EXCESS RISK OF TWO-LAYER RELU NEURAL NETWORKS IN TEACHER-STUDENT SETTINGS AND ITS SUPERIORITY TO KERNEL METHODS

Anonymous authors
Paper under double-blind review

ABSTRACT

While deep learning has outperformed other methods for various tasks, theoretical frameworks that explain its reason have not been fully established. We investigate the excess risk of two-layer ReLU neural networks in a teacher-student regression model, in which a student network learns an unknown teacher network through its outputs. Especially, we consider the student network that has the same width as the teacher network and is trained in two phases: first by noisy gradient descent and then by the vanilla gradient descent. Our result shows that the student network provably reaches a near-global optimal solution and outperforms any kernel methods estimator (more generally, linear estimators), including neural tangent kernel approach, random feature model, and other kernel methods, in a sense of the minimax optimal rate. The key concept inducing this superiority is the non-convexity of the neural network models. Even though the loss landscape is highly non-convex, the student network adaptively learns the teacher neurons.

1 INTRODUCTION

Explaining why deep learning empirically outperforms other methods has been one of the most significant issues. In particular, from the theoretical viewpoint, it is important to reveal the mechanism of how deep learning trained by an optimization method such as gradient descent can achieve superior generalization performance. To this end, we focus on the excess risk of two-layer ReLU neural networks in a nonparametric regression problem and compare its rate to that of kernel methods. One of the difficulties in showing generalization abilities of deep learning is the non-convexity of the associated optimization problem Li et al. (2018), which may let the solution stacked in a bad local minimum. To alleviate the non-convexity of neural network optimization, recent studies focus on over-parameterization as the promising approaches. Indeed, it is fully exploited by (i) Neural Tangent Kernel (NTK) (Jacot et al., 2018; Allen-Zhu et al., 2019; Arora et al., 2019; Du et al., 2019; Weinan et al., 2020; Zou et al., 2020) and (ii) mean field analysis (Nitanda & Suzuki, 2017; Chizat & Bach, 2018; Mei et al., 2019; Tzen & Raginsky, 2020; Chizat, 2021; Suzuki & Akiyama, 2021).

In the setting of NTK, a relatively large-scale initialization is considered. Then the gradient descent related to parameters of neural networks can be reduced to the convex optimization in RKHS, which is easier to analyze. However, in this regime, it is hard to explain the superiority of deep learning because the estimation ability of the obtained estimator is reduced to that of the corresponding kernel. From this perspective, recent works focus on the "beyond kernel" type analysis Allen-Zhu & Li (2019); Bai & Lee (2020); Li et al. (2020); Chen et al. (2020); Refinetti et al. (2021); Abbe et al. (2022). Although their analysis shows the superiority of deep learning to kernel methods in each setting, in terms of the sample size (n), all derived bounds are essentially $\Omega(1/\sqrt{n})$. This bound is known to be sub-optimal for regression problems Caponnetto & De Vito (2007).

In the mean field analysis setting, a kind of continuous limit of the neural network is considered, and its convergence to some specific target functions has been analyzed. This regime is more suitable in terms of a "beyond kernel" perspective, but it essentially deals with a continuous limit and hence is difficult to control the discretization error when considering a teacher network with a finite width. Indeed, the optimization complexity has been exploited recently in some research, but it still requires an exponential time complexity in the worst case (Mei et al., 2018b; Hu et al., 2019; Nitanda et al., 2021a). This problem is mainly due to the lack of landscape analysis that requires closer exploitation

of the problem structure. For example, we may consider the *teacher student setting* where the true function is represented as a neural network. This allows us to use the landscape analysis in the optimization analysis and give a more precise analysis of the statistical performance. In particular, we can obtain a more precise characterization of the *excess risk* (e.g., Suzuki & Akiyama (2021)).

More recently, some studies have focused on the *feature learning* ability of neural networks (Abbe et al., 2021; 2022; Chizat & Bach, 2020; Ba et al., 2022; Nguyen, 2021). Among them, Abbe et al. (2021) considers estimation of the function with staircase property and multi-dimensional Boolean inputs and shows that neural networks can learn that structure through stochastic gradient descent. Moreover, Abbe et al. (2022) studies a similar setting and shows that in a high-dimensional setting, two-layer neural networks with sufficiently smooth activation can outperform the kernel method. However, obtained bound is still $O(1/\sqrt{n})$ and requires a higher smoothness for activation as the dimensionality of the Boolean inputs increases.

The teacher-student setting is one of the most common settings for theoretical studies, e.g., (Tian, 2017; Safran & Shamir, 2018; Goldt et al., 2019; Zhang et al., 2019; Safran et al., 2021; Tian, 2020; Yehudai & Shamir, 2020; Suzuki & Akiyama, 2021; Zhou et al., 2021; Akiyama & Suzuki, 2021) to name a few. Zhong et al. (2017) studies the case where the teacher and student have the same width, shows that the strong convexity holds around the parameters of the teacher network and proposes a special tensor method for initialization to achieve the global convergence to the global optimal. However, its global convergence is guaranteed only for a special initialization which excludes a pure gradient descent method. Safran & Shamir (2018) empirically shows that gradient descent is likely to converge to non-global optimal local minima, even if we prepare a student that has the same size as the teacher. More recently, Yehudai & Shamir (2020) shows that even in the simplest case where the teacher and student have the width *one*, there exist distributions and activation functions in which gradient descent fails to learn. Safran et al. (2021) shows the strong convexity around the parameters of the teacher network in the case where the teacher and student have the same width for Gaussian inputs. They also study the effect of over-parameterization and show that over-parameterization will change the spurious local minima into the saddle points. However, it should be noted that this does not imply that gradient descent can reach the global optima. Akiyama & Suzuki (2021) shows that the gradient descent with a sparse regularization can achieve the global optimal solution for an over-parameterized student network. Thanks to the sparse regularization, the global optimal solution can exactly recover the teacher network. However, this research requires a highly over-parameterized network. Indeed, it requires an exponentially large number of widths in terms of the dimensionality and the sample size. Moreover, they impose quite strong assumptions such that there is no observation noise and the parameter of each neuron in the teacher network should be orthogonal to each other.

