Gradient Descent for Non-convex Problems in Modern Machine Learning

Simon Shaolei Du

APRIL 2019
CMU-ML-19-102

Machine Learning Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Barnabás Póczos , Co-Chair
Aarti Singh, Co-Chair
Ruslan Salakhutdinov
Michael I. Jordan (UC Berkeley)

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

This research was sponsored by Department of Energy award DEAR0000596, Department of the Interior award D17AP00001, Air Force Research Laboratory award FA87501720212 and a grant from the Foxconn Technology Group.

Keywords: machine learning, nonconvex optimization, gradient descent, neural network, saddle point, matrix factorization

To my family and my beautiful girlfriend.

Abstract

Machine learning has become an important tool set for artificial intelligence and data science across many fields. A modern machine learning method can be often reduced to a mathematical optimization problem. Among algorithms to solve the optimization problem, gradient descent and its variants like stochastic gradient descent and momentum methods are the most popular ones. The optimization problem induced from classical machine learning methods is often a convex and smooth one, for which gradient descent is guaranteed to solve it efficiently. On the other hand, modern machine learning methods, like deep neural networks, often require solving a non-smooth and non-convex problem. Theoretically, non-convex mathematical optimization problems cannot be solved efficiently. However, in practice, gradient descent and its variants can find a global optimum efficiently. These competing facts show that often there are special structures in the optimization problems that can make gradient descent succeed in practice.

This thesis presents technical contributions to fill the gap between theory and practice on the gradient descent algorithm. The outline of the thesis is as follows.

- In the first part, we consider applying gradient descent to minimize the empirical risk of a neural network. We will show if a multi-layer neural network with smooth activation function is sufficiently wide, then randomly initialized gradient descent can efficiently find a global minimum of the empirical risk. We will also show the same result for the two-layer neural network with Rectified Linear Unit (ReLU) activation function. It is quite surprising that although the objective function of neural networks is non-convex, gradient descent can still find their global minimum. Lastly, we will study structural property of the trajectory induced by the gradient descent algorithm.
- In the second part, we assume the label is generated from a two-layer teacher convolutional neural network and we consider using gradient descent to recover the teacher convolutional neural network. We will show that if the input distribution is Gaussian, then gradient descent can recovered a one-hidden-layer convolutional neural network in which both the convolutional weights and the output wights are unknown parameters to be recovered. We will also show that the Gaussian input assumption can be relaxed to a general structural assumption if we only need to recover a single convolutional filter.
- In the third part, we study conditions under which gradient descent fails. We will show gradient descent can take exponential time to optimize a smooth function with the strict saddle point property for which the noise-injected gradient can optimize in polynomial time.

While our focus is theoretical, whenever possible, we also present experiments that illustrate our theoretical findings.

Acknowledgments

First and foremost, I would like to thank my two amazing PhD advisors Aarti Singh and Barnabás Póczos. Aarti and Barnabás are brilliant researchers. They are knowledgeable and deep in machine learning and related fields. I learned a lot from them on how to find the problem, formulate the problem and solve the problem with appropriate tools. Aarti and Barnabás are also brilliant mentors. At the beginning of my PhD study, they helped me pick the right project that I could make contributions to and later they gave me the freedom to do research in fields that I am passionate about.

I am fortunate to have Ruslan Salakhutdinov and Michael I. Jordan as my thesis committee members. Russ knows every aspect of deep learning and when I switched my research focus to deep learning theory, Russ gave me tremendous help. I knew Mike when I was a junior undergraduate at Berkeley, taking his statistical machine learning course. To me, his instruction is both technical and philosophical. Mike always suggests me trying to look at the problem from a more general point of view and this really shapes my research taste.

I also want to thank my undergraduate mentors Ming Gu, Lei Li, Michael Mahoney and Stuart Russell for developing my interest in scientific research. Particular thank to Ming Gu, who taught me a lot on numerical linear algebra, which I used frequently throughout my PhD.

During my PhD, I did three wonderful internships, at Microsoft Research Redmond, Facebook AI Research Menlo Park and Microsoft Research NYC. At Microsoft Research, I learned a ton about reinforcement learning from my mentors: Alekh Agarwal, Jianshu Chen, Miro Dudík, Nan Jiang, Akshay Krishnamurthy, John Langford, Lihong Li, Lin Xiao and Dengyong Zhou. They taught me how to develop methods that are both theoretically principled and practically useful. At Facebook AI Research, my mentor Yuandong Tian stirred my interest in deep learning theory. Two chapters (Chapter 5 and Chapter 6) of this thesis are based on our collaborated papers.

I have been extremely fortunate to collaborate with a wonderful set of colleagues: Alekh Agarwal, Dave G. Anderson, Sanjeev Arora, Sivaraman Balakrishnan, Nina Balcan, Jianshu Chen, Miro Dudík, Surbhi Goel, Ming Gu, Quanquan Gu, Wei Hu, Nan Jiang, Chi Jin, Michael I. Jordan, Jayanth Koushik, Akshay Krishnamurthy, John Langford, Jason D. Lee, Haochuan Li, Lihong Li, Zhiyuan Li, Michael Mahoney, Barnabás , Póczos, Pradeep Ravikumar, Ruslan Salakhutdinov, Bin Shi, Paloma Sodhi, Aarti Singh, Shashank Singh, Langxuan Su, Weijie Su, Hanqi Sun, Yuandong Tian, Liwei Wang, Ruosong Wang, Yining Wang, Rachel Ward, David Wettergreen, Xiaoxia Wu, Lin Xiao, Wei Yu, Xiyu Zhai, Xiao Zhang (my middle and high school classmate!) and Dengyong Zhou. Special thank to Sivaraman Balakrishnan, Jason D. Lee, and Yining Wang. I had many intellectual discussions with Siva on various technical problems and I learned a ton about statistics from him. Jason and I have very common research interests, namely non-convex optimization and deep learning theory. We can discuss anytime on various places, in person, on Facebook, on

Google Hangouts, on WeChat and on Slack. Five chapters (Chapter 3 - Chapter 7) in this thesis are based on our collaborated papers. Yining helped me a lot in nearly every aspect of research, e.g., proof techniques, experiments, writings, etc, at my early stage as a PhD student. I still remember he told me that I should add a period after formula when writing a paper (facepalm).

I am very grateful to everyone at CMU MLD for contributing to a great environment for graduate studies. I want to thank Diane Stidle and other amazing administrative staff for making the everyday life at MLD so easy for graduate students. Special thanks to all of my friends and peers at CMU. I can't possibly name all of them; instead, let me thank them for making my PhD years some of the most memorable years of my life.

Last but most importantly, I want to thank my parents Jidong Ma and Jun Du for giving the best education I can have and my girlfriend Yangyi Lu for giving unconditional love and happiness during my PhD study. I could not have done this without their encouragement, support and love. I dedicate this thesis to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Machine learning (ML) is an interdisciplinary field that studies how a system can perform a task through learning. For example, do prediction or make decision by looking at data and keep improving the performance as it sees more data. ML methods have made remarkable impact in real world applications. A variety of these applications, including face recognition, machine translation, self-driving cars, search engines, autonomous robots, recommendation systems, computer games, etc., heavily rely on machine learning methods. Yet, theoretically, we do not have a good understanding of these methods.

We start our discussion by describing the core challenge this thesis aims to solve. A machine learning method often consists of two components. First, it has a model that does prediction. This prediction model depends on parameters and the quality of the parameters determine the performance of this model. Second, to find good-quality parameters, the machine learning method chooses a loss function and improves the quality of the parameters iteratively by minimizing this loss function with an optimization algorithm.

Among optimization algorithms, gradient descent (GD) is perhaps the simplest one and acts as the prototype of many advanced optimization algorithms like stochastic gradient decent (SGD) or momentum methods. These gradient-based methods provide the core optimization methodology in machine learning problems. Given a function $L(\theta)$ where θ denotes the model parameters, the gradient descent method can be written as:

$$\theta(k+1) \leftarrow \theta(k) - \eta \nabla L\left(\theta(k)\right), \tag{1.1}$$

where $\eta > 0$ is a step size, $\nabla L(\theta)$ is the gradient of L at θ and $\theta(0)$ is the initial point.

For classical machine learning methods, the loss function L is often a convex one. For example, for the linear regression, the loss function has the form

$$L(\theta) = \frac{1}{2n} \sum_{i=1}^{n} \left(\mathbf{x}_i^\top \theta - y_i\right)^2 \tag{1.2}$$

where $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^d$ are input data and $y_1, \ldots, y_n \in \mathbb{R}$ are labels that we want to predict based on the inputs. The loss function for linear regression is smooth and convex. Therefore, through standard analysis in convex optimization, one can show gradient descent finds an ϵ-suboptimal solution in $O(1/\epsilon)$ iterations under certain regularity conditions.

However, for modern machine learning methods, models and loss functions become significantly more complicated. For example, for deep fully-connected neural networks, the loss function is

$$L(\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_H) = \frac{1}{2n} \sum_{i=1}^{n} \left(\mathbf{W}_H \sigma\left(\mathbf{W}_{H-1} \sigma\left(\cdots \mathbf{W}_2 \sigma\left(\mathbf{W}_1 \mathbf{x}_i\right)\right)\right) - y_i\right)^2 \tag{1.3}$$

where $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^{d_0}$ are inputs, $y_1, \ldots, y_n \in \mathbb{R}^{d_H}$ are labels we want to predict, $\mathbf{W}_h \in \mathbb{R}^{d_h \times d_{h-1}}$ for $h = 1, \ldots, H$ with $d_H = 1$ are weight matrices we want to optimize, i.e., the parameters in the deep neural network model and $\sigma(\cdot)$ is a point-wise activation function, e.g., Rectified Linear Unit (ReLU): $\sigma(z) = \max\{0, z\}$. Comparing with loss function (1.2), the loss function for deep fully-connected neural network does not have the desired properties like convexity and smoothness. Indeed, in the worst case, it is unlikely an algorithm can find an ϵ-suboptimal solution in polynomial time [9].

While theoretically it is not possible to optimize loss functions like Equation (1.3) efficiently, in practice, simple algorithms like gradient descent can optimize highly non-convex functions that arising in machine learning problems quite successfully. For loss function (1.3), GD and its variants with *random initialization* can often find a global optimal solution efficiently. From traditional view of optimization, these gradient-based methods can easily converge to bad local minima or saddle points but for non-convex loss functions arising from various machine learning problems, these bad scenarios often do not happen. A fundamental theoretical question is

Why randomly initialized gradient descent algorithm can find globally optimal solution of non-convex loss functions arising from machine learning efficiently?

In this thesis, we aim to answer this question in a rigorous manner. Roughly speaking, there are two main approaches for studying the behavior of optimization algorithms.

The first approach is based on the geometry. If one can show that the loss function satisfies certain geometric properties, e.g., convexity, then one can apply a generic theory to derive the convergence rate of gradient descent. Recently, researchers have identified two such geometric properties: 1) all local minima are global and 2) all saddle points are strict, i.e., there exists a negative curvature. They showed if the loss function satisfies these two properties, then the *perturbed* gradient descent algorithm can optimize the loss function efficiently. This is a large function class and includes many non-convex losses that arise in machine learning problems, including tensor decomposition [36], dictionary learning [71], phase retrieval [70], matrix sensing [8, 61], matrix completion [37, 38], and matrix factorization [49]. This thesis will provide new insight on the performance of the vanilla gradient descent algorithm for optimizing functions in this class. While this approach is general and often gives simple analysis, the main drawback is that there are a large number of problems, for example, deep neural networks, that do not belong to this class and so we need other approaches.

The second approach is based on analyzing the trajectory generated by the optimization algorithm. For a specific loss function like (1.3), we can write its gradient in an explicit form. This explicit form allows us to track the trajectory generated by gradient descent in a fine-grained manner. Comparing with the geometric approach, analyzing the dynamical system induced by a first order method is often more complicated because at different stages of the optimization procedure, we need to study different properties of the dynamical system, which requires more

insights for the specific problems. Nevertheless, this approach often gives tighter convergence rate because it is problem-specific and more importantly, we can analyze non-convex problems such as neural networks that do not have the benign geometric properties.

1.1 Overview of Thesis

In this section we give an overview of this thesis. This thesis consists of three parts.

Gradient Descent for Empirical Risk Minimization in Deep Learning In the first part, we study the theoretical properties of gradient descent for minimizing the non-convex empirical risk of neural networks, e.g., the loss function (1.3). As we have discussed, in practice, GD and its variants can minimize the empirical loss of neural network easily. Surprisingly, this property is not correlated with what labels are being used. In [81], authors replaced the true labels with randomly generated labels, but still found that randomly initialized first order methods can always achieve zero training loss when the network is large enough.

A widely believed explanation of why a neural network can fit all training labels is that the neural network is over-parameterized. For example, Wide ResNet [80] uses 100x parameters than the number of training data. Thus there must exist one such neural network of this architecture that can fit all training data. However, the existence does not imply why the network found by a randomly initialized first order method can fit all the data. The objective function is neither smooth nor convex, which makes traditional analysis techniques from convex optimization not useful in this setting.

In Chapter 2, we study two-layer over-parameterized neural networks with rectified linear units (ReLU) activation. Though there are only two layers, the loss is still non-convex and non-smooth. We show that randomly initialized gradient descent can achieve zero training loss, a.k.a a global minimum, with a linear convergence rate. Our analysis relies on relating the trajectory of GD for optimizing the neural network to the trajectory of GD for optimizing a convex function for which GD enjoys a linear convergence rate. We show these two trajectories are close to each other and this implies GD for optimizing the neural network also enjoys a linear convergence rate. This chapter is based on the paper [35].

In Chapter 3, we extend the same proof technique in Chapter 2 to analyze over-parameterized deep neural networks with smooth activation functions. We show that gradient descent achieves zero training loss in polynomial time for a deep over-parameterized neural network with residual connections [44].

We further extend our analysis to deep residual convolutional neural networks and obtain a similar convergence result. This chapter is based on the paper [30].

In Chapter 4, we take a closer look at the dynamics of gradient descent. We prove that gradient flow (i.e. gradient descent with infinitesimal step size) effectively enforces the differences between squared norms across different layers to remain invariant *without any explicit regularization*. This result implies that if the weights are initially small, gradient flow automatically balances the magnitudes of all layers. Using a discretization argument, we analyze gradient descent with positive step size for the non-convex low-rank asymmetric matrix factorization problem, i.e., two-layer neural network with linear activation, without any regularization. We prove that

gradient descent with decreasing step sizes automatically balances two low-rank factors and converges to a bounded global optimum. Furthermore, for rank-1 asymmetric matrix factorization we give a finer analysis showing that gradient descent with constant step size converges to the global minimum at a globally linear rate. Experimentally, we find that for multi-layer neural networks, gradient descent with positive step size automatically balances weight matrices as well. The balancedness shows that gradient descent automatically maintains a certain smoothness on its trajectory. These findings on the invariance could serve as a fundamental building block for understanding optimization in deep models. This chapter is based on the paper [29].

Learning Convolutional Neural Networks by Gradient Descent In the first part, we show that for sufficiently wide neural networks with smooth activation function and sufficiently wide two-layer neural networks with ReLU activation function, gradient descent can find a global minimum of the empirical risk. However, minimizing the empirical risk does not necessarily imply the learned neural network has good generalization ability. In the second part, we study when gradient descent can learn a convolutional neural network with good generalization ability. We assume there is a teacher convolutional neural network and the label is generated according to this convolutional neural network. We show under general conditions, applying gradient descent on the quadratic loss can approximately recover the planted neural network, which implies the learned neural network achieve small generalization error.

There are two main challenges in this part. First, to recover the planted neural network, the models considered in this part are not over-parameterized . Therefore, we need different techniques to analyze the dynamics of gradient descent. Second, to prove the approximate recover guarantee, we need to take the randomness in the dynamics of gradient descent into account.

In Chapter 5, we consider the problem of recovering a one-hidden-layer neural network with non-overlapping convolutional layer and ReLU activation function in which both the convolutional weights and the output weights are parameters to be learned. We prove that with Gaussian input, there is a spurious local minimum that is not a global mininum. Surprisingly, in the presence of local minimum, starting from randomly initialized weights, gradient descent with weight normalization can still be proven to recover the true parameters with constant probability (which can be boosted to arbitrarily high accuracy 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. This chapter is based on the paper [20].

Chapter 5 relies on Gaussian input assumption which may not be satisfied in real world problems. In Chapter 6, we analyze the convergence of (stochastic) gradient descent algorithm for learning a convolutional filter with ReLU activation function. Our analysis does not rely on any specific form of the input distribution, e.g., Gaussian and our proofs only use the definition of ReLU. We show that (stochastic) gradient descent with random initialization can learn the convolutional filter in polynomial time and the convergence rate depends on the smoothness of the input distribution and the closeness of patches. This chapter is based on the paper [31]

When Does Gradient Descent Fail? So far we have discussed the positive results on using gradient descent but it is also important to study the limitation of this algorithm.

In Chapter 7, we show that even if the objective function is smooth and satisfies so-called "strict saddle" property, which are satisfied in many machine learning problems, gradient descent [45] can take exponential time to converge whereas the perturbed gradient descent only requires polynomial time. This result demonstrates that gradient descent may not be the appropriate algorithm for certain problems and some modification, e.g. adding perturbation is needed to ensure the polynomial convergence. This chapter is based on the paper [26].

1.2 Bibliographic Notes

The research presented in this thesis is based on joint work with several co-authors, described below. This thesis only includes works for which this author was the, or one of the, primary contributors.

Chapter 2 is based on joint work with Xiyu Zhai, Barnabás Póczos and Aarti Singh. Chapter 3 is based on joint work with Jason D. Lee, Haochuan Li, Liwei Wang and Xiyu Zhai. Chapter 4 is based on joint work with Wei Hu and Jason D. Lee. Chapter 5 is based on joint work with Jason D. Lee, Yuandong Tian, Barnabás Póczos and Aarti Singh. Chapter 6 is based on joint work with Jason D. Lee and Yuandong Tian. Finally, Chapter 7 is based on joint work with Chi Jin, Jason D. Lee, Michael I. Jordan, Barnabás Póczos and Aarti Singh.

1.2.1 Excluded Research

In an effort to keep this dissertation succinct and coherent, a significant portion of this authors Ph.D. work has been excluded from this document. The excluded research includes:

- Work on other theoretical aspects of neural networks [5, 21, 23, 24, 33].
- Work on reinforcement learning [25, 34].
- Work on matrix analysis [7, 28, 83].
- Work on robust statistics [6, 32].
- Work on transfer learning [27].
- Work on convex-concave saddle point problems [22].

1.3 Notations

Here we list notations that we will use throughout the thesis. For notations that will be used only in a specific chapter, we will introduce them therein. For a positive integer N we denote $[N] = \{1, \ldots, N\}$. For a vector $\mathbf{v}$, we use $\|\mathbf{v}\|_2$ to denote its Euclidean norm and $\|\mathbf{v}\|$ to denote a general norm. For a matrix $\mathbf{A}$, we use $\|\mathbf{A}\|_{\mathrm{op}}$ to denote its operator norm and $\|\mathbf{A}\|_{\mathrm{F}}$ to denote its Frobenius norm. We use $\sigma_i(\mathbf{A})$ to denote the i-th singular value of $\mathbf{A}$ and $\sigma_{\min}(\mathbf{A})$ to denote the smallest singular value. If $\mathbf{A}$ is symmetric, we use $\lambda_i(\mathbf{A})$ to denote its i-th eigenvalue and $\lambda_{\min}(\mathbf{A})$ to denote its smallest eigenvalue. We use $\langle \cdot, \cdot \rangle$ to denote the inner product between two

vectors or matrices. For the gradient descent algorithm considered in this thesis, we will use $\eta > 0$ to denote the step size. For a function f we use ∇f to denote its gradient and $\nabla^2 f$ to denote its Hessian. To measure the performance, we will use $\epsilon > 0$ to denote the target accuracy and $\delta > 0$ to denote the failure probability. Many of our analyses involve probability arguments. We use $\mathbb{I}\{\mathcal{E}\}$ to denote the event $\mathcal{E}$ happens. We use $\mathcal{N}(\mu, \mathbf{\Sigma})$ to denote a Gaussian distribution with mean μ and covariance $\mathbf{\Sigma}$ and $\mathrm{unif}(S)$ to denote a uniform distribution over the set S. Lastly, we use O, Ω, Θ to denote the usual Big-O, Big-Omega and Big-Theta notations. The notations are summarized in Table 1.1.

$[N]$	The set of first N positive integers: $1, \ldots, N$.
$\|\cdot\|$	A general norm
$\|\cdot\|_2$	Euclidean norm of a vector.
$\|\cdot\|_{\mathrm{op}}$	Operator norm of a matrix.
$\|\cdot\|_{\mathrm{F}}$	Frobenius norm of a matrix.
$\langle \cdot, \cdot \rangle$	Inner product.
$\lambda_i(\cdot)$	i-th eigenvalue of a matrix.
$\lambda_{\min}$	The smallest eigenvalue of a matrix.
$\sigma_i(\cdot)$	i-th singular value of a matrix.
$\sigma_{\min}(\cdot)$	The smallest singular value of a matrix.
η	Step size of the gradient descent algorithm.
$\mathbb{I}\{\cdot\}$	Indicator of an event.
$\mathcal{N}(\mu, \mathbf{\Sigma})$	A Gaussian distribution with mean μ and covariance $\mathbf{\Sigma}$.
$\mathrm{unif}(S)$	Uniform distribution over the set S.
$\nabla(\cdot)$	Gradient operator.
$\nabla^2(\cdot)$	Hessian operator.
O	Big-O notation.
Ω	Big-Omega notation.
Θ	Big-Theta notation.

Table 1.1: Notation table.

Part I

Gradient Descent for Empirical Risk Minimization in Deep Learning

Chapter 2

Gradient Descent Provably Optimizes Over-paramterized Two-layer ReLU Neural Networks

2.1 Introduction

In this chapter we show why randomly initialized gradient descent can optimize a two-layer neural networks with rectified linear unit (ReLU) activation. Formally, we consider a neural network of the following form.

$$f(\mathbf{W}, \mathbf{a}, \mathbf{x}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} a_r \sigma\left(\mathbf{w}_r^\top \mathbf{x}\right)$$

where $\mathbf{x} \in \mathbb{R}^d$ is the input, $\mathbf{w}_r \in \mathbb{R}^d$ is the weight vector of the first layer, $a_r \in \mathbb{R}$ is the output weight and $\sigma(\cdot)$ is the ReLU activation function: $\sigma(z) = z$ if $z \geq 0$ and $\sigma(z) = 0$ if $z < 0$. We focus on the empirical risk minimization problem with a quadratic loss. Given a training data set $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, we want to minimize

$$L(\mathbf{W}, \mathbf{a}) = \sum_{i=1}^{n} \frac{1}{2} \left(f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i) - y_i\right)^2. \tag{2.1}$$

Our main focus of this chapter is to analyze the following procedure. We fix the second layer and apply gradient descent (GD) to optimize the first layer

$$\mathbf{W}(k+1) = \mathbf{W}(k) - \eta \frac{\partial L(\mathbf{W}(k), \mathbf{a})}{\partial \mathbf{W}(k)}. \tag{2.2}$$

where $\eta > 0$ is the step size. Here the gradient formula for each weight vector is [1]

$$\frac{\partial L(\mathbf{W}, \mathbf{a})}{\partial \mathbf{w}_r} = \frac{1}{\sqrt{m}} \sum_{i=1}^{n} (f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i) - y_i) \mathbf{a}_r \mathbf{x}_i \mathbb{I}\left\{\mathbf{w}_r^\top \mathbf{x}_i \geq 0\right\}. \tag{2.3}$$

[1] Note ReLU is not continuously differentiable. One can view $\frac{\partial L(\mathbf{W})}{\partial \mathbf{w}_r}$ as a convenient notation for the right hand side of (2.3) and this is the update rule used in practice.

Though this is only a shallow fully connected neural network, the objective function is still non-smooth and non-convex due to the use of ReLU activation function. Even for this simple function, why randomly initialized first order method can achieve zero training error is not known. Many previous works have tried to answer this question or similar ones. Attempts include landscape analysis [68], partial differential equations [55], analysis of the dynamics of the algorithm [51], optimal transport theory [12], to name a few. These results often make strong assumptions on the labels and input distributions or do not imply why randomly initialized first order method can achieve zero training loss.

In this chapter, we rigorously prove that as long as no two inputs are parallel and m is large enough, with randomly initialized $\mathbf{a}$ and $\mathbf{W}(0)$, gradient descent achieves zero training loss at a linear convergence rate. Thus, our theoretical result not only shows the global convergence but also gives a quantitative convergence rate in terms of the desired accuracy.

Analysis Technique Overview Our proof relies on the following insights. First we directly analyze the dynamics of each individual prediction $f(\mathbf{W}, \mathbf{a}, \mathbf{x}_i)$ for $i = 1, \ldots, n$. This is different from many previous work [31, 51] which tried to analyze the dynamics of the parameters $(\mathbf{W})$. Note because the objective function is non-smooth and non-convex, analysis of the parameter space dynamics is very difficult. In contrast, we find the dynamics of prediction space is governed by the spectral property of a Gram matrix (which can vary in each iteration, c.f. Equation (2.5)) and as long as this Gram matrix's least eigenvalue is lower bounded, gradient descent enjoys a linear rate. It is easy to show as long as no two inputs are parallel, in the initialization phase, this Gram matrix has a lower bounded least eigenvalue (c.f. Theorem 2.1). Thus the problem reduces to showing the Gram matrix at later iterations is close to that in the initialization phase. Our second observation is this Gram matrix is only related to the activation patterns ($\mathbb{I}\left\{\mathbf{w}_r^\top \mathbf{x}_i \geq 0\right\}$) and we can use matrix perturbation analysis to show if most of the patterns do not change, then this Gram matrix is close to its initialization. Our third observation is we find over-parameterization, random initialization, and the linear convergence jointly restrict every weight vector $\mathbf{w}_r$ to be close to its initialization. Then we can use this property to show most of the patterns do not change. Notably, our proof only uses linear algebra and standard probability bounds so we believe it can be easily generalized to analyze deep neural networks.

2.2 Continuous Time Analysis

In this section, we present our result for gradient flow, i.e., gradient descent with infinitesimal step size. The analysis of gradient flow is a stepping stone towards understanding discrete algorithms. In the next section, we will modify the proof and give a quantitative bound for gradient descent with positive step size. Formally, we consider the ordinary differential equation defined by: [2]

$$\frac{d\mathbf{w}_r(t)}{dt} = -\frac{\partial L(\mathbf{W}(t), \mathbf{a})}{\partial \mathbf{w}_r(t)}$$

[2] Strictly speaking, this should be differential inclusion [17]

for $r \in [m]$. We denote $u_i(t) = f(\mathbf{W}(t), \mathbf{a}, \mathbf{x}_i)$ the prediction on input $\mathbf{x}_i$ at time t and we let $\mathbf{u}(t) = (u_1(t), \ldots, u_n(t)) \in \mathbb{R}^n$ be the prediction vector at time t. We first state our main assumption.

Assumption 2.1. *Define a matrix*

$$\mathbf{H}^\infty \in \mathbb{R}^{n\times n} \text{ with } \mathbf{H}^\infty_{ij} = \mathbb{E}_{\mathbf{w}\sim\mathcal{N}(\mathbf{0},\mathbf{I})}\left[\mathbf{x}_i^\top \mathbf{x}_j \mathbb{I}\left\{\mathbf{w}^\top \mathbf{x}_i \geq 0, \mathbf{w}^\top \mathbf{x}_j \geq 0\right\}\right].$$

We assume $\lambda_0 \triangleq \lambda_{\min}\left(\mathbf{H}^\infty\right) > 0$.

$\mathbf{H}^\infty$ is the Gram matrix induced by the ReLU activation function and the random initialization. Later we will show that during the training, though the Gram matrix may change (c.f. Equation (2.5)), it is still close to $\mathbf{H}^\infty$. Furthermore, as will be apparent in the proof (c.f. Equation (2.6)), $\mathbf{H}^\infty$ is the fundamental quantity that determines the convergence rate. Interestingly, various properties of this $\mathbf{H}^\infty$ matrix has been studied in previous works [74, 79]. Now to justify this assumption, the following theorem shows if no two inputs are parallel the least eigenvalue is strictly positive.

Theorem 2.1. *If for any* $i \neq j$, $\mathbf{x}_i \nparallel \mathbf{x}_j$, *then* $\lambda_0 > 0$.

