Gradient Descent Learns One-hidden-layer CNN: Don't be Afraid of Spurious Local Minima

Simon S. Du [1] **Jason D. Lee** [2] **Yuandong Tian** [3] **Barnabás Póczos** [1] **Aarti Singh** [1]

Abstract

We consider the problem of learning a one-hidden-layer neural network with non-overlapping convolutional layer and ReLU activation, i.e., $f(\mathbf{Z}, \mathbf{w}, \mathbf{a}) = \sum_j a_j \sigma(\mathbf{w}^T \mathbf{Z}_j)$, in which both the convolutional weights $\mathbf{w}$ and the output weights $\mathbf{a}$ are parameters to be learned. When the labels are the outputs from a teacher network of the same architecture with fixed weights $(\mathbf{w}^*, \mathbf{a}^*)$, we prove that with Gaussian input $\mathbf{Z}$, there is a spurious local minimizer. Surprisingly, in the presence of the spurious local minimizer, gradient descent with weight normalization from randomly initialized weights can still be proven to recover the true parameters with constant probability, which can be boosted to probability 1 with multiple restarts. We also show that with constant probability, the same procedure could also converge to the spurious local minimum, showing that the local minimum plays a non-trivial role in the dynamics of gradient descent. Furthermore, a quantitative analysis shows that the gradient descent dynamics has two phases: it starts off slow, but converges much faster after several iterations.

1. Introduction

Deep convolutional neural networks (DCNN) have achieved the state-of-the-art performance in many applications such as computer vision (Krizhevsky et al., 2012), natural language processing (Dauphin et al., 2016) and reinforcement learning applied in classic games like Go (Silver et al., 2016). Despite the highly non-convex nature of the objective function, simple first-order algorithms like stochastic gradient descent and its variants often train such networks successfully. Why such simple methods in learning DCNN is successful remains elusive from the optimization perspective.

Recently, a line of research (Tian, 2017; Brutzkus & Globerson, 2017; Li & Yuan, 2017; Soltanolkotabi, 2017; Shalev-Shwartz et al., 2017b) assumed the input distribution is Gaussian and showed that stochastic gradient descent with random or $\mathbf{0}$ initialization is able to train a neural network $f(\mathbf{Z}, \{\mathbf{w}_j\}) = \sum_j a_j \sigma(\mathbf{w}_j^T \mathbf{Z})$ with ReLU activation $\sigma(x) = \max(x, 0)$ in polynomial time. However, these results all assume there is only one unknown layer $\{\mathbf{w}_j\}$, while $\mathbf{a}$ is a fixed vector. A natural question thus arises:

Does randomly initialized (stochastic) gradient descent learn neural networks with multiple layers?

In this paper, we take an important step by showing that randomly initialized gradient descent learns a non-linear convolutional neural network with *two* unknown layers $\mathbf{w}$ and $\mathbf{a}$. To our knowledge, our work is the first of its kind.

Formally, we consider the convolutional case in which a filter $\mathbf{w}$ is shared among different hidden nodes. Let $\mathbf{x} \in \mathbb{R}^d$ be an input sample, e.g., an image. We generate k patches from $\mathbf{x}$, each with size p: $\mathbf{Z} \in \mathbb{R}^{p \times k}$ where the i-th column is the i-th patch generated by selecting some coordinates of $\mathbf{x}$: $\mathbf{Z}_i = \mathbf{Z}_i(\mathbf{x})$. We further assume there is no overlap between patches. Thus, the neural network function has the following form:

$$f(\mathbf{Z}, \mathbf{w}, \mathbf{a}) = \sum_{i=1}^{k} a_i \sigma\left(\mathbf{w}^\top \mathbf{Z}_i\right).$$

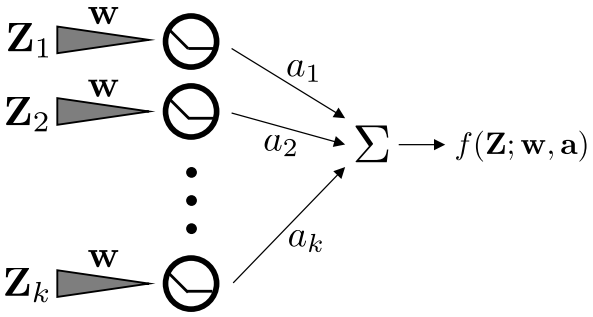
We focus on the realizable case, i.e., the label is generated according to $y = f\left(\mathbf{Z}, \mathbf{w}^*, \mathbf{a}^*\right)$ for some true parameters $\mathbf{w}^*$ and $\mathbf{a}^*$ and use ℓ_2 loss to learn the parameters:

$$\min_{\mathbf{w},\mathbf{a}} \ell(\mathbf{Z}, \mathbf{w}, \mathbf{a}) := \frac{1}{2}\left(f\left(\mathbf{Z}, \mathbf{w}, \mathbf{a}\right) - f\left(\mathbf{Z}, \mathbf{w}^*, \mathbf{a}^*\right)\right)^2.$$

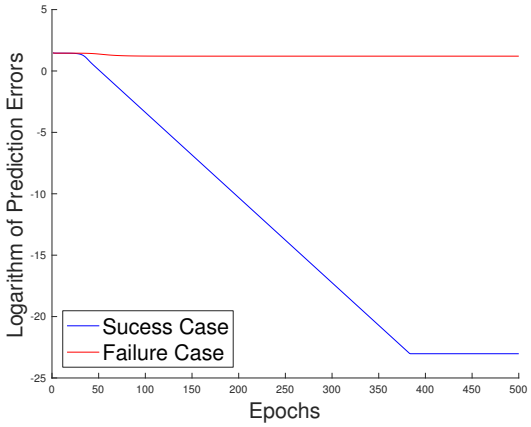
We assume $\mathbf{x}$ is sampled from a Gaussian distribution and there is no overlap between patches. This assumption is equivalent to that each entry of $\mathbf{Z}$ is sampled from a Gaussian distribution (Brutzkus & Globerson, 2017; Zhong et al., 2017b). Following (Zhong et al., 2017a;b; Li & Yuan, 2017;

[1]Machine Learning Department, Carnegie Mellon University [2]Department of Data Sciences and Operations, University of Southern California [3]Facebook Artificial Intelligence Research. Correspondence to: Simon S. Du <ssdu@cs.cmu.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.



(a) Convolutional neural network with an unknown non-overlapping filter and an unknown output layer. In the first (hidden) layer, a filter $\mathbf{w}$ is applied to nonoverlapping parts of the input $\mathbf{x}$, which then passes through a ReLU activation function. The final output is the inner product between an output weight vector $\mathbf{a}$ and the hidden layer outputs.



(b) The convergence of gradient descent for learning a CNN described in Figure 1a with Gaussian input using different initializations. The success case and the failure case correspond to convergence to the global minimum and the spurious local minimum, respectively. In the first ~ 50 iterations the convergence is slow. After that gradient descent converges at a fast linear rate.

Figure 1. Network architecture that we consider in this paper and convergence of gradient descent for learning the parameters of this network.

Tian, 2017; Brutzkus & Globerson, 2017; Shalev-Shwartz et al., 2017b), in this paper, we mainly focus on the population loss:

$$\ell(\mathbf{w}, \mathbf{a}) := \frac{1}{2}\mathbb{E}_{\mathbf{Z}}\left[\left(f(\mathbf{Z}, \mathbf{w}, \mathbf{a}) - f(\mathbf{Z}, \mathbf{w}^*, \mathbf{a}^*)\right)^2\right].$$

We study whether the global convergence $\mathbf{w} \to \mathbf{w}^*$ and $\mathbf{a} \to \mathbf{a}^*$ can be achieved when optimizing $\ell(\mathbf{w}, \mathbf{a})$ using randomly initialized gradient descent.