The superiority of deep learning against kernel methods has also been discussed in the nonparametric statistics literature. They show the minimax optimality of deep learning in terms of excess risk. Especially a line of research (Schmidt-Hieber, 2020; Suzuki, 2018; Hayakawa & Suzuki, 2020; Suzuki & Nitanda, 2021; Suzuki & Akiyama, 2021) shows that deep learning achieves faster rates of convergence than linear estimators in several settings. Here, the linear estimators are a general class of estimators that includes kernel ridge regression, k-NN regression, and Nadaraya-Watson estimator. Among them, Suzuki & Akiyama (2021) treats a tractable optimization algorithm in a teacher-student setting, but they require an exponential computational complexity and smooth activation function, which does not include ReLU.

In this paper, we consider a gradient descent with two phases, a noisy gradient descent first and a vanilla gradient descent next. Our analysis shows that through this method, the student network recovers the teacher network in a polynomial order computational complexity (with respet to the sample size) without using an exponentially wide network, even though we do not need the strong assumptions such as the no-existence of noise and orthogonality. Moreover, we evaluate the excess risk of the trained network and show that the trained network can outperform any linear estimators, including kernel methods, in terms of its dependence on the sample size. More specifically, our contributions can be summarized as follows:

- We show that by two-phase gradient descent, the student network, which has the same width as the teacher network, provably reaches the near-optimal solution. Moreover, we conduct a refined analysis of the excess risk and provide the upper bound for the excess risk of the student network, which is much faster than that obtained by the generalization bound

analysis with the Rademacher complexity argument. Throughout this paper, our analysis does not require the highly over-parameterization and any special initialization schemes.

- We provide a comparison of the excess risk between the student network and linear estimators and show that while the linear estimators much suffer from the curse of dimensionality, the student network less suffers from that. Particularly, in high dimensional settings, the convergence rate of the excess risk of any linear estimators becomes close to $O(n^{-1/2})$, which coincides with the classical bound derived by the Rademacher complexity argument.
- The lower bound of the excess risk derived in this paper is valid for any linear estimator. The analysis is considerably general because the class of linear estimators includes kernel ridge regression with any kernel. This generality implies that the derived upper bound cannot be derived by the argument that uses a fixed kernel, including Neural Tangent Kernel.

2 Problem Settings

Notations For $m \in \mathbb{N}$, let $[m] := \{1, \dots, m\}$. For $x \in \mathbb{R}^d$, $\|x\|$ denotes its Euclidean norm. We denote the inner product between $x, y \in \mathbb{R}^d$ by $\langle x, y \rangle = \sum_{j=1}^d x_i y_i$. $\mathbb{S}^{d-1}$ denotes the unit sphere in $\mathbb{R}^d$. For a matrix W, we denote its operator and Frobenius norm by $\|W\|_2$ and $\|W\|_F$, respectively.

Here, we introduce the problem setting and the model that we consider in this paper. We focus on a regression problem where we observe n training examples $D_n = (x_i, y_i)_{i=1}^n$ generated by the following model for an unknown measurable function $f^\circ : \mathbb{R}^d \to \mathbb{R}$:

$$y_i = f^\circ(x_i) + \epsilon_i,$$

where $(x_i)_{i=1}^n$ is independently identically distributed sequence from P_X that is the uniform distribution over $\Omega = \mathbb{S}^{d-1}$, and ϵ_i are i.i.d. random variables satisfying $\mathbb{E}[\epsilon_i] = 0$, $\mathbb{E}[\epsilon_i^2] = v^2$, and $|\epsilon_i| \le U$ a.s.. Our goal is to estimate the *true* function f° through the training data. To this end, we consider the square loss $\ell(y, f(x)) = (y - f(x))^2$ and define the expected risk and the empirical risk as $\mathcal{L}(f) := \mathbb{E}_{X,Y}[\ell(Y, f(X)]$ and $\widehat{\mathcal{L}}(f) := \frac{1}{n}\ell(y_i, f(x_i))$, respectively. In this paper, we measure the performance of an estimator $\widehat{f}$ by the *excess risk* $\mathcal{L}(\widehat{f}) - \inf_{f:\text{measurable}} \mathcal{L}(f)$. Since $\inf \mathcal{L}(f) = \mathcal{L}(f^\circ) = 0$, we can check that the excess risk coincides with $\|\widehat{f} - f^\circ\|_{L_2(P_X)}^2$, the L_2-distance between $\widehat{f}$ and f°. We remark that the excess risk is different from the generalization gap $\mathcal{L}(\widehat{f}) - \widehat{\mathcal{L}}(\widehat{f})$. Indeed, when considering the convergence rate with respect to n, the generalization gap typically converges to zero with $O(1/\sqrt{n})$ Wainwright (2019). On the other hand, the excess risk can converge with the rate faster than $O(1/\sqrt{n})$, which is known as *fast learning rate*.

2.1 Model of true functions

To evaluate the excess risk, we introduce a function class in which the true function f° is included. In this paper, we focus on the teacher-student setting with two-layer ReLU neural networks, in which the true function (called *teacher*) is given by

$$f_{a^\circ, W^\circ}(x) = \sum_{j=1}^m a_j^\circ \sigma(\langle w_j^\circ, x \rangle),$$

where $\sigma(u) = \max\{u, 0\}$ is the ReLU activation, m is the width of the teacher model satisfying $m \le d$, and $a_j^\circ \in \mathbb{R}$, $w_j^\circ \in \mathbb{R}^d$ for $j \in [m]$ are its parameters. We impose several conditions for the parameters of the teacher networks. Let $W^\circ = (w_1^\circ\ w_2^\circ\ \cdots\ w_m^\circ) \in \mathbb{R}^{d \times m}$ and $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_m$ be the singular values of W°. First, we assume that $a_j^\circ \in \{\pm 1\}$ for any $j \in [m]$. Note that by 1-homogeneity of the ReLU activation[1], this condition does not restrict the generality of the teacher networks. Moreover, we assume that there exists $\sigma_{\min} > 0$ such that $\sigma_m > \sigma_{\min}$. If $\sigma_m = 0$, there exists an example in which f_{a°, W° has multiple representations. Indeed, Zhou et al. (2021) shows that in the case $a_j^\circ = 1$ for all $j \in [m]$ and $\sum w_j^\circ = 0$, it holds that $f_{a^\circ, W^\circ} = \sum_{j=1}^m \sigma(\langle w_j^\circ, x \rangle) = \sum_{j=1}^m \sigma(\langle -w_j^\circ, x \rangle)$. Hence, throughout this paper, we focus on the estimation problem in which the true function is included in the following class:

$$\mathcal{F}^\circ := \{f_{a^\circ, W^\circ} \mid a^\circ \in \{\pm 1\}^m, \|W^\circ\|_2 \le 1, \sigma_m > \sigma_{\min}\}. \tag{1}$$