Note for most real world datasets, no two inputs are parallel, so our assumption holds in general. Now we are ready to state our main theorem in this section.

Theorem 2.2 (Convergence Rate of Gradient Flow)**.** *Suppose Assumption 2.1 holds and for all* $i \in [n]$, $\|\mathbf{x}_i\|_2 = 1$ *and* $|y_i| \leq C$ *for some constant* C. *Then if we set the number of hidden nodes* $m = \Omega\left(\frac{n^6}{\lambda_0^4 \delta^3}\right)$ *and we i.i.d. initialize* $\mathbf{w}_r \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $a_r \sim \mathrm{unif}\left[\{-1, 1\}\right]$ *for* $r \in [m]$, *with probability at least* $1 - \delta$ *over the initialization, we have*

$$\|\mathbf{u}(t) - \mathbf{y}\|_2^2 \leq \exp(-\lambda_0 t) \|\mathbf{u}(0) - \mathbf{y}\|_2^2.$$

This theorem establishes that if m is large enough, the training error converges to 0 at a linear rate. Here we assume $\|\mathbf{x}_i\|_2 = 1$ only for simplicity and it is not hard to relax this condition. The bounded label condition also holds for most real world data set. The number of hidden nodes m required is $\Omega\left(\frac{n^6}{\lambda_0^4 \delta^3}\right)$, which depends on the number of samples n, λ_0, and the failure probability δ. Over-parameterization, i.e., the fact $m = \mathrm{poly}(n, 1/\lambda_0, 1/\delta)$, plays a crucial role in guaranteeing gradient descent to find the global minimum. Lastly, we note the specific convergence rate depends on λ_0 but independent of the number of hidden nodes m.

2.2.1 Proof of Theorem 2.2

Our first step is to calculate the dynamics of each prediction.

$$\begin{aligned}
\frac{d}{dt} u_i(t) &= \sum_{r=1}^m \langle \frac{\partial f(\mathbf{W}(t), \mathbf{a}, \mathbf{x}_i)}{\partial \mathbf{w}_r(t)}, \frac{d\mathbf{w}_r(t)}{dt} \rangle \\
&= \sum_{j=1}^n (y_j - u_j) \sum_{r=1}^m \langle \frac{\partial f(\mathbf{W}(t), \mathbf{a}, \mathbf{x}_i)}{\partial \mathbf{w}_r(t)}, \frac{\partial f(\mathbf{W}(t), \mathbf{a}, \mathbf{x}_j)}{\partial \mathbf{w}_r(t)} \rangle \\
&\triangleq \sum_{j=1}^n (y_j - u_j) \mathbf{H}_{ij}(t) \qquad (2.4)
\end{aligned}$$

where $\mathbf{H}(t)$ is an $n \times n$ matrix with (i,j)-th entry

$$\mathbf{H}_{ij}(t) = \frac{1}{m}\mathbf{x}_i^\top \mathbf{x}_j \sum_{r=1}^{m} \mathbb{I}\left\{\mathbf{x}_i^\top \mathbf{w}_r(t) \geq 0, \mathbf{x}_j^\top \mathbf{w}_r(t) \geq 0\right\}. \tag{2.5}$$

With this $\mathbf{H}(t)$ matrix, we can write the dynamics of predictions in a compact way:

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{H}(t)(\mathbf{y} - \mathbf{u}(t)). \tag{2.6}$$

$\mathbf{H}(t)$ is a time-dependent symmetric matrix. We first analyze its property when $t = 0$. The following lemma shows if m is large then $\mathbf{H}(0)$ has a lower bounded least eigenvalue with high probability. The proof is by the standard concentration bound so we defer it to the appendix.

Lemma 2.1. *If $m = \Omega\left(\frac{n^2}{\lambda_0^2}\log\left(\frac{n}{\delta}\right)\right)$, we have with probability at least $1-\delta$, $\|\mathbf{H}(0) - \mathbf{H}^\infty\|_2 \leq \frac{\lambda_0}{4}$ and $\lambda_{\min}(\mathbf{H}(0)) \geq \frac{3}{4}\lambda_0$.*

Our second step is to show $\mathbf{H}(t)$ is stable in terms of $\mathbf{W}(t)$. Formally, the following lemma shows for any $\mathbf{W}$ close to $\mathbf{W}(0)$, the induced Gram matrix $\mathbf{H}$ is close to $\mathbf{H}(0)$ and has a lower bounded least eigenvalue.

Lemma 2.2. *If $\mathbf{w}_1, \ldots, \mathbf{w}_m$ are i.i.d. generated from $\mathcal{N}(\mathbf{0}, \mathbf{I})$, then with probability at least $1-\delta$, the following holds. For any set of weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_m \in \mathbb{R}^d$ that satisfy for any $r \in [m]$, $\|\mathbf{w}_r(0) - \mathbf{w}_r\|_2 \leq \frac{c\delta\lambda_0}{n^2} \triangleq R$ for some small positive constant c, then the matrix $\mathbf{H} \in \mathbb{R}^{n\times n}$ defined by*

$$\mathbf{H}_{ij} = \frac{1}{m}\mathbf{x}_i^\top \mathbf{x}_j \sum_{r=1}^{m} \mathbb{I}\left\{\mathbf{w}_r^\top \mathbf{x}_i \geq 0, \mathbf{w}_r^\top \mathbf{x}_j \geq 0\right\}$$

satisfies $\|\mathbf{H} - \mathbf{H}(0)\|_2 < \frac{\lambda_0}{4}$ and $\lambda_{\min}(\mathbf{H}) > \frac{\lambda_0}{2}$.

This lemma plays a crucial role in our analysis so we give the proof below.

Proof of Lemma 2.2. We define the event

$$A_{ir} = \left\{\exists \mathbf{w} : \|\mathbf{w} - \mathbf{w}_r(0)\| \leq R, \mathbb{I}\left\{\mathbf{x}_i^\top \mathbf{w}_r(0) \geq 0\right\} \neq \mathbb{I}\left\{\mathbf{x}_i^\top \mathbf{w} \geq 0\right\}\right\}.$$

Note this event happens if and only if $\left|\mathbf{w}_r(0)^\top \mathbf{x}_i\right| < R$. Recall $\mathbf{w}_r(0) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. By anti-concentration inequality of Gaussian, we have $P(A_{ir}) = P_{z\sim\mathcal{N}(0,1)}(|z| < R) \leq \frac{2R}{\sqrt{2\pi}}$. Therefore, for any set of weight vectors $\mathbf{w}_1, \ldots, \mathbf{w}_m$ that satisfy the assumption in the lemma, we can bound the entry-wise deviation on their induced matrix $\mathbf{H}$: for any $(i,j) \in [n] \times [n]$

$$\begin{aligned}
&\mathbb{E}\left[|\mathbf{H}_{ij}(0) - \mathbf{H}_{ij}|\right] \\
=&\mathbb{E}\left[\frac{1}{m}\left|\mathbf{x}_i^\top \mathbf{x}_j \sum_{r=1}^{m}\left(\mathbb{I}\left\{\mathbf{w}_r(0)^\top \mathbf{x}_i \geq 0, \mathbf{w}_r(0)^\top \mathbf{x}_j \geq 0\right\} - \mathbb{I}\left\{\mathbf{w}_r^\top \mathbf{x}_i \geq 0, \mathbf{w}_r^\top \mathbf{x}_j \geq 0\right\}\right)\right|\right] \\
\leq&\frac{1}{m}\sum_{r=1}^{m}\mathbb{E}\left[\mathbb{I}\left\{A_{ir} \cup A_{jr}\right\}\right] \leq \frac{4R}{\sqrt{2\pi}}
\end{aligned}$$

where the expectation is taken over the random initialization of $\mathbf{w}_1(0), \ldots, \mathbf{w}_m(0)$. Summing over (i,j), we have $\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(n,n)}|\mathbf{H}_{ij} - \mathbf{H}_{ij}(0)|\right] \leq \frac{4n^2R}{\sqrt{2\pi}}$. Thus by Markov's inequality, with

probability $1-\delta$, we have $\sum_{(i,j)=(1,1)}^{(n,n)} |\mathbf{H}_{ij} - \mathbf{H}_{ij}(0)| \leq \frac{4n^2R}{\sqrt{2\pi\delta}}$. Next, we use matrix perturbation theory to bound the deviation from the initialization

$$\|\mathbf{H} - \mathbf{H}(0)\|_2 \leq \|\mathbf{H} - \mathbf{H}(0)\|_F \leq \sum_{(i,j)=(1,1)}^{(n,n)} |\mathbf{H}_{ij} - \mathbf{H}_{ij}(0)| \leq \frac{4n^2R}{\sqrt{2\pi\delta}}.$$

Lastly, we lower bound the smallest eigenvalue by plugging in R

$$\lambda_{\min}(\mathbf{H}) \geq \lambda_{\min}(\mathbf{H}(0)) - \frac{4n^2R}{\sqrt{2\pi\delta}} \geq \frac{\lambda_0}{2}.$$

□

The next lemma shows two facts if the least eigenvalue of $\mathbf{H}(t)$ is lower bounded. First, the loss converges to 0 at a linear convergence rate. Second, $\mathbf{w}_r(t)$ is close to the initialization for every $r \in [m]$. This lemma clearly demonstrates the power of over-parameterization.

Lemma 2.3. *Suppose for* $0 \leq s \leq t$, $\lambda_{\min}(\mathbf{H}(s)) \geq \frac{\lambda_0}{2}$. *Then we have* $\|\mathbf{y} - \mathbf{u}(t)\|_2^2 \leq \exp(-\lambda_0 t)\|\mathbf{y} - \mathbf{u}(0)\|_2^2$ *and for any* $r \in [m]$, $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq \frac{\sqrt{n}\|\mathbf{y}-\mathbf{u}(0)\|_2}{\sqrt{m}\lambda_0} \triangleq R'$.

Proof of Lemma 2.3. Recall we can write the dynamics of predictions as $\frac{d}{dt}\mathbf{u}(t) = \mathbf{H}(\mathbf{y} - \mathbf{u}(t))$. We can calculate the loss function dynamics

$$\begin{aligned}\frac{d}{dt}\|\mathbf{y} - \mathbf{u}(t)\|_2^2 =& -2(\mathbf{y} - \mathbf{u}(t))^\top \mathbf{H}(t)(\mathbf{y} - \mathbf{u}(t)) \\ \leq& -\lambda_0 \|\mathbf{y} - \mathbf{u}(t)\|_2^2.\end{aligned}$$

Thus we have $\frac{d}{dt}\left(\exp(\lambda_0 t)\|\mathbf{y} - \mathbf{u}(t)\|_2^2\right) \leq 0$ and $\exp(\lambda_0 t)\|\mathbf{y} - \mathbf{u}(t)\|_2^2$ is a decreasing function with respect to t. Using this fact we can bound the loss

$$\|\mathbf{y} - \mathbf{u}(t)\|_2^2 \leq \exp(-\lambda_0 t)\|\mathbf{y} - \mathbf{u}(0)\|_2^2.$$

Therefore, $\mathbf{u}(t) \to \mathbf{y}$ exponentially fast. Now we bound the gradient norm. Recall for $0 \leq s \leq t$,

$$\begin{aligned}\left\|\frac{d}{ds}\mathbf{w}_r(s)\right\|_2 =& \left\|\sum_{i=1}^n (y_i - u_i)\frac{1}{\sqrt{m}} a_r \mathbf{x}_i \mathbb{I}\left\{\mathbf{w}_r(s)^\top \mathbf{x}_i \geq 0\right\}\right\|_2 \\ \leq& \frac{1}{\sqrt{m}}\sum_{i=1}^n |y_i - u_i(s)| \\ \leq& \frac{\sqrt{n}}{\sqrt{m}}\|\mathbf{y} - \mathbf{u}(s)\|_2 \leq \frac{\sqrt{n}}{\sqrt{m}}\exp(-\lambda_0 s)\|\mathbf{y} - \mathbf{u}(0)\|_2.\end{aligned}$$

Integrating the gradient, we can bound the distance from the initialization

$$\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq \int_0^t \left\|\frac{d}{ds}\mathbf{w}_r(s)\right\|_2 ds \leq \frac{\sqrt{n}\|\mathbf{y} - \mathbf{u}(0)\|_2}{\sqrt{m}\lambda_0}.$$

□

The next lemma shows if $R' < R$, the conditions in Lemma 2.2 and 2.3 hold for all $t \geq 0$. The proof is by contradiction and we defer it to appendix.

Lemma 2.4. *If $R' < R$, we have for all $t \geq 0$, $\lambda_{\min}(\mathbf{H}(t)) \geq \frac{1}{2}\lambda_0$, for all $r \in [m]$,*

$$\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2 \leq R'$$

and

$$\|\mathbf{y} - \mathbf{u}(t)\|_2^2 \leq \exp(-\lambda_0 t) \|\mathbf{y} - \mathbf{u}(0)\|_2^2 .$$

Thus it is sufficient to show $R' < R$ which is equivalent to $m = \Omega\left(\frac{n^5\|\mathbf{y}-\mathbf{u}(0)\|_2^2}{\lambda_0^4\delta^2}\right)$.

We bound

$$\begin{aligned}\mathbb{E}\left[\|\mathbf{y} - \mathbf{u}(0)\|_2^2\right] &= \sum_{i=1}^{n}(y_i^2 + y_i\mathbb{E}\left[f(\mathbf{W}(0), \mathbf{a}, \mathbf{x}_i)\right] + \mathbb{E}\left[f(\mathbf{W}(0), \mathbf{a}, \mathbf{x}_i)^2\right]) \\ &= \sum_{i=1}^{n}(y_i^2 + 1) = O(n).\end{aligned}$$

Thus by Markov's inequality, we have with probability at least $1 - \delta$, $\|\mathbf{y} - \mathbf{u}(0)\|_2^2 = O(\frac{n}{\delta})$. Plugging in this bound we prove the theorem.

2.3 Discrete Time Analysis

In this section, we show randomly initialized gradient descent with a constant positive step size converges to the global minimum at a linear rate. We first present our main theorem.

Theorem 2.3 (Convergence Rate of Gradient Descent)**.** *Under the same assumptions as in Theorem 2.2, if we set the number of hidden nodes $m = \Omega\left(\frac{n^6}{\lambda_0^4\delta^3}\right)$, we i.i.d. initialize $\mathbf{w}_r \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, $a_r \sim \text{unif}\left[\{-1, 1\}\right]$ for $r \in [m]$, and we set the step size $\eta = O\left(\frac{\lambda_0}{n^2}\right)$ then with probability at least $1 - \delta$ over the random initialization we have for $k = 0, 1, 2, \ldots$*

$$\|\mathbf{u}(k) - \mathbf{y}\|_2^2 \leq \left(1 - \frac{\eta\lambda_0}{2}\right)^k \|\mathbf{u}(0) - \mathbf{y}\|_2^2 .$$

Theorem 2.3 shows even though the objective function is non-smooth and non-convex, gradient descent with a constant step size still enjoys a linear convergence rate. Our assumptions on the least eigenvalue and the number of hidden nodes are exactly the same as the theorem for gradient flow.

2.3.1 Proof of Theorem 2.3

We prove Theorem 2.3 by induction. Our induction hypothesis is just the following convergence rate of the empirical loss.

Condition 2.1. *At the k-th iteration, we have $\|\mathbf{y} - \mathbf{u}(k)\|_2^2 \leq (1 - \frac{\eta\lambda_0}{2})^k \|\mathbf{y} - \mathbf{u}(0)\|_2^2$.*

A directly corollary of this condition is the following bound of deviation from the initialization. The proof is similar to that of Lemma 2.3 so we defer it to appendix.

Corollary 2.1. *If Condition 2.1 holds for $k' = 0, \ldots, k$, then we have for every $r \in [m]$*

$$\|\mathbf{w}_r(k+1) - \mathbf{w}_r(0)\|_2 \leq \frac{4\sqrt{n}\,\|\mathbf{y} - \mathbf{u}(0)\|_2}{\sqrt{m}\lambda_0} \triangleq R'. \tag{2.7}$$

Now we show Condition 2.1 holds for every $k = 0, 1, \ldots$. For the base case $k = 0$, by definition Condition 2.1 holds. Suppose for $k' = 0, \ldots, k$, Condition 2.1 holds and we want to show Condition 2.1 holds for $k' = k + 1$.

Our strategy is similar to the proof of Theorem 2.2. We define the event

$$A_{ir} = \left\{\exists \mathbf{w} : \|\mathbf{w} - \mathbf{w}_r(0)\| \leq R, \mathbb{I}\left\{\mathbf{x}_i^\top \mathbf{w}_r(0) \geq 0\right\} \neq \mathbb{I}\left\{\mathbf{x}_i^\top \mathbf{w} \geq 0\right\}\right\}.$$

where $R = \frac{c\lambda_0}{n^2}$ for some small positive constant c. Different from gradient flow, for gradient descent we need a more refined analysis. We let $S_i = \{r \in [m] : \mathbb{I}\{A_{ir}\} = 0\}$ and $S_i^\perp = [m] \backslash S_i$. The following lemma bounds the sum of sizes of $S_i^\perp$. The proof is similar to the analysis used in Lemma 2.2. See Section 2.6 for the whole proof.

Lemma 2.5. *With probability at least $1 - \delta$ over the initialization, we have $\sum_{i=1}^n \left|S_i^\perp\right| \leq \frac{Cmn R}{\delta}$ for some positive constant $C > 0$.*

Next, we calculate the difference of predictions between two consecutive iterations, analogue to $\frac{du_i(t)}{dt}$ term in Section 2.2.

$$\begin{aligned} u_i(k+1) - u_i(k) =& \frac{1}{\sqrt{m}} \sum_{r=1}^m a_r \left(\sigma\left(\mathbf{w}_r(k+1)^\top \mathbf{x}_i\right) - \sigma\left(\mathbf{w}_r(k)^\top \mathbf{x}_i\right)\right) \\ =& \frac{1}{\sqrt{m}} \sum_{r=1}^m a_r \left(\sigma\left(\left(\mathbf{w}_r(k) - \eta \frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)}\right)^\top \mathbf{x}_i\right) - \sigma\left(\mathbf{w}_r(k)^\top \mathbf{x}_i\right)\right). \end{aligned}$$

Here we divide the right hand side into two parts. I_1^i accounts for terms that the pattern does not change and I_2^i accounts for terms that pattern may change.

$$\begin{aligned} I_1^i \triangleq& \frac{1}{\sqrt{m}} \sum_{r \in S_i} a_r \left(\sigma\left(\left(\mathbf{w}_r(k) - \eta \frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)}\right)^\top \mathbf{x}_i\right) - \sigma\left(\mathbf{w}_r(k)^\top \mathbf{x}_i\right)\right) \\ I_2^i \triangleq& \frac{1}{\sqrt{m}} \sum_{r \in S_i^\perp} a_r \left(\sigma\left(\left(\mathbf{w}_r(k) - \eta \frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)}\right)^\top \mathbf{x}_i\right) - \sigma\left(\mathbf{w}_r(k)^\top \mathbf{x}_i\right)\right) \end{aligned}$$

We view I_2^i as a perturbation and bound its magnitude. Because ReLU is a 1-Lipschitz function and $|a_r| = 1$, we have

$$\left|I_2^i\right| \leq \frac{\eta}{\sqrt{m}} \sum_{r \in S_i^\perp} \left| \left(\frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)}\right)^\top \mathbf{x}_i \right| \leq \frac{\eta \left|S_i^\perp\right|}{\sqrt{m}} \max_{r \in [m]} \left\| \frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)} \right\|_2 \leq \frac{\eta \left|S_i\right|^\perp \sqrt{n}\,\|\mathbf{u}(k) - \mathbf{y}\|_2}{m}.$$

To analyze I_1^i, by Corollary 2.1, we know $\|\mathbf{w}_r(k) - \mathbf{w}_r(0)\| \leq R'$ and $\|\mathbf{w}_r(k) - \mathbf{w}_r(0)\| \leq R'$ for all $r \in [m]$. Furthermore, because $R' < R$, we know $\mathbb{I}\left\{\mathbf{w}_r(k+1)^\top \mathbf{x}_i \geq 0\right\} = \mathbb{I}\left\{\mathbf{w}_r(k)^\top \mathbf{x}_i \geq 0\right\}$ for $r \in S_i$. Thus we can find a more convenient expression of I_1^i for analysis

$$\begin{aligned} I_1^i =& -\frac{\eta}{m}\sum_{j=1}^n \mathbf{x}_i^\top \mathbf{x}_j (u_j - y_j) \sum_{r\in S_i} \mathbb{I}\left\{\mathbf{w}_r(k)^\top \mathbf{x}_i \geq 0, \mathbf{w}_r(k)^\top \mathbf{x}_j \geq 0\right\} \\ =& -\eta \sum_{j=1}^n (u_j - y_j)(\mathbf{H}_{ij}(k) - \mathbf{H}_{ij}^\perp(k)) \end{aligned}$$

where $\mathbf{H}_{ij}(k) = \frac{1}{m}\sum_{r=1}^m \mathbf{x}_i^\top \mathbf{x}_j \mathbb{I}\left\{\mathbf{w}_r(k)^\top \mathbf{x}_i \geq 0, \mathbf{w}_r(k)^\top \mathbf{x}_j \geq 0\right\}$ is just the (i,j)-th entry of a discrete version of Gram matrix defined in Section 2.2 and

$$\mathbf{H}_{ij}^\perp(k) = \frac{1}{m}\sum_{r\in S_i^\perp} \mathbf{x}_i^\top \mathbf{x}_j \mathbb{I}\left\{\mathbf{w}_r(k)^\top \mathbf{x}_i \geq 0, \mathbf{w}_r(k)^\top \mathbf{x}_j \geq 0\right\}$$

is a perturbation matrix. Let $\mathbf{H}^\perp(k)$ be the $n \times n$ matrix with (i,j)-th entry being $\mathbf{H}_{ij}^\perp(k)$. Using Lemma 2.5, we obtain an upper bound of the operator norm

$$\left\|\mathbf{H}^\perp(k)\right\|_2 \leq \sum_{(i,j)=(1,1)}^{(n,n)} \left|\mathbf{H}_{ij}^\perp(k)\right| \leq \frac{n\sum_{i=1}^n \left|S_i^\perp\right|}{m} \leq \frac{Cn^2 mR}{\delta m} \leq \frac{Cn^2 R}{\delta}.$$

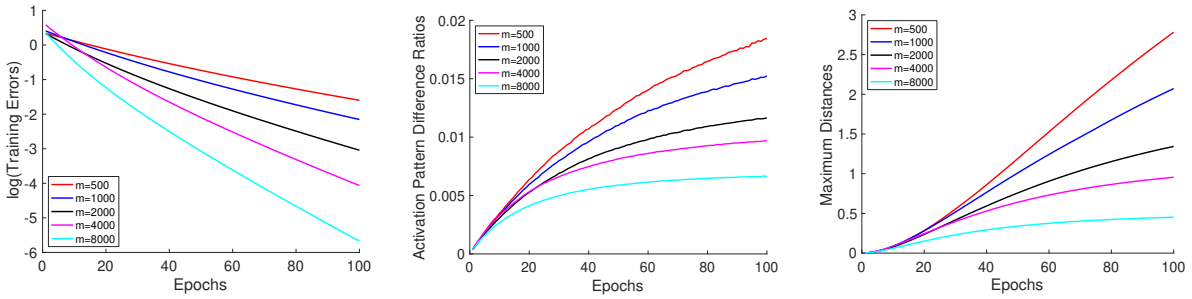
Similar to the classical analysis of gradient descent, we also need bound the quadratic term

$$\|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \leq \eta^2 \sum_{i=1}^n \frac{1}{m}\left(\sum_{r=1}^m \left\|\frac{\partial L(\mathbf{W}(k))}{\partial \mathbf{w}_r(k)}\right\|_2\right)^2 \leq \eta^2 n^2 \|\mathbf{y} - \mathbf{u}(k)\|_2^2 .$$

With these estimates at hand, we are ready to prove the induction hypothesis.

$$\begin{aligned} \|\mathbf{y} - \mathbf{u}(k+1)\|_2^2 =& \|\mathbf{y} - \mathbf{u}(k) - (\mathbf{u}(k+1) - \mathbf{u}(k))\|_2^2 \\ =& \|\mathbf{y} - \mathbf{u}(k)\|_2^2 - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top (\mathbf{u}(k+1) - \mathbf{u}(k)) + \|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \\ =& \|\mathbf{y} - \mathbf{u}(k)\|_2^2 - 2\eta\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{H}(k)\left(\mathbf{y} - \mathbf{u}(k)\right) \\ &+ 2\eta\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{H}(k)^\perp \left(\mathbf{y} - \mathbf{u}(k)\right) - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{I}_2 \\ &+ \|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \\ \leq& (1 - \eta\lambda_0 + \frac{2C\eta n^2 R}{\delta} + \frac{2C\eta n^{3/2} R}{\delta} + \eta^2 n^2)\|\mathbf{y} - \mathbf{u}(k)\|_2^2 \\ \leq& (1 - \frac{\eta\lambda_0}{2})\|\mathbf{y} - \mathbf{u}(k)\|_2^2 . \end{aligned}$$

The third equality we used the decomposition of $\mathbf{u}(k+1) - \mathbf{u}(k)$. The first inequality we used the Lemma 2.2, the bound on the step size, the bound on $\mathbf{I}_2$, the bound on $\left\|\mathbf{H}(k)^\perp\right\|_2$ and the bound on $\|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2$. The last inequality we used the bound of the step size and the bound of R. Therefore Condition 2.1 holds for $k' = k+1$. Now by induction, we prove Theorem 2.3.



(a) Convergence rates.

(b) Percentiles of pattern changes.

(c) Maximum distances from initialization.

Figure 2.1: Results on synthetic data.

2.4 Experiments

In this section, we use synthetic data to corroborate our theoretical findings. We use the initialization and training procedure described in Section 2.1. For all experiments, we run 100 epochs of gradient descent and use a fixed step size. We uniformly generate $n = 1000$ data points from a $d = 1000$ dimensional unit sphere and generate labels from a one-dimensional standard Gaussian distribution.

We test three metrics with different widths (m). First, we test how the amount of over-parameterization affects the convergence rates. Second, we test the relation between the amount of over-parameterization and the number of pattern changes. Formally, at a given iteration k, we check $\frac{\sum_{i=1}^{m}\sum_{r=1}^{m}\mathbb{I}\{\text{sign}(\mathbf{w}_r(0)^\top \mathbf{x}_i)\neq\text{sign}(\mathbf{w}_r(k)^\top \mathbf{x}_i)\}}{mn}$ (there are mn patterns). This aims to verify Lemma 2.2. Last, we test the relation between the amount of over-parameterization and the maximum of the distances between weight vectors and their initializations. Formally, at a given iteration k, we check $\max_{r\in[m]} \|\mathbf{w}_r(k) - \mathbf{w}_r(0)\|_2$. This aims to verify Lemma 2.3 and Corollary 2.1.

Figure 2.1a shows as m becomes larger, we have better convergence rate. We believe the reason is as m becomes larger, $\mathbf{H}(t)$ matrix becomes more stable, and thus has larger least eigenvalue. Figure 2.1b and Figure 2.1c show as m becomes larger, the percentiles of pattern changes and the maximum distance from the initialization become smaller. These empirical findings are consistent with our theoretical results.

2.5 Conclusion

In this chapter we show with over-parameterization, gradient descent provably converges to the global minimum of the empirical loss at a linear convergence rate. The key proof idea is to show the over-parameterization makes Gram matrix remain positive definite for all iterations, which in turn guarantees the linear convergence. In the next chapter, we will generalize this idea to analyze the convergence of gradient descent for deep neural networks.