A crucial difference between our two-layer network and previous one-layer models is there is a positive-homogeneity issue. That is, for any $c > 0$, $f\left(\mathbf{Z}, c\mathbf{w}, \frac{\mathbf{a}}{c}\right) = f(\mathbf{Z}, \mathbf{w}, \mathbf{a})$. This interesting property allows the network to be rescaled without changing the function computed by the network. As reported by (Neyshabur et al., 2015), it is desirable to have scaling-invariant learning algorithm to stabilize the training process.

One commonly used technique to achieve stability is *weight-normalization* introduced by Salimans & Kingma (2016). As reported in (Salimans & Kingma, 2016), this re-parametrization improves the conditioning of the gradient because it couples the magnitude of the weight vector from the direction of the weight vector and empirically accelerates stochastic gradient descent optimization.

In our setting, we re-parametrize the first layer as $\mathbf{w} = \frac{\mathbf{v}}{\|\mathbf{v}\|_2}$ and the prediction function becomes

$$f(\mathbf{Z}, \mathbf{v}, \mathbf{a}) = \sum_{i=1}^{k} a_i \frac{\sigma\left(\mathbf{Z}_i^\top \mathbf{v}\right)}{\|\mathbf{v}\|_2}. \tag{1}$$

The loss function is

$$\ell(\mathbf{v}, \mathbf{a}) = \frac{1}{2}\mathbb{E}_{\mathbf{Z}}\left[\left(f(\mathbf{Z}, \mathbf{v}, \mathbf{a}) - f(\mathbf{Z}, \mathbf{v}^*, \mathbf{a}^*)\right)^2\right]. \tag{2}$$

In this paper we focus on using randomly initialized gradient descent for learning this convolutional neural network. The pseudo-code is listed in Algorithm 1.[1]

Main Contributions. Our paper have three contributions. First, we show if $(\mathbf{v}, \mathbf{a})$ is initialized by a specific *random initialization*, then with high probability, gradient descent from $(\mathbf{v}, \mathbf{a})$ converges to teacher's parameters $(\mathbf{v}^*, \mathbf{a}^*)$. We can further boost the success rate with more trials.

Second, perhaps surprisingly, we prove that the objective function (Equation (2)) *does* have a spurious local minimum: using the same random initialization scheme, there exists a pair $(\tilde{\mathbf{v}}^0, \tilde{\mathbf{a}}^0) \in S_{\pm}(\mathbf{v}, \mathbf{a})$ so that gradient descent from $(\tilde{\mathbf{v}}^0, \tilde{\mathbf{a}}^0)$ converges to this bad local minimum. In contrast to previous works on guarantees for non-convex objective functions whose landscape satisfies "no spurious local minima" property (Li et al., 2016; Ge et al., 2017a; 2016; Bhojanapalli et al., 2016; Ge et al., 2017b; Kawaguchi, 2016), our result provides a concrete counter-example and highlights a conceptually surprising phenomenon:

Randomly initialized local search can find a global

[1] With some simple calculations, we can see the optimal solution for $\mathbf{a}$ is unique, which we denote as $\mathbf{a}^*$ whereas the optimal for $\mathbf{v}$ is not because for every optimal solution $\mathbf{v}^*$, $c\mathbf{v}^*$ for $c > 0$ is also an optimal solution. In this paper, with a little abuse of the notation, we use $\mathbf{v}^*$ to denote the equivalent class of optimal solutions.

Algorithm 1 Gradient Descent for Learning One-Hidden-Layer CNN with Weight Normalization

Input: Initialization $\mathbf{v}_0 \in \mathbb{R}^p$, $\mathbf{a}_0 \in \mathbb{R}^k$, learning rate η.
for $t = 1, 2, \ldots$ **do**
  $\mathbf{v}^{t+1} \leftarrow \mathbf{v}^t - \eta \frac{\partial \ell(\mathbf{v}^t, \mathbf{a}^t)}{\partial \mathbf{v}^t}$,
  $\mathbf{a}^{t+1} \leftarrow \mathbf{a}^t - \eta \frac{\partial \ell(\mathbf{v}^t, \mathbf{a}^t)}{\partial \mathbf{a}^t}$.
end for

minimum in the presence of spurious local minima.

Finally, we conduct a quantitative study of the dynamics of gradient descent. We show that the dynamics of Algorithm 1 has two phases. At the beginning (around first 50 iterations in Figure 1b), because the magnitude of initial signal (angle between $\mathbf{v}$ and $\mathbf{w}^*$) is small, the prediction error drops slowly. After that, when the signal becomes stronger, gradient descent converges at a much faster rate and the prediction error drops quickly.

Technical Insights. The main difficulty of analyzing the convergence is the presence of local minima. Note that local minimum and the global minimum are disjoint (c.f. Figure 1b). The key technique we adopt is to characterize the attraction basin for each minimum. We consider the sequence $\{(\mathbf{v}^t, \mathbf{a}^t)\}_{t=0}^{\infty}$ generated by Algorithm 1 with step size η using initialization point $(\mathbf{v}^0, \mathbf{a}^0)$. The attraction basin for a minimum $(\mathbf{v}^*, \mathbf{a}^*)$ is defined as the

$$\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*) = \left\{ (\mathbf{v}^0, \mathbf{a}^0), \lim_{t\to\infty} (\mathbf{v}^t, \mathbf{a}^t) \to (\mathbf{v}^*, \mathbf{a}^*) \right\}$$

The goal is to find a distribution $\mathcal{G}$ for weight initialization so that the probability that the initial weights are in $\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)$ of the global minimum is bounded below:

$$\mathbb{P}_{(\mathbf{v}^0, \mathbf{a}^0)\sim\mathcal{G}} [\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)] \geq c$$

for some absolute constant $c > 0$.

While it is hard to characterize $\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)$, we find that the set $\tilde{\mathbf{B}}(\mathbf{v}^*, \mathbf{a}^*) \equiv \{(\mathbf{v}^0, \mathbf{a}^0) : (\mathbf{v}^0)^\top \mathbf{v}^* \geq 0, (\mathbf{a}^0)^\top \mathbf{a}^* \geq 0, |\mathbf{1}^\top \mathbf{a}^0| \leq |\mathbf{1}^\top \mathbf{a}^*|\}$ is a subset of $\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)$ (c.f. Lemma 5.2-Lemma 5.4). Furthermore, when the learning rate η is sufficiently small, we can design a specific distribution $\mathcal{G}$ so that:

$$\mathbb{P}_{(\mathbf{v}^0, \mathbf{a}^0)\sim\mathcal{G}} [\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)] \geq \mathbb{P}_{(\mathbf{v}^0, \mathbf{a}^0)\sim\mathcal{G}} \left[\tilde{\mathbf{B}}(\mathbf{v}^*, \mathbf{a}^*)\right] \geq c$$

This analysis emphasizes that for non-convex optimization problems, we need to carefully characterize both the trajectory of the algorithm and the initialization. We believe that this idea is applicable to other non-convex problems.

To obtain the convergence rate, we propose a potential function (also called Lyapunov function in the literature). For this problem we consider the quantity $\sin^2 \phi^t$ where $\phi^t = \theta(\mathbf{v}^t, \mathbf{v}^*)$ and we show it shrinks at a geometric rate (c.f. Lemma 5.5).

Organization This paper is organized as follows. In Section 3 we introduce the necessary notations and analytical formulas of gradient updates in Algorithm 1. In Section 4, we provide our main theorems on the performance of the algorithm and their implications. In Section 6, we use simulations to verify our theories. In Section 5, we give a proof sketch of our main theorem. We conclude and list future directions in Section 7. We place most of our detailed proofs in the appendix.