This class represents the two-layer neural networks with the ReLU activation whose width is at most the dimensionality of the inputs. The constraint $\|W^\circ\|_2 \le 1$ is assumed only for the analytical simplicity and can be extended to any positive constants.

[1] $\sigma(\langle w, x \rangle) = \|w\| \sigma(\langle w/\|w\|, x \rangle)$ for any $w \in \mathbb{R}^d / \{\mathbf{0}\}$ and $x \in \mathbb{R}^d$.

3 ESTIMATORS

In this section, we introduce the classes of estimators: linear estimators and neural networks (student networks) trained by two-phase gradient descent. The linear estimator is introduced as a generalization of the kernel method. We will show separation between *any* linear estimator and neural networks by giving a suboptimal rate of the excess risk for the linear estimators (Theorem 4.1), which simultaneously gives separation between the kernel methods and the neural network approach. A detailed comparison of the excess risk of these estimators will be conducted in Section 4.

3.1 LINEAR ESTIMATORS

Given observation $(x_1, y_1), \dots, (x_n, y_n)$, an estimator $\widehat{f}$ is called *linear* if it is represented by $\widehat{f}(x) = \sum_{i=1}^n y_i \varphi_i(x_1, \dots, x_n, x)$, where $(\varphi_i)_{i=1}^n$ is a sequence of measurable and $L_2(P_{\mathcal{X}})$-integrable functions. The most important example in this study is the kernel ridge regression estimator. We note that the kernel ridge estimator is given by $\widehat{f}(x) = Y^\mathsf{T}(K_X + \lambda I)^{-1}\mathbf{k}(x)$, where $K_X = (\mathbf{k}(x_i, x_j))_{i,j=1}^n \in \mathbb{R}^{n\times n}$, $\mathbf{k}(x) = [\mathbf{k}(x, x_1), \dots, \mathbf{k}(x, x_n)]^\mathsf{T} \in \mathbb{R}^n$ and $Y = [y_1, \dots, y_n]^\mathsf{T} \in \mathbb{R}^n$ for a kernel function $\mathbf{k} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, which is linear to the output observation Y. Since this form is involved in the definition of linear estimators, the kernel ridge regression with any kernel function can be seen as one of the linear estimators. The choice of φ_i is arbitrary, and thus the choice of the kernel function is also arbitrary. Therefore, we may choose the best kernel function before we observe the data. However, as we will show in Theorem 4.1, it suffers from a suboptimal rate. Other examples include the k-NN estimator and the Nadaraya-Watson estimator. Thus our analysis gives a suboptimality of not only the kernel method but also these well-known linear estimators. Suzuki (2018); Hayakawa & Suzuki (2020) utilized such an argument to show the superiority of deep learning but did not present any tractable optimization algorithm.

3.2 STUDENT NETWORKS TRAINED BY TWO-PHASE GRADIENT DESCENT

We prepare the neural network trained through the observation data (called *student*), defined by

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

where $\theta = ((a_1, w_1), \dots (a_m, w_m)) \in \mathbb{R}^{(d+1)m} =: \Theta$. We assume that the student and teacher networks have the same width. Based on this formulation, we aim to train the parameter θ that will be provably close to that of the teacher network. To this end, we introduce the training algorithm, two-phase gradient descent, which we consider in this paper.

Phase I: noisy gradient descent For $r \in \mathbb{R}$, let $\bar{r} := R \cdot \tanh(r|r|/2R)$ be a clipping of r, where $R > 1$ is a fixed constant. In the first phase, we conduct a noisy gradient descent with the weight decay regularization. The objective function used to train the student network is given as follows:

$$\widehat{\mathcal{R}}_\lambda(\theta) := \tfrac{1}{2n} \textstyle\sum_{i=1}^n (y_i - f(x_i; \bar{\theta}))^2 + \lambda \sum_{j=1}^m \Big(|a_j|^2 + \|w_j\|^2 \Big),$$

where $\bar{\theta}$ is the element-wise clipping of θ and $\lambda > 0$ is a regularization parameter. The parameter clipping ensures the bounded objective value and smoothness of the expected risk around the origin, which will be helpful in our analysis. Then, the parameters of the student network are updated by

$$\theta^{(k+1)} = \theta^{(k)} - \eta^{(1)} \nabla \widehat{\mathcal{R}}_\lambda\big(\theta^{(k)}\big) + \sqrt{\tfrac{2\eta^{(1)}}{\beta}} \zeta^{(k)},$$

where $\eta^{(1)} > 0$ is a step-size, $\{\zeta^{(k)}\}_{k=1}^\infty$ are independently identically distributed noises from the standard normal distribution, and $\beta > 0$ is a constant called *inverse temperature*. This type of noisy gradient descent is called *gradient Langevin dynamics*. It is known that by letting β be large, we can ensure that the smooth objective function will decrease. On the other hand, because of the non-smoothness of the ReLU activation, the objective function $\widehat{\mathcal{R}}_\lambda$ is also non-smooth. Hence it is difficult to guarantee the small objective value. To overcome this problem, we evaluate the expected one instead in the theoretical analysis, which is given by

$$\mathcal{R}_\lambda(\theta) := \tfrac{1}{2}\mathbb{E}_x \Big[\big(f_{a^\circ, W^\circ}(x) - f(x; \bar{\theta}) \big)^2 \Big] + \lambda \textstyle\sum_{j=1}^m \Big(|a_j|^2 + \|w_j\|^2 \Big).$$

We can ensure a small $\mathcal{R}_\lambda(\theta)$ after a sufficient number of iterations (see Section 4.2 for the detail).