Appendix: Omitted Proofs

2.6 Proofs for Section 2.2

Proof of Theorem 2.1. The proof of this lemma just relies on standard real and functional analysis. Let $\mathcal{H}$ be the Hilbert space of integrable d-dimensional vector fields on $\mathbb{R}^d$: $f \in \mathcal{H}$ if $\mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},\mathbf{I})}\left[|f(\mathbf{w})|^2\right] < \infty$. The inner product of this space is then

$$\langle f, g\rangle_{\mathcal{H}} = \mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},\mathbf{I})}\left[f(\mathbf{w})^\top g(\mathbf{w})\right].$$

ReLU activation induces an infinite-dimensional feature map ϕ which is defined as for any $\mathbf{x} \in \mathbb{R}^d$, $(\phi(\mathbf{x}))(\mathbf{w}) = \mathbf{x}\mathbb{I}\left\{\mathbf{w}^\top\mathbf{x} \geq 0\right\}$ where $\mathbf{w}$ can be viewed as the index. Now to prove $\mathbf{H}^\infty$ is strictly positive definite, it is equivalent to show $\phi(\mathbf{x}_1), \ldots, \phi(\mathbf{x}_n) \in \mathcal{H}$ are linearly independent. Suppose that there are $\alpha_1, \cdots, \alpha_n \in \mathbb{R}$ such that

$$\alpha_1\phi(\mathbf{x}_i) + \cdots + \alpha_n\phi(\mathbf{x}_n) = 0 \text{ in } \mathcal{H}.$$

This means that

$$\alpha_1\phi(\mathbf{x}_1)(\mathbf{w}) + \cdots + \alpha_n\phi(\mathbf{x}_n)(\mathbf{w}) = 0 \text{ a.e.}$$

Now we prove $\alpha_i = 0$ for all i.

We define $D_i = \left\{\mathbf{w} \in \mathbb{R}^d : \mathbf{w}^\top\mathbf{x}_i = 0\right\}$. This is set of discontinuities of $\phi(\mathbf{x}_i)$. The following lemma characterizes the basic property of these discontinuity sets.

Lemma 2.6. *If for any $i \neq j$, $\mathbf{x}_i \nparallel \mathbf{x}_j$, then for any $i \in [m]$, $D_i \not\subset \bigcup_{j\neq i} D_j$.*

Now for a fixed $i \in [n]$, since $D_i \not\subset \bigcup_{j\neq i} D_j$, we can choose $\mathbf{z} \in D_i \setminus \bigcup_{j\neq i} D_j$. Note $D_j, j \neq i$ are closed sets. We can pick $r_0 > 0$ small enough such that $B(\mathbf{z}, r) \cap D_j = \emptyset, \forall j \neq i, r \leq r_0$. Let $B(\mathbf{z}, r) = B_r^+ \sqcup B_r^-$ where

$$B_r^+ = B(\mathbf{z}, r) \cap \{\mathbf{w} \in \mathbb{R}^d : \mathbf{w}^\top\mathbf{x}_i > 0\}.$$

For $j \neq i$, $\phi(\mathbf{x}_j)(\mathbf{w})$ is continuous in a neighborhood of $\mathbf{z}$, then for any $\epsilon > 0$ there is a small enough $r > 0$ such that

$$\forall \mathbf{w} \in B(\mathbf{z}, r), |\phi(\mathbf{x}_j)(\mathbf{w}) - \phi(\mathbf{x}_j)(\mathbf{z})| < \epsilon.$$

Let μ be the Lebesgue measure on $\mathbb{R}^d$. We have

$$\left|\frac{1}{\mu(B_r^+)}\int_{B_r^+}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w} - \phi(\mathbf{x}_j)(\mathbf{z})\right| \leq \frac{1}{\mu(B_r^+)}\int_{B_r^+}|\phi(\mathbf{x}_j)(\mathbf{w}) - \phi(\mathbf{x}_j)(\mathbf{z})|\,d\mathbf{w} < \epsilon$$

and similarly

$$\left|\frac{1}{\mu(B_r^-)}\int_{B_r^-}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w} - \phi(\mathbf{x}_j)(\mathbf{z})\right| \leq \frac{1}{\mu(B_r^-)}\int_{B_r^-}|\phi(\mathbf{x}_j)(\mathbf{w}) - \phi(\mathbf{x}_j)(\mathbf{z})|\,d\mathbf{w} < \epsilon.$$

Thus, we have

$$\lim_{r\to 0+}\frac{1}{\mu(B_r^+)}\int_{B_r^+}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}=\lim_{r\to 0+}\frac{1}{\mu(B_r^-)}\int_{B_r^-}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}=\phi(\mathbf{x}_j)(\mathbf{z}).$$

Therefore, as $r\to 0+$, by continuity, we have

$$\forall j\neq i,\frac{1}{\mu(B_r^+)}\int_{B_r^+}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}-\frac{1}{\mu(B_r^-)}\int_{B_r^-}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}\to 0 \tag{2.8}$$

Next recall that $(\phi(\mathbf{x}))(\mathbf{w})=\mathbf{x}\mathbb{I}\left\{\mathbf{x}^\top\mathbf{w}>0\right\}$, so for $\mathbf{w}\in B_r^+$ and $\mathbf{x}_i$, $(\phi(\mathbf{x}_i))(\mathbf{w})=\mathbf{x}_i$. Then, we have

$$\lim_{r\to 0+}\frac{1}{\mu(B_r^+)}\int_{B_r^+}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}=\lim_{r\to 0+}\frac{1}{\mu(B_r^+)}\int_{B_r^+}\mathbf{x}_i d\mathbf{w}=\mathbf{x}_i. \tag{2.9}$$

For $\mathbf{w}\in B_r^-$ and $\mathbf{x}_i$, we know $(\phi(\mathbf{x}_i))(\mathbf{w})=0$. Then we have

$$\lim_{r\to 0+}\frac{1}{\mu(B_r^-)}\int_{B_r^-}\phi(\mathbf{x}_i)(\mathbf{w})d\mathbf{w}=\lim_{r\to 0+}\frac{1}{\mu(B_r^-)}\int_{B_r^-}0 d\mathbf{w}=0 \tag{2.10}$$

Now recall $\sum_i\alpha_i\phi(\mathbf{x}_i)\equiv 0$. Using Equation (2.8), (2.9) and (2.10), we have

$$\begin{aligned}
0&=\lim_{r\to 0+}\frac{1}{\mu(B_r^+)}\int_{B_r^+}\sum_j\alpha_j\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}-\lim_{r\to 0+}\frac{1}{\mu(B_r^-)}\int_{B_r^-}\sum_j\alpha_j\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}\\
&=\sum_j\alpha_j\left(\lim_{r\to 0+}\frac{1}{\mu(B_r^+)}\int_{B_r^+}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}-\lim_{r\to 0+}\frac{1}{\mu(B_r^-)}\int_{B_r^-}\phi(\mathbf{x}_j)(\mathbf{w})d\mathbf{w}\right)\\
&=\sum_j\alpha_j\left(\delta_{ij}\mathbf{x}_i\right)\\
&=\alpha_i\mathbf{x}_i
\end{aligned}$$

Since $\mathbf{x}_i\neq 0$, we must have $\alpha_i=0$. We complete the proof. □

Proof of Lemma 2.6. Let μ be the canonical Lebesgue measure on D_i. We have $\sum_{j\neq i}\mu(D_i\cap D_j)=0$ because $D_i\cap D_j$ is a hyperplane in D_i. Now we bound

$$\mu(D_i\cap\bigcup_{j\neq i}D_j)\leq\sum_{j\neq i}\mu(D_i\cap D_j)=0.$$

This implies our desired result. □

Proof of Lemma 2.1. For every fixed (i,j) pair, $\mathbf{H}_{ij}(0)$ is an average of independent random variables. Therefore, by Hoeffding inequality, we have with probability $1-\delta'$,

$$\left|\mathbf{H}_{ij}(0)-\mathbf{H}_{ij}^\infty\right|\leq\frac{2\sqrt{\log(1/\delta')}}{\sqrt{m}}.$$

Setting $\delta' = n^2\delta$ and applying union bound over (i,j) pairs, we have for every (i,j) pair with probability at least $1-\delta$

$$\left|\mathbf{H}_{ij}(0) - \mathbf{H}_{ij}^{\infty}\right| \leq \frac{4\sqrt{\log(n/\delta)}}{\sqrt{m}}.$$

Thus we have

$$\|\mathbf{H}(0) - \mathbf{H}^{\infty}\|_2^2 \leq \|\mathbf{H}(0) - \mathbf{H}^{\infty}\|_F^2 \leq \sum_{i,j}\left|\mathbf{H}_{ij}(0) - \mathbf{H}_{ij}^{\infty}\right|^2 \leq \frac{16n^2\log(n/\delta)}{m}.$$

Thus if $m = \Omega\left(\frac{n^2\log(n/\delta)}{\lambda_0^2}\right)$ we have the desired result. □

Proof of Lemma 2.4. Suppose the conclusion does not hold at time t. If there exists $r \in [m]$, $\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\| \geq R'$ or $\|\mathbf{y} - \mathbf{u}(t)\|_2^2 > \exp(-\lambda_0 t)\|\mathbf{y} - \mathbf{u}(0)\|_2^2$, then by Lemma 2.3 we know there exists $s \leq t$ such that $\lambda_{\min}(\mathbf{H}(s)) < \frac{1}{2}\lambda_0$. By Lemma 2.2 we know there exists

$$t_0 = \inf\left\{t > 0 : \max_{r\in[m]}\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2^2 \geq R\right\}.$$

Thus at t_0, there exists $r \in [m]$, $\|\mathbf{w}_r(t_0) - \mathbf{w}_r(0)\|_2^2 = R$. Now by Lemma 2.2, we know $\mathbf{H}(t_0) \geq \frac{1}{2}\lambda_0$ for $t' \leq t_0$. However, by Lemma 2.3, we know $\|\mathbf{w}_r(t_0) - \mathbf{w}_r(0)\|_2 < R' < R$. Contradiction.

For the other case, at time t, $\lambda_{\min}(\mathbf{H}(t)) < \frac{1}{2}\lambda_0$ we know there exists

$$t_0 = \inf\left\{t \geq 0 : \max_{r\in[m]}\|\mathbf{w}_r(t) - \mathbf{w}_r(0)\|_2^2 \geq R\right\}.$$

The rest of the proof is the same as the previous case. □

2.7 Proofs for Section 2.3

Proof of Corollary 2.1. We use the norm of gradient to bound this distance.

$$\begin{aligned}
\|\mathbf{w}_r(k+1) - \mathbf{w}_r(0)\|_2 \leq& \eta\sum_{k'=0}^{k}\left\|\frac{\partial L(\mathbf{W}(k'))}{\partial \mathbf{w}_r(k')}\right\|_2 \\
\leq& \eta\sum_{k'=0}^{k}\frac{\sqrt{n}\,\|\mathbf{y} - \mathbf{u}(k')\|_2}{\sqrt{m}} \\
\leq& \eta\sum_{k'=0}^{k}\frac{\sqrt{n}(1-\frac{\eta\lambda}{2})^{k'/2}}{\sqrt{m}}\|\mathbf{y} - \mathbf{u}(k')\|_2 \\
\leq& \eta\sum_{k'=0}^{\infty}\frac{\sqrt{n}(1-\frac{\eta\lambda_0}{2})^{k'/2}}{\sqrt{m}}\|\mathbf{y} - \mathbf{u}(k')\|_2
\end{aligned}$$

$$= \frac{4\sqrt{n}\,\|\mathbf{y} - \mathbf{u}(0)\|_2}{\sqrt{m}\lambda_0}.$$

□

Proof of Lemma 2.5. For a fixed $i \in [n]$ and $r \in [m]$, by anti-concentration inequality, we know $\mathbf{P}(A_{ir}) \leq \frac{2R}{\sqrt{2\pi}}$. Thus we can bound the size of $S_i^\perp$ in expectation.

$$\mathbb{E}\left[\left|S_i^\perp\right|\right] = \sum_{r=1}^{m} \mathbf{P}(A_{ir}) \leq \frac{2mR}{\sqrt{2\pi}}. \tag{2.11}$$

Summing over $i = 1, \ldots, n$, we have

$$\mathbb{E}\left[\sum_{i=1}^{n} \left|S_i^\perp\right|\right] \leq \frac{2mnR}{\sqrt{2\pi}}.$$

Thus by Markov's inequality, we have with probability at least $1 - \delta$

$$\sum_{i=1}^{n} \left|S_i^\perp\right| \leq \frac{CmnR}{\delta}. \tag{2.12}$$

for some large positive constant $C > 0$. □

Chapter 3

Gradient Descent Provably Optimizes Over-parameterized Deep Neural Networks with Smooth Activation

3.1 Introduction

In the previous chapter we showed for a two-layer over-parameterized ReLU-activated neural network, randomly initialized gradient descent can find a global minimum. A natural question is whether we can prove the same statement for deep neural networks. In this chapter, we give positive answer to this question. We consider the setting where there are n data points, and the neural network has H layers with width m. We focus on the least-squares loss and assume the activation function is Lipschitz and smooth. This assumption holds for many activation functions including the soft-plus and sigmoid. We summarize the results below:

- We first consider a fully-connected feedforward network. We show if $m = \Omega\left(\mathrm{poly}(n)2^{O(H)}\right)$[1], then randomly initialized gradient descent converges to zero training loss at a linear rate.
- Next, we consider the ResNet architecture [44]. We show as long as $m = \Omega\left(\mathrm{poly}(n, H)\right)$, then randomly initialized gradient descent converges to zero training loss at a linear rate. Comparing with the first result, the dependence on the number of layers improves exponentially for ResNet.
- Lastly, we apply the same technique to analyze convolutional ResNet. We show if $m = \mathrm{poly}(n, p, H)$ where p is the number of patches, then randomly initialized gradient descent achieves zero training loss.

Our proof follows the same high level idea of the previous chapter. However, in analyzing deep neural networks, we need to exploit more structural properties of deep neural networks and develop new techniques for analyzing both the initialization and gradient descent dynamics. In Section 3.3 we give an overview of our proof technique.

[1]The precise polynomials and data-dependent parameters are stated in Section 3.4, 3.5, 3.6.

3.2 Preliminaries

3.2.1 Activation Function

In this chapter we impose some technical conditions on the activation function. The guiding example is softplus: $\sigma(z) = \log(1 + \exp(z))$.

Condition 3.1 (Lipschitz and Smooth)**.** *There exists a constant $c > 0$ such that $|\sigma(0)| \leq c$ and for any $z, z' \in \mathbb{R}$,*

$$|\sigma(z) - \sigma(z')| \leq c|z - z'| \text{ and } |\sigma'(z) - \sigma'(z)| \leq c|z - z'|.$$

These two conditions will be used to show the stability of the training process. Note for softplus both Lipschitz constant and smoothness constant are 1. In this chapter, we view all activation function related parameters as constants.

Condition 3.2. *$\sigma(\cdot)$ is analytic and is not a polynomial function.*

This assumption is used to guarantee the positive-definiteness of certain Gram matrices which we will define later. Softplus function satisfies this assumption by definition.

3.2.2 Problem Setup

In this chapter, we focus on the empirical risk minimization problem with the quadratic loss function

$$\min_{\theta} L(\theta) = \frac{1}{2}\sum_{i=1}^{n}(f(\theta, \mathbf{x}_i) - y_i)^2 \tag{3.1}$$

where $\{\mathbf{x}_i\}_{i=1}^n$ are the training inputs, $\{y_i\}_{i=1}^n$ are the labels, θ is the parameter we optimize over and f is the prediction function, which in our case is a neural network. We consider the following architectures.

- **Multilayer fully-connected neural networks:** Let $\mathbf{x} \in \mathbb{R}^d$ be the input, $\mathbf{W}^{(1)} \in \mathbb{R}^{m\times d}$ is the first weight matrix, $\mathbf{W}^{(h)} \in \mathbb{R}^{m\times m}$ is the weight at the h-th layer for $2 \leq h \leq H$, $\mathbf{a} \in \mathbb{R}^m$ is the output layer and $\sigma(\cdot)$ is the activation function.[2] We define the prediction function recursively (for simplicity we let $\mathbf{x}^{(0)} = \mathbf{x}$).

$$\begin{aligned}\mathbf{x}^{(h)} &= \sqrt{\frac{c_\sigma}{m}}\sigma\left(\mathbf{W}^{(h)}\mathbf{x}^{(h-1)}\right), 1 \leq h \leq H \\ f(\mathbf{x}, \theta) &= \mathbf{a}^\top \mathbf{x}^{(H)}.\end{aligned} \tag{3.2}$$

  where $c_\sigma = \left(\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\sigma(x)^2\right]\right)^{-1}$ is a scaling factor to normalize the input in the initialization phase.

[2]We assume intermediate layers are square matrices for simplicity. It is not difficult to generalize our analysis to rectangular weight matrices.

- **ResNet**[3]: We use the same notations as the multilayer fully connected neural networks. We define the prediction recursively.

$$\begin{aligned}\mathbf{x}^{(1)} &= \sqrt{\frac{c_\sigma}{m}}\sigma\left(\mathbf{W}^{(1)}\mathbf{x}\right),\\ \mathbf{x}^{(h)} &= \mathbf{x}^{(h-1)} + \frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}\mathbf{x}^{(h-1)}\right) 2 \le h \le H,\\ f_{res}(\mathbf{x},\theta) &= \mathbf{a}^\top \mathbf{x}^{(H)}\end{aligned} \tag{3.3}$$

where $0 < c_{res} < 1$ is a small constant. Note here we use a $\frac{c_{res}}{H\sqrt{m}}$ scaling. This scaling plays an important role in guaranteeing the width per layer only needs to scale polynomially with H. In practice, the small scaling is enforced by a small initialization of the residual connection [41, 82], which obtains state-of-the-art performance for deep residual networks. We choose to use an explicit scaling, instead of altering the initialization scheme for notational convenience.

- **Convolutional ResNet**: Lastly, we consider the convolutional ResNet architecture. Again we define the prediction function in a recursive way. Let $\mathbf{x}^{(0)} \in \mathbb{R}^{d_0 \times p}$ be the input, where d_0 is the number of input channels and p is the number of pixels. For $h \in [H]$, we let the number of channels be $d_h = m$ and number of pixels be p. Given $\mathbf{x}^{(h-1)} \in \mathbb{R}^{d_{h-1}\times p}$ for $h \in [H]$, we first use an operator $\phi_h(\cdot)$ to divide $\mathbf{x}^{(h-1)}$ into p patches. Each patch has size qd_{h-1} and this implies a map $\phi_h(\mathbf{x}^{(h-1)}) \in \mathbb{R}^{qd_{h-1}\times p}$. For example, when the stride is 1 and $q = 3$

$$\phi_h(\mathbf{x}^{(h-1)}) = \begin{pmatrix} \left(\mathbf{x}^{(h-1)}_{1,0:2}\right)^\top, & \dots & , \left(\mathbf{x}^{(h-1)}_{1,p-1:p+1}\right)^\top \\ \dots, & \dots, & \dots \\ \left(\mathbf{x}^{(h-1)}_{d_{h-1},0:2}\right)^\top, & \dots, & \left(\mathbf{x}^{(h-1)}_{d_{h-1},p-1:p+1}\right)^\top \end{pmatrix}$$

where we let $\mathbf{x}^{(h-1)}_{:,0} = \mathbf{x}^{(h-1)}_{:,p+1} = \mathbf{0}$, i.e., zero-padding. Note this operator has the property

$$\left\|\mathbf{x}^{(h-1)}\right\|_F \le \left\|\phi_h(\mathbf{x}^{(h-1)})\right\|_F \le \sqrt{q}\left\|\mathbf{x}^{(h-1)}\right\|_F.$$

because each element from $\mathbf{x}^{(h-1)}$ at least appears once and at most appears q times. In practice, q is often small like 3×3, so throughout the chapter we view q as a constant in our theoretical analysis. To proceed, let $\mathbf{W}^{(h)} \in \mathbb{R}^{d_h \times qd_{h-1}}$, we have

$$\begin{aligned}\mathbf{x}^{(1)} =& \sqrt{\frac{c_\sigma}{m}}\sigma\left(\mathbf{W}^{(1)}\phi_1(\mathbf{x})\right) \in \mathbb{R}^{m\times p},\\ \mathbf{x}^{(h)} =& \mathbf{x}^{(h-1)} + \frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}\phi_h(\mathbf{x}^{(h-1))}\right) \in \mathbb{R}^{m\times p} \text{ for } 2 \le h \le H,\end{aligned}$$

[3]We will refer to this architecture as ResNet, although this differs by the standard ResNet architecture since the skip-connections at every layer, instead of every two layers. This architecture was previously studied in [41]. We study this architecture for the ease of presentation and analysis. It is not hard to generalize our analysis to architectures with skip-connections are every two or more layers.

where $0 < c_{res} < 1$ is a small constant. Finally, for $\mathbf{a} \in \mathbb{R}^{m \times p}$, the output is defined as

$$f_{cnn}(\mathbf{x}, \theta) = \langle \mathbf{a}, \mathbf{x}^{(H)} \rangle.$$

Note here we use the similar scaling $O(\frac{1}{H\sqrt{m}})$ as ResNet.

To learn the deep neural network, we consider the randomly initialized gradient descent algorithm to find the global minimizer of the empirical loss (3.1). Specifically, we use the following random initialization scheme. For every level $h \in [H]$, each entry is sampled from a standard Gaussian distribution, $\mathbf{W}_{ij}^{(h)} \sim \mathcal{N}(0, 1)$ and each entry of the output layer $\mathbf{a}$ is also sampled from $\mathcal{N}(0, 1)$. In this chapter, we train all layers by gradient descent, for $k = 1, 2, \ldots,$ and $h \in [H]$

$$\mathbf{W}^{(h)}(k) = \mathbf{W}^{(h)}(k-1) - \eta \frac{\partial L(\theta(k-1))}{\partial \mathbf{W}^{(h)}(k-1)},$$
$$\mathbf{a}(k) = \mathbf{a}(k-1) - \eta \frac{\partial L(\theta(k-1))}{\partial \mathbf{a}(k-1)}$$

where $\eta > 0$ is the step size.

3.3 Technique Overview

In this section, we describe our main idea of proving the global convergence of gradient descent. Our proof technique follows that of the previous chapter. Here the individual prediction at the k-th iteration is

$$u_i(k) = f(\theta(k), \mathbf{x}_i)$$

and we denote $\mathbf{u}(k) = (u_1(k), \ldots, u_n(k))^\top \in \mathbb{R}^n$. We consider the sequence $\{\mathbf{y} - \mathbf{u}(k)\}_{k=0}^{\infty}$, which admits the dynamics

$$\mathbf{y} - \mathbf{u}(k+1) = (\mathbf{I} - \eta \mathbf{G}(k)) (\mathbf{y} - \mathbf{u}(k))$$

where

$$\begin{aligned}
&\mathbf{G}_{ij}(k) \\
=&\left\langle \frac{\partial u_i(k)}{\partial \theta(k)}, \frac{\partial u_j(k)}{\partial \theta(k)} \right\rangle \\
=&\sum_{h=1}^{H} \left\langle \frac{\partial u_i(k)}{\partial \mathbf{W}^{(h)}(k)}, \frac{\partial u_j(k)}{\partial \mathbf{W}^{(h)}(k)} \right\rangle + \left\langle \frac{\partial u_i(k)}{\partial \mathbf{a}(k)}, \frac{\partial u_j(k)}{\partial \mathbf{a}(k)} \right\rangle \\
\triangleq&\sum_{h=1}^{H+1} \mathbf{G}_{ij}^{(h)}(k).
\end{aligned}$$

Here we define $\mathbf{G}^{(h)} \in \mathbb{R}^{n \times n}$ with $\mathbf{G}_{ij}^{(h)}(k) = \left\langle \frac{\partial u_i(k)}{\partial \mathbf{W}^{(h)}(k)}, \frac{\partial u_j(k)}{\partial \mathbf{W}^{(h)}(k)} \right\rangle$ for $h = 1, \ldots, H$ and $\mathbf{G}_{ij}^{(H+1)}(k) = \left\langle \frac{\partial u_i(k)}{\partial \mathbf{a}(k)}, \frac{\partial u_j(k)}{\partial \mathbf{a}(k)} \right\rangle$. Note for all $h \in [H+1]$, each entry of $\mathbf{G}^{(h)}(k)$ is an inner

product. Therefore, $\mathbf{G}^{(h)}(k)$ is a positive semi-definite (PSD) matrix for $h \in [H+1]$. Furthermore, if there exists one $h \in [H]$ that $\mathbf{G}^{(h)}(k)$ is strictly positive definite, then if one chooses the step size η to be sufficiently small, the loss decreases at the k-th iteration according the analysis of power method. In this chapter we focus on $\mathbf{G}^{(H)}(k)$, the gram matrix induced by the weights from H-th layer for simplicity at the cost of a minor degradation in convergence rate.[4]

We use the similar observation in the previous chapter that we show if the width is large enough for all layers, for all $k = 0, 1, \ldots$, $\mathbf{G}^{(H)}(k)$ is close to a fixed matrix $\mathbf{K}^{(H)} \in \mathbb{R}^{n \times n}$ which depends on the input data, neural network architecture and the activation but does not depend on neural network parameters θ. According to the analysis of the power method, once we establish this, as long as $\mathbf{K}^{(H)}$ is strictly positive definite, then the gradient descent enjoys a linear convergence rate. We will show for $\mathbf{K}^{(H)}$ is strictly positive definite as long as the training data is not degenerate (c.f. Proposition 3.1 and 3.2).

While following the similar high-level analysis framework as the previous chapter, analyzing the convergence of gradient descent for *deep* neural network is significantly more involved and requires new technical tools. To show $\mathbf{G}^{(H)}(k)$ is close to $\mathbf{K}^{(H)}$, we have two steps. First, we show in the initialization phase $\mathbf{G}^{(H)}(0)$ is close to $\mathbf{K}^{(H)}$. Second, we show during training $\mathbf{G}^{(H)}(k)$ is close to $\mathbf{G}^{(H)}(0)$ for $k = 1, 2, \ldots$. Below we give overviews of these two steps.

Analysis of Random Initialization Unlike the previous chapter in which we showed $\mathbf{H}(0)$ is close to $\mathbf{H}^{\infty}$ via a simple concentration inequality, showing $\mathbf{G}^{(H)}(0)$ is close to $\mathbf{K}^{(H)}$ requires more subtle calculations. First, as will be clear in the following sections, $\mathbf{K}^{(H)}$ is a recursively defined matrix. Therefore, we need to analyze how the perturbation (due to randomness of initialization and finite m) from lower layers propagates to the H-th layer. Second, this perturbation propagation involves non-linear operations due to the activation function. To quantitatively characterize this perturbation propagation dynamics, we use induction and leverage techniques from Malliavin calculus [54]. We derive a general framework that allows us to analyze the initialization behavior for the fully-connected neural network, ResNet, convolutional ResNet and other potential neural network architectures in a unified way.

One important finding in our analysis is that ResNet architecture makes the "perturbation propagation" more stable. The high level intuition is the following. For fully connected neural network, suppose we have some perturbation $\left\|\mathbf{G}^{(1)}(0) - \mathbf{K}^{(1)}\right\|_2 \leq \mathcal{E}_1$ in the first layer. This perturbation propagates to the H-th layer admits the form

$$\left\|\mathbf{G}^{(H)}(0) - \mathbf{K}^{(H)}\right\|_2 \triangleq \mathcal{E}_H \lesssim 2^{O(H)}\mathcal{E}_1. \tag{3.4}$$

Therefore, we need to have $\mathcal{E}_1 \leq \frac{1}{2^{O(H)}}$ and this makes m have exponential dependency on H.[5]

On the other hand, for ResNet the perturbation propagation admits the form

$$\mathcal{E}_H \lesssim \left(1 + O\left(\frac{1}{H}\right)\right)^H \epsilon_1 = O\left(\epsilon_1\right) \tag{3.5}$$

[4]Using the contribution of all the gram matrices to the minimum eigenvalue can potentially improve the convergence rate.

[5]We do not mean to imply that fully-connected networks necessarily depend exponentially on H, but simply to illustrate in our analysis why the exponential dependence arises. For specific activations such as ReLU and careful initialization schemes, this exponential dependence may be avoided [3].

Therefore we do not have the exponential explosion problem for ResNet. We refer readers to Section 3.12 for details.

Analysis of Perturbation of During Training The next step is to show $\mathbf{G}^{(H)}(k)$ is close to $\mathbf{G}^{(H)}(0)$ for $k = 0, 1, \ldots$. Note $\mathbf{G}^{(H)}$ depends on weight matrices from all layers, so to establish that $\mathbf{G}^{(H)}(k)$ is close to $\mathbf{G}^{(H)}(0)$, we need to show $\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)$ is small for all $h \in [H]$ and $\mathbf{a}(k) - \mathbf{a}(0)$ is small.