2. Related Works

From the point of view of learning theory, it is well known that training a neural network is hard in the worst cases (Blum & Rivest, 1989; Livni et al., 2014; Šíma, 2002; Shalev-Shwartz et al., 2017a;b) and recently, Shamir (2016) showed that assumptions on *both* the target function and the input distribution are needed for optimization algorithms used in practice to succeed.

Solve NN without gradient descent. With some additional assumptions, many works tried to design algorithms that provably learn a neural network with polynomial time and sample complexity (Goel et al., 2016; Zhang et al., 2015; Sedghi & Anandkumar, 2014; Janzamin et al., 2015; Goel & Klivans, 2017a;b). However these algorithms are specially designed for certain architectures and cannot explain why (stochastic) gradient based optimization algorithms work well in practice.

Gradient-based optimization with Gaussian Input. Focusing on gradient-based algorithms, a line of research analyzed the behavior of (stochastic) gradient descent for *Gaussian* input distribution. Tian (2017) showed that population gradient descent is able to find the true weight vector with random initialization for one-layer one-neuron model. Soltanolkotabi (2017) later improved this result by showing the true weights can be exactly recovered by empirical projected gradient descent with enough samples in linear time. Brutzkus & Globerson (2017) showed population gradient descent recovers the true weights of a convolution filter with non-overlapping input in polynomial time. Zhong et al. (2017b;a) proved that with sufficiently good initialization, which can be implemented by tensor method, gradient descent can find the true weights of a one-hidden-layer fully connected and convolutional neural network. Li & Yuan (2017) showed SGD can recover the true weights of a one-layer ResNet model with ReLU activation under the assumption that the spectral norm of the true weights is within a small constant of the identity mapping. (Panigrahy et al., 2018) also analyzed gradient descent for learning

a two-layer neural network but with different activation functions. This paper also follows this line of approach that studies the behavior of gradient descent algorithm with Gaussian inputs.

Local minimum and Global minimum. Finding the optimal weights of a neural network is non-convex problem. Recently, researchers found that if the objective functions satisfy the following two key properties, (1) all saddle points and local maxima are strict (i.e., there exists a direction with negative curvature), and (2) all local minima are global (no spurious local minmum), then perturbed (stochastic) gradient descent (Ge et al., 2015) or methods with second order information (Carmon et al., 2016; Agarwal et al., 2017) can find a global minimum in polynomial time. [2] Combined with geometric analyses, these algorithmic results have shown a large number problems, including tensor decomposition (Ge et al., 2015), dictionary learning (Sun et al., 2017), matrix sensing (Bhojanapalli et al., 2016; Park et al., 2017), matrix completion (Ge et al., 2017a; 2016) and matrix factorization (Li et al., 2016) can be solved in polynomial time with local search algorithms.

This motivates the research of studying the landscape of neural networks (Kawaguchi, 2016; Choromanska et al., 2015; Hardt & Ma, 2016; Haeffele & Vidal, 2015; Mei et al., 2016; Freeman & Bruna, 2016; Safran & Shamir, 2016; Zhou & Feng, 2017; Nguyen & Hein, 2017a;b; Ge et al., 2017b; Zhou & Feng, 2017; Safran & Shamir, 2017). In particular, Kawaguchi (2016); Hardt & Ma (2016); Zhou & Feng (2017); Nguyen & Hein (2017a;b); Feizi et al. (2017) showed that under some conditions, all local minima are global. Recently, Ge et al. (2017b) showed using a modified objective function satisfying the two properties above, one-hidden-layer neural network can be learned by noisy perturbed gradient descent. However, for nonlinear activation function, where the number of samples larger than the number of nodes at every layer, which is usually the case in most deep neural network, and natural objective functions like ℓ_2, it is still unclear whether the strict saddle and "all locals are global" properties are satisfied. In this paper, we show that even for a one-hidden-layer neural network with ReLU activation, there exists a spurious local minimum. However, we further show that randomly initialized local search can achieve *global* minimum with constant probability.

[2] Lee et al. (2016) showed vanilla gradient descent only converges to minimizers with no convergence rates guarantees. Recently, Du et al. (2017a) gave an exponential time lower bound for the vanilla gradient descent. In this paper, we give polynomial convergence guarantee on vanilla gradient descent.

3. Preliminaries

We use bold-faced letters for vectors and matrices. We use $\|\cdot\|_2$ to denote the Euclidean norm of a finite-dimensional vector. We let $\mathbf{w}^t$ and $\mathbf{a}^t$ be the parameters at the t-th iteration and $\mathbf{w}^*$ and $\mathbf{a}^*$ be the optimal weights. For two vector $\mathbf{w}_1$ and $\mathbf{w}_2$, we use $\theta(\mathbf{w}_1, \mathbf{w}_2)$ to denote the angle between them. a_i is the i-th coordinate of a and $\mathbf{Z}_i$ is the transpose of the i-th row of $\mathbf{Z}$ (thus a column vector). We denote $\mathcal{S}^{p-1}$ the $(p-1)$-dimensional unit sphere and $\mathcal{B}(\mathbf{0}, r)$ the ball centered at $\mathbf{0}$ with radius r.

In this paper we assume every patch $\mathbf{Z}_i$ is vector of i.i.d Gaussian random variables. The following theorem gives an explicit formula for the population loss. The proof uses basic rotational invariant property and polar decomposition of Gaussian random variables. See Section A for details.

Theorem 3.1. *If every entry of* $\mathbf{Z}$ *is i.i.d. sampled from a Gaussian distribution with mean* 0 *and variance* 1*, then population loss is*

$$\begin{aligned}\ell(\mathbf{v}, \mathbf{a}) =& \frac{1}{2}\left[\frac{(\pi-1)\|\mathbf{w}^*\|_2^2}{2\pi}\|\mathbf{a}^*\|_2^2 + \frac{(\pi-1)}{2\pi}\|\mathbf{a}\|_2^2\right. \\ &- \frac{2(g(\phi)-1)\|\mathbf{w}^*\|_2}{2\pi}\mathbf{a}^\top\mathbf{a}^* + \frac{\|\mathbf{w}^*\|_2^2}{2\pi}\left(\mathbf{1}^\top\mathbf{a}^*\right)^2 \\ &\left.+ \frac{1}{2\pi}\left(\mathbf{1}^\top\mathbf{a}\right)^2 - 2\|\mathbf{w}^*\|_2\mathbf{1}^\top\mathbf{a}\cdot\mathbf{1}^\top\mathbf{a}^*\right] \end{aligned} \tag{3}$$

where $\phi = \theta(\mathbf{v}, \mathbf{w}^*)$ *and* $g(\phi) = (\pi-\phi)\cos\phi + \sin\phi$.

Using similar techniques, we can show the gradient also has an analytical form.

Theorem 3.2. *Suppose every entry of* $\mathbf{Z}$ *is i.i.d. sampled from a Gaussian distribution with mean* 0 *and variance* 1*. Denote* $\phi = \theta(\mathbf{w}, \mathbf{w}^*)$*. Then the expected gradient of* $\mathbf{w}$ *and* $\mathbf{a}$ *can be written as*

$$\begin{aligned}&\mathbb{E}_{\mathbf{Z}}\left[\frac{\partial\ell(\mathbf{Z}, \mathbf{v}, \mathbf{a})}{\partial\mathbf{v}}\right] \\ =& -\frac{1}{2\pi\|\mathbf{v}\|_2}\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{a}^\top\mathbf{a}^*(\pi-\phi)\mathbf{w}^* \\ &\mathbb{E}_{\mathbf{Z}}\left[\frac{\partial\ell(\mathbf{Z}, \mathbf{v}, \mathbf{a})}{\partial\mathbf{a}}\right] \\ =& \frac{1}{2\pi}\left(\mathbf{1}\mathbf{1}^\top + (\pi-1)\mathbf{I}\right)\mathbf{a} \\ &- \frac{1}{2\pi}\left(\mathbf{1}\mathbf{1}^\top + (g(\phi)-1)\mathbf{I}\right)\|\mathbf{w}^*\|_2\mathbf{a}^*\end{aligned}$$

As a remark, if the second layer is fixed, upon proper scaling, the formulas for the population loss and gradient of $\mathbf{v}$ are equivalent to the corresponding formulas derived in (Brutzkus & Globerson, 2017; Cho & Saul, 2009). However, when the second layer is not fixed, the gradient of $\mathbf{v}$ depends on $\mathbf{a}^\top\mathbf{a}^*$, which plays an important role in deciding whether converging to the global or the local minimum.