Algorithm 1 Two-Phase Gradient Descent

Input: max iteration $k_{\max}^{(1)}$ and $k_{\max}^{(2)}$, stepsize parameter $\eta^{(1)}$, $\eta^{(2)} > 0$, regularization parameter $\lambda > 0$, inverse temperature $\beta > 0$.

Initialization: $\theta^{(0)} \sim \rho_0$.
for $k = 1, 2, \ldots, k_{\max}^{(1)}$ **do**
  $\zeta^{(k)} \sim \mathcal{N}(0, I_{m(d+1)})$
  $\theta^{(k+1)} = \theta^{(k)} - \eta^{(1)} \nabla \widehat{\mathcal{R}}_\lambda(\theta^{(k)}) + \sqrt{\frac{2\eta^{(1)}}{\beta}} \zeta^{(k)}$
end for
Reparameterization: $a_j^{(k)} = \mathrm{sgn}(\bar{a}_j^{(k)})$, $w_j^{(k)} = \left|\bar{a}_j^{(k)}\right| \bar{w}_j^{(k)}$
for $k = k_{\max}^{(1)} + 1, k_{\max}^{(1)} + 2, \ldots, k_{\max}^{(2)}$ **do**
  $W^{(k+1)} = W^{(k)} - \eta^{(2)} \nabla_W \widehat{\mathcal{R}}(W^{(k)})$
end for

Phase II: vanilla gradient descent After phase I, we can ensure that for each node of the student network, there is a node of the teacher network that is relatively close to each other. Then we move to conduct the vanilla gradient descent to estimate the parameters of the teacher more precisely. Before conducting the gradient descent, we rescale the parameters as follows:

$$a_j^{(k)} \leftarrow \mathrm{sgn}(\bar{a}^{(k)}),\ w_j^{(k)} \leftarrow \left|\bar{a}^{(k)}\right| \bar{w}_j{}^{(k)}, \qquad \forall j \in [m].$$

We note this transformation does not change the output of the student network thanks to the 1-homogeneity of the ReLU activation. After that, we update the parameters of the first layer by

$$W^{(k+1)} = W^{(k)} - \eta^{(2)} \nabla_W \widehat{\mathcal{R}}(W^{(k)}),$$

where $\eta^{(2)} > 0$ is a step-size different from $\eta^{(1)}$ and $\widehat{\mathcal{R}}(W) := \frac{1}{2n} \sum_{i=1}^n (y_i - f(x_i; \theta))^2$. In this phase, we no longer need to update the parameters of both layers. Moreover, the regularization term and the gradient noise added in phase I are also unnecessary. These simplifications of the optimization algorithm are based on the strong convexity of $\widehat{\mathcal{R}}(W)$ around W°, the parameters of the teacher network. The analysis for this local convergence property is based on that of Zhang et al. (2019), and eventually, we can evaluate the excess risk of the student network.

The overall training algorithm can be seen in Algorithm 1. In summary, we characterize the role of each phase as follows: in phase I, the student network explore the parameter space globally and finds the parameters that are relatively close to that of teachers, and in phase II, the vanilla gradient descent for the first layer outputs more precise parameters, as we analyze in Section 4.2.

Remark 3.1. *Akiyama & Suzuki (2021) also considered the convergence of the gradient descent in a teacher-student model. They considered a sparse regularization, $\sum_{j=1}^m |a_j| \|w_j\|$, for the ReLU activation while we consider the L_2-regularization given by $\sum_{j=1}^m (|a_j|^2 + \|w_j\|^2)$. These two regularizations are essentially the same since the minimum of the later regularization under the constraint of $|a_j| \|w_j\| = $ const. is given by $2 \sum_{j=1}^m |a_j| \|w_j\|$ by the arithmetic-geometric mean relation. On the other hand, Akiyama & Suzuki (2021) consider a vanilla gradient descent instead of the noisy gradient descent. This makes it difficult to reach the local region around the optimal solution, and their analysis required an exponentially large width to find the region. We may use a narrow network in this paper with the same width as the teacher network. This is due to the ability of the gradient Langevin dynamics to explore the entire space and find the near global optimal solution.*

4 Excess Risk Analysis and Its Comparison

This section provides the excess risk bounds for linear estimators and the deep learning estimator (the trained student network). More precisely, we give its lower bound for linear estimators and upper bound for the student network. As a consequence, it will be provided that the student network achieves a faster learning rate and less hurt from a curse of dimensionality than linear estimators.

4.1 Minimax Lower Bound for Linear Estimators

Here, we analyze the excess risk of linear estimators and introduce its lower bound. More specifically, we consider the minimax excess risk over the class of linear estimators given as follows:

$$R_{\mathrm{lin}}(\mathcal{F}^\circ) = \inf_{\widehat{f}:\text{linear}} \sup_{f^\circ \in \mathcal{F}^\circ} \mathbb{E}_{D_n}[\|\widehat{f} - f^\circ\|_{L_2(P_{\mathcal{X}})}^2],$$

where the infimum is taken over all linear estimators, and the expectation is taken for the training data. This expresses the infimum of worst-case error over the class of linear estimators to estimate a function class $\mathcal{F}^\circ$. Namely, any class of linear estimators cannot achieve a faster excess risk than $R_{\mathrm{lin}}(\mathcal{F}^\circ)$. Based on this concept, we provide our result about the excess risk bound for linear estimators. Under the definition of $\mathcal{F}^\circ$ by Eq. (1), we can obtain the lower bound as follows:

Theorem 4.1. *For arbitrary small $\kappa > 0$, we have that*

$$R_{\mathrm{lin}}(\mathcal{F}^\circ) \gtrsim n^{-\frac{d+2}{2d+2}} n^{-\kappa}.$$

The proof can be seen in Appendix A. This theorem implies that under $d \geq 2$, the convergence rate of excess risk is at least slower than $n^{-\frac{2+2}{2\cdot 2+2}} = n^{-2/3}$. Moreover, since $-\frac{d+2}{2d+2} \to -1/2$ as $d \to \infty$, the convergence rate of excess risk will be close to $n^{-1/2}$ in high dimensional settings, which coincides with the generalization bounds derived by the Rademacher complexity argument. Hence, we can conclude that the linear estimators suffer from the curse of dimensionality.