In the two-layer neural network setting (Chapter 2), we showed *every* weight vector of the first layer is close to its initialization, i.e., $\left\|\mathbf{W}^{(1)}(k) - \mathbf{W}^{(1)}(0)\right\|_{2,\infty}$ is small for $k = 0, 1, \ldots$. While establishing this condition for two-layer neural network is not hard, this condition may not hold for multi-layer neural networks. In this chapter, we show instead, the averaged Frobenius norm

$$\frac{1}{\sqrt{m}} \left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \tag{3.6}$$

is small for all $k = 0, 1, \ldots$.

Similar to the analysis in the initialization, showing Equation (3.6) is small is highly involved because again, we need to analyze how the perturbation propagates. We develop a unified proof strategy for the fully-connected neural network, ResNet and convolutional ResNet. Our analysis in this step again sheds light on the benefit of using ResNet architecture for training. The high-level intuition is similar to Equation (3.5). See Section 3.9, 3.10, and 3.11 for details.

3.4 Convergence Result of GD for Deep Fully-connected Neural Networks

In this section, as a warm up, we show gradient descent with a constant positive step size converges to the global minimum at a linear rate. As we discussed in Section 3.3, the convergence rate depends on least eigenvalue of the Gram matrix $\mathbf{K}^{(H)}$.

Definition 3.1. *The Gram matrix $\mathbf{K}^{(H)}$ is recursively defined as follows, for $(i, j) \in [n] \times [n]$, and $h = 1, \ldots, H-1$*

$$\begin{aligned}
\mathbf{K}_{ij}^{(0)} =& \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \\
\mathbf{A}_{ij}^{(h)} =& \begin{pmatrix} \mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)} \end{pmatrix}, \\
\mathbf{K}_{ij}^{(h)} =& c_\sigma \mathbb{E}_{(u,v)^\top \sim \mathcal{N}\mathbf{0}, \mathbf{A}_{ij}^{(h)})} \left[\sigma(u)\, \sigma(v)\right], \\
\mathbf{K}_{ij}^{(H)} =& c_\sigma \mathbf{K}_{ij}^{(H-1)} \mathbb{E}_{(u,v)^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{A}_{ij}^{(H-1)})} \left[\sigma'(u)\sigma'(v)\right].
\end{aligned} \tag{3.7}$$

The derivation of this Gram matrix is deferred to Section 3.12. The convergence rate and the amount of over-parameterization depends on the least eigenvalue of this Gram matrix. In Section 3.13.1 we show as long as the input training data is not degenerate, then $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ is

strictly positive. We remark that if $H = 1$, then $\mathbf{K}^{(H)}$ is the same the Gram matrix defined in the previous chapter.

Now we are ready to state our main convergence result of gradient descent for deep fully-connected neural networks.

Theorem 3.1 (Convergence Rate of Gradient Descent for Deep Fully-connected Neural Networks)**.** *Assume for all $i \in [n]$, $\|\mathbf{x}_i\|_2 = 1$, $|y_i| = O(1)$ and the number of hidden nodes per layer*

$$m = \Omega\left(2^{O(H)} \max\left\{\frac{n^4}{\lambda_{\min}^4\left(\mathbf{K}^{(H)}\right)}, \frac{n}{\delta}, \frac{n^2\log(\frac{Hn}{\delta})}{\lambda_{\min}^2\left(\mathbf{K}^{(H)}\right)}\right\}\right)$$

where $\mathbf{K}^{(H)}$ is defined in Equation (3.7)*. If we set the step size*

$$\eta = O\left(\frac{\lambda_{\min}\left(\mathbf{K}^{(H)}\right)}{n^2 2^{O(H)}}\right),$$

then with probability at least $1 - \delta$ over the random initialization the loss, for $k = 1, 2, \ldots$, the loss at each iteration satisfies

$$L(\theta(k)) \leq \left(1 - \frac{\eta\lambda_{\min}\left(\mathbf{K}^{(H)}\right)}{2}\right)^k L(\theta(0)).$$

This theorem states that if the width m is large enough and we set step size appropriately then gradient descent converges to the global minimum with zero loss at linear rate. The main assumption of the theorem is that we need a large enough width of each layer. The width m depends on n, H and $1/\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$. The dependency on n is only polynomial, which is the same as the previous chapter. Furthermore, m also polynomially depends on $1/\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$. However, the dependency on the number of layers H is exponential. As we discussed in Section 3.9.1, this exponential comes from the instability of the fully-connected architecture (c.f. Equation (3.4)). In the next section, we show with ResNet architecture, we can reduce the dependency on H from $2^{(H)}$ to $\mathrm{poly}(H)$.

Note the requirement of m has three terms. The first term is used to show the Gram matrix is stable during training. The second term is used to guarantee the output in each layer is approximately normalized at the initialization phase. The third term is used to show the perturbation of Gram matrix at the initialization phase is small. See Section 3.9 for proofs.

The convergence rate depends step size η and $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$, similar to the previous chapter Here we require $\eta = O\left(\frac{\lambda_{\min}\left(\mathbf{K}^{(H)}\right)}{n^2 2^{O(H)}}\right)$. When $H = 1$, this requirement is the same as the one used in the previous chapter. However, for deep fully-connected neural network, we require η to be exponentially small in terms of number of layers. The reason is similar to that we require m to be exponentially large. Again, this will be improved in the next section.

3.5 Convergence Result of GD for ResNet

In this section we consider the convergence of gradient descent for training a ResNet. We will focus on how much over-parameterization is needed to ensure the global convergence of gradient descent and compare it with fully-connected neural networks. Again we first define the key Gram matrix whose least eigenvalue will determine the convergence rate.

Definition 3.2. *The Gram matrix $\mathbf{K}^{(H)}$ is recursively defined as follows, for $(i,j) \in [n] \times [n]$ and $h = 2, \ldots, H-1$:*

$$
\begin{aligned}
\mathbf{K}_{ij}^{(0)} =& \langle \mathbf{x}_i, \mathbf{x}_j \rangle, \\
\mathbf{K}_{ij}^{(1)} =& \mathbb{E}_{(u,v)^\top \sim N\left(\mathbf{0}, \begin{pmatrix} \mathbf{K}_{ii}^{(0)} & \mathbf{K}_{ij}^{(0)} \\ \mathbf{K}_{ji}^{(0)} & \mathbf{K}_{jj}^{(0)} \end{pmatrix}\right)} c_\sigma \sigma(u) \sigma(v), \\
\mathbf{b}_i^{(1)} =& \sqrt{c_\sigma} \mathbb{E}_{u \sim \mathcal{N}(0, \mathbf{K}_{ii}^{(0)})} [\sigma(u)], \\
\mathbf{A}_{ij}^{(h)} =& \begin{pmatrix} \mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)} \end{pmatrix} \\
\mathbf{K}_{ij}^{(h)} =& \mathbf{K}_{ij}^{(h-1)} + \mathbb{E}_{(u,v)^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{A}_{ij}^{(h)})} \left[\frac{c_{res} \mathbf{b}_i^{(h-1)} \sigma(u)}{H} \frac{c_{res} \mathbf{b}_j^{(h-1)} \sigma(v)}{H} + \frac{c_{res}^2 \sigma(u) \sigma(v)}{H^2} \right], \\
\mathbf{b}_i^{(h)} =& \mathbf{b}_i^{(h-1)} + \frac{c_{res}}{H} \mathbb{E}_{u \sim \mathcal{N}(0, \mathbf{K}_{ii}^{(h-1)})} [\sigma(u)], \\
\mathbf{K}_{ij}^{(H)} =& \frac{c_{res}^2}{H^2} \mathbf{K}_{ij}^{(H-1)} \mathbb{E}_{(u,v)^\top \sim \mathcal{N}(\mathbf{0}, \mathbf{A}_{ij}^{(H-1)})} [\sigma'(u) \sigma'(v)].
\end{aligned} \tag{3.8}
$$

Comparing $\mathbf{K}^{(H)}$ of the ResNet and the one of the fully-connect neural network, the definition of $\mathbf{K}^{(H)}$ also depends on a series of $\{\mathbf{b}^{(h)}\}_{h=1}^{H-1}$. This dependency is comes from the skip connection block in the ResNet architecture. See Section 3.12. In Section 3.13.2, we show as long as the input training data is not degenerate, then $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ is strictly positive. Furthermore, $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ does not depend inversely exponentially in H.

Now we are ready to state our main theorem for ResNet.

Theorem 3.2 (Convergence Rate of Gradient Descent for ResNet)**.** *Assume for all $i \in [n]$, $\|\mathbf{x}_i\|_2 = 1$, $|y_i| = O(1)$ and the number of hidden nodes per layer*

$$
m = \Omega\left(\max\left\{ \frac{n^4}{\lambda_{\min}^4\left(\mathbf{K}^{(H)}\right) H^6}, \frac{n^2}{\lambda_{\min}^2(\mathbf{K}^{(H)}) H^2}, \frac{n}{\delta}, \frac{n^2 \log\left(\frac{Hn}{\delta}\right)}{\lambda_{\min}^2\left(\mathbf{K}^{(H)}\right)} \right\} \right).
$$

If we set the step size $\eta = O\left(\frac{\lambda_{\min}\left(\mathbf{K}^{(H)}\right) H^2}{n^2}\right)$, then with probability at least $1-\delta$ over the random initialization we have for $k = 1, 2, \ldots$

$$
L(\theta(k)) \leq \left(1 - \frac{\eta \lambda_{\min}\left(\mathbf{K}^{(H)}\right)}{2}\right)^k L(\theta(0)).
$$

In sharp contrast to Theorem 3.1, this theorem is fully polynomial in the sense that both the number of neurons and the convergence rate is polynomially in n and H. Note the amount of over-parameterization depends on $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ which is the smallest eigenvalue of the H-th layer's Gram matrix. The main reason that we do not have any exponential factor here is that the skip connection block makes the overall architecture more stable in both the initialization phase and the training phase.

Note the requirement on m has 4 terms. The first two terms are used to show the Gram matrix stable during training. The third term is used to guarantee the output in each layer is approximately normalized at the initialization phase. The fourth term is used to show bound the size of the perturbation of the Gram matrix at the initialization phase. See Section 3.10 for details.

3.6 Convergence Result of GD for Convolutional ResNet

In this section we generalize the convergence result of gradient descent for ResNet to convolutional ResNet. Again, we focus on how much over-parameterization is needed to ensure the global convergence of gradient descent. Similar to previous sections, we first define the $\mathbf{K}^{(H)}$ for this architecture.

Definition 3.3. *The Gram matrix $\mathbf{K}^{(H)}$ is recursively defined as follows, for $(i,j) \in [n] \times [n]$, $(l,r) \in [p] \times [p]$ and $h = 2, \ldots, H-1$,*

$$
\begin{aligned}
\mathbf{K}_{ij}^{(0)} =& \phi_1\left(\mathbf{x}_i\right)^\top \phi_1\left(\mathbf{x}_j\right) \in \mathbb{R}^{p\times p}, \\
\mathbf{K}_{ij}^{(1)} =& \mathbb{E}_{(\mathbf{u},\mathbf{v})\sim\mathcal{N}(\mathbf{0},\begin{pmatrix}\mathbf{K}_{ii}^{(0)} & \mathbf{K}_{ij}^{(0)} \\ \mathbf{K}_{ji}^{(0)} & \mathbf{K}_{jj}^{(0)}\end{pmatrix})} c_\sigma \sigma\left(\mathbf{u}\right)^\top \sigma\left(\mathbf{v}\right), \\
\mathbf{b}_i^{(1)} =& \sqrt{c_\sigma}\mathbb{E}_{\mathbf{u}\sim\mathcal{N}(\mathbf{0},\mathbf{K}_{ii}^{(0)})}\left[\sigma\left(\mathbf{u}\right)\right], \\
\mathbf{A}_{ij}^{(h)} =& \begin{pmatrix}\mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)}\end{pmatrix}, \\
\mathbf{H}_{ij}^{(h)} =& \mathbf{K}_{ij}^{(h-1)} + \mathbb{E}_{(\mathbf{u},\mathbf{v})\sim\mathcal{N}(\mathbf{0},\mathbf{A}_{ij}^{(h-1)})}\left[\frac{c_{res}\mathbf{b}_i^{(h-1)\top}\sigma\left(\mathbf{u}\right)}{H} + \frac{c_{res}\mathbf{b}_j^{(h-1)\top}\sigma\left(\mathbf{v}\right)}{H} + \frac{c_{res}^2\sigma\left(\mathbf{u}\right)^\top\sigma\left(\mathbf{v}\right)}{H^2}\right], \\
\mathbf{K}_{ij,lr}^{(h)} =& \mathrm{tr}\left(\mathbf{H}_{ij,D_\ell^{(h)}D_r^{(h)}}^{(h)}\right), \qquad (3.9) \\
\mathbf{b}_i^{(h)} =& \mathbf{b}_i^{(h-1)} + \frac{c_{res}}{H}\mathbb{E}_{\mathbf{u}\sim\mathcal{N}(\mathbf{0},\mathbf{K}_{ii}^{(h-1)})}\left[\sigma\left(\mathbf{u}\right)\right] \\
\mathbf{M}_{ij,lr}^{(H)} =& \mathbf{K}_{ij,lr}^{(H-1)}\mathbb{E}_{(\mathbf{u},\mathbf{v})\sim\mathcal{N}(\mathbf{0},\mathbf{A}_{ij}^{(H-1)})}\left[\sigma'(u_l)\sigma'(v_r)\right] \\
\mathbf{K}_{ij}^{(H)} =& \mathrm{tr}(\mathbf{M}_{ij}^{(H)})
\end{aligned}
$$

where $\mathbf{u}$ and $\mathbf{v}$ are both random row vectors and $D_l^{(h)} \triangleq \{s : \mathbf{x}_{:,s}^{(h-1)} \in$ the l^{th} patch$\}$.

Note here $\mathbf{K}_{ij}^{(h)}$ has dimension $p \times p$ for $h = 0, \ldots, H-1$ and $\mathbf{K}_{ij,lr}$ denotes the (l,r)-th entry. Now we state our main convergence theorem for the convolutional ResNet.

Theorem 3.3 (Convergence Rate of Gradient Descent for Convolutional ResNet)**.** *Assume for all* $i \in [n]$, $\|\mathbf{x}_i\|_F = 1$, $|y_i| = O(1)$ *and the number of hidden nodes per layer*

$$m = \Omega\left(\max\left\{\frac{n^4}{\lambda_0^4 H^6}, \frac{n^4}{\lambda_0^4 H^2}, \frac{n}{\delta}, \frac{n^2 \log\left(\frac{Hn}{\delta}\right)}{\lambda_0^2}\right\} \mathrm{poly}(p)\right). \tag{3.10}$$

If we set the step size $\eta = O\left(\frac{\lambda_0 H^2}{n^2 \mathrm{poly}(p)}\right)$*, then with probability at least* $1 - \delta$ *over the random initialization we have for* $k = 1, 2, \ldots$

$$L(\theta(k)) \leq \left(1 - \frac{\eta \lambda_{\min}\left(\mathbf{K}^{(H)}\right)}{2}\right)^k L(\theta(0)).$$

This theorem is similar to that of ResNet. The number of neurons required per layer is only polynomial in the depth and the number of data points and step size is only polynomially small. The only extra term is $\mathrm{poly}(p)$ in the requirement of m and η. The analysis is also similar to ResNet and we refer readers to Section 3.11 for details.

3.7 Conclusion and Future Work

In this chapter, we show that gradient descent on deep overparametrized networks can obtain zero training loss. Our proof builds on a careful analysis of the random initialization scheme and a perturbation analysis which shows that the Gram matrix is increasingly stable under over-parametrization. These techniques allow us to show that every step of gradient descent decreases the loss at a geometric rate. We believe the analysis techniques developed in this chapter may be applicable to other problems.

We list some directions for future research:

1. The current chapter focuses on the training loss, but does not address the test loss. It would be an important problem to show that gradient descent can also find solutions of low test loss. In particular, existing work only demonstrate that gradient descent works under the same situations as kernel methods and random feature methods [2, 5, 16, 50].
2. The width of the layers m is polynomial in all the parameters for the ResNet architecture, but still very large. Realistic networks have number of parameters, not width, a large constant multiple of n. We consider improving the analysis to cover commonly utilized networks an important open problem.
3. The current analysis is for gradient descent, instead of stochastic gradient descent. We believe the analysis can be extended to stochastic gradient, while maintaining the linear convergence rate.
4. The convergence rate can be potentially improved if the minimum eigenvalue takes into account the contribution of all Gram matrices, but this would considerably complicate the initialization and perturbation analysis.

Appendix: Omitted Proofs

In the proof we will use the geometric series function $g_\alpha(n) = \sum_{i=0}^{n-1} \alpha^i$ extensively. Some constants we will define below may be different for different network structures, such as c_x, $c_{w,0}$ and $c_{x,0}$. We will also use c to denote a small enough constant, which may be different in different lemmas. For simplicity, we use λ_0 to denote $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ in the proofs.

3.8 Proof Sketch

Note we can write the loss as

$$L(\theta(k)) = \frac{1}{2}\left\|\mathbf{y} - \mathbf{u}(k)\right\|_2^2.$$

Our proof is by induction. Our induction hypothesis is just the following convergence rate of empirical loss.

Condition 3.3. *At the k-th iteration, we have*

$$\left\|\mathbf{y} - \mathbf{u}(k)\right\|_2^2 \leq (1 - \frac{\eta\lambda_0}{2})^k \left\|\mathbf{y} - \mathbf{u}(0)\right\|_2^2.$$

Note this condition implies the conclusions we want to prove. To prove Condition 3.3, we consider one iteration on the loss function.

$$\begin{aligned}
&\left\|\mathbf{y} - \mathbf{u}(k+1)\right\|_2^2 \\
=&\left\|\mathbf{y} - \mathbf{u}(k) - (\mathbf{u}(k+1) - \mathbf{u}(k))\right\|_2^2 \\
=&\left\|\mathbf{y} - \mathbf{u}(k)\right\|_2^2 - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top (\mathbf{u}(k+1) - \mathbf{u}(k)) + \left\|\mathbf{u}(k+1) - \mathbf{u}(k)\right\|_2^2. \qquad (3.11)
\end{aligned}$$

This equation shows if $2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top (\mathbf{u}(k+1) - \mathbf{u}(k)) > \left\|\mathbf{u}(k+1) - \mathbf{u}(k)\right\|_2^2$, the loss decreases. Note both terms involves $\mathbf{u}(k+1) - \mathbf{u}(k)$, which we will carefully analyze. To simplify notations, we define

$$\begin{aligned}
&u_i'(\theta) \triangleq \frac{\partial u_i}{\partial \theta}, \quad u_i'^{(h)}(\theta) \triangleq \frac{\partial u_i}{\partial \mathbf{W}^{(h)}}, \quad u_i'^{(a)}(\theta) \triangleq \frac{\partial u_i}{\partial \mathbf{a}}, \qquad \text{and} \\
&L'(\theta) \triangleq \frac{\partial L(\theta)}{\partial \theta}, \quad L'^{(h)}(\mathbf{W}^{(h)}) \triangleq \frac{\partial L(\theta)}{\partial \mathbf{W}^{(h)}}, \quad L'^{(a)}(\theta) \triangleq \frac{\partial L}{\partial \mathbf{a}}.
\end{aligned}$$

We look one coordinate of $\mathbf{u}(k+1) - \mathbf{u}(k)$.

Using Taylor expansion, we have

$$\begin{aligned}
&u_i(k+1) - u_i(k) \\
=&u_i\left(\theta(k) - \eta L'(\theta(k))\right) - u_i\left(\theta(k)\right) \\
=&-\int_{s=0}^{\eta} \langle L'(\theta(k)), u_i'\left(\theta(k) - sL'(\theta(k))\right)\rangle ds \\
=&-\int_{s=0}^{\eta} \langle L'(\theta(k)), u_i'\left(\theta(k)\right)\rangle ds + \int_{s=0}^{\eta} \langle L'(\theta(k)), u_i'\left(\theta(k)\right) - u_i'\left(\theta(k) - sL'(\theta(k))\right)\rangle ds
\end{aligned}$$

$\triangleq I_1^i(k) + I_2^i(k)$.

Denote $\mathbf{I}_1(k) = (I_1^1(k), \ldots, I_1^n(k))^\top$ and $\mathbf{I}_2(k) = (I_2^1(k), \ldots, I_2^n(k))^\top$ and so $\mathbf{u}(k+1) - \mathbf{u}(k) = \mathbf{I}_1(k) + \mathbf{I}_2(k)$. We will show the $\mathbf{I}_1(k)$ term, which is proportional to η, drives the loss function to decrease and the $\mathbf{I}_2(k)$ term, which is a perturbation term but it is proportional to η^2 so it is small. We further unpack the $I_1^i(k)$ term,

$$\begin{aligned} I_1^i =& -\eta \langle L'(\theta(k)), u_i'(\theta(k)) \rangle \\ =& -\eta \sum_{j=1}^n (u_j - y_j) \langle u_j'(\theta(k)), u_i'(\theta(k)) \rangle \\ \triangleq& -\eta \sum_{j=1}^n (u_j - y_j) \sum_{h=1}^{H+1} \mathbf{G}_{ij}^{(h)}(k) \end{aligned}$$

According to Section 3.3, we will only look at $\mathbf{G}^{(H)}$ matrix which has the following form

$$\mathbf{G}_{i,j}^{(H)}(k) = (\mathbf{x}_i^{(H-1)}(k))^\top \mathbf{x}_j^{(H-1)}(k) \cdot \frac{c_\sigma}{m} \sum_{r=1}^m a_r^2 \sigma'((\theta_r^{(H)}(k))^\top \mathbf{x}_i^{(H-1)}(k)) \sigma'((\theta_r^{(H)}(k))^\top \mathbf{x}_j^{(H-1)}(k)).$$

Now we analyze $\mathbf{I}_1(k)$. We can write $\mathbf{I}_1$ in a more compact form with $\mathbf{G}(k)$.

$$\mathbf{I}_1(k) = -\eta \mathbf{G}(k) \left(\mathbf{u}(k) - \mathbf{y} \right).$$

Now observe that

$$\begin{aligned} (\mathbf{y} - \mathbf{u}(k))^\top \mathbf{I}_1(k) =& \eta \left(\mathbf{y} - \mathbf{u}(k) \right)^\top \mathbf{G}(k) (\mathbf{y} - \mathbf{u}(k)) \\ \geq& \lambda_{\min} \left(\mathbf{G}(k) \right) \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2 \\ \geq& \lambda_{\min} \left(\mathbf{G}^{(H)}(k) \right) \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2 \end{aligned}$$

Now recall the progress of loss function in Equation (3.11):

$$\begin{aligned} & \left\| \mathbf{y} - \mathbf{u}(k+1) \right\|_2^2 \\ =& \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2 - 2 \left(\mathbf{y} - \mathbf{u}(k) \right)^\top \mathbf{I}_1(k) - 2 \left(\mathbf{y} - \mathbf{u}(k) \right)^\top \mathbf{I}_2(k) + \left\| \mathbf{u}(k+1) - \mathbf{u}(k) \right\|_2^2 \\ \leq& \left(1 - \eta \lambda_{\min} \left(\mathbf{G}^{(H)}(k) \right) \right) \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2 - 2 \left(\mathbf{y} - \mathbf{u}(k) \right)^\top \mathbf{I}_2(k) + \left\| \mathbf{u}(k+1) - \mathbf{u}(k) \right\|_2^2. \end{aligned}$$

For the perturbation terms, through standard calculations, we can show both $-2 \left(\mathbf{y} - \mathbf{u}(k) \right)^\top \mathbf{I}_2(k)$ and $\left\| \mathbf{u}(k+1) - \mathbf{u}(k) \right\|_2$ are proportional to $\eta^2 \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2$ so if we set η sufficiently small, this term is smaller than $\eta \lambda_{\min} \left(\mathbf{G}^{(H)}(k) \right) \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2$ and thus the loss function decreases with a linear rate.

Therefore, to prove the induction hypothesis, it suffices to prove $\lambda_{\min} \left(\mathbf{G}^{(H)}(k) \right) \geq \frac{\lambda_0}{2}$ for $k' = 0, \ldots, k$, where λ_0 is independent of m. To analyze the least eigenvalue, we first look at the initialization. Using assumptions of the population Gram matrix and concentration inequalities, we can show at the beginning $\left\| \mathbf{G}^{(H)}(0) - \mathbf{K}^{(H)}(0) \right\|_{\text{op}} \leq \frac{1}{4} \lambda_0$, which implies

$$\lambda_{\min} \left(\mathbf{G}^{(H)}(0) \right) \geq \frac{3}{4} \lambda_0.$$

Now for the k-th iteration, by matrix perturbation analysis, we know it is sufficient to show $\left\|\mathbf{G}^{(H)}(k)-\mathbf{G}^{(H)}(0)\right\|_{\text{op}} \leq \frac{1}{4}\lambda_0$. To do this, we use a similar approach as in the previous chapter. We show as long as m is large enough, every weight matrix is close its initialization in a relative error sense. Ignoring all other parameters except m, $\left\|\mathbf{W}^{(h)}(k)-\mathbf{W}^{(h)}(0)\right\|_F \lesssim 1$, and thus the average per-neuron distance from initialization is $\frac{\left\|\mathbf{W}^{(h)}(k)-\mathbf{W}^{(h)}(0)\right\|_F}{\sqrt{m}} \lesssim \frac{1}{\sqrt{m}}$ which tends to zero as m increases. See Lemma 3.5 for precise statements with all the dependencies.

This fact in turn shows $\left\|\mathbf{G}^{(H)}(k)-\mathbf{G}^{(H)}(0)\right\|_{\text{op}}$ is small. The main difference from the previous chapter is that we are considering deep neural networks, and when translating the small deviation, $\left\|\mathbf{W}^{(h)}(k)-\mathbf{W}^{(h)}(0)\right\|_F$ to $\left\|\mathbf{G}^{(H)}(k)-\mathbf{G}^{(H)}(0)\right\|_{\text{op}}$, there is an amplification factor which depends on the neural network architecture.

For deep fully connected neural networks, we show this amplification factor is exponential in H. On the other hand, for ResNet and convolutional ResNet we show this amplification factor is only polynomial in H. We further show the width m required is proportional to this amplification factor.

3.9 Proofs for Section 3.4

We first derive the formula of the gradient for the multilayer fully connected neural network

$$\frac{\partial L(\theta)}{\partial \mathbf{W}^{(h)}} = \left(\frac{c_\sigma}{m}\right)^{\frac{H-h+1}{2}} \sum_{i=1}^{n} \left(f(\mathbf{x}_i,\theta)-y_i\right) \mathbf{x}_i^{(h-1)} \mathbf{a}^\top \left(\prod_{k=h+1}^{H} \mathbf{J}_i^{(k)} \mathbf{W}^{(k)}\right) \mathbf{J}_i^{(h)}$$

where

$$\mathbf{J}^{(h')} \triangleq \mathbf{diag}\left(\sigma'\left((\mathbf{w}_1^{(h')})^\top \mathbf{x}^{(h'-1)}\right), \ldots, \sigma'\left((\mathbf{w}_m^{(h')})^\top \mathbf{x}^{(h'-1)}\right)\right) \in \mathbb{R}^{m\times m}$$

are the derivative matrices induced by the activation function and

$$\mathbf{x}^{(h')} = \sqrt{\frac{c_\sigma}{m}}\sigma\left(\mathbf{W}^{(h')}\mathbf{x}^{(h'-1)}\right).$$

is the output of the h'-th layer.

Through standard calculation, we can get the expression of $\mathbf{G}_{i,j}^{(H)}$ of the following form

$$\mathbf{G}_{i,j}^{(H)} = (\mathbf{x}_i^{(H-1)})^\top \mathbf{x}_j^{(H-1)} \cdot \frac{c_\sigma}{m} \sum_{r=1}^{m} a_r^2 \sigma'((\mathbf{w}_r^{(H)})^\top \mathbf{x}_i^{(H-1)}) \sigma'((\mathbf{w}_r^{(H)})^\top \mathbf{x}_j^{(H-1)}). \tag{3.12}$$

We first present a lemma which shows with high probability the feature of each layer is approximately normalized.

Lemma 3.1 (Lemma on Initialization Norms)**.** *If $\sigma(\cdot)$ is $L-$Lipschitz and $m = \Omega\left(\frac{nHg_C(H)^2}{\delta}\right)$, where $C \triangleq c_\sigma L\left(2\left|\sigma(0)\right|\sqrt{\frac{2}{\pi}}+2L\right)$, then with probability at least $1-\delta$ over random initialization, for every $h \in [H]$ and $i \in [n]$, we have*

$$\frac{1}{c_{x,0}} \leq \left\|\mathbf{x}_i^{(h)}(0)\right\|_2 \leq c_{x,0}$$

where $c_{x,0} = 2$.