4. Main Result

We begin with our main theorem about the convergence of gradient descent.

Theorem 4.1. *Suppose the initialization satisfies* $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* > 0$, $\left|\mathbf{1}^\top \mathbf{a}^0\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$, $\phi^0 < \pi/2$ *and step size satisfies*

$$\eta = O\left(\min\left\{\frac{\left(\mathbf{a}^0\right)^\top \mathbf{a}^* \cos\phi^0}{\left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right)\left\|\mathbf{w}^*\right\|_2^2}, \frac{\left(g(\phi_0)-1\right)\left\|\mathbf{a}^*\right\|_2^2 \cos\phi^0}{\left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right)\left\|\mathbf{w}^*\right\|_2^2}, \frac{\cos\phi^0}{\left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right)\left\|\mathbf{w}^*\right\|_2^2}, \frac{1}{k}\right\}\right).$$

Then the convergence of gradient descent has two phases. ***(Phase I: Slow Initial Rate)*** *There exists* $T_1 = O\left(\frac{1}{\eta \cos\phi^0 \beta^0} + \frac{1}{\eta}\right)$ *such that we have* $\phi^{T_1} = \Theta(1)$ *and* $\left(\mathbf{a}^{T_1}\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\|_2 = \Theta\left(\left\|\mathbf{a}^*\right\|_2^2 \left\|\mathbf{w}^*\right\|_2^2\right)$ *where* $\beta^0 = \min\left\{\left(\mathbf{a}^0\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\|_2, \left(g(\phi^0)-1\right)\left\|\mathbf{a}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2\right\}$. ***(Phase II: Fast Rate)*** *Suppose at the* T_1*-th iteration,* $\phi^{T_1} = \Theta(1)$ *and* $\left(\mathbf{a}^{T_1}\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\|_2 = \Theta\left(\left\|\mathbf{a}^*\right\|_2^2 \left\|\mathbf{w}^*\right\|_2^2\right)$*, then there exists* $T_2 = \widetilde{O}(\left(\frac{1}{\eta\|\mathbf{w}^*\|_2^2\|\mathbf{a}^*\|_2^2} + \frac{1}{\eta}\right)\log\left(\frac{1}{\epsilon}\right))$[3] *such that* $\ell\left(\mathbf{v}^{T_1+T_2}, \mathbf{a}^{T_1+T_2}\right) \leq \epsilon \left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2$.

Theorem 4.1 shows under certain conditions of the initialization, gradient descent converges to the global minimum. The convergence has two phases, at the beginning because the initial signal ($\cos\phi^0\beta^0$) is small, the convergence is quite slow. After T_1 iterations, the signal becomes stronger and we enter a regime with a faster convergence rate. See Lemma 5.5 for technical details.

Initialization plays an important role in the convergence. First, Theorem 4.1 needs the initialization satisfy $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* > 0$, $\left|\mathbf{1}^\top \mathbf{a}^0\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$ and $\phi^0 < \pi/2$. Second, the step size η and the convergence rate in the first phase also depends on the initialization. If the initial signal is very small, for example, $\phi^0 \approx \pi/2$ which makes $\cos\phi^0$ close to 0, we can only choose a very small step size and because T_1 depends on the inverse of $\cos\phi^0$, we need a large number of iterations to enter phase II. We provide the following initialization scheme which ensures the conditions required by Theorem 4.1 and a large enough initial signal.

Theorem 4.2. *Let* $\mathbf{v} \sim \mathrm{unif}\left(\mathcal{S}^{p-1}\right)$ *and* $\mathbf{a} \sim \mathrm{unif}\left(\mathcal{B}\left(\mathbf{0}, \frac{\left|\mathbf{1}^\top \mathbf{a}^*\right|\left\|\mathbf{w}^*\right\|_2}{\sqrt{k}}\right)\right)$*, then exists*

$$\left(\mathbf{v}^0, \mathbf{a}^0\right) \in \left\{\left(\mathbf{v}, \mathbf{a}\right), \left(\mathbf{v}, -\mathbf{a}\right), \left(-\mathbf{v}, \mathbf{a}\right), \left(-\mathbf{v}, -\mathbf{a}\right)\right\}$$

that $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* > 0$, $\left|\mathbf{1}^\top \mathbf{a}^0\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$ *and* $\phi^0 < \pi/2$. *Further, with high probability, the initialization satisfies* $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\|_2 = \Theta\left(\frac{\left|\mathbf{1}^\top \mathbf{a}^*\right|\left\|\mathbf{a}^*\right\|_2\left\|\mathbf{w}^*\right\|_2^2}{k}\right)$, *and* $\phi^0 = \Theta\left(\frac{1}{\sqrt{p}}\right)$.

Theorem 4.2 shows after generating a pair of random vectors $(\mathbf{v}, \mathbf{a})$, trying out all 4 sign combinations of $(\mathbf{v}, \mathbf{a})$, we can find the global minimum by gradient descent. Further, because the initial signal is not too small, we only need to set the step size to be $O(1/\mathrm{poly}(k, p, \left\|\mathbf{w}^*\right\|_2 \left\|\mathbf{a}\right\|_2))$ and the number of iterations in phase I is at most $O(\mathrm{poly}(k, p, \left\|\mathbf{w}^*\right\|_2 \left\|\mathbf{a}\right\|_2))$. Therefore, Theorem 4.1 and Theorem 4.2 together show that randomly initialized gradient descent learns an one-hidden-layer convolutional neural network in polynomial time. The proof of the first part of Theorem 4.2 uses the symmetry of unit sphere and ball and the second part is a standard application of random vector in high-dimensional spaces. See Lemma 2.5 of (Hardt & Price, 2014) for example.

Remark 1: For the second layer we use $O\left(\frac{1}{\sqrt{k}}\right)$ type initialization, verifying common initialization techniques (Glorot & Bengio, 2010; He et al., 2015; LeCun et al., 1998).

Remark 2: The Gaussian input assumption is not necessarily true in practice, although this is a common assumption appeared in the previous papers (Brutzkus & Globerson, 2017; Li & Yuan, 2017; Zhong et al., 2017a;b; Tian, 2017; Xie et al., 2017; Shalev-Shwartz et al., 2017b) and also considered plausible in (Choromanska et al., 2015). Our result can be easily generalized to rotation invariant distributions. However, extending to more general distributional assumption, e.g., structural conditions used in (Du et al., 2017b) remains a challenging open problem.

Remark 3: Since we only require initialization to be smaller than some quantities of $\mathbf{a}^*$ and $\mathbf{w}^*$. In practice, if the optimization fails, i.e., the initialization is too large, one can halve the initialization size, and eventually these conditions will be met.