The key strategy to show this theorem is the following "convex-hull argument" given as follows:

$$R_{\mathrm{lin}}(\mathcal{F}^\circ) = R_{\mathrm{lin}}(\overline{\mathrm{conv}}(\mathcal{F}^\circ)),$$

where $\mathrm{conv}(\mathcal{F}^\circ) := \{\sum_{j=1}^N \lambda_j f_j \mid N \in \mathbb{N}, f_j \in \mathcal{F}^\circ, \lambda_j \geq 0, \sum_{j=1}^N \lambda_j = 1\}$ and $\overline{\mathrm{conv}}(\cdot)$ is the closure of $\mathrm{conv}(\cdot)$ in $L_2(P_{\mathcal{X}})$. By combining this argument with the minimax optimal rate analysis exploited in Zhang et al. (2002) for linear estimators, we obtain the rate in Theorem 4.1.

This equality implies that the linear estimators cannot distinguish the original class $\mathcal{F}^\circ$ and its convex hull $\overline{\mathrm{conv}}(\mathcal{F}^\circ)$. Therefore, if the function class $\mathcal{F}^\circ$ is highly non-convex, then the linear estimators result in a much slower convergence rate since $\overline{\mathrm{conv}}(\mathcal{F}^\circ)$ will be much larger than that of the original class $\mathcal{F}^\circ$. Indeed, we can show that the convex hull of the teacher network class is *considerably larger* than the original function class, which causes the curse of dimensionality. For example, the mean of two teacher networks with a width m can be a network with width $2m$, which shows that $\mathrm{conv}(\mathcal{F}^\circ)$ can consist of much wider networks. See Appendix A for more details.

4.2 Excess risk of the neural networks

Here, we give an upper bound of the excess risk of the student network trained by Algorithm 1. The main result is shown in Theorem 4.5, which states that the student network can achieve the excess risk with $O(n^{-1})$. To address this consequence, we provide a convergence guarantee for phase I and phase II in Algorithm 1. We first show that by phase I, the value of $\mathcal{R}_\lambda(\theta^{(k)})$ will be sufficiently small (see Proposition 4.3). Then, we can show that the parameters of the student network and the teacher networks are close to each other as Proposition 4.4. By using the strong convexity around the parameters of the teacher network, the convergence of phase II is ensured.

Convergence in phase I: First, we provide a convergence result and theoretical strategy of the proof for phase I. Since the ReLU activation is non-smooth, the loss function $\widehat{\mathcal{R}}_\lambda(\cdot)$ is also non-smooth. Therefore it is difficult to ensure the convergence of the gradient Langevin dynamics. To overcome this problem, we evaluate the value of $\mathcal{R}_\lambda(\cdot)$ instead by considering the following update:

$$\theta^{(k+1)} = \theta^{(k)} - \eta^{(1)} \nabla \mathcal{R}_\lambda\big(\theta^{(k)}\big) + \sqrt{\tfrac{2\eta^{(1)}}{\beta}} \zeta^{(k)},$$

and bound the residual due to using the gradient of $\widehat{\mathcal{R}}_\lambda(\cdot)$. This update can be interpreted as the discretization of the following stochastic differential equation:

$$\mathrm{d}\theta = -\beta \nabla \mathcal{R}_\lambda(\theta)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t,$$

where $(B_t)_{t\geq 0}$ is the standard Brownian motion in $\Theta(= \mathbb{R}^{(d+1)m})$. It is known that this process has a unique invariant distribution π_∞ that satisfies $\frac{\mathrm{d}\pi_\infty}{\mathrm{d}\theta}(\theta) \propto \exp(-\beta \mathcal{R}_\lambda(\theta))$. Intuitively, as $\beta \to \infty$, this invariant measure concentrates around the minimizer of $\mathcal{R}_\lambda$. Hence, by letting β sufficiently large, obtaining a near-optimal solution will be guaranteed.

Such a technique for optimization is guaranteed in recent works (Raginsky et al., 2017; Erdogdu et al., 2018). However, as we stated above, they require a smooth objective function. Therefore we cannot use the same technique here directly. To overcome this difficulty, we evaluate the difference between $\nabla \widehat{\mathcal{R}}_\lambda$ and $\nabla \mathcal{R}_\lambda$ as follows:

Lemma 4.2. *There exists a constant $C > 0$ such that with probability at least $1 - \delta$, it holds that*

$$V_{grad} := \sup_{\theta} \left\| \nabla \mathcal{R}_\lambda(\theta) - \nabla \widehat{\mathcal{R}}_\lambda(\theta) \right\| \leq C R^3 m \sqrt{\frac{d \log(mdn/\delta)}{n}}.$$

This lemma implies that with high probability, the difference between $\nabla \widehat{\mathcal{R}}_\lambda$ and $\nabla \mathcal{R}_\lambda$ will vanish as $n \to \infty$. Thanks to this lemma, we can connect the dynamics of the non-smooth objective with that of the smooth objective and import the convergence analysis developed so far in the smooth objective. In particular, we utilize the technique developed by Vempala & Wibisono (2019) (see Appendix C for more details). We should note that our result extends the existing one Vempala & Wibisono (2019) in the sense that it gives the convergence for the non-differential objective function $\widehat{\mathcal{R}}_\lambda(\cdot)$. As a consequence, we obtain the following convergence result as for phase I.