We follow the proof sketch described in Section 3.8. We first analyze the spectral property of $\mathbf{G}^{(H)}(0)$ at the initialization phase. The following lemma lower bounds its least eigenvalue. This lemma is a direct consequence of results in Section 3.12.

Lemma 3.2 (Least Eigenvalue at the Initialization)**.** *If* $m = \Omega\left(\frac{n^2 \log(Hn/\delta) 2^{O(H)}}{\lambda_0^2}\right)$*, we have*

$$\lambda_{\min}(\mathbf{G}^{(H)}(0)) \geq \frac{3}{4}\lambda_0.$$

Now we proceed to analyze the training process. We prove the following lemma which characterizes how the perturbation from weight matrices propagates to the input of each layer. This Lemma is used to prove the subsequent lemmas.

Lemma 3.3. *Suppose for every* $h \in [H]$, $\left\|\mathbf{W}^{(h)}(0)\right\|_{\text{op}} \leq c_{w,0}\sqrt{m}$, $\left\|\mathbf{x}^{(h)}(0)\right\|_2 \leq c_{x,0}$ *and* $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \leq \sqrt{m}R$ *for some constant* $c_{w,0}, c_{x,0} > 0$ *and* $R \leq c_{w,0}$*. If* $\sigma(\cdot)$ *is* $L-$*Lipschitz, we have*

$$\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \leq \sqrt{c_\sigma} L c_{x,0} g_{c_x}(h) R$$

where $c_x = 2\sqrt{c_\sigma} L c_{w,0}$*.*

Here the assumption of $\left\|\mathbf{W}^{(h)}(0)\right\|_2 \leq c_{w,0}\sqrt{m}$ can be shown using Lemma 3.27 and taking union bound over $h \in [H]$, where $c_{w,0}$ is a universal constant. Next, we show with high probability over random initialization, perturbation in weight matrices leads to small perturbation in the Gram matrix.

Lemma 3.4. *Suppose* $\sigma(\cdot)$ *is* $L-$*Lipschitz and* $\beta-$*smooth. Suppose for* $h \in [H]$, $\left\|\mathbf{W}^{(h)}(0)\right\|_2 \leq c_{w,0}\sqrt{m}$, $\|\mathbf{a}(0)\|_2 \leq a_{2,0}\sqrt{m}$, $\|\mathbf{a}(0)\|_4 \leq a_{4,0}m^{1/4}$, $\frac{1}{c_{x,0}} \leq \left\|\mathbf{x}^{(h)}(0)\right\|_2 \leq c_{x,0}$,
if $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F$ *and* $\|\mathbf{a}(k) - \mathbf{a}(0)\|_2 \leq \sqrt{m}R$ *where* $R \leq c g_{c_x}(H)^{-1}\lambda_0 n^{-1}$ *and* $R \leq c g_{c_x}(H)^{-1}$ *for some small constant* c *and* $c_x = 2\sqrt{c_\sigma} L c_{w,0}$*, we have*

$$\left\|\mathbf{G}^{(H)}(k) - \mathbf{G}^{(H)}(0)\right\|_{\text{op}} \leq \frac{\lambda_0}{4}.$$

Here the assumption of $\|\mathbf{a}(0)\|_2 \leq a_{2,0}\sqrt{m}$, $\|\mathbf{a}(0)\|_4 \leq a_{4,0}m^{1/4}$ can be easily obtained using standard concentration inequalities, where $a_{2,0}$ and $a_{4,0}$ are both universal constants. The following lemma shows if the induction holds, we have every weight matrix close to its initialization.

Lemma 3.5. *If Condition 3.3 holds for* $k' = 1, \ldots, k$*, we have for any* $s = 1, \ldots, k+1$

$$\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(0)\|_2 \leq R'\sqrt{m}$$
$$\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(s-1)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(s-1)\|_2 \leq \eta Q'(s-1)$$

where $R' = \frac{16 c_{x,0} a_{2,0} (c_x)^H \sqrt{n} \|\mathbf{y} - \mathbf{u}(0)\|_2}{\lambda_0 \sqrt{m}} \leq c g_{c_x}(H)^{-1}$ *for some small constant* c *with* $c_x = \max\{2\sqrt{c_\sigma} L c_{w,0}, 1\}$ *and* $Q'(s) = 4 c_{x,0} a_{2,0} (c_x)^H \sqrt{n} \|\mathbf{y} - \mathbf{u}(s)\|_2$*.*

Now we proceed to analyze the perturbation terms.

Lemma 3.6. *If Condition 3.3 holds for* $k' = 1, \ldots, k$*, suppose* $\eta \leq c\lambda_0 \left(n^2 H^2 (c_x)^{3H} g_{2c_x}(H)\right)^{-1}$ *for some small constant* c*, we have*

$$\|\mathbf{I}_2(k)\|_2 \leq \frac{1}{8}\eta\lambda_0 \|\mathbf{y} - \mathbf{u}(k)\|_2.$$

Lemma 3.7. *If Condition 3.3 holds for $k' = 1, \ldots, k$, suppose $\eta \leq c\lambda_0 \left(n^2 H^2 (c_x)^{2H} g_{2c_x}(H)\right)^{-1}$ for some small constant c, then we have $\|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \leq \frac{1}{8}\eta\lambda_0 \|\mathbf{y} - \mathbf{u}(k)\|_2^2$.*

We now proceed with the proof of Theorem 3.1. By induction, we assume Condition 3.3 for all $k' < k$. Using Lemma 3.5, this establishes

$$\begin{aligned}\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F &\leq R'\sqrt{m} \\ &\leq R\sqrt{m} \qquad \text{(using the choice of } m \text{ in the theorem.)}\end{aligned}$$

By Lemma 3.4, this establishes $\lambda_{\min}(\mathbf{G}^{(H)}(k)) \geq \frac{\lambda_0}{2}$.

With these estimates in hand, we are ready to prove the induction hypothesis of Condition 3.3.

$$\begin{aligned}&\|\mathbf{y} - \mathbf{u}(k+1)\|_2^2 \\ &= \|\mathbf{y} - \mathbf{u}(k)\|_2^2 - 2\eta \left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{G}(k) \left(\mathbf{y} - \mathbf{u}(k)\right) - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{I}_2 + \|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \\ &\leq \|\mathbf{y} - \mathbf{u}(k)\|_2^2 - 2\eta \left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{G}^{(H)}(k) \left(\mathbf{y} - \mathbf{u}(k)\right) - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{I}_2 + \|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \\ &\leq (1 - \eta\lambda_0) \|\mathbf{y} - \mathbf{u}(k)\|_2^2 - 2\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \mathbf{I}_2 + \|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \\ &\leq (1 - \frac{\eta\lambda_0}{2}) \|\mathbf{y} - \mathbf{u}(k)\|_2^2 .\end{aligned}$$

The first inequality drops the positive terms $\left(\mathbf{y} - \mathbf{u}(k)\right)^\top \sum_{h \in [H+1], h \neq H} \mathbf{G}^{(h)}(k) \left(\mathbf{y} - \mathbf{u}(k)\right)$. The second inequality uses the argument above that establishes $\lambda_{\min}(\mathbf{G}^{(H)}(k)) \geq \frac{\lambda_0}{2}$. The third inequality uses Lemmas 3.6 and 3.7.

3.9.1 Proofs of Lemmas

Proof of Lemma 3.1. We will bound $\left\|\mathbf{x}_i^{(h)}(0)\right\|_2$ by induction on layers. The induction hypothesis is that with probability at least $1 - (h-1)\frac{\delta}{nH}$ over $\mathbf{W}^{(1)}(0), \ldots, \mathbf{W}^{(h-1)}(0)$, for every $1 \leq h' \leq h-1$, $\frac{1}{2} \leq 1 - \frac{g_C(h')}{2g_C(H)} \leq \left\|\mathbf{x}_i^{(h')}(0)\right\|_2 \leq 1 + \frac{g_C(h')}{2g_C(H)} \leq 2$. Note that it is true for $h = 1$. We calculate the expectation of $\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2$ over the randomness from $\mathbf{W}^{(h)}(0)$. Recall

$$\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2 = \frac{c_\sigma}{m} \sum_{r=1}^m \sigma\left(\mathbf{w}_r^{(h)}(0)^\top \mathbf{x}_i^{(h-1)}(0)\right)^2 .$$

Therefore we have

$$\begin{aligned}\mathbb{E}\left[\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2\right] =& c_\sigma \mathbb{E}\left[\sigma\left(\mathbf{w}_r^{(h)}(0)^\top \mathbf{x}_i^{(h-1)}(0)\right)^2\right] \\ =& c_\sigma \mathbb{E}_{X \sim N(0,1)} \sigma(\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2 X)^2 .\end{aligned}$$

Note that $\sigma(\cdot)$ is $L-$Lipschitz, for any $\frac{1}{2} \leq \alpha \leq 2$, we have

$$\left|\mathbb{E}_{X \sim N(0,1)} \sigma(\alpha X)^2 - \mathbb{E}_{X \sim N(0,1)} \sigma(X)^2\right|$$

$$
\begin{aligned}
\leq & \mathbb{E}_{X\sim N(0,1)} \left|\sigma(\alpha X)^2 - \sigma(X)^2\right| \\
\leq & L\left|\alpha-1\right| \mathbb{E}_{X\sim N(0,1)} \left|X\left(\sigma(\alpha X)+\sigma(X)\right)\right| \\
\leq & L\left|\alpha-1\right| \mathbb{E}_{X\sim N(0,1)} \left|X\right| \left(\left|2\sigma(0)\right| + L\left|(\alpha+1)X\right|\right) \\
\leq & L\left|\alpha-1\right| \left(2\left|\sigma(0)\right| \mathbb{E}_{X\sim N(0,1)} \left|X\right| + L\left|\alpha+1\right| \mathbb{E}_{X\sim N(0,1)} X^2\right) \\
= & L\left|\alpha-1\right| \left(2\left|\sigma(0)\right| \sqrt{\frac{2}{\pi}} + L\left|\alpha+1\right|\right) \\
\leq & \frac{C}{c_\sigma}\left|\alpha-1\right|,
\end{aligned}
$$

where $C \triangleq c_\sigma L\left(2\left|\sigma(0)\right|\sqrt{\frac{2}{\pi}}+2L\right)$, which implies

$$
1 - \frac{Cg_C(h-1)}{2g_C(H)} \leq \mathbb{E}\left[\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2\right] \leq 1 + \frac{Cg_C(h-1)}{2g_C(H)}.
$$

For the variance we have

$$
\begin{aligned}
\mathrm{Var}\left[\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2\right] = & \frac{c_\sigma^2}{m}\mathrm{Var}\left[\sigma\left(\mathbf{w}_r^{(h)}(0)^\top \mathbf{x}_i^{(h-1)}(0)\right)^2\right] \\
\leq & \frac{c_\sigma^2}{m}\mathbb{E}\left[\sigma\left(\mathbf{w}_r^{(h)}(0)^\top \mathbf{x}_i^{(h-1)}(0)\right)^4\right] \\
\leq & \frac{c_\sigma^2}{m}\mathbb{E}\left[\left(\left|\sigma(0)\right| + L\left|\mathbf{w}_r^{(h)}(0)^\top \mathbf{x}_i^{(h-1)}(0)\right|\right)^4\right] \\
\leq & \frac{C_2}{m}.
\end{aligned}
$$

where $C_2 \triangleq \sigma(0)^4 + 8\left|\sigma(0)\right|^3 L\sqrt{2/\pi} + 24\sigma(0)^2L^2 + 64\sigma(0)L^3\sqrt{2/\pi} + 512L^4$ and the last inequality we used the formula for the first four absolute moments of Gaussian.

Applying Chebyshev's inequality and plugging in our assumption on m, we have with probability $1-\frac{\delta}{nH}$ over $\mathbf{W}^{(h)}$,

$$
\left|\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2 - \mathbb{E}\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2\right| \leq \frac{1}{2g_C(H)}.
$$

Thus with probability $1 - h\frac{\delta}{nH}$ over $\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(h)}$,

$$
\left|\left\|\mathbf{x}_i^{(h)}(0)\right\|_2 - 1\right| \leq \left|\left\|\mathbf{x}_i^{(h)}(0)\right\|_2^2 - 1\right| \leq \frac{Cg_C(h-1)}{2g_C(H)} + \frac{1}{2g(H)} = \frac{g_C(h)}{2g_C(H)}.
$$

Using union bounds over $[n]$, we prove the lemma. □

Proof of Lemma 3.3. We prove this lemma by induction. Our induction hypothesis is

$$
\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \leq \sqrt{c_\sigma} LRc_{x,0} g_{c_x}(h),
$$

where $c_x = 2\sqrt{c_\sigma} L c_{w,0}$. For $h = 0$, since the input data is fixed, we know the induction hypothesis holds. Now suppose the induction hypothesis holds for $h' = 0, \ldots, h-1$, we consider $h' = h$.

$$
\begin{aligned}
&\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \\
=&\sqrt{\frac{c_\sigma}{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(k)\right) - \sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\
\leq&\sqrt{\frac{c_\sigma}{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(k)\right) - \sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\
&+ \sqrt{\frac{c_\sigma}{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(0)\right) - \sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\
\leq&\sqrt{\frac{c_\sigma}{m}} L \left(\left\|\mathbf{W}^{(h)}(0)\right\|_2 + \left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F\right) \cdot \left\|\mathbf{x}^{(h-1)}(k) - \mathbf{x}^{(h-1)}(0)\right\|_2 \\
&+ \sqrt{\frac{c_\sigma}{m}} L \left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \left\|\mathbf{x}^{h-1}(0)\right\|_2 \\
\leq&\sqrt{\frac{c_\sigma}{m}} L \left(c_{w,0}\sqrt{m} + R\sqrt{m}\right) \sqrt{c_\sigma} L R c_{x,0} g_{c_x}(h-1) + \sqrt{\frac{c_\sigma}{m}} L \sqrt{m} R c_{x,0} \\
\leq&\sqrt{c_\sigma} L R c_{x,0} \left(c_x g_{c_x}(h-1) + 1\right) \\
\leq&\sqrt{c_\sigma} L R c_{x,0} g_{c_x}(h).
\end{aligned}
$$

□

Proof of Lemma 3.4. Because Frobenius-norm of a matrix is bigger than the operator norm, it is sufficient to bound $\left\|\mathbf{G}^{(H)}(k) - \mathbf{G}^{(H)}(0)\right\|_F$. For simplicity define $z_{i,r}(k) = \mathbf{w}_r^{(H)}(k)^\top \mathbf{x}_i^{(H-1)}(k)$, we have

$$
\begin{aligned}
&\left|\mathbf{G}_{i,j}^{(H)}(k) - \mathbf{G}_{i,j}^{(H)}(0)\right| \\
=&\Big|\mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k) \frac{c_\sigma}{m} \sum_{r=1}^m a_r(k)^2 \sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right) \\
&- \mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0) \frac{c_\sigma}{m} \sum_{r=1}^m a_r(0)^2 \sigma'\left(z_{i,r}(0)\right) \sigma'\left(z_{j,r}(0)\right) \Big| \\
\leq&\left|\mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0)\right| \frac{c_\sigma}{m} \sum_{r=1}^m a_r(0)^2 \left|\sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right)\right| \\
&+ \left|\mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0)\right| \frac{c_\sigma}{m} \left|\sum_{r=1}^m a_r(0)^2 \left(\sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right) - \sigma'\left(z_{i,r}(0)\right) \sigma'\left(z_{j,r}(0)\right)\right)\right| \\
&+ \left|\mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k)\right| \frac{c_\sigma}{m} \left|\sum_{r=1}^m \left(a_r(k)^2 - a_r(0)^2\right) \sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right)\right| \\
\leq&L^2 c_\sigma a_{2,0}^2 \left|\mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0)\right|
\end{aligned}
$$

$$
\begin{aligned}
&+ c_{x,0}^2 \frac{c_\sigma}{m} \left| \sum_{r=1}^m a_r(0)^2 \left(\sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right) - \sigma'\left(z_{i,r}(0)\right) \sigma'\left(z_{j,r}(0)\right) \right) \right| \\
&+ 4L^2 c_{x,0}^2 \frac{c_\sigma}{m} \sum_{r=1}^m \left| a_r(k)^2 - a_r(0)^2 \right| \\
\triangleq & I_1^{i,j} + I_2^{i,j} + I_3^{i,j}.
\end{aligned}
$$

For $I_1^{i,j}$, using Lemma 3.3, we have

$$
\begin{aligned}
I_1^{i,j} =& L^2 c_\sigma a_{2,0}^2 \left| \mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0) \right| \\
\leq & L^2 c_\sigma a_{2,0}^2 \left| (\mathbf{x}_i^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0))^\top \mathbf{x}_j^{(H-1)}(k) \right| + L^2 c_\sigma a_{2,0}^2 \left| \mathbf{x}_i^{(H-1)}(0)^\top (\mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_j^{(H-1)}(0)) \right| \\
\leq & c_\sigma a_{2,0}^2 \sqrt{c_\sigma} L^3 c_{x,0} g_{c_x}(H) R \cdot (c_{x,0} + \sqrt{c_\sigma} L c_{x,0} g_{c_x}(H) R) + c_\sigma \sqrt{c_\sigma} a_{2,0}^2 L^3 c_{x,0} g_{c_x}(H) R c_{x,0} \\
\leq & 3 c_\sigma a_{2,0}^2 c_{x,0}^2 \sqrt{c_\sigma} L^3 g_{c_x}(H) R.
\end{aligned}
$$

For $I_2^{i,j}$, we have

$$
\begin{aligned}
& I_2^{i,j} \\
=& c_{x,0}^2 \frac{c_\sigma}{m} \left| \sum_{r=1}^m a_r(0)^2 \sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right) - a_r(0)^2 \sigma'\left(z_{i,r}(0)\right) \sigma'\left(z_{j,r}(0)\right) \right| \\
\leq & c_{x,0}^2 \frac{c_\sigma}{m} \sum_{r=1}^m a_r(0)^2 \left| \left(\sigma'\left(z_{i,r}(k)\right) - \sigma'\left(z_{i,r}(0)\right)\right) \sigma'\left(z_{j,r}(k)\right) \right| \\
& + \sum_{r=1}^m a_r(0)^2 \left| \left(\sigma'\left(z_{j,r}(k)\right) - \sigma'\left(z_{j,r}(0)\right)\right) \sigma'\left(z_{i,r}(0)\right) \right| \\
\leq & \frac{\beta L c_\sigma c_{x,0}^2}{m} \left(\sum_{r=1}^m a_r(0)^2 \left| z_{i,r}(k) - z_{i,r}(0) \right| + a_r(0)^2 \left| z_{j,r}(k) - z_{j,r}(0) \right| \right) \\
\leq & \frac{\beta L c_\sigma a_{4,0}^2 c_{x,0}^2}{\sqrt{m}} \left(\sqrt{\sum_{r=1}^m \left| z_{i,r}(k) - z_{i,r}(0) \right|^2} + \sqrt{\sum_{r=1}^m \left| z_{j,r}(k) - z_{j,r}(0) \right|^2} \right).
\end{aligned}
$$

Using the same proof for Lemma 3.3, it is easy to see

$$
\sum_{r=1}^m \left| z_{i,r}(t) - z_{i,r}(0) \right|^2 \leq c_{x,0}^2 g_{c_x}(H)^2 m R^2.
$$

Thus

$$
I_2^{i,j} \leq 2\beta c_\sigma a_{4,0}^2 c_{x,0}^3 L g_{c_x}(H) R.
$$

For $I_3^{i,j}$,

$$
I_3^{i,j} = 4L^2 c_{x,0}^2 \frac{c_\sigma}{m} \sum_{r=1}^m \left| a_r(k)^2 - a_r(0)^2 \right|
$$

$$\leq 4L^2 c_{x,0}^2 \frac{c_\sigma}{m} \sum_{r=1}^{m} |a_r(k) - a_r(0)| \, |a_r(k)| + |a_r(k) - a_r(0)| \, |a_r(0)|$$
$$\leq 12L^2 c_{x,0}^2 c_\sigma a_{2,0} R.$$

Therefore we can bound the perturbation

$$\left\|\mathbf{G}^{(H)}(t) - \mathbf{G}^{(H)}(0)\right\|_F = \sqrt{\sum_{(i,j)}^{n,n} \left|\mathbf{G}_{i,j}^{(H)}(t) - \mathbf{G}_{i,j}^{(H)}(0)\right|^2}$$
$$\leq \left[\left(2\beta c_{x,0} a_{4,0}^2 + 3\sqrt{c_\sigma} L^2\right) L c_\sigma c_{x,0}^2 a_{2,0}^2 g_{c_x}(H) + 12L^2 c_{x,0}^2 c_\sigma a_{2,0}\right] nR.$$

Plugging in the bound on R, we have the desired result.

□

Proof of Lemma 3.5. We will prove this corollary by induction. The induction hypothesis is

$$\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F \leq \sum_{s'=0}^{s-1} (1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m} \leq R'\sqrt{m}, s \in [k+1],$$
$$\left\|\mathbf{a}(s) - \mathbf{a}(0)\right\|_2 \leq \sum_{s'=0}^{s-1} (1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m} \leq R'\sqrt{m}, s \in [k+1].$$

First it is easy to see it holds for $s' = 0$. Now suppose it holds for $s' = 0, \ldots, s$, we consider $s' = s + 1$. We have

$$\left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(s)\right\|_F$$
$$= \eta \left\| \left(\frac{c_\sigma}{m}\right)^{\frac{H-h+1}{2}} \sum_{i=1}^{n} (y_i - u_i(s)) \mathbf{x}_i^{(h-1)}(s) \left(\mathbf{a}(s)^\top \left(\prod_{k=h+1}^{H} \mathbf{J}_i^{(k)}(s) \mathbf{W}^{(k)}(s) \right) \mathbf{J}_i^{(h)}(s) \right) \right\|_F$$
$$\leq \eta \left(\frac{c_\sigma}{m}\right)^{\frac{H-h+1}{2}} \left\|\mathbf{a}(s)\right\|_2 \sum_{i=1}^{n} |y_i - u_i(s)| \left\|\mathbf{x}_i^{(h-1)}(s)\right\|_2 \prod_{k=h+1}^{H} \left\|\mathbf{W}^{(k)}(s)\right\|_2 \prod_{k=h}^{H} \left\|\mathbf{J}^{(k)}(s)\right\|_2,$$
$$\left\|\mathbf{a}(s+1) - \mathbf{a}(s)\right\|_2 = \eta \left\| \sum_{i=1}^{n} (y_i - u_i(s)) \mathbf{x}_i^{(H)}(s) \right\|_2.$$

To bound $\left\|\mathbf{x}_i^{(h-1)}(s)\right\|_2$, we can just apply Lemma 3.3 and get

$$\left\|\mathbf{x}_i^{(h-1)}(s)\right\|_2 \leq \sqrt{c_\sigma} L c_{x,0} g_{c_x}(h) R' + c_{x,0} \leq 2c_{x,0}.$$

To bound $\left\|\mathbf{W}^{(k)}(s)\right\|_2$, we use our assumption

$$\prod_{k=h+1}^{H} \left\|\mathbf{W}^{(k)}(s)\right\|_2 \leq \prod_{k=h+1}^{H} \left(\left\|\mathbf{W}^{(k)}(0)\right\|_2 + \left\|\mathbf{W}^{(k)}(s) - \mathbf{W}^{(k)}(0)\right\|_2\right)$$

$$
\begin{aligned}
&\leq \prod_{k=h+1}^{H} (c_{w,0}\sqrt{m} + R'\sqrt{m}) \\
&= (c_{w,0} + R')^{H-h} m^{\frac{H-h}{2}} \\
&\leq (2c_{w,0})^{H-h} m^{\frac{H-h}{2}}.
\end{aligned}
$$

Note that $\left\|\mathbf{J}^{(k)}(s)\right\|_2 \leq L$. Plugging in these two bounds back, we obtain

$$
\begin{aligned}
\left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(s)\right\|_F \leq & 4\eta c_{x,0} a_{2,0} c_x^H \sum_{i=1}^{n} |y_i - u(s)| \\
\leq & 4\eta c_{x,0} a_{2,0} c_x^H \sqrt{n} \left\|\mathbf{y} - \mathbf{u}(s)\right\|_2 \\
= & \eta Q'(s) \\
\leq & (1 - \frac{\eta\lambda_0}{2})^{s/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m}.
\end{aligned}
$$

Similarly, we have

$$
\begin{aligned}
\left\|\mathbf{a}(s+1) - \mathbf{a}(s)\right\|_2 \leq & 2\eta c_{x,0} \sum_{i=1}^{n} |y_i - u(s)| \\
\leq & \eta Q'(s) \\
\leq & (1 - \frac{\eta\lambda_0}{2})^{s/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m}.
\end{aligned}
$$

Thus

$$
\begin{aligned}
&\left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(0)\right\|_F \\
\leq & \left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(s)\right\|_F + \left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F \\
\leq & \sum_{s'=0}^{s} \eta (1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m}.
\end{aligned}
$$

Similarly,

$$
\begin{aligned}
&\left\|\mathbf{a}(s+1) - \mathbf{a}(0)\right\|_2 \\
\leq & \sum_{s'=0}^{s} \eta (1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4} \eta\lambda_0 R' \sqrt{m}.
\end{aligned}
$$

□

Proof of Lemma 3.6. Fix $i \in [n]$, we bound

$$
\left|I_2^i(k)\right| \leq \eta \max_{0\leq s\leq \eta} \sum_{h=1}^{H} \left\|L'^{(h)}(\theta(k))\right\|_F \left\|u_i'^{(h)}(\theta(k)) - u_i'^{(h)}\left(\theta(k) - sL'^{(h)}(\theta(k))\right)\right\|_F.
$$

For the gradient norm, we have

$$\begin{aligned}&\left\|L'^{(h)}(\theta(k))\right\|_F\\=&\left\|\left(\frac{c_\sigma}{m}\right)^{\frac{H-h+1}{2}}\sum_{i=1}^{n}(y_i-u_i(k))\mathbf{x}_i^{(h-1)}(k)\left(\mathbf{a}(k)^\top\left(\prod_{l=h+1}^{H}\mathbf{J}_i^{(l)}(k)\mathbf{W}^{(l)}(k)\right)\mathbf{J}_i^{(h)}(k)\right)\right\|_F.\end{aligned}$$

Similar to the proof for Lemma 3.5, we have

$$\left\|L'^{(h)}(\theta(k))\right\|_F\leq Q'(k).$$

Let $\theta(k,s)=\theta(k)-sL'(\theta(k))$,

$$\begin{aligned}&\left\|u_i'^{(h)}\left(\theta(k)\right)-u_i'^{(h)}\left(\theta(k,s)\right)\right\|_F\\=&\left(\frac{c_\sigma}{m}\right)^{\frac{H-h+1}{2}}\left\|\mathbf{x}_i^{(h-1)}(k)\left(\mathbf{a}(k)^\top\left(\prod_{l=h+1}^{H}\mathbf{J}_i^{(l)}(k)\mathbf{W}^{(l)}(k)\right)\mathbf{J}_i^{(h)}(k)\right)\right.\\&\left.-\mathbf{x}_i^{(h-1)}(k,s)\left(\mathbf{a}(k,s)^\top\left(\prod_{l=h+1}^{H}\mathbf{J}_i^{(l)}(k,s)\mathbf{W}^{(l)}(k,s)\right)\mathbf{J}_i^{(h)}(k,s)\right)\right\|_F\end{aligned}$$

Through standard calculations, we have

$$\begin{aligned}\left\|\mathbf{W}^{(l)}(k)-\mathbf{W}^{(l)}(k,s)\right\|_F\leq&\eta Q'(k),\\\left\|\mathbf{a}(k)-\mathbf{a}(k,s)\right\|_2\leq&\eta Q'(k),\\\left\|\mathbf{x}_i^{(h-1)}(k)-\mathbf{x}_i^{(h-1)}(k,s)\right\|_F\leq&2\eta\sqrt{c_\sigma}Lc_{x,0}g_{2c_x}(H)\frac{Q'(k)}{\sqrt{m}},\\\left\|\mathbf{J}_i^{(l)}(k)-\mathbf{J}_i^{(l)}(k,s)\right\|_F\leq&2\eta\beta\sqrt{c_\sigma}Lc_{x,0}g_{2c_x}(H)Q'(k).\end{aligned}$$

According to Lemma 3.26, we have

$$\begin{aligned}&\left\|u_i'^{(h)}\left(\mathbf{w}(k)\right)-u_i'^{(h)}\left(\mathbf{w}(k,s)\right)\right\|_F\\\leq&4c_{x,0}a_{2,0}c_x^H\eta\frac{Q'(k)}{\sqrt{m}}\left(\frac{H}{2}+\left[\frac{1}{2c_{x,0}}+\frac{H\beta\sqrt{m}}{L}\right]2\sqrt{c_\sigma}Lc_{x,0}g_{2c_x}(H)\right)\\\leq&16H\sqrt{c_\sigma}c_{x,0}^2a_{2,0}c_x^Hg_{2c_x}(H)\beta\eta Q'(k).\end{aligned}$$

Thus we have

$$\left|I_2^i\right|\leq 16H^2\sqrt{c_\sigma}c_{x,0}^2a_{2,0}c_x^Hg_{2c_x}(H)\beta\eta^2Q'(k)^2.$$

Since this holds for all $i\in[n]$, plugging in η and noting that $\left\|\mathbf{y}-\mathbf{u}(0)\right\|_2=O(\sqrt{n})$, we have

$$\left\|\mathbf{I}_2(k)\right\|_2\leq\frac{1}{8}\eta\lambda_0\left\|\mathbf{y}-\mathbf{u}(k)\right\|_2.$$

□

Proof of Lemma 3.7.

$$
\begin{aligned}
&\|\mathbf{u}(k+1)-\mathbf{u}(k)\|_2^2 \\
=&\sum_{i=1}^n \left(\mathbf{a}(k+1)^\top \mathbf{x}_i^{(H)}(k+1)-\mathbf{a}(k)^\top \mathbf{x}_i^{(H)}(k)\right)^2 \\
=&\sum_{i=1}^n \left([\mathbf{a}(k+1)-\mathbf{a}(k)]^\top \mathbf{x}_i^{(H)}(k+1)+\mathbf{a}(k)^\top\left[\mathbf{x}_i^{(H)}(k+1)-\mathbf{x}_i^{(H)}(k)\right]\right)^2 \\
\le& 2\|\mathbf{a}(k+1)-\mathbf{a}(k)\|_2^2 \sum_{i=1}^n \left\|\mathbf{x}_i^{(H)}(k+1)\right\|_2^2 + 2\|\mathbf{a}(k)\|_2^2 \sum_{i=1}^n \left\|\mathbf{x}_i^{(H)}(k+1)-\mathbf{x}_i^{(H)}(k)\right\|_2^2 \\
\le& 8n\eta^2 c_{x,0}^2 Q'(k)^2 + 4n\left(2\eta\sqrt{c_\sigma} L c_{x,0} a_{2,0}^2 g_{2c_x}(H) Q'(k)\right)^2 \\
\le& \frac{1}{8}\eta\lambda_0\|\mathbf{y}-\mathbf{u}(k)\|_2^2 .
\end{aligned}
$$

□

3.10 Proofs for Section 3.5

The gradient for ResNet is

$$
\frac{\partial L}{\partial \mathbf{W}^{(h)}} = \frac{c_{res}}{H\sqrt{m}} \sum_{i=1}^n (y_i-u_i)\mathbf{x}_i^{(h-1)} \cdot \left[\mathbf{a}^\top \prod_{l=h+1}^H \left(\mathbf{I}+\frac{c_{res}}{H\sqrt{m}}\mathbf{J}_i^{(l)}\mathbf{W}^{(l)}\right)\mathbf{J}_i^{(h)}\right]
$$

For ResNets, $\mathbf{G}^{(H)}$ has the following form:

$$
\mathbf{G}_{ij}^{(H)} = \frac{c_{res}^2}{H^2 m}(\mathbf{x}_i^{(H-1)})^\top \mathbf{x}_j^{(H-1)} \sum_{r=1}^m a_r^2 \sigma'((\mathbf{w}_r^{(H)})^\top \mathbf{x}_i^{(H-1)})\sigma'((\mathbf{w}_r^{(H)})^\top \mathbf{x}_j^{(H-1)}). \tag{3.13}
$$

Similar to Lemma 3.1, we can show with high probability the feature of each layer is approximately normalized.