4.1. Gradient Descent Can Converge to the Spurious Local Minimum

Theorem 4.2 shows that among $\{(\mathbf{v}, \mathbf{a}), (\mathbf{v}, -\mathbf{a}), (-\mathbf{v}, \mathbf{a}), (-\mathbf{v}, -\mathbf{a})\}$, there is a pair that enables gradient descent to converge to the global minimum. Perhaps surprisingly, the next theorem shows that under some conditions of the underlying truth, there is also a pair that makes gradient descent converge to the

[3] $\widetilde{O}(\cdot)$ hides logarithmic factors on $\left|\mathbf{1}^\top \mathbf{a}^*\right|\left\|\mathbf{w}^*\right\|_2$ and $\left\|\mathbf{a}^*\right\|_2\left\|\mathbf{w}^*\right\|_2$

spurious local minimum.

Theorem 4.3. *Without loss of generality, we let* $\|\mathbf{w}^*\|_2 = 1$. *Suppose* $\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 < \frac{1}{\mathrm{poly}(p)} \|\mathbf{a}^*\|_2^2$ *and* η *is sufficiently small. Let* $\mathbf{v} \sim \mathrm{unif}\left(\mathcal{S}^{p-1}\right)$ *and* $\mathbf{a} \sim \mathrm{unif}\left(\mathcal{B}\left(\mathbf{0}, \frac{\left|\mathbf{1}^\top \mathbf{a}^*\right|}{\sqrt{k}}\right)\right)$, *then with high probability, there exists* $\left(\mathbf{v}^0, \mathbf{a}^0\right) \in \{(\mathbf{v}, \mathbf{a}), (\mathbf{v}, -\mathbf{a}), (-\mathbf{v}, \mathbf{a}), (-\mathbf{v}, -\mathbf{a})\}$ *that* $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* < 0$, $\left|\mathbf{1}^\top \mathbf{a}^0\right| \le \left|\mathbf{1}^\top \mathbf{a}^*\right|$, $g\left(\phi^0\right) \le \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2} + 1$. *If* $\left(\mathbf{v}^0, \mathbf{a}^0\right)$ *is used as the initialization, when Algorithm 1 converges, we have*

$$\theta\left(\mathbf{v}, \mathbf{w}^*\right) = \pi, \mathbf{a} = \left(\mathbf{1}\mathbf{1}^\top + (\pi - 1)\mathbf{I}\right)^{-1}\left(\mathbf{1}\mathbf{1}^\top - \mathbf{I}\right)\mathbf{a}^*$$

and $\ell\left(\mathbf{v}, \mathbf{a}\right) = \Omega\left(\|\mathbf{a}^*\|_2^2\right)$.

Unlike Theorem 4.1 which requires no assumption on the underlying truth $\mathbf{a}^*$, Theorem 4.3 assumes $\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 < \frac{1}{\mathrm{poly}(p)} \|\mathbf{a}^*\|_2^2$. This technical condition comes from the proof which requires invariance $g(\phi^t) \le \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$ for all iterations. To ensure there exists $\left(\mathbf{v}^0, \mathbf{a}^0\right)$ which makes $g(\phi^0) \le \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$, we need $\frac{\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$ relatively small. See Section E for more technical insights.

A natural question is whether the ratio $\frac{\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$ becomes larger, the probability randomly gradient descent converging to the global minimum, becomes larger as well. We verify this phenomenon empirically in Section 6.

5. Proof Sketch

In Section 5.1, we give qualitative high level intuition on why the initial conditions are sufficient for gradient descent to converge to the global minimum. In Section 5.2, we explain why the gradient descent has two phases.

5.1. Qualitative Analysis of Convergence

The convergence to global optimum relies on a geometric characterization of saddle points and a series of invariants throughout the gradient descent dynamics. The next lemma gives the analysis of stationary points. The main step is to check the first order condition of stationary points using Theorem 3.2.

Lemma 5.1 (Stationary Point Analysis)**.** *When the gradient descent converges,* $\mathbf{a}^\top \mathbf{a}^* \neq 0$ *and* $\|\mathbf{v}\|_2 < \infty$, *we have either*

$$\theta\left(\mathbf{v}, \mathbf{w}^*\right) = 0, \mathbf{a} = \|\mathbf{w}^*\|_2 \mathbf{a}^*$$

or $\theta\left(\mathbf{v}, \mathbf{w}^*\right) = \pi$,

$$\mathbf{a} = \left(\mathbf{1}\mathbf{1}^\top + (\pi - 1)\mathbf{I}\right)^{-1}\left(\mathbf{1}\mathbf{1}^\top - \mathbf{I}\right)\|\mathbf{w}^*\|_2 \mathbf{a}^*.$$

This lemma shows that when the algorithm converges, and $\mathbf{a}$ and $\mathbf{a}^*$ are not orthogonal, then we arrive at either a global optimal point or a local minimum. Now recall the gradient formula of $\mathbf{v}$: $\frac{\partial \ell(\mathbf{v}, \mathbf{a})}{\partial \mathbf{v}} = -\frac{1}{2\pi\|\mathbf{v}\|_2}\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{a}^\top \mathbf{a}^* (\pi - \phi)\mathbf{w}^*$. Notice that $\phi \le \pi$ and $\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)$ is just the projection matrix onto the complement of $\mathbf{v}$. Therefore, the sign of inner product between $\mathbf{a}$ and $\mathbf{a}^*$ plays a crucial role in the dynamics of Algorithm 1 because if the inner product is positive, the gradient update will decrease the angle between $\mathbf{v}$ and $\mathbf{w}^*$ and if it is negative, the angle will increase. This observation is formalized in the lemma below.

Lemma 5.2 (Invariance I: Tje Angle between $\mathbf{v}$ and $\mathbf{w}^*$ always decreases.)**.** *If* $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* > 0$, *then* $\phi^{t+1} \le \phi^t$.

This lemma shows that when $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* > 0$ for all t, gradient descent converges to the global minimum. Thus, we need to study the dynamics of $\left(\mathbf{a}^t\right)^\top \mathbf{a}^*$. For the ease of presentation, without loss of generality, we assume $\|\mathbf{w}^*\|_2 = 1$. By the gradient formula of $\mathbf{a}$, we have

$$\begin{aligned}
&\left(\mathbf{a}^{t+1}\right)^\top \mathbf{a}^* \\
=&\left(1 - \frac{\eta(\pi - 1)}{2\pi}\right)\left(\mathbf{a}^t\right)^\top \mathbf{a}^* + \frac{\eta(g(\phi^t) - 1)}{2\pi}\left\|\mathbf{a}^t\right\|_2^2 \\
&+ \frac{\eta}{2\pi}\left(\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 - \left(\mathbf{1}^\top \mathbf{a}^t\right)\left(\mathbf{1}^\top \mathbf{a}^*\right)\right). \qquad (4)
\end{aligned}$$

We can use induction to prove the invariance. If $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* > 0$ and $\phi^t < \frac{\pi}{2}$ the first term of Equation (4) is non-negative. For the second term, notice that if $\phi^t < \frac{\pi}{2}$, we have $g(\phi^t) > 1$, so the second term is non-negative. Therefore, as long as $\left(\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 - \left(\mathbf{1}^\top \mathbf{a}^t\right)\left(\mathbf{1}^\top \mathbf{a}^*\right)\right)$ is also non-negative, we have the desired invariance. The next lemma summarizes the above analysis.

Lemma 5.3 (Invariance II: Positive Signal from the Second Layer.)**.** *If* $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* > 0$, $0 \le \mathbf{1}^\top \mathbf{a}^* \cdot \mathbf{1}^\top \mathbf{a}^t \le \left(\mathbf{1}^\top \mathbf{a}^*\right)^2$, $0 < \phi^t < \pi/2$ *and* $\eta < 2$, *then* $\left(\mathbf{a}^{t+1}\right)^\top \mathbf{a}^* > 0$.