Proposition 4.3. *Let $\mathcal{R}_\lambda^*$ be the minimum value of $\mathcal{R}_\lambda$ in Θ, q be a density function of π_∞ (i.e., $q(\theta) \propto \exp(-\beta \mathcal{R}_\lambda(\theta))$) and $H_q(p) := \int_{\mathbb{R}} p(\theta) \log \frac{p(\theta)}{q(\theta)} \mathrm{d}x$ be the KL-divergence. There exists a constant c, $C > 0$ and the log-Sobolev constant α (defined in Lemma C.4) such that with step-size $0 < \eta^{(1)} < c \frac{\delta \lambda \alpha}{\beta R^3 m^3 d}$, after $k^{(1)} \geq \frac{\beta}{\alpha \eta^{(1)}} \log \frac{2 H_q(\rho_0)}{\delta}$ iteration, the output $\theta^{(k)}$ satisfies*

$$\mathbb{E}[\mathcal{R}_\lambda(\theta^{(k)})] - \mathcal{R}_\lambda^* \leq C \Big[(\lambda + m) \exp\big(m^2 \beta\big) \sqrt{\delta + \tfrac{1}{3n\lambda}} + \tfrac{d}{2\beta} \log\Big(\tfrac{m^3 d \beta}{\lambda} \Big) \Big]$$

with probability at least $1 - \delta$, where the expectation is taken over the initialization and Gaussian random variables added in the algorithm.

Therefore, we can see that phase I optimization can find a *near optimal solution* with a polynomial time complexity (with respect to n) even though the objective function is non-smooth. It also may be considered to use the gradient Langevin dynamics to reach the global optimal solution by using higher β. However, it requires increasing the inverse temperature β exponentially related to n and other parameters, which leads to exponential computational complexity. To overcome this difficulty, we utilize the local landscape of the objective function. We can show the objective function will be strongly convex around the teacher parameters and we do not need to use the gradient noise and any regularization. Indeed, we can show that the vanilla gradient descent can reach the global optimal solution in phase II, as shown in the following.

Convergence in phase II and excess risk of the student network: Next, we prove the convergence guarantee of phase II and provide an upper bound of the excess risk. The convergence result is based on the fact that when $\mathcal{R}_\lambda(\theta)$ is small enough (guaranteed in Proposition 4.3), the parameters of the student network will be close to those of the teacher network, as the following proposition:

Proposition 4.4. *There exists a threshold $\epsilon_0 = \mathrm{poly}(m^{-1}, \sigma_{\min})$ such that by letting $\lambda \leq \epsilon_0 / m$, if $\epsilon = \mathcal{R}_\lambda(\theta) - \mathcal{R}_\lambda^* \leq \epsilon_0$, it holds that for every $j \in [m]$, there exists $k_j \in [m]$ such that $\mathrm{sgn}(a_{k_j}) = a_j^\circ$ and $\big\| |a_{k_j}| w_{k_j} - w_j^\circ \big\| \leq c \sigma_m / \kappa^3 m^3$.*

The proof of this proposition can be seen in Appendix D. We utilize the technique in Zhou et al. (2021), which give the same results to the cases when the activation is the absolute value function. In this proposition, we compare the parameters of the teacher network with the *normalized* student parameters. This normalization is needed because of the 1-homogeneity of the ReLU activation. The inequality $\big\| |a_{k_j}| w_{k_j} - w_j^\circ \big\| \leq c \sigma_m / \kappa^3 m^3$ ensures the closeness of parameters in the sense of the *direction* and the *amplitude*. Combining this with the equality $\mathrm{sgn}(a_{k_j}) = a_j^\circ$, we can conclude the closeness and move to ensure the local convergence. Thanks to this closeness and local strong convexity, we can ensure the convergence in phase II as follows:

Theorem 4.5. *There exists $\epsilon_0 = \mathrm{poly}(m^{-1}, \sigma_{\min})$ and constants C and $C' > 0$ such that for any $0 < b < 1$, under $n \geq \lambda \sigma_b^{-3} \exp\big(\sigma_b^{-1} m^2\big)$ where $\sigma_b := b \epsilon_0$, let $k^{(1)} = C \lambda^{-2} \beta^{-1} \exp\big(m^2 \beta\big)$ and $k^{(2)} = k^{(1)} + \log\big(C' n \eta^{(2)-2}\big)$, then the output of Algorithm 1 with $\lambda = \sigma_b d^{-1}$, $\beta = \Omega(\sigma_b^{-1} d)$, $\eta^{(1)} = O(\lambda \sigma_b^3 \exp\big(\sigma_b^{-1} m^2\big))$ and $\eta^{(2)} = O(\sigma_{\min} m^{-2})$ satisfies*

$$\|\widehat{f} - f^\circ\|_{L_2(P_{\mathcal{X}})}^2 \lesssim \frac{\tilde{\sigma}^2 \sigma_{\min}^{-4} m^5 \log n}{n}$$

with probability at least $1 - b - d^{-10}$, where $\tilde{\sigma} = (\prod_{j=1}^m \sigma_j) / \sigma_m^m$.

The proof of this theorem also can be seen in Appendix D. This theorem implies that for fixed m, the excess risk of the student networks is bounded by $\mathbb{E}_{D_n}[\|\widehat{f} - f^\circ\|_{L_2(P_{\mathcal{X}})}^2] \lesssim n^{-1}$. As compared

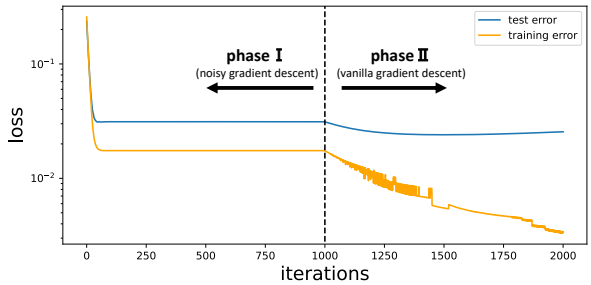

Figure 1: Convergence of the training loss and test loss.

to the lower bound derived for linear estimators in Theorem 4.1, we get the faster rate n^{-1} in terms of the sample size. Moreover, the dependence of the excess risk on the dimensionality d does not appear explicitly. Therefore we can conclude that the student network less suffers from the curse of dimensionality than linear estimators. As we pointed out in the previous subsection, the convex hull argument causes the curse of dimensionality for linear estimators since they only prepare a fixed basis. On the other hand, the student network can "find" the basis of the teacher network via noisy gradient descent in phase I and eventually avoid the curse of dimensionality.