Lemma 3.8 (Lemma on Initialization Norms)**.** *If $\sigma(\cdot)$ is $L-$Lipschitz and $m=\Omega\left(\frac{n}{\delta}\right)$, assuming $\left\|\mathbf{W}^{(h)}(0)\right\|_{\mathrm{op}} \le c_{w,0}\sqrt{m}$ for $h\in[2,H]$ and $1.99\le c_{w,0}\le 2.01$ for Gaussian initialization. We have with probability at least $1-\delta$ over random initialization, for every $h\in[H]$ and $i\in[n]$,*

$$
\frac{1}{c_{x,0}} \le \left\|\mathbf{x}_i^{(h)}(0)\right\|_2 \le c_{x,0}
$$

for some universal constant $c_{x,0}>1$ (only depends on σ).

The following lemma lower bounds $\mathbf{G}^{(H)}(0)$'s least eigenvalue. This lemma is a direct consequence of results in Section 3.12.

Lemma 3.9 (Least Eigenvalue at the Initialization)**.** *If $m=\Omega\left(\frac{n^2\log(Hn/\delta)}{\lambda_0^2}\right)$, we have*

$$
\lambda_{\min}(\mathbf{G}^{(H)}(0)) \ge \frac{3}{4}\lambda_0.
$$

Next, we characterize how the perturbation on the weight matrices affects the input of each layer.

Lemma 3.10. *Suppose $\sigma(\cdot)$ is L-Lipschitz and for $h \in [H]$, $\left\|\mathbf{W}^{(h)}(0)\right\|_{\text{op}} \leq c_{w,0}\sqrt{m}$, $\left\|\mathbf{x}^{(h)}(0)\right\|_2 \leq c_{x,0}$ and $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \leq \sqrt{m}R$ for some constant $c_{w,0}, c_{x,0} > 0$ and $R \leq c_{w,0}$. Then we have*

$$\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \leq \left(\sqrt{c_\sigma}L + \frac{c_{x,0}}{c_{w,0}}\right) e^{2c_{res}c_{w,0}L}R.$$

Next, we characterize how the perturbation on the weight matrices affect $\mathbf{G}^{(H)}$.

Lemma 3.11. *Suppose $\sigma(\cdot)$ is differentiable, $L-$Lipschitz and $\beta-$smooth. Using the same notations in Lemma 3.4, if $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F, \|\mathbf{a}(k) - \mathbf{a}(0)\|_2 \leq \sqrt{m}R$ where $R \leq c\lambda_0 H^2 n^{-1}$ and $R \leq c$ for some small constant c, we have*

$$\left\|\mathbf{G}^{(H)}(k) - \mathbf{G}^{(H)}(0)\right\|_{\text{op}} \leq \frac{\lambda_0}{2}.$$

We prove Theorem 3.2 by induction. Our induction hypothesis is just the following convergence rate of empirical loss.

A directly corollary of this condition is the following bound of deviation from the initialization. The proof only involves standard calculations so we defer it to appendix.

Lemma 3.12. *If Condition 3.3 holds for $k' = 1, \ldots, k$, we have for any $s \in [k+1]$*

$$\begin{aligned}
&\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(0)\|_2 \leq R'\sqrt{m}, \\
&\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(s-1)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(s-1)\|_2 \leq \eta Q'(s-1),
\end{aligned}$$

where $R' = \frac{16c_{res}c_{x,0}a_{2,0}Le^{2c_{res}c_{w,0}L}\sqrt{n}\|\mathbf{y}-\mathbf{u}(0)\|_2}{H\lambda_0\sqrt{m}} < c$ for some small constant c and $Q'(s) = 4c_{res}c_{x,0}a_{2,0}Le^{2c_{res}c_{w,0}L}\sqrt{n}\left\|\mathbf{y} - \mathbf{u}(s)\right\|_2 / H$.

The next lemma bounds the $\mathbf{I}_2$ term.

Lemma 3.13. *If Condition 3.3 holds for $k' = 1, \ldots, k$ and $\eta \leq c\lambda_0 H^2 n^{-2}$ for some small constant c, we have*

$$\|\mathbf{I}_2(k)\|_2 \leq \frac{1}{8}\eta\lambda_0 \left\|\mathbf{y} - \mathbf{u}(k)\right\|_2.$$

Next we bound the quadratic term.

Lemma 3.14. *If Condition 3.3 holds for $k' = 1, \ldots, k$ and $\eta \leq c\lambda_0 H^2 n^{-2}$ for some small constant c, we have $\|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \leq \frac{1}{8}\eta\lambda_0 \left\|\mathbf{y} - \mathbf{u}(k)\right\|_2^2$.*

Now using the same argument as in the proof for multilayer fully connected neural network, we finish our proof for ResNet.

3.10.1 Proofs of Lemmas

Proof of Lemma 3.8. We will bound $\left\|\mathbf{x}_i^{(h)}(0)\right\|_2$ layer by layer. For the first layer, we can calculate

$$\mathbb{E}\left[\left\|\mathbf{x}_i^{(1)}(0)\right\|_2^2\right] = c_\sigma \mathbb{E}\left[\sigma\left(\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_i\right)^2\right]$$

$$
\begin{aligned}
=&c_\sigma \mathbb{E}_{X\sim N(0,1)}\sigma(X)^2\\
=&1.
\end{aligned}
$$

$$
\begin{aligned}
\mathrm{Var}\left[\left\|\mathbf{x}_i^{(1)}(0)\right\|_2^2\right] =&\frac{c_\sigma^2}{m}\mathrm{Var}\left[\sigma\left(\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_i(0)\right)^2\right]\\
\le&\frac{c_\sigma^2}{m}\mathbb{E}_{X\sim N(0,1)}\sigma(X)^4\\
\le&\frac{c_\sigma^2}{m}\mathbb{E}\left[\left(|\sigma(0)| + L\left|\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_i\right|\right)^4\right]\\
\le&\frac{C_2}{m},
\end{aligned}
$$

where $C_2 \triangleq \sigma(0)^4 + 4\left|\sigma(0)\right|^3 L\sqrt{2/\pi} + 6\sigma(0)^2L^2 + 8\left|\sigma(0)\right| L^3\sqrt{2/\pi} + 32L^4$. We have with probability at least $1-\frac{\delta}{n}$,

$$
\frac{1}{2} \le \left\|\mathbf{x}_i^{(1)}(0)\right\|_2 \le 2.
$$

By definition we have for $2 \le h \le H$,

$$
\begin{aligned}
\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2 - \left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}_i^{(h-1)}(0)\right)\right\|_2 \le& \left\|\mathbf{x}^{(h)}(0)\right\|_2\\
\le& \left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2 + \left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}^{(h-1)}(0)\right)\right\|_2,
\end{aligned}
$$

where

$$
\left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}_i^{(h-1)}(0)\right)\right\|_2 \le \frac{c_{res}c_{w,0}L}{H}\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2.
$$

Thus

$$
\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2\left(1-\frac{c_{res}c_{w,0}L}{H}\right) \le \left\|\mathbf{x}^{(h)}(0)\right\|_2 \le \left\|\mathbf{x}_i^{(h-1)}(0)\right\|_2\left(1+\frac{c_{res}c_{w,0}L}{H}\right),
$$

which implies

$$
\frac{1}{2}e^{-c_{res}c_{w,0}L} \le \left\|\mathbf{x}^{(h)}(0)\right\|_2 \le 2e^{c_{res}c_{w,0}L}.
$$

Choosing $c_{x,0} = 2e^{c_{res}c_{w,0}L}$ and using union bounds over $[n]$, we prove the lemma.

□

Proof of Lemma 3.10. We prove this lemma by induction. Our induction hypothesis is

$$
\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \le g(h),
$$

where

$$
g(h) = g(h-1)\left[1+\frac{2c_{res}c_{w,0}L}{H}\right] + \frac{L}{H}Rc_{x,0}.
$$

For $h = 1$, we have

$$\begin{aligned}\left\|\mathbf{x}^{(1)}(k) - \mathbf{x}^{(1)}(0)\right\|_2 &\leq \sqrt{\frac{c_\sigma}{m}} \left\|\sigma\left(\mathbf{W}^{(1)}(k)\mathbf{x}\right) - \sigma\left(\mathbf{W}^{(1)}(0)\mathbf{x}\right)\right\|_2 \\ &\leq \sqrt{\frac{c_\sigma}{m}} \left\|\mathbf{W}^{(1)}(k) - \mathbf{W}^{(1)}(0)\right\|_F \leq \sqrt{c_\sigma} LR,\end{aligned}$$

which implies $g(1) = \sqrt{c_\sigma} LR$, for $2 \leq h \leq H$, we have

$$\begin{aligned}&\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_2 \leq \frac{c_{res}}{H\sqrt{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(k)\right) - \sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\ &+ \left\|\mathbf{x}^{(h-1)}(k) - \mathbf{x}^{(h-1)}(0)\right\|_2 \\ &\leq \frac{c_{res}}{H\sqrt{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(k)\right) - \sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\ &+ \frac{c_{res}}{H\sqrt{m}} \left\|\sigma\left(\mathbf{W}^{(h)}(k)\mathbf{x}^{(h-1)}(0)\right) - \sigma\left(\mathbf{W}^{(h)}(0)\mathbf{x}^{(h-1)}(0)\right)\right\|_2 \\ &+ \left\|\mathbf{x}^{(h-1)}(k) - \mathbf{x}^{(h-1)}(0)\right\|_2 \\ &\leq \frac{c_{res}L}{H\sqrt{m}} \left(\left\|\mathbf{W}^{(h)}(0)\right\|_2 + \left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F\right) \cdot \left\|\mathbf{x}^{(h-1)}(k) - \mathbf{x}^{(h-1)}(0)\right\|_2 \\ &+ \frac{c_{res}L}{H\sqrt{m}} \left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \left\|\mathbf{x}^{h-1}(0)\right\|_2 + \left\|\mathbf{x}^{(h-1)}(k) - \mathbf{x}^{(h-1)}(0)\right\|_2 \\ &\leq \left[1 + \frac{c_{res}L}{H\sqrt{m}} \left(c_{w,0}\sqrt{m} + R\sqrt{m}\right)\right] g(h-1) + \frac{c_{res}L}{H\sqrt{m}} \sqrt{m} R c_{x,0} \\ &\leq \left(1 + \frac{2c_{res}c_{w,0}L}{H}\right) g(h-1) + \frac{c_{res}}{H} L c_{x,0} R.\end{aligned}$$

Lastly, simple calculations show $g(h) \leq \left(\sqrt{c_\sigma}L + \frac{c_{x,0}}{c_{w,0}}\right) e^{2c_{res}c_{w,0}L} R$.

□

Proof of Lemma 3.11. Similar to the proof of Lemma 3.4, we can obtain

$$\left|\mathbf{G}_{i,j}^{(H)}(k) - \mathbf{G}_{i,j}^{(H)}(0)\right| \leq \frac{c_{res}^2}{H^2} \left(I_1^{i,j} + I_2^{i,j} + I_3^{i,j}\right).$$

For $I_1^{i,j}$, using Lemma 3.10, we have

$$\begin{aligned}I_1^{i,j} =& L^2 a_{2,0}^2 \left|\mathbf{x}_i^{(H-1)}(k)^\top \mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0)^\top \mathbf{x}_j^{(H-1)}(0)\right| \\ \leq& L^2 a_{2,0}^2 \left|(\mathbf{x}_i^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0))^\top \mathbf{x}_j^{(H-1)}(k)\right| + L^2 a_{2,0}^2 \left|\mathbf{x}_i^{(H-1)}(0)^\top (\mathbf{x}_i^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0))\right| \\ \leq& c_x L^2 a_{2,0}^2 R \cdot (c_{x,0} + c_x R) + c_{x,0} c_x L^2 a_{2,0}^2 R \\ \leq& 3 c_{x,0} c_x L^2 a_{2,0}^2 R,\end{aligned}$$

where $c_x \triangleq \left(\sqrt{c_\sigma}L + \frac{c_{x,0}}{c_{w,0}}\right) e^{2c_{res}c_{w,0}L}$. To bound $I_2^{i,j}$, we have

$$I_2^{i,j} = c_{x,0}^2 \frac{1}{m} \left|\sum_{r=1}^m a_r(0)^2 \sigma'\left(z_{i,r}(k)\right) \sigma'\left(z_{j,r}(k)\right) - a_r(0)^2 \sigma'\left(z_{i,r}(0)\right) \sigma'\left(z_{j,r}(0)\right)\right|$$

$$
\begin{aligned}
\leq & c_{x,0}^2 \frac{1}{m} \sum_{r=1}^{m} a_r(0)^2 \left| \left(\sigma'\left(z_{i,r}(k) \right) - \sigma'\left(z_{i,r}(0) \right) \right) \sigma'\left(z_{j,r}(k) \right) \right| \\
& + \sum_{r=1}^{m} a_r(0)^2 \left| \left(\sigma'\left(z_{j,r}(k) \right) - \sigma'\left(z_{j,r}(0) \right) \right) \sigma'\left(z_{i,r}(0) \right) \right| \\
\leq & \frac{\beta L c_{x,0}^2}{m} \left(\sum_{r=1}^{m} a_r(0)^2 \left| z_{i,r}(k) - z_{i,r}(0) \right| + a_r(0)^2 \left| z_{j,r}(k) - z_{j,r}(0) \right| \right) \\
\leq & \frac{\beta L a_{4,0}^2 c_{x,0}^2}{\sqrt{m}} \left(\sqrt{\sum_{r=1}^{m} \left| z_{i,r}(k) - z_{i,r}(0) \right|^2} + \sqrt{\sum_{r=1}^{m} \left| z_{j,r}(k) - z_{j,r}(0) \right|^2} \right).
\end{aligned}
$$

Using the same proof for Lemma 3.10, it is easy to see

$$
\sum_{r=1}^{m} \left| z_{i,r}(k) - z_{i,r}(0) \right|^2 \leq \left(2 c_x c_{w,0} + c_{x,0} \right)^2 L^2 m R^2.
$$

Thus

$$
I_2^{i,j} \leq 2\beta c_{x,0}^2 \left(2 c_x c_{w,0} + c_{x,0} \right) L^2 R.
$$

The bound of $I_3^{i,j}$ is similar to that in Lemma 3.4,

$$
I_3^{i,j} \leq 12 L^2 c_{x,0}^2 a_{2,0} R.
$$

Therefore we can bound the perturbation

$$
\begin{aligned}
\left\| \mathbf{G}^{(H)}(k) - \mathbf{G}^{(H)}(0) \right\|_F = & \sqrt{\sum_{(i,j)}^{n,n} \left| \mathbf{G}_{i,j}^{(H)}(k) - \mathbf{G}_{i,j}^{(H)}(0) \right|^2} \\
\leq & \frac{c_{res}^2 L^2 n R}{H^2} \left[3 c_{x,0} c_x a_{2,0}^2 + 2\beta c_{x,0}^2 \left(2 c_x c_{w,0} + c_{x,0} \right) a_{4,0}^2 + 12 c_{x,0}^2 a_{2,0} \right].
\end{aligned}
$$

Plugging in the bound on R, we have the desired result.

□

Proof of Lemma 3.12. We will prove this corollary by induction. The induction hypothesis is

$$
\begin{aligned}
\left\| \mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0) \right\|_F \leq \sum_{s'=0}^{s-1} (1 - \frac{\eta \lambda_0}{2})^{s'/2} \frac{1}{4} \eta \lambda_0 R' \sqrt{m} \leq R' \sqrt{m}, s \in [k+1], \\
\left\| \mathbf{a}(s) - \mathbf{a}(0) \right\|_2 \leq \sum_{s'=0}^{s-1} (1 - \frac{\eta \lambda_0}{2})^{s'/2} \frac{1}{4} \eta \lambda_0 R' \sqrt{m} \leq R' \sqrt{m}, s \in [k+1].
\end{aligned}
$$

First it is easy to see it holds for $s' = 0$. Now suppose it holds for $s' = 0, \ldots, s$, we consider $s' = s + 1$. Similar to Lemma 3.5, we have

$$\begin{aligned}
&\left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(s)\right\|_F \\
\leq&\eta \frac{Lc_{res}}{H\sqrt{m}} \|\mathbf{a}\|_2 \sum_{i=1}^{n} |y_i - u_i(s)| \left\|\mathbf{x}_i^{(h-1)}(s)\right\|_2 \prod_{k=h+1}^{H} \left\|\mathbf{I} + \frac{c_{res}\lambda^{3/2}}{H\sqrt{m}} \mathbf{J}_i^{(k)}(s)\mathbf{W}^{(k)}(s)\right\|_2 \\
\leq&2\eta c_{res} c_{x,0} L a_{2,0} e^{2c_{res}c_{w,0}L} \sqrt{n} \|\mathbf{y} - \mathbf{u}(s)\|_2 / H \\
=&\eta Q'(s) \\
\leq&(1 - \frac{\eta\lambda_0}{2})^{s/2} \frac{1}{4}\eta\lambda_0 R'\sqrt{m},
\end{aligned}$$

Similarly, we have

$$\begin{aligned}
\|\mathbf{a}(s+1) - \mathbf{a}(s)\|_2 \leq&2\eta c_{x,0} \sum_{i=1}^{n} |y_i - u(s)| \\
\leq&\eta Q'(s) \\
\leq&(1 - \frac{\eta\lambda_0}{2})^{s/2} \frac{1}{4}\eta\lambda_0 R'\sqrt{m}.
\end{aligned}$$

Thus

$$\begin{aligned}
&\left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(0)\right\|_F \\
\leq& \left\|\mathbf{W}^{(h)}(s+1) - \mathbf{W}^{(h)}(s)\right\|_F + \left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F \\
\leq& \sum_{s'=0}^{s} \eta(1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4}\eta\lambda_0 R'\sqrt{m}.
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\|\mathbf{a}(s+1) - \mathbf{a}(0)\|_2 \\
\leq& \sum_{s'=0}^{s} \eta(1 - \frac{\eta\lambda_0}{2})^{s'/2} \frac{1}{4}\eta\lambda_0 R'\sqrt{m}.
\end{aligned}$$

□

Proof of Lemma 3.13. Similar to Lemma 3.6, we first bound the gradient norm.

$$\begin{aligned}
&\left\|L'^{(h)}(\mathbf{w}(k))\right\|_F \\
=&\left\| \frac{c_{res}}{H\sqrt{m}} \sum_{i=1}^{n} (y_i - u_i(k)) \mathbf{x}_i^{(h-1)}(k) \cdot \left[\mathbf{a}(k)^\top \prod_{l=h+1}^{H} \left(\mathbf{I} + \frac{c_{res}}{H\sqrt{m}} \mathbf{J}_i^{(l)}(k)\mathbf{W}^{(l)}(k)\right) \mathbf{J}_i^{(h)}(k)\right] \right\|_F
\end{aligned}$$

$$\leq \frac{c_{res}L}{H\sqrt{m}} \left\|\mathbf{a}(k)\right\|_2 \sum_{i=1}^{n} \left|y_i - u_i(k)\right| \left\|\mathbf{x}^{(h-1)}(k)\right\|_2 \prod_{k=h+1}^{H} \left\|\mathbf{I} + \frac{c_{res}}{H\sqrt{m}} \mathbf{J}_i^{(k)}(k)\mathbf{W}^{(k)}(k)\right\|_2.$$

We have bounded the RHS in the proof for Lemma 3.12, thus

$$\left\|L'^{(h)}(\theta(k))\right\|_F \leq \lambda_0 Q'(k).$$

Let $\theta(k,s) = \theta(k) - sL'(\theta(k))$, we have

$$\begin{aligned}
&\left\|u_i'^{(h)}\left(\theta(k)\right) - u_i'^{(h)}\left(\theta(k,s)\right)\right\|_F = \\
&\frac{c_{res}}{H\sqrt{m}} \Bigg\| \mathbf{x}_i^{(h-1)}(k)\mathbf{a}(k)^\top \prod_{l=h+1}^{H} \left(\mathbf{I} + \frac{c_{res}}{H\sqrt{m}} \mathbf{J}_i^{(l)}(k)\mathbf{W}^{(l)}(k)\right) \mathbf{J}_i^{(h)}(k) \\
&-\mathbf{x}_i^{(h-1)}(k,s)\mathbf{a}(k,s)^\top \prod_{l=h+1}^{H} \left(\mathbf{I} + \frac{c_{res}}{H\sqrt{m}} \mathbf{J}_i^{(l)}(k,s)\mathbf{W}^{(l)}(k,s)\right) \mathbf{J}_i^{(h)}(k,s) \Bigg\|_F.
\end{aligned}$$

Through standard calculations, we have

$$\begin{aligned}
\left\|\mathbf{W}^{(l)}(k) - \mathbf{W}^{(l)}(k,s)\right\|_F \leq& \eta Q'(k), \\
\left\|\mathbf{a}(k) - \mathbf{a}(k,s)\right\|_F \leq& \eta Q'(k), \\
\left\|\mathbf{x}_i^{(h-1)}(k) - \mathbf{x}_i^{(h-1)}(k,s)\right\|_F \leq& \eta c_x \frac{Q'(k)}{\sqrt{m}}, \\
\left\|\mathbf{J}^{(l)}(k) - \mathbf{J}^{(l)}(k,s)\right\|_F \leq& 2\left(c_{x,0} + c_{w,0}c_x\right)\eta\beta Q'(k),
\end{aligned}$$

where $c_x \triangleq \left(\sqrt{c_\sigma}L + \frac{c_{x,0}}{c_{w,0}}\right) e^{3c_{res}c_{w,0}L}$. According to Lemma 3.26, we have

$$\begin{aligned}
&\left\|u_i'^{(h)}\left(\theta(k)\right) - u_i'^{(h)}\left(\theta(k,s)\right)\right\|_F \\
\leq& \frac{4}{H} c_{res}c_{x,0}La_{2,0}e^{2Lc_{w,0}}\eta\frac{Q'(k)}{\sqrt{m}} \left(\frac{c_x}{c_{x,0}} + \frac{2}{L}\left(c_{x,0} + c_{w,0}c_x\right)\beta\sqrt{m} + 4c_{w,0}\left(c_{x,0} + c_{w,0}c_x\right)\beta + L + 1\right) \\
\leq& \frac{32}{H} c_{res}c_{x,0}a_{2,0}e^{2Lc_{w,0}}\left(c_{x,0} + c_{w,0}c_x\right)\beta\eta Q'(k).
\end{aligned}$$

Thus we have

$$\left|I_2^i\right| \leq 32c_{res}c_{x,0}a_{2,0}e^{2Lc_{w,0}}\left(c_{x,0} + c_{w,0}c_x\right)\beta\eta^2 Q'(k)^2 \leq \frac{1}{8}\eta\lambda_0 \left\|\mathbf{y} - \mathbf{u}(k)\right\|_2,$$

where we used the bound of η and that $\left\|\mathbf{y} - \mathbf{u}(0)\right\|_2 = O(\sqrt{n})$,. □

Proof of Lemma 3.14.

$$\left\|\mathbf{u}(k+1) - \mathbf{u}(k)\right\|_2^2 = \sum_{i=1}^{n} \left(\mathbf{a}(k+1)^\top \mathbf{x}_i^{(H)}(k+1) - \mathbf{a}(k)^\top \mathbf{x}_i^{(H)}(k)\right)^2$$

$$
\begin{aligned}
=&\sum_{i=1}^{n}\left(\left[\mathbf{a}(k+1)-\mathbf{a}(k)\right]^{\top}\mathbf{x}_i^{(H)}(k+1)+\mathbf{a}(k)^{\top}\left[\mathbf{x}_i^{(H)}(k+1)-\mathbf{x}_i^{(H)}(k)\right]\right)^2\\
\leq&2\left\|\mathbf{a}(k+1)-\mathbf{a}(k)\right\|_2^2\sum_{i=1}^{n}\left\|\mathbf{x}_i^{(H)}(k+1)\right\|_2^2\\
&+2\left\|\mathbf{a}(k)\right\|_2^2\sum_{i=1}^{n}\left\|\mathbf{x}_i^{(H)}(k+1)-\mathbf{x}_i^{(H)}(k)\right\|_2^2\\
\leq&8n\eta^2c_{x,0}^2Q'(k)^2+4n\left(\eta a_{2,0}c_xQ'(k)\right)^2\\
\leq&\frac{1}{8}\eta\lambda_0\left\|\mathbf{y}-\mathbf{u}(k)\right\|_2^2.
\end{aligned}
$$

□

3.11 Proofs for Section 3.6

For CNN, denote $\mathbf{x}_{i,l}=\phi\left(\mathbf{x}_{i,l}\right)_{:,l}$, $\mathbf{G}^{(H)}$ has the following form:

$$
\mathbf{G}_{ij}^{(H)}=\frac{c_{res}^2}{H^2m}\sum_{r=1}^{m}\left[\sum_{l=1}^{p}a_{l,r}\mathbf{x}_{i,l}^{(H-1)}\sigma'\left(\left(\mathbf{w}_r^{(H)}\right)^{\top}\mathbf{x}_{i,l}^{(H-1)}\right)\right]^{\top}\left[\sum_{k=1}^{p}a_{k,r}\mathbf{x}_{j,k}^{(H-1)}\sigma'\left(\left(\mathbf{w}_r^{(H)}\right)^{\top}\mathbf{x}_{j,k}^{(H-1)}\right)\right]. \tag{3.14}
$$

We define a constant $c_{\sigma,c_0}=\left(\min_{c_0\leq\alpha\leq1}\mathbb{E}_{X\sim N(0,1)}\sigma(\alpha X)^2\right)^{-1}>0$, where $0<c_0\leq1$. In particular, it is easy to see for smooth ReLU, $c_{\sigma,\frac{1}{\sqrt{p}}}=\mathrm{poly}(p)$.