It remains to prove $\left(\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 - \left(\mathbf{1}^\top \mathbf{a}^t\right)\left(\mathbf{1}^\top \mathbf{a}^*\right)\right) > 0$. Again, we study the dynamics of this quantity. Using the gradient formula and some algebra, we have

$$\begin{aligned}
\mathbf{1}^\top \mathbf{a}^{t+1} \cdot \mathbf{1}^\top \mathbf{a}^* &\le \left(1 - \frac{\eta\left(k - \pi - 1\right)}{2\pi}\right)\mathbf{1}^\top \mathbf{a}^t \cdot \mathbf{1}^\top \mathbf{a}^* \\
&+ \frac{\eta\left(k + g(\phi^t) - 1\right)}{2}\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 \\
&\le \left(1 - \frac{\eta\left(k - \pi - 1\right)}{2\pi}\right)\mathbf{1}^\top \mathbf{a}^t \cdot \mathbf{1}^\top \mathbf{a}^* \\
&+ \frac{\eta\left(k + \pi - 1\right)}{2}\left(\mathbf{1}^\top \mathbf{a}^*\right)^2
\end{aligned}$$

where have used the fact that $g(\phi) \leq \pi$ for all $0 \leq \phi \leq \frac{\pi}{2}$. Therefore we have

$$\begin{aligned}&\left(\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^{t+1}\right) \cdot \mathbf{1}^\top \mathbf{a}^* \\ \geq &\left(1 - \frac{\eta(k+\pi-1)}{2\pi}\right)\left(\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^t\right)\mathbf{1}^\top \mathbf{a}^*.\end{aligned}$$

These imply the third invariance.

Lemma 5.4 (Invariance III: Summation of Second Layer Always Small.)**.** *If* $\mathbf{1}^\top \mathbf{a}^* \cdot \mathbf{1}^\top \mathbf{a}^t \leq \left(\mathbf{1}^\top \mathbf{a}^*\right)^2$ *and* $\eta < \frac{2\pi}{k+\pi-1}$ *then* $\mathbf{1}^\top \mathbf{a}^* \cdot \mathbf{1}^\top \mathbf{a}^{t+1} \leq \left(\mathbf{1}^\top \mathbf{a}^*\right)^2$.

To sum up, if the initialization satisfies (1) $\phi^0 < \frac{\pi}{2}$, (2) $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* > 0$ and (3) $\mathbf{1}^\top \mathbf{a}^* \cdot \mathbf{1}^\top \mathbf{a}^0 \leq \left(\mathbf{1}^\top \mathbf{a}^*\right)^2$, with Lemma 5.2, 5.3, 5.4, by induction we can show the convergence to the global minimum. Further, Theorem 4.2 shows these three conditions are true with constant probability using random initialization.

5.2. Quantitative Analysis of Two Phase Phenomenon

In this section we demonstrate why there is a two-phase phenomenon. Throughout this section, we assume the conditions in Section 5.1 hold. We first consider the convergence of the first layer. Because we are using weight-normalization, only the angle between $\mathbf{v}$ and $\mathbf{w}^*$ will affect the prediction. Therefore, in this paper, we study the dynamics $\sin^2 \phi^t$. The following lemma quantitatively characterize the shrinkage of this quantity of one iteration.

Lemma 5.5 (Convergence of Angle between $\mathbf{v}$ and $\mathbf{w}^*$)**.** *Under the same assumptions as in Theorem 4.1. Let* $\beta^0 = \min\left\{\left(\mathbf{a}^0\right)^\top \mathbf{a}^*, \left(g(\phi^0)-1\right)\|\mathbf{a}^*\|_2^2\right\}\|\mathbf{w}^*\|_2^2$. *If the step size satisfies* $\eta = O(\min\{\frac{\beta^0 \cos\phi^0}{\left(\|\mathbf{a}^*\|_2^2 + (\mathbf{1}^\top \mathbf{a}^*)^2\right)\|\mathbf{w}^*\|_2^2}, \frac{\cos\phi^0}{\left(\|\mathbf{a}^*\|_2^2 + (\mathbf{1}^\top \mathbf{a}^*)^2\right)\|\mathbf{w}^*\|_2^2}, \frac{1}{k}\})$, *we have*

$$\sin^2 \phi^{t+1} \leq \left(1 - \eta \cos\phi^t \lambda^t\right) \sin^2 \phi^t$$

where $\lambda^t = \frac{\|\mathbf{w}^*\|_2 \left(\pi - \phi^t\right)\left(\mathbf{a}^t\right)^\top \mathbf{a}^*}{2\pi \|\mathbf{v}^t\|_2^2}$.

This lemma shows the convergence rate depends on two crucial quantities, $\cos\phi^t$ and λ^t. At the beginning, both $\cos\phi^t$ and λ^t are small. Nevertheless, Lemma C.3 shows λ^t is universally lower bounded by $\Omega\left(\beta^0\right)$. Therefore, after $O(\frac{1}{\eta \cos\phi^0 \beta^0})$ we have $\cos\phi^t = \Omega(1)$. Once $\cos\phi^t = \Omega(1)$, Lemma C.2 shows, after $O\left(\frac{1}{\eta}\right)$ iterations, $\left(\mathbf{a}^t\right)\mathbf{a}^* \|\mathbf{w}^*\| = \Omega\left(\|\mathbf{w}^*\|_2^2 \|\mathbf{a}^*\|_2^2\right)$. Combining the facts $\|\mathbf{v}^t\|_2 \leq 2$ (Lemma C.3) and $\phi^t < \pi/2$, we have $\cos\phi^t \lambda^t = \Omega\left(\|\mathbf{w}^*\|_2^2 \|\mathbf{a}^*\|_2^2\right)$. Now we enter phase II.

In phase II, Lemma 5.5 shows

$$\sin^2 \phi^{t+1} \leq \left(1 - \eta C \|\mathbf{w}^*\|_2^2 \|\mathbf{a}^*\|_2^2\right) \sin^2 \phi^t$$

for some positive absolute constant C. Therefore, we have much faster convergence rate than that in the Phase I. After only $\widetilde{O}\left(\frac{1}{\eta \|\mathbf{w}^*\|_2^2 \|\mathbf{a}\|_2^2} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations, we obtain $\phi \leq \epsilon$.

Once we have this, we can use Lemma C.4 to show $\left|\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}\right| \leq O\left(\epsilon \|\mathbf{a}^*\|_2\right)$ after $\widetilde{O}(\frac{1}{\eta k} \log\left(\frac{1}{\epsilon}\right))$ iterations. Next, using Lemma C.5, we can show after $\widetilde{O}\left(\frac{1}{\eta} \log\frac{1}{\epsilon}\right)$ iterations, $\|\mathbf{a} - \mathbf{a}^*\|_2 = O\left(\epsilon \|\mathbf{a}^*\|_2\right)$. Lastly, Lemma C.6 shows if $\|\mathbf{a} - \mathbf{a}^*\|_2 = O\left(\epsilon \|\mathbf{a}^*\|_2\right)$ and $\phi = O(\epsilon)$ we have we have $\ell(\mathbf{v}, \mathbf{a}) = O\left(\epsilon \|\mathbf{a}^*\|_2^2\right)$.

6. Experiments

In this section, we illustrate our theoretical results with numerical experiments. Again without loss of generality, we assume $\|\mathbf{w}^*\|_2 = 1$ in this section.

6.1. Multi-phase Phenomenon

In Figure 2, we set $k = 20$, $p = 25$ and we consider 4 key quantities in proving Theorem 4.1, namely, angle between $\mathbf{v}$ and $\mathbf{w}^*$ (c.f. Lemma 5.5), $\|\mathbf{a} - \mathbf{a}^*\|$ (c.f. Lemma C.5), $\left|\mathbf{1}^\top \mathbf{a} - \mathbf{1}^\top \mathbf{a}^*\right|$ (c.f. Lemma C.4) and prediction error (c.f. Lemma C.6).