Remark 4.6. *Akiyama & Suzuki (2021) establishes the local convergence theory for the student wider than the teacher. However, it cannot apply here since they only consider the teacher whose parameters are orthogonal to each other. Suzuki & Akiyama (2021) also showed the benefit of the neural network and showed the superiority of deep learning in a teacher-student setting where the teacher has infinite width. They assume that the teacher has decaying importance; that is, the teacher can be written as $f^\circ(x) = \sum_{j=1}^{\infty} a_j^\circ \sigma(\langle w_j^\circ, x\rangle)$ where $a_j^\circ \lesssim j^{-a}$ and $w_j^\circ \lesssim j^{-b}$ (with an exponent $a, b > 0$) for a bounded smooth activation σ. Our analysis does not assume the decay of importance, and the activation function is the non-differential ReLU function. Moreover, Suzuki & Akiyama (2021) considers a pure gradient Langevin dynamics instead of the two-stage algorithm, which results in the exponential computational complexity in contrast to our analysis.*

5 Numerical Experiment

In this section, we conduct a numerical experiment to justify our theoretical results. We apply Algorithm 1 to the settings $d = m = 50$. For the teacher network, we employ $a_j^\circ = 1$ for $1 \le j \le 25$, $a_j^\circ = -1$ for $26 \le j \le 50$ and $(w_1^\circ, \dots, w_{50}^\circ) = I_{50}$ as its parameters. The parameters of the student network are initialized by $\theta^{(0)} \sim \mathcal{N}(0, I_{m(d+1)})$. We use the sample with $n = 1000$ as the training data. Hyperparameters are set to $\eta^{(1)} = \eta^{(2)} = 0.01$, $\beta = 100$, $\lambda = 0.01$, $k_{\max}^{(1)} = 1000$ and $k_{\max}^{(2)} = 2000$. Figure 1 shows the experimental result. The orange line represents the training loss without the regularization term. The blue line represents the test loss. Since we can compute the generalization error analytically (see Appendix B), we utilize its value as the test loss.

We can see that in phase I, both the training and test losses decrease first and then fall flat. At the beginning of phase II, we can observe that both the training and test losses decrease linearly. This reflects the strong convexity around the parameters of the teacher network, as we stated in the convergence guarantee of phase II. While the training loss keeps going up and down, the curve of the test loss is relatively smooth. This difference is due to the smoothness of the generalization loss (or $\mathcal{R}_\lambda$), which we use in the convergence analysis in phase I. The test loss does not keep decreasing and converges to a constant. The existence of the sample noise causes this phenomenon: even if the parameters of the student coincide with that of the teacher, its training loss will not be zero. Thus we can say that the numerical experiment is consistent with our theoretical results.

6 Conclusion

In this paper, we focus on the nonparametric regression problem, in which a true function is given by a two-layer neural network with the ReLU activation, and evaluate the excess risks of linear estimators and neural networks trained by two-phase gradient descent. Our analysis revealed that while any linear estimator suffers from the curse of dimensionality, deep learning can avoid it and outperform linear estimators, which include the neural tangent kernel approach, random feature model, and other kernel methods. Essentially, the non-convexity of the model induces this difference.

REFERENCES

Emmanuel Abbe, Enric Boix-Adsera, Matthew S Brennan, Guy Bresler, and Dheeraj Nagaraj. The staircase property: How hierarchical structure can guide deep learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 26989–27002. Curran Associates, Inc., 2021.

Emmanuel Abbe, Enric Boix-Adsera, and Theodor Misiakiewicz. The merged-staircase property: a necessary and nearly sufficient condition for sgd learning of sparse functions on two-layer neural networks. *arXiv preprint arXiv:2202.08658*, 2022.

Shunta Akiyama and Taiji Suzuki. On learnability via gradient method for two-layer relu neural networks in teacher-student setting. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 152–162. PMLR, 2021.

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

Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. In *International Conference on Machine Learning*, pp. 242–252. PMLR, 2019.

Sanjeev Arora, Simon 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*, pp. 322–332. PMLR, 2019.

Jimmy Ba, Murat A Erdogdu, Taiji Suzuki, Zhichao Wang, Denny Wu, and Greg Yang. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. *arXiv preprint arXiv:2205.01445*, 2022.

Yu Bai and Jason D. Lee. Beyond linearization: On quadratic and higher-order approximation of wide neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkllGyBFPH.

D. Bakry and M. Émery. Diffusions hypercontractives. In *Séminaire de Probabilités XIX 1983/84*, pp. 177–206. Springer, 1985.

Alon Brutzkus and Amir Globerson. Globally optimal gradient descent for a ConvNet with Gaussian inputs. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 605–614. PMLR, 2017.

T Tony Cai, Jianqing Fan, and Tiefeng Jiang. Distributions of angles in random packing on spheres. *Journal of Machine Learning Research*, 14:1837–1864, 2013.

Andrea Caponnetto and Ernesto De Vito. Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7(3):331–368, 2007.

Minshuo Chen, Yu Bai, Jason D Lee, Tuo Zhao, Huan Wang, Caiming Xiong, and Richard Socher. Towards understanding hierarchical learning: Benefits of neural representations. In *Advances in Neural Information Processing Systems*, volume 33, pp. 22134–22145, 2020. URL https://proceedings.neurips.cc/paper/2020/file/fb647ca6672b0930e9d00dc384d8b16f-Paper.pdf.

Lenaic Chizat. Sparse optimization on measures with over-parameterized gradient descent. *Mathematical Programming*, pp. 1–46, 2021.

Lenaic Chizat and Francis Bach. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems*, volume 31, pp. 3036–3046, 2018.

Lenaic Chizat and Francis Bach. Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss. In *Conference on Learning Theory*, pp. 1305–1338. PMLR, 2020.

Youngmin Cho and Lawrence Saul. Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, volume 22, 2009.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685, 2019.