Similar to Lemma 3.1, we can show with high probability the feature of each layer is approximately normalized.

Lemma 3.15 (Lemma on Initialization Norms)**.** *If $\sigma(\cdot)$ is $L-$Lipschitz and $m=\Omega\left(\frac{p^2n}{c_{\sigma,\frac{1}{\sqrt{p}}}^2\delta}\right)$, assuming $\left\|\mathbf{W}^{(h)}(0)\right\|_{\mathrm{op}}\leq c_{w,0}\sqrt{m}$ for $h\in[H]$, we have with probability at least $1-\delta$ over random initialization, for every $h\in[H]$ and $i\in[n]$,*

$$
\frac{1}{c_{x,0}}\leq\left\|\mathbf{x}_i^{(h)}(0)\right\|_F\leq c_{x,0}
$$

for some constant $c_{x,0}=poly(p)>1$.

The following lemma lower bounds $\mathbf{G}^{(H)}(0)$'s least eigenvalue. This lemma is a direct consequence of results in Section 3.12.

Lemma 3.16 (Least Eigenvalue at the Initialization)**.** *If $m=\Omega\left(\frac{n^2p^2\log(Hn/\delta)}{\lambda_0^2}\right)$, we have*

$$
\lambda_{\min}(\mathbf{G}^{(H)}(0))\geq\frac{3}{4}\lambda_0.
$$

Next, we prove the following lemma which characterizes how the perturbation from weight matrices propagates to the input of each layer.

Lemma 3.17. *Suppose $\sigma(\cdot)$ is $L-$Lipschitz and for $h \in [H]$, $\left\|\mathbf{W}^{(h)}(0)\right\|_{\text{op}} \leq c_{w,0}\sqrt{m}$, $\left\|\mathbf{x}^{(h)}(0)\right\|_F \leq c_{x,0}$ and $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F \leq \sqrt{m}R$ for some constant $c_{w,0}, c_{x,0} > 1$ and $R \leq c_{w,0}$. Then we have*

$$\left\|\mathbf{x}^{(h)}(k) - \mathbf{x}^{(h)}(0)\right\|_F \leq \left(\sqrt{c_\sigma}L\sqrt{q} + \frac{c_{x,0}}{c_{w,0}}\right) e^{2c_{w,0}L\sqrt{q}c_{res}}R.$$

Next, we show with high probability over random initialization, perturbation in weight matrices leads to small perturbation in the Gram matrix.

Lemma 3.18. *Suppose $\sigma(\cdot)$ is differentaible, $L-$Lipschitz and $\beta-$smooth. Using the same notations in Lemma 3.4, if $\|\mathbf{a}_{:,i}\|_2 \leq a_{2,0}\sqrt{m}$ and $\|\mathbf{a}_{:,i}\|_4 \leq a_{4,0}m^{1/4}$ for any $i \in [p]$, $\left\|\mathbf{W}^{(h)}(k) - \mathbf{W}^{(h)}(0)\right\|_F$ and $\|\mathbf{a}(k) - \mathbf{a}(0)\|_F \leq \sqrt{m}R$ where $R \leq c\lambda_0 H^2 (n)^{-1} poly(p)^{-1}$ for some small constant c, we have*

$$\left\|\mathbf{G}^{(H)}(k) - \mathbf{G}^{(H)}(0)\right\|_{\text{op}} \leq \frac{\lambda_0}{2}.$$

Lemma 3.19. *If Condition 3.3 holds for $k' = 1, \ldots, k$, we have for any $s \in [k+1]$*

$$\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(0)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(0)\|_F \leq R'\sqrt{m},$$
$$\left\|\mathbf{W}^{(h)}(s) - \mathbf{W}^{(h)}(s-1)\right\|_F, \|\mathbf{a}(s) - \mathbf{a}(s-1)\|_F \leq \eta Q'(s-1),$$

where $R' = \frac{16c_{res}c_{x,0}L\sqrt{pq}e^{2c_{res}c_{w,0}La_{2,0}\sqrt{q}}\sqrt{n}\|\mathbf{y}-\mathbf{u}(0)\|_2}{H\lambda_0\sqrt{m}} < c$ for some small constant c and

$$Q'(s) = 4c_{res}c_{x,0}La_{2,0}\sqrt{pq}e^{2c_{res}c_{w,0}L\sqrt{q}}\sqrt{n}\left\|\mathbf{y} - \mathbf{u}(s)\right\|_2 / H.$$

The follow lemma bounds the norm of $\mathbf{I}_2$.

Lemma 3.20. *If Condition 3.3 holds for $k' = 1, \ldots, k$ and $\eta \leq c\lambda_0 H^2 n^{-2} poly(1/p)$ for some small constant c, we have*

$$\left\|\mathbf{I}_2(k)\right\|_2 \leq \frac{1}{8}\eta\lambda_0\left\|\mathbf{y} - \mathbf{u}(k)\right\|_2.$$

Next we also bound the quadratic term.

Lemma 3.21. *If Condition 3.3 holds for $k' = 1, \ldots, k$ and $\eta \leq c\lambda_0 H^2 n^{-2} poly(1/p)$ for some small constant c, we have $\|\mathbf{u}(k+1) - \mathbf{u}(k)\|_2^2 \leq \frac{1}{8}\eta\lambda_0\|\mathbf{y} - \mathbf{u}(k)\|_2^2$.*

Now using the same argument as in the proof for multilayer fully connected neural network, we finish our proof for CNN.

3.11.1 Proofs of Lemmas

Proof of Lemma 3.15. We will bound $\left\|\mathbf{x}_i^{(h)}(0)\right\|_F$ layer by layer. For the first layer, we can calculate

$$\mathbb{E}\left[\left\|\mathbf{x}_i^{(1)}(0)\right\|_F^2\right] = c_\sigma \sum_{l=1}^{p_1} \mathbb{E}\left[\sigma\left(\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_{i,l}\right)^2\right]$$

$$\geq \frac{c_\sigma}{c_{\sigma,\frac{1}{\sqrt{p}}}},$$

where the inequality we use the definition of $c_{\sigma,\frac{1}{\sqrt{p}}}$ and the fact that there must exist $l' \in [p]$ such that $\|\mathbf{x}_{i,l'}\|_2^2 \geq \frac{1}{p_1} \geq \frac{1}{p}$. For the variance,

$$\begin{aligned}
\text{Var}\left[\left\|\mathbf{x}_i^{(1)}(0)\right\|_F^2\right] =& \frac{c_\sigma^2}{m}\text{Var}\left[\sum_{l=1}^{p_1}\sigma\left(\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_{i,l}\right)^2\right] \\
\leq& \frac{c_\sigma^2}{m}\mathbb{E}\left[\left(\sum_{l=1}^{p_1}\left(|\sigma(0)| + L\left|\mathbf{w}_r^{(1)}(0)^\top \mathbf{x}_{i,l}\right|\right)^2\right)^2\right] \\
\leq& \frac{p^2 C_2}{m},
\end{aligned}$$

where $C_2 \triangleq \sigma(0)^4 + 4|\sigma(0)|^3 L\sqrt{2/\pi} + 6\sigma(0)^2 L^2 + 8|\sigma(0)| L^3\sqrt{2/\pi} + 32L^4$. We have with probability at least $1 - \frac{\delta}{n}$,

$$\left\|\mathbf{x}_i^{(1)}(0)\right\|_F^2 \geq \frac{c_\sigma}{2c_{\sigma,\frac{1}{\sqrt{p}}}}.$$

It is easy to get its upper bound

$$\left\|\mathbf{x}_i^{(1)}(0)\right\|_F^2 = \frac{c_\sigma}{m}\left\|\sigma\left(\mathbf{W}^{(1)}\phi(\mathbf{x}_i)\right)\right\|_F^2 \leq qL^2 c_\sigma c_{w,0}^2.$$

By definition we have for $2 \leq h \leq H$

$$\begin{aligned}
\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_F - \left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\phi\left(\mathbf{x}_i^{(h-1)}(0)\right)\right)\right\|_F \leq& \left\|\mathbf{x}_i^{(h)}(0)\right\|_F \\
\leq& \left\|\mathbf{x}_i^{(h-1)}(0)\right\|_F + \left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\phi\left(\mathbf{x}_i^{(h-1)}(0)\right)\right)\right\|_F,
\end{aligned}$$

where

$$\left\|\frac{c_{res}}{H\sqrt{m}}\sigma\left(\mathbf{W}^{(h)}(0)\phi\left(\mathbf{x}_i^{(h-1)}(0)\right)\right)\right\|_F \leq \frac{\sqrt{q}c_{res}c_{w,0}L}{H}\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_F.$$

Thus

$$\left\|\mathbf{x}_i^{(h-1)}(0)\right\|_F\left(1 - \frac{\sqrt{q}c_{res}c_{w,0}L}{H}\right) \leq \left\|\mathbf{x}^{(h)}(0)\right\|_F \leq \left\|\mathbf{x}_i^{(h-1)}(0)\right\|_F\left(1 + \frac{\sqrt{q}c_{res}c_{w,0}L}{H}\right),$$

which implies

$$\sqrt{\frac{c_\sigma}{2c_{\sigma,\frac{1}{\sqrt{p}}}}}e^{-\sqrt{q}c_{res}c_{w,0}L} \leq \left\|\mathbf{x}^{(h)}(0)\right\|_F \leq \sqrt{qL^2 c_\sigma c_{w,0}^2}e^{\sqrt{q}c_{res}c_{w,0}L}.$$

Choosing $c_{x,0} = \max\{\sqrt{qL^2 c_\sigma c_{w,0}^2}, \sqrt{\frac{2c_{\sigma,\frac{1}{\sqrt{p}}}}{c_\sigma}}\}e^{\sqrt{q}c_{res}c_{w,0}L}$ and using union bounds over $[n]$, we prove the lemma.

□

Proof of Lemma 3.17. We prove this lemma by induction. Our induction hypothesis is

$$\left\|\mathbf{x}^{(h)}(k)-\mathbf{x}^{(h)}(0)\right\|_F \leq g(h),$$

where

$$g(h)=g(h-1)\left[1+\frac{2c_{res}c_{w,0}L\sqrt{q}}{H}\right]+\frac{c_{res}L\sqrt{q}}{H}Rc_{x,0}.$$

For $h=1$, we have

$$\begin{aligned}\left\|\mathbf{x}^{(1)}(k)-\mathbf{x}^{(1)}(0)\right\|_F \leq & \sqrt{\frac{c_\sigma}{m}}\left\|\sigma\left(\mathbf{W}^{(1)}(k)\phi_1(\mathbf{x})\right)-\sigma\left(\mathbf{W}^{(1)}(0)\phi_1(\mathbf{x})\right)\right\|_F \\ \leq & \sqrt{\frac{c_\sigma}{m}}L\sqrt{q}\left\|\mathbf{W}^{(1)}(k)-\mathbf{W}^{(1)}(0)\right\|_F \leq \sqrt{c_\sigma}L\sqrt{q}R,\end{aligned}$$

which implies $g(1)=\sqrt{c_\sigma}L\sqrt{q}R$, for $2\leq h\leq H$, we have

$$\begin{aligned}&\left\|\mathbf{x}^{(h)}(k)-\mathbf{x}^{(h)}(0)\right\|_F \\ \leq & \frac{c_{res}}{H\sqrt{m}}\left\|\sigma\left(\mathbf{W}^{(h)}(k)\phi_h\left(\mathbf{x}^{(h-1)}(k)\right)\right)-\sigma\left(\mathbf{W}^{(h)}(0)\phi_h\left(\mathbf{x}^{(h-1)}(0)\right)\right)\right\|_F+\left\|\mathbf{x}^{(h-1)}(k)-\mathbf{x}^{(h-1)}(0)\right\|_F \\ \leq & \frac{c_{res}}{H\sqrt{m}}\left\|\sigma\left(\mathbf{W}^{(h)}(k)\phi_h\left(\mathbf{x}^{(h-1)}(k)\right)\right)-\sigma\left(\mathbf{W}^{(h)}(k)\phi_h\left(\mathbf{x}^{(h-1)}(0)\right)\right)\right\|_F \\ &+\frac{c_{res}}{H\sqrt{m}}\left\|\sigma\left(\mathbf{W}^{(h)}(k)\phi_h\left(\mathbf{x}^{(h-1)}(0)\right)\right)-\sigma\left(\mathbf{W}^{(h)}(0)\phi_h\left(\mathbf{x}^{(h-1)}(0)\right)\right)\right\|_F \\ &+\left\|\mathbf{x}^{(h-1)}(k)-\mathbf{x}^{(h-1)}(0)\right\|_F \\ \leq & \frac{L\sqrt{q}c_{res}}{H\sqrt{m}}\left(\left\|\mathbf{W}^{(h)}(0)\right\|_2+\left\|\mathbf{W}^{(h)}(k)-\mathbf{W}^{(h)}(0)\right\|_F\right)\cdot\left\|\mathbf{x}^{(h-1)}(k)-\mathbf{x}^{(h-1)}(0)\right\|_F \\ &+\frac{L\sqrt{q}c_{res}}{H\sqrt{m}}\left\|\mathbf{W}^{(h)}(k)-\mathbf{W}^{(h)}(0)\right\|_F\left\|\mathbf{x}^{h-1}(0)\right\|_F+\left\|\mathbf{x}^{(h-1)}(k)-\mathbf{x}^{(h-1)}(0)\right\|_F \\ \leq & \left[1+\frac{L\sqrt{q}c_{res}}{H\sqrt{m}}\left(c_{w,0}\sqrt{m}+R\sqrt{m}\right)\right]g(h-1)+\frac{L\sqrt{q}c_{res}}{H\sqrt{m}}\sqrt{m}Rc_{x,0} \\ \leq & \left(1+\frac{2c_{w,0}L\sqrt{q}c_{res}}{H}\right)g(h-1)+\frac{1}{H}L\sqrt{q}c_{res}c_{x,0}R.\end{aligned}$$

Lastly, simple calculations show $g(h)\leq\left(\sqrt{c_\sigma}L\sqrt{q}+\frac{c_{x,0}}{c_{w,0}}\right)e^{2c_{w,0}L\sqrt{q}c_{res}}R$.

□

Proof of Lemma 3.18. Similar to Lemma 3.11, define $z_{i,l,r}=\left(\mathbf{w}_r^{(H)}\right)^\top\mathbf{x}_{i,l}^{(H-1)}$, we have

$$\begin{aligned}&\left|\mathbf{G}_{i,j}^{(H)}(k)-\mathbf{G}_{i,j}^{(H)}(0)\right| \\ =&\frac{c_{res}^2}{H^2}\Big|\sum_{l=1}^p\sum_{k=1}^p\mathbf{x}_{i,l}^{(H-1)}(k)^\top\mathbf{x}_{j,k}^{(H-1)}(k)\frac{1}{m}\sum_{r=1}^m a_{r,l}(k)a_{r,k}(k)\sigma'\left(z_{i,l,r}(k)\right)\sigma'\left(z_{j,k,r}(k)\right)\end{aligned}$$

$$
\begin{aligned}
&- \sum_{l=1}^{p}\sum_{k=1}^{p} \mathbf{x}_{i,l}^{(H-1)}(0)^\top \mathbf{x}_{j,k}^{(H-1)}(0) \frac{1}{m}\sum_{r=1}^{m} a_{r,l}(0)a_{r,k}(0)\sigma'\left(z_{i,l,r}(0)\right)\sigma'\left(z_{j,k,r}(0)\right)\Big| \\
\leq & \frac{c_{res}^2 L^2 a_{2,0}^2}{H^2}\left|\sum_{l=1}^{p}\sum_{k=1}^{p} \mathbf{x}_{i,l}^{(H-1)}(k)^\top \mathbf{x}_{j,k}^{(H-1)}(k) - \mathbf{x}_{i,l}^{(H-1)}(0)^\top \mathbf{x}_{j,k}^{(H-1)}(0)\right| \\
&+ \frac{c_{res}^2}{H^2}\sum_{l=1}^{p}\sum_{k=1}^{p}\left|\mathbf{x}_{i,l}^{(H-1)}(0)^\top \mathbf{x}_{j,k}^{(H-1)}(0)\right| \\
&\cdot \left(\frac{1}{m}\sum_{r=1}^{m}\left|a_{r,l}(0)a_{r,k}(0)\right|\left|\sigma'\left(z_{i,l,r}(k)\right)\sigma'\left(z_{j,k,r}(k)\right) - \sigma'\left(z_{i,l,r}(0)\right)\sigma'\left(z_{j,k,r}(0)\right)\right|\right) \\
&+ \frac{c_{res}^2}{H^2}L^2\sum_{l=1}^{p}\sum_{k=1}^{p}\left|\mathbf{x}_{i,l}^{(H-1)}(k)^\top \mathbf{x}_{j,k}^{(H-1)}(k)\right|\frac{1}{m}\sum_{r=1}^{m}\left|a_{r,l}(k)a_{r,k}(k) - a_{r,l}(0)a_{r,k}(0)\right| \\
\triangleq & \frac{c_{res}^2}{H^2}\left(I_1^{i,j} + I_2^{i,j} + I_3^{i,j}\right).
\end{aligned}
$$

For $I_1^{i,j}$, using Lemma 3.17, we have

$$
\begin{aligned}
I_1^{i,j} =& L^2 a_{2,0}^2\left|\sum_{l=1}^{p}\sum_{k=1}^{p} \mathbf{x}_{i,l}^{(H-1)}(k)^\top \mathbf{x}_{j,k}^{(H-1)}(k) - \mathbf{x}_{i,l}^{(H-1)}(0)^\top \mathbf{x}_{j,k}^{(H-1)}(0)\right| \\
\leq & L^2 a_{2,0}^2\sum_{l=1}^{p}\sum_{k=1}^{p}\left|\left(\mathbf{x}_{i,l}^{(H-1)}(k) - \mathbf{x}_{i,l}^{(H-1)}(0)\right)^\top \mathbf{x}_{j,k}^{(H-1)}(k)\right| \\
&+ L^2 a_{2,0}^2\sum_{l=1}^{p}\sum_{k=1}^{p}\left|\mathbf{x}_{i,l}^{(H-1)}(0)^\top\left(\mathbf{x}_{j,k}^{(H-1)}(k) - \mathbf{x}_{j,k}^{(H-1)}(0)\right)\right| \\
\leq & L^2 a_{2,0}^2\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{x}_{i,l}^{(H-1)}(k) - \mathbf{x}_{i,l}^{(H-1)}(0)\right\|_2^2}\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{x}_{j,k}^{(H-1)}(k)\right\|_2^2} \\
&+ L^2 a_{2,0}^2\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{x}_{i,l}^{(H-1)}(0)\right\|_2^2}\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{x}_{j,k}^{(H-1)}(k) - \mathbf{x}_{j,k}^{(H-1)}(0)\right\|_2^2} \\
\leq & L^2 a_{2,0}^2 p\left\|\mathbf{x}_i^{(H-1)}(k) - \mathbf{x}_i^{(H-1)}(0)\right\|_F\left\|\mathbf{x}_j^{(H-1)}(k)\right\|_F \\
&+ L^2 a_{2,0}^2 p\left\|\mathbf{x}_i^{(H-1)}(0)\right\|_F\left\|\mathbf{x}_j^{(H-1)}(k) - \mathbf{x}_j^{(H-1)}(0)\right\|_F \\
\leq & 3c_{x,0}c_x L^2 a_{2,0}^2 pR,
\end{aligned}
$$

where $c_x \triangleq \left(\sqrt{c_\sigma}L\sqrt{q} + \frac{c_{x,0}}{c_{w,0}}\right)e^{2c_{res}c_{w,0}L\sqrt{q}}$. To bound $I_2^{i,j}$, we have

$$
\begin{aligned}
& I_2^{i,j} \\
=& \sum_{l=1}^{p}\sum_{k=1}^{p}\left|\mathbf{x}_{i,l}^{(H-1)}(0)^\top \mathbf{x}_{j,k}^{(H-1)}(0)\right|
\end{aligned}
$$

$$
\begin{aligned}
&\cdot \left(\frac{1}{m} \sum_{r=1}^{m} |a_{r,l}(0) a_{r,k}(0)| \left| \sigma'\left(z_{i,l,r}(k)\right) \sigma'\left(z_{j,k,r}(k)\right) - \sigma'\left(z_{i,l,r}(0)\right) \sigma'\left(z_{j,k,r}(0)\right) \right| \right) \\
\leq & \sum_{l=1}^{p} \sum_{k=1}^{p} \left| \mathbf{x}_{i,l}^{(H-1)}(0)^{\top} \mathbf{x}_{j,k}^{(H-1)}(0) \right| \\
&\cdot \left(\frac{\beta L}{m} \left(\sum_{r=1}^{m} |a_{r,l}(0) a_{r,k}(0)| \left(|z_{i,l,r}(k) - z_{i,l,r}(0)| + |z_{j,k,r}(k) - z_{j,k,r}(0)| \right) \right) \right) \\
\leq & \frac{\beta L}{m} \sqrt{\sum_{l=1}^{p} \sum_{k=1}^{p} \left\| \mathbf{x}_{i,l}^{(H-1)}(0) \right\|_2^2 \left\| \mathbf{x}_{j,k}^{(H-1)}(0) \right\|_2^2} \\
&\left(\sqrt{\sum_{l=1}^{p} \sum_{k=1}^{p} \left(\sum_{r=1}^{m} |a_{r,l}(0) a_{r,k}(0)| \left| z_{i,l,r}(k) - z_{i,l,r}(0) \right| \right)^2} \right. \\
&\left. + \sqrt{\sum_{l=1}^{p} \sum_{k=1}^{p} \left(\sum_{r=1}^{m} |a_{r,l}(0) a_{r,k}(0)| \left| z_{j,k,r}(k) - z_{j,k,r}(0) \right| \right)^2} \right) \\
\leq & \frac{\beta L c_{x,0}^2 a_{4,0}^2}{m} \left(\sqrt{m \sum_{l=1}^{p} \sum_{k=1}^{p} \sum_{r=1}^{m} \left| z_{i,l,r}(k) - z_{i,l,r}(0) \right|^2} + \sqrt{m \sum_{l=1}^{p} \sum_{k=1}^{p} \sum_{r=1}^{m} \left| z_{j,k,r}(k) - z_{j,k,r}(0) \right|^2} \right) \\
\leq & \frac{\beta L a_{4,0}^2 \sqrt{p} c_{x,0}^2}{\sqrt{m}} \left(\|\mathbf{z}_i\|_F + \|\mathbf{z}_j\|_F \right).
\end{aligned}
$$

Using the same proof for Lemma 3.17, it is easy to see

$$
\|\mathbf{z}_i\|_F \leq \left(2 c_x c_{w,0} \sqrt{q} + c_{x,0}\right) R \sqrt{m}.
$$

Thus

$$
I_2^{i,j} \leq 2 \beta L a_{4,0}^2 \sqrt{p} c_{x,0}^2 \left(2 c_x c_{w,0} \sqrt{q} + c_{x,0}\right) R.
$$

Similarly for $I_3^{i,j}$, we have

$$
\begin{aligned}
I_3^{i,j} = & \frac{c_{res}^2}{H^2} L^2 \sum_{l=1}^{p} \sum_{k=1}^{p} \left| \mathbf{x}_{i,l}^{(H-1)}(k)^{\top} \mathbf{x}_{j,k}^{(H-1)}(k) \right| \frac{1}{m} \sum_{r=1}^{m} |a_{r,l}(k) a_{r,k}(k) - a_{r,l}(0) a_{r,k}(0)| \\
\leq & \frac{c_{res}^2}{H^2} L^2 \sum_{l=1}^{p} \sum_{k=1}^{p} \left| \mathbf{x}_{i,l}^{(H-1)}(k)^{\top} \mathbf{x}_{j,k}^{(H-1)}(k) \right| \\
&\cdot \left(\frac{1}{m} \sum_{r=1}^{m} \left(|a_{r,l}(k) - a_{r,l}(0)| \, |a_{r,k}(k)| + |a_{r,k}(k) - a_{r,k}(0)| \, |a_{r,l}(0)| \right) \right) \\
\leq & \frac{c_{res}^2}{H^2} L^2 \sum_{l=1}^{p} \sum_{k=1}^{p} \left| \mathbf{x}_{i,l}^{(H-1)}(k)^{\top} \mathbf{x}_{j,k}^{(H-1)}(k) \right|
\end{aligned}
$$

$$
\begin{aligned}
&\cdot\left(\frac{1}{m}\left(\left\|\mathbf{a}_{:,l}(k)-\mathbf{a}_{:,l}(0)\right\|_2\left\|\mathbf{a}_{:,k}(k)\right\|_2+\left\|\mathbf{a}_{:,k}(k)-\mathbf{a}_{:,k}(0)\right\|_2\left\|\mathbf{a}_{:,l}(0)\right\|_2\right)\right)\\
\leq&\frac{c_{res}^2}{H^2m}L^2\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{x}_{i,l}^{(H-1)}(k)\right\|_2^2\left\|\mathbf{x}_{j,k}^{(H-1)}(k)\right\|_2^2}\\
&\left(\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{a}_{:,l}(k)-\mathbf{a}_{:,l}(0)\right\|_2^2\left\|\mathbf{a}_{:,k}(k)\right\|_2^2}+\sqrt{\sum_{l=1}^{p}\sum_{k=1}^{p}\left\|\mathbf{a}_{:,k}(k)-\mathbf{a}_{:,k}(0)\right\|_2^2\left\|\mathbf{a}_{:,l}(0)\right\|_2^2}\right)\\
\leq&\frac{c_{res}^2}{H^2m}L^2\left\|\mathbf{x}_i^{(H-1)}(k)\right\|_F\left\|\mathbf{x}_j^{(H-1)}(k)\right\|_F\left(\left\|\mathbf{a}(k)-\mathbf{a}(0)\right\|_F\left\|\mathbf{a}(k)\right\|_F+\left(\left\|\mathbf{a}(k)-\mathbf{a}(0)\right\|_F\left\|\mathbf{a}(0)\right\|_F\right)\right.\\
\leq&\frac{12a_{2,0}c_{res}^2c_{x,0}^2L^2\sqrt{p}R}{H^2}.
\end{aligned}
$$

Therefore we can bound the perturbation

$$
\begin{aligned}
&\left\|\mathbf{G}^{(H)}(k)-\mathbf{G}^{(H)}(0)\right\|_2\\
\leq&\left\|\mathbf{G}^{(H)}(k)-\mathbf{G}^{(H)}(0)\right\|_F\\
=&\sqrt{\sum_{(i,j)}^{n,n}\left|\mathbf{G}_{i,j}^{(H)}(k)-\mathbf{G}_{i,j}^{(H)}(0)\right|^2}\\
\leq&\frac{c_{res}^2}{H^2}\left[3c_{x,0}c_xLa_{2,0}^2p+2\beta c_{x,0}^2a_{4,0}^2\sqrt{p}\left(2c_xc_{w,0}\sqrt{q}+c_{x,0}\right)+12c_{x,0}^2La_{2,0}\sqrt{p}\right]LnR.
\end{aligned}
$$

