How Much Over-parameterization Is Sufficient to Learn Deep ReLU Networks?

Zixiang Chen, Yuan Cao, Difan Zou, Quanquan Gu
Department of Computer Science, UCLA
{chenzx19,yuancao,knowzou,qgu}@cs.ucla.edu

Abstract

A recent line of research on deep learning focuses on the extremely over-parameterized setting, and shows that when the network width is larger than a high degree polynomial of the training sample size $n$ and the inverse of the target accuracy $\epsilon^{-1}$, deep neural networks learned by (stochastic) gradient descent enjoy nice optimization and generalization guarantees. Very recently, it is shown that under certain margin assumption on the training data, a polylogarithmic width condition suffices for two-layer ReLU networks to converge and generalize [2]. However, how much over-parameterization is sufficient to guarantee optimization and generalization for deep neural networks still remains an open question. In this work, we establish sharp optimization and generalization guarantees for deep ReLU networks. Under various assumptions made in previous work, our optimization and generalization guarantees hold with network width polylogarithmic in $n$ and $\epsilon^{-1}$.

Our results push the study of over-parameterized deep neural networks towards more practical settings.

1 Introduction

Deep neural networks have become one of the most important and prevalent machine learning models due to their remarkable power in various real-world applications. Very recently, Ji and Telgarsky [2] showed that for two-layer ReLU networks, when the training data are well separated, polylogarithmic width is sufficient to guarantee good optimization and generalization performance of neural networks trained by GD/SGD. However, it remains unclear whether similar results can be developed for deep neural networks. In this paper, we study the optimization and generalization of deep ReLU networks for a wider range of scaling and give a firm answer.

- We establish the global convergence guarantee of GD for training deep ReLU networks for binary classification. Specifically, we prove that for any positive constant $R$, if there exists a good neural network weight configuration within distance $R \cdot m^{-\alpha}$ to the initialization, and the neural network width satisfies $\tilde{O}((\text{poly}(L,R))^{1/\alpha})$, GD can achieve $\epsilon$-training loss within $T = O(L^2 R^2 \epsilon^{-1})$ iterations, where $m$ is the neural network width, $\alpha \in (0, 1/2]$ is the scaling factor of the neural network and $L$ is the neural network depth.
- We also establish the generalization guarantees for both GD and SGD in the same setting. Specifically, for GD, we establish a $\tilde{O}(\epsilon^{-2})$ sample complexity for a wide range of network width. For SGD, we prove a $\tilde{O}(\epsilon^{-1})$ sample complexity. For both algorithms, our results provide tighter sample complexities based on milder network width conditions compared with existing results.
- Our theoretical results can be generalized to the scenarios with different data separability assumptions studied in the literature, and therefore can cover and improve many existing results in the NTK regime. Specifically, under the data separability assumptions studied in Cao and Gu [1], Ji and Telgarsky [2], our results hold with $R = \mathcal{O}(\log(n/\delta) + \log(1/\epsilon))$, where $\delta$ is the failure probability parameter. This suggests that a neural network with width $m = \text{poly}(L, \log(n/\delta), \log(1/\epsilon))$ can
be learned by GD/SGD with good optimization and generalization guarantees. Moreover, we also show that under a very mild data nondegeneration assumption in Zou et al. [3], our theoretical result can lead to a sharper over-parameterization condition, which improves the existing results in Zou et al. [3] if the neural network depth satisfies \( L \leq \tilde{O}(n^{1/3} \lor \epsilon^{-1/6}) \).

## 2 Main Results

In the following we summarize the problem setting and our main results.

**Neural network function.** Given an input \( x \in \mathbb{R}^d \), the output of deep fully-connected ReLU network is defined as follows,

\[
    f_W(x) = m^a W_L \sigma(W_{L-1} \cdots \sigma(W_1 x) \cdots),
\]

where \( \alpha \in (0, 1/2] \) is a scaling parameter, \( W_1 \in \mathbb{R}^{m \times d}, W_2, \cdots, W_{L-1} \in \mathbb{R}^{m \times m} \) and \( W_L \in \mathbb{R}^{1 \times m} \). We denote the collection of all weight matrices as \( W = \{W_1, \ldots, W_L\} \). The entries in \( W_1^{(0)}, \cdots, W_{L-1}^{(0)} \) are generated independently from univariate Gaussian distribution \( N(0, 2/m) \) and the entries in \( W_L^{(0)} \) are generated independently from \( N(0, 1/m) \).

**Loss function.** Given training dataset \( \{x_i, y_i\}_{i=1, \ldots, n} \) with input \( x_i \in \mathbb{R}^d \) and output \( y_i \in \{-1, +1\} \), we define the training loss function as

\[
    L_S(W) = \frac{1}{n} \sum_{i=1}^{n} L_i(W),
\]

where \( L_i(W) = \ell(y_i, f_W(x_i)) = \log (1 + \exp(-y_i f_W(x_i))) \) is defined as the cross-entropy loss.

**Assumptions.** All training data points satisfy \( \|x_i\|_2 = 1 \), \( i = 1, \ldots, n \) and there exists a positive constant \( R \) and \( W^* \in \mathcal{B}(W^{(0)}, R \cdot m^{-\alpha}) \) such that \( L_i(W^*) < \epsilon \) for all \( i \in [n] \).

**Theorem 2.1.** For any \( \epsilon, \delta > 0 \), there exists \( m^*(\delta, R, L, \alpha) \) that satisfies

\[
    m^*(\delta, R, L, \alpha) = \tilde{O}([\text{poly}(R, L)]^{1/\alpha} \cdot \log^{2/(2-\alpha)}(n/\delta)),
\]

such that if \( m \geq m^*(\delta, R, L, \alpha) \), with probability at least \( 1 - \delta \) over the initialization, GD with step size \( \eta = \Theta(L^{-1} m^{-2\alpha}) \) can train a neural network to achieve at most \( 3\epsilon \) training loss within \( T = \tilde{O}(L^2 R^2 \epsilon^{-1}) \) iterations.

**Theorem 2.2.** Under the same assumptions as Theorem 2.1 with probability at least \( 1 - \delta \), the iterate \( W(t) \) of GD satisfies that

\[
    L_{D(t)}^{0-1}(W(t)) \leq 2L_S(W(t)) + \tilde{O} \left( \min \left\{ 4^4 L^2 R \sqrt{\frac{m}{n}}, \frac{L^{3/2} R}{\sqrt{n}}, \frac{L^{1/3} R^{4/3}}{m^{\alpha/3}} \right\} \right) + \tilde{O} \left( \frac{\log(1/\delta)}{n} \right)
\]

for all \( t = 0, \ldots, T \).

**Theorem 3.3.** For any \( \epsilon, \delta > 0 \), there exists \( m^*(\delta, R, L, \alpha) \) that satisfies

\[
    m^*(\delta, R, L, \alpha) = \tilde{O}([\text{poly}(R, L)]^{1/\alpha} \cdot \log^{2/(2-\alpha)}(n/\delta)),
\]

such that if \( m \geq m^*(\delta, R, L, \alpha) \), with probability at least \( 1 - \delta \), SGD with step size \( \eta = \Theta\left(m^{-2\alpha} \cdot (LR^2 n^{-1} \epsilon^{-1} \land 1^-)\right) \) achieves

\[
    \mathbb{E}[L_{D(t)}^{0-1}(\hat{W})] \leq \frac{8L^2 R^2}{n} + \frac{8 \log(1/\delta)}{n} + 24\epsilon,
\]

where the expectation is taken over the uniform draw of \( \hat{W} \) from \( \{W^{(0)}, \ldots, W^{(n-1)}\} \).

## References