When we achieve the global minimum, all these quantities are 0. At the beginning (first ~ 10 iterations), $\left|\mathbf{1}^\top \mathbf{a} - \mathbf{1}^\top \mathbf{a}^*\right|$ and the prediction error drop quickly. This is because for the gradient of $\mathbf{a}$, $\mathbf{1}\mathbf{1}^\top \mathbf{a}^*$ is the dominating term which will make $\mathbf{1}\mathbf{1}^\top \mathbf{a}$ closer to $\mathbf{1}\mathbf{1}^\top \mathbf{a}^*$ quickly.

After that, for the next ~ 200 iterations, all quantities decrease at a slow rate. This phenomenon is explained to the Phase I stage in Theorem 4.1. The rate is slow because the initial signal is small.

After ~ 200 iterations, all quantities drop at a much faster rate. This is because the signal is very strong and since the convergence rate is proportional to this signal, we have a much faster convergence rate (c.f. Phase II of Theorem 4.1).

6.2. Probability of Converging to the Global Minimum

In this section we test the probability of converging to the global minimum using the random initialization scheme described in Theorem 4.2. We set $p = 6$ and vary k and $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$. We run 5000 random initializations for each $(k, \frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2})$ and compute the probability of converging to the global minimum.

In Theorem 4.3, we showed if $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ is sufficiently small, randomly initialized gradient descent converges to the spurious local minimum with constant probability. Table 1 empirically verifies the importance of this assumption. For

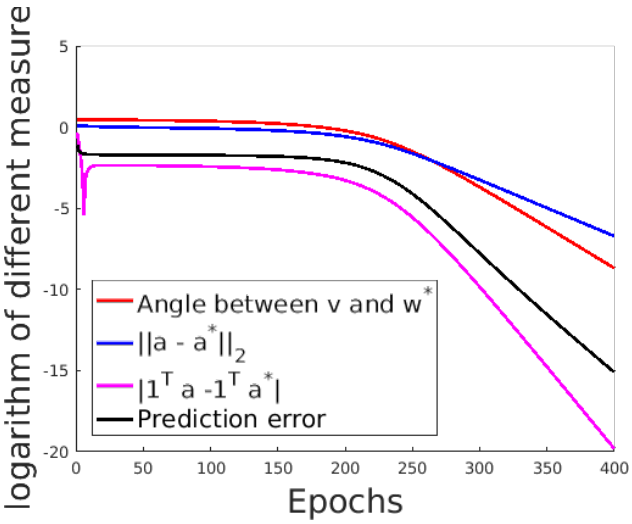


Figure 2. Convergence of different measures we considered in proving Theorem 4.1. In the first $\sim$ 200 iterations, all quantities drop slowly. After that, these quantities converge at much faster linear rates.

every fixed k if $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ becomes larger, the probability of converging to the global minimum becomes larger.

An interesting phenomenon is for every fixed ratio $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ when k becomes lager, the probability of converging to the global minimum becomes smaller. How to quantitatively characterize the relationship between the success probability and the dimension of the second layer is an open problem.

7. Conclusion and Future Works

In this paper we proved the first polynomial convergence guarantee of randomly initialized gradient descent algorithm for learning a one-hidden-layer convolutional neural network. Our result reveals an interesting phenomenon that randomly initialized local search algorithm can converge to a global minimum or a spurious local minimum. We give a quantitative characterization of gradient descent dynamics to explain the two-phase convergence phenomenon. Experimental results also verify our theoretical findings. Here we list some future directions.

Our analysis focused on the population loss with Gaussian input. In practice one uses (stochastic) gradient descent on the empirical loss. Concentration results in (Mei et al., 2016; Soltanolkotabi, 2017) are useful to generalize our results to the empirical version. A more challenging question is how to extend the analysis of gradient dynamics beyond rotationally invariant input distributions. Du et al. (2017b) proved the convergence of gradient descent under some structural input distribution assumptions in the one-layer convolutional neural network. It would be interesting to bring their insights to our setting.

k \ $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}^*\|_2^2}$	0	1	4	9	16	25
25	0.50	0.55	0.73	1	1	1
36	0.50	0.53	0.66	0.89	1	1
49	0.50	0.53	0.61	0.78	1	1
64	0.50	0.51	0.59	0.71	0.89	1
81	0.50	0.53	0.57	0.66	0.81	0.97
100	0.50	0.50	0.57	0.63	0.75	0.90

Table 1. Probability of converging to the global minimum with different $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ and k. For every fixed k, when $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ becomes larger, the probability of converging to the global minimum becomes larger and for every fixed ratio $\frac{(\mathbf{1}^\top \mathbf{a}^*)^2}{\|\mathbf{a}\|_2^2}$ when k becomes lager, the probability of converging to the global minimum becomes smaller.

Another interesting direction is to generalize our result to deeper and wider architectures. Specifically, an open problem is under what conditions randomly initialized gradient descent algorithms can learn one-hidden-layer fully connected neural network or a convolutional neural network with multiple kernels. Existing results often require sufficiently good initialization (Zhong et al., 2017a;b). We believe the insights from this paper, especially the invariance principles in Section 5.1 are helpful to understand the behaviors of gradient-based algorithms in these settings.

8. Acknowledgment

This research was partly funded by NSF grant IIS1563887, AFRL grant FA8750-17-2-0212 DARPA D17AP00001. J.D.L. acknowledges support of the ARO under MURI Award W911NF-11-1-0303. This is part of the collaboration between US DOD, UK MOD and UK Engineering and Physical Research Council (EPSRC) under the Multidisciplinary University Research Initiative.

References

Agarwal, Naman, Allen-Zhu, Zeyuan, Bullins, Brian, Hazan, Elad, and Ma, Tengyu. Finding Approximate Local Minima Faster Than Gradient Descent. In *STOC*, 2017. Full version available at http://arxiv.org/abs/1611.01146.

Bhojanapalli, Srinadh, Neyshabur, Behnam, and Srebro, Nati. Global optimality of local search for low rank matrix recovery. In *Advances in Neural Information Processing Systems*, pp. 3873–3881, 2016.

Blum, Avrim and Rivest, Ronald L. Training a 3-node

neural network is NP-complete. In *Advances in neural information processing systems*, pp. 494–501, 1989.

Brutzkus, Alon and Globerson, Amir. Globally optimal gradient descent for a Convnet with Gaussian inputs. *arXiv preprint arXiv:1702.07966*, 2017.

Carmon, Yair, Duchi, John C, Hinder, Oliver, and Sidford, Aaron. Accelerated methods for non-convex optimization. *arXiv preprint arXiv:1611.00756*, 2016.

Cho, Youngmin and Saul, Lawrence K. Kernel methods for deep learning. In *Advances in neural information processing systems*, pp. 342–350, 2009.

Choromanska, Anna, Henaff, Mikael, Mathieu, Michael, Arous, Gérard Ben, and LeCun, Yann. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pp. 192–204, 2015.

Dauphin, Yann N, Fan, Angela, Auli, Michael, and Grangier, David. Language modeling with gated convolutional networks. *arXiv preprint arXiv:1612.08083*, 2016.

Du, Simon S, Jin, Chi, Lee, Jason D, Jordan, Michael I, Poczos, Barnabas, and Singh, Aarti. Gradient descent can take exponential time to escape saddle points. *arXiv preprint arXiv:1705.10412*, 2017a.