Murat A Erdogdu, Lester Mackey, and Ohad Shamir. Global non-convex optimization with discretized diffusions. In *Advances in Neural Information Processing Systems 31*, pp. 9671–9680, 2018.

Sebastian Goldt, Madhu Advani, Andrew M Saxe, Florent Krzakala, and Lenka Zdeborová. Dynamics of stochastic gradient descent for two-layer neural networks in the teacher-student setup. In *Advances in Neural Information Processing Systems*, pp. 6981–6991, 2019.

Satoshi Hayakawa and Taiji Suzuki. On the minimax optimality and superiority of deep neural network learning over sparse parameter spaces. *Neural Networks*, 123:343–361, 2020.

Richard Holley and Daniel W Stroock. Logarithmic sobolev inequalities and stochastic ising models. *Journal of Statistical Physics*, 46:1159–1194, 1987.

Kaitong Hu, Zhenjie Ren, David Siska, and Lukasz Szpruch. Mean-field langevin dynamics and energy landscape of neural networks. *arXiv preprint arXiv:1905.07769*, 2019.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pp. 8571–8580, 2018.

Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, pp. 6389–6399, 2018.

Yuanzhi Li, Tengyu Ma, and Hongyang R. Zhang. Learning over-parametrized two-layer neural networks beyond NTK. In *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 2613–2682. PMLR, 2020.

Song Mei, Yu Bai, and Andrea Montanari. The landscape of empirical risk for nonconvex losses. *The Annals of Statistics*, 46(6A):2747–2774, 2018a. ISSN 00905364, 21688966.

Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671, 2018b.

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*, pp. 2388–2464. PMLR, 2019.

Phan-Minh Nguyen. Analysis of feature learning in weight-tied autoencoders via the mean field lens. *arXiv preprint arXiv:2102.08373*, 2021.

Atsushi Nitanda and Taiji Suzuki. Stochastic particle gradient descent for infinite ensembles. *arXiv preprint arXiv:1712.05438*, 2017.

Atsushi Nitanda, Denny Wu, and Taiji Suzuki. Particle dual averaging: Optimization of mean field neural networks with global convergence rate analysis. In *Advances in Neural Information Processing Systems*, volume 34. Curran Associates, Inc., 2021a. to appear.

Atsushi Nitanda, Denny Wu, and Taiji Suzuki. Particle dual averaging: Optimization of mean field neural network with global convergence rate analysis. *Advances in Neural Information Processing Systems*, 34, 2021b.

Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. Non-convex learning via stochastic gradient langevin dynamics: a nonasymptotic analysis. In *Conference on Learning Theory*, pp. 1674–1703. PMLR, 2017.

Maria Refinetti, Sebastian Goldt, Florent Krzakala, and Lenka Zdeborová. Classifying high-dimensional gaussian mixtures: Where kernel methods fail and neural networks succeed. In *International Conference on Machine Learning*, pp. 8936–8947. PMLR, 2021.

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

Itay M Safran, Gilad Yehudai, and Ohad Shamir. The effects of mild over-parameterization on the optimization landscape of shallow relu neural networks. In *Conference on Learning Theory*, pp. 3889–3934. PMLR, 2021.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *The Annals of Statistics*, 48(4):1875–1897, 2020.

Taiji Suzuki. Adaptivity of deep relu network for learning in besov and mixed smooth besov spaces: optimal rate and curse of dimensionality. In *International Conference on Learning Representations*, 2018.

Taiji Suzuki and Shunta Akiyama. Benefit of deep learning with non-convex noisy gradient descent: Provable excess risk bound and superiority to kernel methods. In *International Conference on Learning Representations*, 2021.

Taiji Suzuki and Atsushi Nitanda. Deep learning is adaptive to intrinsic dimensionality of model smoothness in anisotropic Besov space. In *Advances in Neural Information Processing Systems*, volume 34, pp. 3609–3621, 2021.

Yuandong Tian. An analytical formula of population gradient for two-layered ReLU network and its applications in convergence and critical point analysis. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 3404–3413, 2017.

Yuandong Tian. Student specialization in deep rectified networks with finite width and input dimension. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pp. 9470–9480. PMLR, 2020.

Belinda Tzen and Maxim Raginsky. A mean-field theory of lazy training in two-layer neural nets: entropic regularization and controlled McKean-Vlasov dynamics. *arXiv preprint arXiv:2002.01987*, 2020.

Santosh Vempala and Andre Wibisono. Rapid convergence of the unadjusted langevin algorithm: Isoperimetry suffices. *Advances in neural information processing systems*, 32:8094–8106, 2019.

Martin J. Wainwright. *High-Dimensional Statistics: A Non-Asymptotic Viewpoint*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019. doi: 10.1017/9781108627771.

E Weinan, Chao Ma, and Lei Wu. A comparative analysis of optimization and generalization properties of two-layer neural network and random feature models under gradient descent dynamics. *Science China Mathematics*, pp. 1–24, 2020.

Gilad Yehudai and Ohad Shamir. Learning a single neuron with gradient methods. In *Proceedings of the 33rd Conference on Learning Theory*, volume 125, pp. 3756–3786, 2020.

Shuanglin Zhang, Man-Yu Wong, and Zhongguo Zheng. Wavelet threshold estimation of a regression function with random design. *Journal of multivariate analysis*, 80(2):256–284, 2002.

Xiao Zhang, Yaodong Yu, Lingxiao Wang, and Quanquan Gu. Learning one-hidden-layer relu networks via gradient descent. In *Proceedings of Machine Learning Research*, volume 89, pp. 1524–1534. PMLR, 2019.

Kai Zhong, Zhao Song, Prateek Jain, Peter L Bartlett, and Inderjit S Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *International conference on machine learning*, pp. 4140–4149. PMLR, 2017.

Mo Zhou, Rong Ge, and Chi Jin. A local convergence theory for mildly over-parameterized two-layer neural network. In *Conference on Learning Theory*, pp. 4577–4632. PMLR, 2021.

Difan Zou, Yuan Cao, Dongruo Zhou, and Quanquan Gu. Gradient descent optimizes over-parameterized deep ReLU networks. *Machine Learning*, 109(3):467–492, 2020.