Bias of Gradient Descent based Adversarial Training on Separable Data.” In *International Conference on Learning Representations (ICLR)*, 2020.

[Mar15] John Markoff. “A Learning Advance in Artificial Intelligence Rivals Human Abilities.” *The New York Times*, 2015.

[MBM18] Song Mei, Yu Bai, and Andrea Montanari. “The landscape of empirical risk for nonconvex losses.” *The Annals of Statistics*, **46**(6A):2747–2774, 12 2018.

[MGW20] Edward Moroshko, Suriya Gunasekar, Blake Woodworth, Jason D. Lee, Nathan Srebro, and Daniel Soudry. “Implicit Bias in Deep Linear Classification: Initialization Scale vs Training Accuracy.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[MM20] Anirbit Mukherjee and Ramchandran Muthukumar. “A Study of Neural Training with Non-Gradient and Noise Assisted Gradient Methods.” *Preprint, arXiv:2005.0421*, 2020.

[MMM19] Song Mei, Theodor Misiakiewicz, and Andrea Montanari. “Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit.” In *Conference on Learning Theory (COLT)*, 2019.

[MMS18] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. “Towards Deep Learning Models Resistant to Adversarial Attacks.” In *International Conference on Learning Representations (ICLR)*, 2018.

[MN06] Pascal Massart, Élodie Nédélec, et al. “Risk bounds for statistical learning.” *The Annals of Statistics*, **34**(5):2326–2366, 2006.

[NBS18] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. “A PAC-Bayesian Approach to Spectrally-Normalized Margin Bounds for Neural Networks.” In *International Conference on Learning Representations (ICLR)*, 2018.

[NKK19] Preetum Nakkiran, Gal Kaplun, Dimitris Kalimeris, Tristan Yang, Benjamin L. Edelman, Fred Zhang, and Boaz Barak. "SGD on Neural Networks Learns Functions of Increasing Complexity." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[PDX20] Hieu Pham, Zihang Dai, Qizhe Xie, Minh-Thang Luong, and Quoc V. Le. "Meta Pseudo Labels." *Preprint, arXiv:2003.10580*, 2020.

[PS86] I. F. Pinelis and A. I. Sakhanenko. "Remarks on Inequalities for Large Deviation Probabilities." *Theory of Probability & Its Applications*, **30**(1):143–148, 1986.

[Ros58] Frank Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." *Psychological review*, **65**(6):386, 1958.

[RR08] Ali Rahimi and Benjamin Recht. "Weighted Sums of Random Kitchen Sinks: Replacing minimization with randomization in learning." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2008.

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms.* Cambridge University Press, New York, NY, USA, 2014.

[Ser99] Rocco A. Servedio. "On PAC Learning Using Winnow, Perceptron, and a Perceptron-like Algorithm." In *Conference on Computational Learning Theory*, p. 296–307, 1999.

[Sha15] Ohad Shamir. "The Sample Complexity of Learning Linear Predictors with the Squared Loss." *Journal of Machine Learning Research*, **16**(108):3475–3486, 2015.

[Sha18] Ohad Shamir. "Are ResNets Provably Better than Linear Predictors?" In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

[Sha20] Ohad Shamir. "Gradient Methods Never Overfit On Separable Data." *Preprint, arXiv:2007.00028*, 2020.

[SHN18] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. "The Implicit Bias of Gradient Descent on Separable Data." *Journal of Machine Learning Research (JMLR)*, **19**(70):1–57, 2018.

[SJL19] Mahdi Soltanolkotabi, Adel Javanmard, and Jason D. Lee. "Theoretical Insights Into the Optimization Landscape of Over-Parameterized Shallow Neural Networks." *IEEE Transactions on Information Theory*, **65**(2):742–769, 2019.

[Slo88] Robert Sloan. "Types of Noise in Data for Concept Learning." In *Conference on Learning Theory (COLT)*, 1988.

[Sol17] Mahdi Soltanolkotabi. “Learning ReLUs via Gradient Descent.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[SSS09] Karthik Sridharan, Shai Shalev-Shwartz, and Nathan Srebro. “Fast rates for regularized objectives.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2009.

[SST10] Nathan Srebro, Karthik Sridharan, and Ambuj Tewari. “Smoothness, Low Noise and Fast Rates.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2010.

[STR20] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. “The Pitfalls of Simplicity Bias in Neural Networks.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[Tia17] Yuandong Tian. “Symmetry-Breaking Convergence Analysis of Certain Two-layered Neural Networks with ReLU nonlinearity.” In *International Conference on Learning Representations (ICLR)*, 2017.

[TL18] Raphael Tang and Jimmy Lin. “Deep Residual Learning for Small-Footprint Keyword Spotting.” In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[Tsy04] Alexander B Tsybakov et al. “Optimal aggregation of classifiers in statistical learning.” *The Annals of Statistics*, **32**(1):135–166, 2004.

[Ver10] Roman Vershynin. “Introduction to the non-asymptotic analysis of random matrices.” *Preprint, arXiv:1011.3027*, 2010.

[VW19] Santosh S. Vempala and John Wilmes. “Gradient Descent for One-Hidden-Layer Neural Networks: Polynomial Convergence and SQ Lower Bounds.” In *Conference on Learning Theory (COLT)*, 2019.

[WGL20] Blake Woodworth, Suriya Gunasekar, Jason D. Lee, Edward Moroshko, Pedro Savarese, Itay Golan, Daniel Soudry, and Nathan Srebro. “Kernel and Rich Regimes in Overparametrized Models.” In *Conference on Learning Theory (COLT)*, 2020.

[WLL19] Colin Wei, Jason D. Lee, Qiang Liu, and Tengyu Ma. “Regularization Matters: Generalization and Optimization of Neural Nets v.s. their Induced Kernel.” In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[Yar17] Dmitry Yarotsky. “Error bounds for approximations with deep ReLU networks.” *Neural Networks*, **94**:103–114, 2017.

[YS20] Gilad Yehudai and Ohad Shamir. "Learning a Single Neuron with Gradient Methods." In *Conference on Learning Theory (COLT)*, 2020.

[ZBH17] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. "Understanding deep learning requires rethinking generalization." In *International Conference on Learning Representations (ICLR)*, 2017.

[ZCZ19] Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. "Gradient descent optimizes over-parameterized deep ReLU networks." *Machine Learning*, 2019.

[ZG19] Difan Zou and Quanquan Gu. "An Improved Analysis of Training Over-parameterized Deep Neural Networks." In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[ZYC19] Huishuai Zhang, Da Yu, Wei Chen, and Tie-Yan Liu. "Training Over-parameterized Deep ResNet Is almost as Easy as Training a Two-layer Network." *Preprint, arXiv:1903.07120*, 2019.

[ZYW19] Xiao Zhang, Yaodong Yu, Lingxiao Wang, and Quanquan Gu. "Learning one-hidden-layer relu networks via gradient descent." In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2019.