Du, Simon S, Lee, Jason D, and Tian, Yuandong. When is a convolutional filter easy to learn? *arXiv preprint arXiv:1709.06129*, 2017b.

Feizi, Soheil, Javadi, Hamid, Zhang, Jesse, and Tse, David. Porcupine neural networks:(almost) all local optima are global. *arXiv preprint arXiv:1710.02196*, 2017.

Freeman, C Daniel and Bruna, Joan. Topology and geometry of half-rectified network optimization. *arXiv preprint arXiv:1611.01540*, 2016.

Ge, Rong, Huang, Furong, Jin, Chi, and Yuan, Yang. Escaping from saddle points − online stochastic gradient for tensor decomposition. In *Proceedings of The 28th Conference on Learning Theory*, pp. 797–842, 2015.

Ge, Rong, Lee, Jason D, and Ma, Tengyu. Matrix completion has no spurious local minimum. In *Advances in Neural Information Processing Systems*, pp. 2973–2981, 2016.

Ge, Rong, Jin, Chi, and Zheng, Yi. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 1233–1242, 2017a.

Ge, Rong, Lee, Jason D, and Ma, Tengyu. Learning one-hidden-layer neural networks with landscape design. *arXiv preprint arXiv:1711.00501*, 2017b.

Glorot, Xavier and Bengio, Yoshua. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

Goel, Surbhi and Klivans, Adam. Eigenvalue decay implies polynomial-time learnability for neural networks. *arXiv preprint arXiv:1708.03708*, 2017a.

Goel, Surbhi and Klivans, Adam. Learning depth-three neural networks in polynomial time. *arXiv preprint arXiv:1709.06010*, 2017b.

Goel, Surbhi, Kanade, Varun, Klivans, Adam, and Thaler, Justin. Reliably learning the ReLU in polynomial time. *arXiv preprint arXiv:1611.10258*, 2016.

Haeffele, Benjamin D and Vidal, René. Global optimality in tensor factorization, deep learning, and beyond. *arXiv preprint arXiv:1506.07540*, 2015.

Hardt, Moritz and Ma, Tengyu. Identity matters in deep learning. *arXiv preprint arXiv:1611.04231*, 2016.

Hardt, Moritz and Price, Eric. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems*, pp. 2861–2869, 2014.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Janzamin, Majid, Sedghi, Hanie, and Anandkumar, Anima. Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods. *arXiv preprint arXiv:1506.08473*, 2015.

Kawaguchi, Kenji. Deep learning without poor local minima. In *Advances In Neural Information Processing Systems*, pp. 586–594, 2016.

Krizhevsky, Alex, Sutskever, Ilya, and Hinton, Geoffrey E. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105, 2012.

LeCun, Yann, Bottou, Léon, Orr, Genevieve B, and Müller, Klaus-Robert. Efficient backprop. In *Neural networks: Tricks of the trade*, pp. 9–50. Springer, 1998.

Lee, Jason D, Simchowitz, Max, Jordan, Michael I, and Recht, Benjamin. Gradient descent only converges to minimizers. In *Conference on Learning Theory*, pp. 1246–1257, 2016.

Li, Xingguo, Wang, Zhaoran, Lu, Junwei, Arora, Raman, Haupt, Jarvis, Liu, Han, and Zhao, Tuo. Symmetry, saddle points, and global geometry of nonconvex matrix factorization. *arXiv preprint arXiv:1612.09296*, 2016.

Li, Yuanzhi and Yuan, Yang. Convergence analysis of two-layer neural networks with ReLU activation. *arXiv preprint arXiv:1705.09886*, 2017.

Livni, Roi, Shalev-Shwartz, Shai, and Shamir, Ohad. On the computational efficiency of training neural networks. In *Advances in Neural Information Processing Systems*, pp. 855–863, 2014.

Mei, Song, Bai, Yu, and Montanari, Andrea. The landscape of empirical risk for non-convex losses. *arXiv preprint arXiv:1607.06534*, 2016.

Neyshabur, Behnam, Salakhutdinov, Ruslan R, and Srebro, Nati. Path-SGD: Path-normalized optimization in deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 2422–2430, 2015.

Nguyen, Quynh and Hein, Matthias. The loss surface of deep and wide neural networks. *arXiv preprint arXiv:1704.08045*, 2017a.

Nguyen, Quynh and Hein, Matthias. The loss surface and expressivity of deep convolutional neural networks. *arXiv preprint arXiv:1710.10928*, 2017b.

Panigrahy, Rina, Rahimi, Ali, Sachdeva, Sushant, and Zhang, Qiuyi. Convergence results for neural networks via electrodynamics. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 94. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2018.

Park, Dohyung, Kyrillidis, Anastasios, Carmanis, Constantine, and Sanghavi, Sujay. Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach. In *Artificial Intelligence and Statistics*, pp. 65–74, 2017.

Safran, Itay and Shamir, Ohad. On the quality of the initial basin in overspecified neural networks. In *International Conference on Machine Learning*, pp. 774–782, 2016.

Safran, Itay and Shamir, Ohad. Spurious local minima are common in two-layer relu neural networks. *arXiv preprint arXiv:1712.08968*, 2017.

Salimans, Tim and Kingma, Diederik P. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 901–909, 2016.

Sedghi, Hanie and Anandkumar, Anima. Provable methods for training neural networks with sparse connectivity. *arXiv preprint arXiv:1412.2693*, 2014.

Shalev-Shwartz, Shai, Shamir, Ohad, and Shammah, Shaked. Failures of gradient-based deep learning. In *International Conference on Machine Learning*, pp. 3067–3075, 2017a.

Shalev-Shwartz, Shai, Shamir, Ohad, and Shammah, Shaked. Weight sharing is crucial to succesful optimization. *arXiv preprint arXiv:1706.00687*, 2017b.

Shamir, Ohad. Distribution-specific hardness of learning neural networks. *arXiv preprint arXiv:1609.01037*, 2016.

Silver, David, Huang, Aja, Maddison, Chris J, Guez, Arthur, Sifre, Laurent, Van Den Driessche, George, Schrittwieser, Julian, Antonoglou, Ioannis, Panneershelvam, Veda, Lanctot, Marc, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587): 484–489, 2016.

Šíma, Jiří. Training a single sigmoidal neuron is hard. *Neural Computation*, 14(11):2709–2728, 2002.

Soltanolkotabi, Mahdi. Learning ReLUs via gradient descent. *arXiv preprint arXiv:1705.04591*, 2017.

Sun, Ju, Qu, Qing, and Wright, John. Complete dictionary recovery over the sphere I: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63 (2):853–884, 2017.

Tian, Yuandong. An analytical formula of population gradient for two-layered ReLU network and its applications in convergence and critical point analysis. *arXiv preprint arXiv:1703.00560*, 2017.

Xie, Bo, Liang, Yingyu, and Song, Le. Diverse neural network learns true target functions. In *Artificial Intelligence and Statistics*, pp. 1216–1224, 2017.

Zhang, Yuchen, Lee, Jason D, Wainwright, Martin J, and Jordan, Michael I. Learning halfspaces and neural networks with random initialization. *arXiv preprint arXiv:1511.07948*, 2015.

Zhong, Kai, Song, Zhao, and Dhillon, Inderjit S. Learning non-overlapping convolutional neural networks with multiple kernels. *arXiv preprint arXiv:1711.03440*, 2017a.

Zhong, Kai, Song, Zhao, Jain, Prateek, Bartlett, Peter L, and Dhillon, Inderjit S. Recovery guarantees for one-hidden-layer neural networks. *arXiv preprint arXiv:1706.03175*, 2017b.

Zhou, Pan and Feng, Jiashi. The landscape of deep learning algorithms. *arXiv preprint arXiv:1705.07038*, 2017.