Plugging in the bound on R, we have the desired result. □

Proof of Lemma 3.19. We will prove this corollary by induction. The induction hypothesis is

$$
\begin{aligned}
\left\|\mathbf{W}^{(h)}(s)-\mathbf{W}^{(h)}(0)\right\|_F\leq\sum_{s'=0}^{s-1}(1-\frac{\eta\lambda_0}{2})^{s'/2}\frac{1}{4}\eta\lambda_0R'\sqrt{m}\leq R'\sqrt{m},s\in[k+1],\\
\left\|\mathbf{a}(s)-\mathbf{a}(0)\right\|_F\leq\sum_{s'=0}^{s-1}(1-\frac{\eta\lambda_0}{2})^{s'/2}\frac{1}{4}\eta\lambda_0R'\sqrt{m}\leq R'\sqrt{m},s\in[k+1].
\end{aligned}
$$

First it is easy to see it holds for $s'=0$. Now suppose it holds for $s'=0,\ldots,s$, we consider $s'=s+1$. Similar to Lemma 3.5, we have

$$
\begin{aligned}
&\left\|\mathbf{W}^{(h)}(s+1)-\mathbf{W}^{(h)}(s)\right\|_F\\
\leq&\eta\frac{c_{res}L}{H\sqrt{m}}\left\|\mathbf{a}\right\|_F\sum_{i=1}^{n}\left|y_i-u(s)\right|\left\|\phi_h\left(\mathbf{x}^{(h-1)}(s)\right)\right\|_F\prod_{k=h+1}^{H}\left\|\mathbf{I}+\frac{c_{res}}{H\sqrt{m}}\mathbf{W}^{(k)}(s)\phi_k\right\|_{op}\\
\leq&2\eta c_{res}c_{x,0}La_{2,0}\sqrt{pq}e^{2c_{res}c_{w,0}L\sqrt{q}}\sqrt{n}\left\|\mathbf{y}-\mathbf{u}(s)\right\|_2/H\\
=&\eta Q'(s)\\
\leq&(1-\frac{\eta\lambda_0}{2})^{s/2}\frac{1}{4}\eta\lambda_0R'\sqrt{m},
\end{aligned}
$$

where $\|\cdot\|_{op}$ denotes the operator norm. Similarly, we have

$$\begin{aligned}\|\mathbf{a}(s+1)-\mathbf{a}(s)\|_2 \leq & 2\eta c_{x,0}\sum_{i=1}^{n}|y_i-u(s)| \\ \leq & \eta Q'(s) \\ \leq & (1-\frac{\eta\lambda_0}{2})^{s/2}\frac{1}{4}\eta\lambda_0 R'\sqrt{m}.\end{aligned}$$

Thus

$$\begin{aligned}&\left\|\mathbf{W}^{(h)}(s+1)-\mathbf{W}^{(h)}(0)\right\|_F \\ \leq &\left\|\mathbf{W}^{(h)}(s+1)-\mathbf{W}^{(h)}(s)\right\|_F+\left\|\mathbf{W}^{(h)}(s)-\mathbf{W}^{(h)}(0)\right\|_F \\ \leq &\sum_{s'=0}^{s}\eta(1-\frac{\eta\lambda_0}{2})^{s'/2}\frac{1}{4}\eta\lambda_0 R'\sqrt{m}.\end{aligned}$$

Similarly,

$$\begin{aligned}&\|\mathbf{a}(s+1)-\mathbf{a}(0)\|_2 \\ \leq &\sum_{s'=0}^{s}\eta(1-\frac{\eta\lambda_0}{2})^{s'/2}\frac{1}{4}\eta\lambda_0 R'\sqrt{m}.\end{aligned}$$

□

Proof of Lemma 3.20.

$$\left|I_2^i\right| \leq \eta\max_{0\leq s\leq\eta}\sum_{h=1}^{H}\left\|L'^{(h)}(\theta(k))\right\|_F\left\|u_i'^{(h)}\left(\theta(k)\right)-u_i'^{(h)}\left(\theta(k)-sL'^{(h)}(\theta(k))\right)\right\|_F.$$

For the gradient norm, we have

$$\begin{aligned}&\left\|L'^{(h)}(\theta(k))\right\|_F \\ \leq &\frac{Lc_{res}}{H\sqrt{m}}\|\mathbf{a}(k)\|_F\sum_{i=1}^{n}|y_i-u_i(k)|\left\|\phi_h\left(\mathbf{x}_i^{(h-1)}(k)\right)\right\|_F\prod_{k=h+1}^{H}\left\|\mathbf{I}+\frac{c_{res}}{H\sqrt{m}}\mathbf{J}_i^{(k)}(k)\mathbf{W}^{(k)}(k)\phi_k\right\|_{op},\end{aligned}$$

which we have bounded in Lemma 3.19, thus

$$\left\|L'^{(h)}(\theta(k))\right\|_F \leq Q'(k).$$

Let $\theta(k,s)=\theta(k)-sL'(\theta(k))$. Similar to the proof of Lemma 3.6, we have

$$\left\|u_i'^{(h)}\left(\theta(k)\right)-u_i'^{(h)}\left(\theta(k,s)\right)\right\|_F$$

$$
\begin{aligned}
\leq & \frac{2}{H} c_{res} c_{x,0} L a_{2,0} \sqrt{q} e^{2 c_{res} L c_{w,0} \sqrt{q}} \eta \frac{Q'(k)}{\sqrt{m}} \\
& \cdot \left(\frac{c_x}{c_{x,0}} + \frac{2}{L} \left(c_{x,0} + c_{w,0} c_x \right) \beta \sqrt{m} + 4 \sqrt{q} c_{w,0} \left(c_{x,0} + c_{w,0} c_x \right) \beta \sqrt{m} + (L+1) \sqrt{q} \right) \\
\leq & \frac{24}{H} c_{res} c_{x,0} L a_{2,0} \sqrt{q} c_{w,0} e^{2 c_{res} L c_{w,0} \sqrt{q}} \left(c_{x,0} + c_{w,0} c_x \right) \beta \eta Q'(k).
\end{aligned}
$$

Thus

$$
\left| I_2^i \right| \leq 24 c_{res} c_{x,0} L a_{2,0} \sqrt{q} c_{w,0} e^{2 c_{res} L c_{w,0}} \left(c_{x,0} + c_{w,0} c_x \right) \beta \eta^2 \lambda_0 \sqrt{m} Q'(k) R' \leq \frac{1}{8} \eta \lambda_0 \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2 .
$$

where we used the bound of η and that $\left\| \mathbf{y} - \mathbf{u}(0) \right\|_2 = O(\sqrt{n})$. □

Proof of Lemma 3.21.

$$
\begin{aligned}
\left\| \mathbf{u}(k+1) - \mathbf{u}(k) \right\|_2^2 = & \sum_{i=1}^{n} \left(\langle \mathbf{a}(k+1), \mathbf{x}_i^{(H)}(k+1) \rangle - \langle \mathbf{a}(k), \mathbf{x}_i^{(H)}(k+1) \rangle \right)^2 \\
\leq & \sum_{i=1}^{n} \left(\langle \mathbf{a}(k+1) - \mathbf{a}(k), \mathbf{x}_i^{(H)}(k+1) \rangle + \langle \mathbf{a}(k), \mathbf{x}_i^{(H)}(k+1) - \mathbf{x}_i^{(H)}(k) \rangle \right)^2 \\
\leq & 2 \left\| \mathbf{a}(k+1) - \mathbf{a}(k) \right\|_F^2 \sum_{i=1}^{n} \left\| \mathbf{x}_i^{(H)}(k+1) \right\|_F^2 \\
& + 2 \left\| \mathbf{a}(k) \right\|_F^2 \sum_{i=1}^{n} \left\| \mathbf{x}_i^{(H)}(k+1) - \mathbf{x}_i^{(H)}(k) \right\|_F^2 \\
\leq & 8 n \eta^2 c_{x,0}^2 Q'(k)^2 + 4 n p \left(\eta a_{2,0} c_x Q'(k) \right)^2 \\
\leq & \frac{1}{8} \eta \lambda_0 \left\| \mathbf{y} - \mathbf{u}(k) \right\|_2^2 .
\end{aligned}
$$

□

3.12 Analysis of Random Initialization

3.12.1 A General Framework for Analyzing Random Initialization in First $(H-1)$ Layers

In this section we provide a self-contained framework to analyze the Gram matrix at the initialization phase. There are two main objectives. First, we provide the expression of the Gram matrix as $m \to \infty$, i.e., the population Gram matrix. Second, we quantitatively study how much over-parameterization is needed to ensure the Gram matrix generated by the random initialization. The bound will depend on number of samples n and properties of the activation function. This analysis framework is fully general that it can explain fully connected neural network, ResNet, convolutional neural considered in this chapter and other neural network architectures that satisfy the general setup defined below.

We begin with some notations. Suppose that we have a sequence of real vector spaces

$$\mathbb{R}^{p^{(0)}} \to \mathbb{R}^{p^{(1)}} \to \cdots \to \mathbb{R}^{p^{(H)}}.$$

Remark 3.1. *For fully-connected neural network and ResNet,* $p^{(0)} = p^{(1)} = \ldots = p^{(H)} = 1$. *For convolutional neural network,* $p^{(h)}$ *is the number of patches of the* h*-th layer.*

For each pair $(\mathbb{R}^{p^{(h-1)}}, \mathbb{R}^{p^{(h)}})$, let $\mathcal{W} \subset \mathcal{L}(\mathbb{R}^{p^{(h-1)}}, \mathbb{R}^{p^{(h)}}) = \mathbb{R}^{p^{(h)} \times p^{(h-1)}}$ be a linear subspace.

Remark 3.2. *For convolutional neural network, the dimension of* $\mathcal{W}$ *is the filter size.*

In this section, by Gaussian distribution $\mathcal{G}$ over a q-dimensional subspace $\mathcal{W}$, we mean that for a basis $\{\mathbf{e}_1, \ldots, \mathbf{e}_q\}$ of $\mathcal{W}$ and $(v_1, \ldots, v_q) \sim \mathcal{N}\mathbf{0}, \mathbf{I})$ such that $\sum_{i=1}^q v_i \mathbf{e}_i \sim \mathcal{G}$. In this section, we equip one Gaussian distribution $\mathcal{G}^{(h)}$ with each linear subspace $\mathcal{W}^{(h)}$. By an abuse of notation, we also use $\mathcal{W}$ to denote a transformation. For $\mathbf{K} \in \mathbb{R}^{p^{(h-1)} \times p^{(h-1)}}$, we let

$$\mathcal{W}^{(h)}(\mathbf{K}) = \mathbb{E}_{\mathbf{W} \sim \mathcal{G}^{(h)}} \left[\mathbf{W}\mathbf{K}\mathbf{W}^\top \right].$$

We also consider a deterministic linear mapping $\mathcal{D}^{(h)} : \mathbb{R}^{n^{(h-1)}} \to \mathbb{R}^{n^{(h)}}$. For this section, we denote $\mathcal{D}^{(1)} = \mathbf{0}$, i.e., the zero mapping.

Remark 3.3. *For full-connected neural networks, we take* $\mathcal{D}^{(h)}$ *to be zero. For ResNet and convolutional ResNet, we take* $\mathcal{D}^{(h)}$ *to be the identity mapping.*

Let $\rho^{(1)}, \cdots, \rho^{(H)}$ be a sequence of activation functions over $\mathbb{R}$. Note here we use ρ instead of σ to denote the activation function because we will incorporate the scaling in ρ for the ease of presentation and the full generality.

Now we recursively define the output of each layer in this setup. In the following, we use $h \in [H]$ to index layers, $i \in [n]$ to index data points, $\alpha, \beta, \gamma \in [m]$ or $[d]$ to index channels (for CNN) or weight vectors (for fully connected neural networks or ResNet).

Remark 3.4. $d = 1$ *for fully connected neural network and ResNet and* $d \geq 1$ *for convolutional neural network because* d *represents the number of input channels.*

We denote $\mathbf{X}_i^{(h),[\alpha]}$ an $p^{(h)}$-dimensional vector which is the output at $(h-1)$-th layer. We have the following recursive formula

$$\begin{aligned}
\mathbf{X}_i^{(1),(\alpha)} =& \rho^{(h)} \left(\sum_\beta \mathbf{W}_{(\beta)}^{(h),(\alpha)} \mathbf{X}_i^{(h-1),(\beta)} \right) \\
\mathbf{X}_i^{(h),(\alpha)} =& \mathcal{D}^{(h)}(\mathbf{X}_i^{(h-1),(\alpha)}) + \rho^{(h)} \left(\frac{\sum_\beta \mathbf{W}_{(\beta)}^{(h),(\alpha)} \mathbf{X}_i^{(h-1),(\beta)}}{\sqrt{m}} \right)
\end{aligned}$$

where $\mathbf{W}_{(\beta)}^{(h),(\alpha)}$ is $p^{(h)} \times p^{(h-1)}$ matrix generated according to the following rule

- for $h = 1$, $\mathbf{W}_{[\beta]}^{(h),(\alpha)}$ is defined for $1 \leq \alpha \leq m$ and $1 \leq \beta \leq d$; for $h > 1$, $\mathbf{W}_{(\beta)}^{(h),(\alpha)}$ is defined for $1 \leq \alpha \leq m$ and $1 \leq \beta \leq m$;
- the set of random variables $\{\mathbf{W}_{(\beta)}^{(h),(\alpha)}\}_{h,\alpha,\beta}$ are independently generated;
- for fixed h, α, β, $\mathbf{W}_{(\beta)}^{(h),(\alpha)} \sim \mathcal{G}^{(h)}$.

Remark 3.5. *Choosing $\rho^{(h)}(z)$ to be $\sigma(z)$ and $\mathcal{D}^{(h)}$ to be the zero mapping, we recover the fully-connected architecture. Choosing $\rho^{(h)}(z)$ to be $\frac{c_{res}}{H}\sigma(z)$ and $\mathcal{D}^{(h)}$ to be the identity mapping, we recover ResNet architecture.*

Remark 3.6. *Note here $\mathbf{X}_i^{(h)} = \mathbf{x}_i^{(h)}\sqrt{m}$ for $h \geq 1$ and $\mathbf{X}_i^{(h)} = \mathbf{x}_i^{(h)}$ for $h = 0$ in the main text. We change the scaling here to simplify the calculation of expectation and the covariance in this section.*

With these notations, we first define the population Gram matrices recursively.

Definition 3.4. *We fix $(i,j) \in [n] \times [n]$, for $h = 1, \ldots, H$. The population Gram matrices are defined according to the following formula*

$$\begin{aligned}
\mathbf{K}_{ij}^{(0)} =& \sum_{\gamma} (\mathbf{X}_i^{(0),[\gamma]})^\top \mathbf{X}_j^{(0),[\gamma]}, \\
\mathbf{b}_i^{(0)} =& \mathbf{0}, \\
\mathbf{K}_{ij}^{(h)} =& \mathcal{D}^{(h)} \mathbf{K}_{ij}^{(h-1)} \mathcal{D}^{(h)\top} + \mathbb{E}_{(\mathbf{U},\mathbf{V})} \left(\rho(\mathbf{U}) \mathcal{D}^{(h)} (\mathbf{b}_j^{(h-1)})^\top + (\mathcal{D}^{(h)}(\mathbf{b}_i^{(h-1)})) \rho(\mathbf{V})^\top + \rho(\mathbf{U})\rho(\mathbf{V})^\top \right), \\
\mathbf{b}_i^{(h)} =& \mathcal{D}^{(h)}(\mathbf{b}_i^{(h-1)}) + \mathbb{E}_{\mathbf{U}} \rho^{(h)}(\mathbf{U}),
\end{aligned} \tag{3.15}$$

where

$$(\mathbf{U}, \mathbf{V}) \sim N\left(\mathbf{0}, \begin{pmatrix} \mathcal{W}\left(\mathbf{K}_{ii}^{(h-1)}\right) & \mathcal{W}\left(\mathbf{K}_{ij}^{(h-1)}\right) \\ \mathcal{W}\left(\mathbf{K}_{ji}^{(h-1)}\right) & \mathcal{W}\left(\mathbf{K}_{jj}^{(h-1)}\right) \end{pmatrix} \right). \tag{3.16}$$

Notice that the Gram matrix of the next layer $\mathbf{K}^{(h)}$ not only depends on the previous layer's Gram matrix $\mathbf{K}^{(h-1)}$ but also depends on a "bias" term $\mathbf{b}^{(h-1)}$.

Given the population Gram matrices defined in Equation (3.15) and (3.16), we derive the following quantitative bounds which characterizes how much over-parameterization, i.e., how large m is needed to ensure the randomly generated Gram matrices is close to the population Gram matrices.

Theorem 3.4. *With probability $1-\delta$ over the $\left\{\mathbf{W}_{(\beta)}^{(h),(\alpha)}\right\}_{h,\alpha,\beta}$, for any $1 \leq h \leq H-1, 1 \leq i,j \leq n$,*

$$\left\| \frac{1}{m} \sum_{\alpha=1}^{m} (\mathbf{X}_i^{(h),(\alpha)})^\top \mathbf{X}_j^{(h),(\alpha)} - \mathbf{K}_{ij}^{(h)} \right\|_\infty \leq \mathcal{E} \sqrt{\frac{\log(Hn \max_h p^{(h)}/\delta)}{m}} \tag{3.17}$$

and any $h \in [H-1], \forall 1 \leq i \leq n$,

$$\left\| \frac{1}{m} \sum_{\alpha=1}^{m} \mathbf{X}_i^{(h),(\alpha)} - \mathbf{b}_i^{(h)} \right\|_\infty \leq \mathcal{E} \sqrt{\frac{\log(Hn \max_h p^{(h)}/\delta)}{m}} \tag{3.18}$$

The error constant $\mathcal{E}$ satisfies there exists an absolute constant $C > 0$ such that

$$\begin{aligned}
\mathcal{E} \leq & C \left(\prod_{h=2}^{H-1} \left(A_{(h)} + \Lambda_{(h)} \mathfrak{W} + C_{(h)} A_{(h)} B \mathfrak{W} + C_{(h)} A_{(h)} \sqrt{\mathfrak{W}_{(h)} M} \right) \right) \\
& \cdot \max\{ \mathfrak{W} \sqrt{(1 + C_{(1)}^2) M^2}, \sqrt{C_{(1)}^2 M} \}
\end{aligned}$$

where $M, B, \Lambda_{(h)}, C_{(h)}, A_{(h)}, \mathfrak{W}_{(h)}$ are defined by:

- $M = 1 + 100 \max_{i,j,p,q,h} |\mathcal{W}^{(h)}(\mathbf{K}_{ij}^{(h-1)})_{pq}|$,
- $A_{(h)} = 1 + \max\left\{\|\mathcal{D}^{(h)}\|_{L^\infty \to L^\infty}, \|\mathcal{D}^{(h)}(\cdot)\mathcal{D}^{(h)\top}\|_{L^\infty \to L^\infty}\right\}$,
- $B = 1 + 100 \max_{i,p,h} |\mathbf{b}_{ip}^{(h)}|$,
- $C_{(h)} = |\rho(0)| + \sup_{x \in \mathbb{R}} |\rho'(x)|$,
- $\Lambda_{(h)}$ *is a constant that only depends on* $\rho^{(h)}$,
- $\mathfrak{W}_{(h)} = 1 + \|\mathcal{W}^{(h)}\|_{L^\infty \to L^\infty}$.

Remark 3.7. *For fully-connected neural networks, we have* $M = O(1), A_{(h)} = 0, B = O(1), C_{(h)} = O(1), \Lambda_{(h)} = O(1), \mathfrak{W}_{(h)} = O(1)$*, so we need* $m = \Omega\left(\frac{n^2 \log(Hn/\delta) 2^{O(H)}}{\lambda_0^2}\right)$*.. For ResNet, we have* $M = O(1), A_{(h)} = 1, B = O(1), C_{(h)} = O(\frac{1}{H}), \Lambda_{(h)} = O(\frac{1}{H}), \mathfrak{W}_{(h)} = O(1)$*, so we need* $m = \Omega\left(\frac{n^2 \log(Hn/\delta)}{\lambda_0^2}\right)$*. The convolutional ResNet has the same parameters as ResNet but because the Gram matrix is* $np \times np$*, so we need* $m = \Omega\left(\frac{n^2 p^2 \log(Hnp/\delta)}{\lambda_0^2}\right)$.

Proof of Theorem 3.4. The proof is by induction. For the base case, $h = 1$, recall

$$\mathbf{X}_i^{(1),[\alpha]} = \rho^{(1)}(\sum_\beta \mathbf{W}_{(\beta)}^{(1),(\alpha)} \mathbf{X}_i^{(0),(\beta)}).$$

We define

$$\mathbf{U}_i^{(1),(\alpha)} = \sum_\beta W_{(\beta)}^{(1),(\alpha)} \mathbf{X}_i^{(0),(\beta)}.$$

By our generating process of $\left\{\mathbf{W}_{(\beta)}^{(h),(\alpha)}\right\}_{h,\alpha,\beta}$, the collection $\{\mathbf{U}_i^{(1),(\beta)}\}_{1 \le i \le n, 1 \le \beta \le m}$ is a mean-zero Gaussian variable with covariance matrix:

$$\begin{aligned}
&\mathbb{E}\mathbf{U}_i^{(1),(\alpha)} \left(\mathbf{U}_j^{(1),(\beta)}\right)^\top \\
=&\mathbb{E} \sum_{\gamma,\gamma'} \mathbf{W}_{(\gamma)}^{(1),(\alpha)} \mathbf{X}_i^{(0),(\gamma)} \left(\mathbf{X}_i^{(0),(\gamma')\top}\right)^\top \left(\mathbf{W}_{(\gamma')}^{(1),(\beta)}\right)^\top \\
=&\delta_{\alpha\beta} \mathcal{W}^{(1)} \left(\sum_\gamma \left(\mathbf{X}_i^{(0),(\gamma)} \mathbf{X}_j^{(0),(\gamma)}\right)^\top\right) \\
=&\delta_{\alpha\beta} \mathcal{W}^{(1)}(\mathbf{K}_{ij}^{(0)})
\end{aligned}$$

Therefore, we have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{m}\sum_{i=1}^m \mathbf{X}_i^{(1),(\alpha)} \mathbf{X}_j^{(1),(\alpha)\top}\right] =& \mathbf{K}_{ij}^{(1)} \\
\mathbb{E}\left[\frac{1}{m}\sum_{i=1}^m \mathbf{X}_i^{(1),(\alpha)}\right] =& \mathbf{b}_i^{(1)}.
\end{aligned}$$

Now we have calculated the expectation. Note since inside the expectation is an average, we can apply standard standard Bernstein bounds and Hoeffding bound and obtain the following

concentration inequalities. With probability at least $1 - \frac{\delta}{H}$, we have

$$\max_{i,j} \left\| \frac{1}{m} \sum_{i=1}^{m} \mathbf{X}_i^{(1),(\alpha)} \mathbf{X}_j^{(1),(\alpha)\top} - \mathbf{K}_{ij}^{(1)} \right\|_\infty \leq \sqrt{\frac{16(1 + 2C_{(1)}^2/\sqrt{\pi}) M^2 \log(4Hn^2(p^{(1)})^2/\delta)}{m}},$$

$$\max_{i,p} \left| \frac{1}{m} \sum_{\alpha=1}^{m} \mathbf{X}_{ip}^{(1),(\alpha)} - \mathbf{b}_{ip}^{(1)} \right| \leq \sqrt{\frac{2C_{(1)}^2 M \log(2np^{(1)}H/\delta)}{m}}$$

Now we prove the induction step. Define for $1 \leq h \leq H$

$$\hat{\mathbf{K}}_{ij}^{(h)} = \frac{1}{m} \sum_{\gamma} \mathbf{X}_i^{(h),(\gamma)} \left(\mathbf{X}_j^{(h),(\gamma)} \right)^\top$$

$$\hat{\mathbf{b}}_i^{(h)} = \frac{1}{m} \sum_{\gamma} \mathbf{X}_i^{(1),(\gamma)}$$

In the following, by $\mathbb{E}^{(h)}$ we mean taking expectation conditioned on first $(h-1)$ layers.

Now suppose that Equation (3.17) and (3.18) hold for $1 \leq l \leq h$ with probability at least $1 - \frac{h}{H}\delta$, now we want to show the equations holds for $h+1$ with probability at least $1 - \delta/H$ conditioned on previous layers satisfying Equation (3.17) and (3.18). Let $l = h + 1$. recall

$$\mathbf{X}_i^{(l),(\alpha)} = \mathcal{D}^{(l)}(\mathbf{X}^{(l-1)}) + \rho^{(l)} \left(\frac{\sum_\beta \mathbf{W}_{(\beta)}^{(l),(\alpha)} \mathbf{X}_i^{(l-1),(\beta)}}{\sqrt{m}} \right).$$

Similar to the base case, denote

$$\mathbf{U}_i^{(l),(\alpha)} = \frac{\sum_\beta \mathbf{W}_{(\beta)}^{(l),(\alpha)} \mathbf{X}_i^{(l-1),(\beta)}}{\sqrt{m}}.$$

Again note that $\{\mathbf{U}_i^{(l),(\beta)}\}_{1 \leq i \leq n, 1 \leq \beta \leq m}$ is a collection of mean-zero Gaussian variables with covariance matrix:

$$\mathbb{E}\left[\mathbf{U}_i^{(1),(\alpha)} \left(\mathbf{U}_j^{(1),(\beta)} \right)^\top \right] = \delta_{\alpha\beta} \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ij}^{(l-1)})$$

Now we get the following formula for the expectation:

$$\begin{aligned}
\mathbb{E}^{(l)}[\hat{\mathbf{K}}_{ij}^{(l)}] =& \mathcal{D}^{(l)} \hat{\mathbf{K}}_{ij}^{(l-1)} \left(\mathcal{D}^{(l)} \right)^\top \\
&+ \mathbb{E}_{(\mathbf{U},\mathbf{V})} \left(\rho^{(l)}(\mathbf{U})^\top \mathcal{D}^{(l)}(\hat{\mathbf{b}}_j^{(l-1)}) + (\mathcal{D}^{(l)}(\hat{\mathbf{b}}_i^{(l-1)}))^\top \rho^{(l)}(\mathbf{V}) + \rho^{(l)}(\mathbf{U})^\top \rho^{(l)}(\mathbf{V}) \right) \\
\mathbb{E}^{(l)} \hat{\mathbf{b}}_i^{(l)} =& \mathcal{D}^{(l)}(\hat{\mathbf{b}}_i^{(l-1)}) + \mathbb{E}_{\mathbf{U}} \rho^{(l)}(\mathbf{U})
\end{aligned}$$

with

$$(\mathbf{U}, \mathbf{V}) \sim N\left(\mathbf{0}, \begin{pmatrix} \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ii}^{(l-1)}) & \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ij}^{(l-1)}) \\ \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ji}^{(l-1)}) & \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{jj}^{(l-1)}) \end{pmatrix} \right)$$

Same as the base case, applying concentration inequalities, we have with probability at least $1-\delta/H$,

$$\max_{ij}\|\mathbb{E}^{(l)}\hat{\mathbf{K}}_{ij}^{(l)}-\hat{\mathbf{K}}_{ij}^{(l)}\|_\infty \leq \sqrt{\frac{16(1+2C_{(l)}^2/\sqrt{\pi})M^2\log(4Hn^2(p^{(l)})^2/\delta)}{m}},$$
$$\max_{i}\|\mathbb{E}^{(l)}\hat{\mathbf{b}}_{i}^{(l)}-\hat{\mathbf{b}}_{i}^{(l)}\|_\infty \leq \sqrt{\frac{2C_{(l)}^2M\log(2np^{(1)}H/\delta)}{m}}$$

Now it remains to bound the differences

$$\max_{ij}\left\|\mathbb{E}^{(l)}\hat{\mathbf{K}}_{ij}^{(l)}-\mathbf{K}_{ij}^{(l)}\right\|_\infty \text{ and } \max_{i}\left\|\mathbb{E}^{(l)}\hat{\mathbf{b}}_{ij}^{(l)}-\mathbf{b}_{ij}^{(l)}\right\|_\infty$$

which determine how the error propagates through layers.

We analyze the error directly.

$$\begin{aligned}
&\left\|\mathbb{E}^{(l)}\hat{\mathbf{K}}_{ij}^{(l)}-\mathbf{K}_{ij}^{(l)}\right\|_\infty\\
\leq&\left\|\mathcal{D}^{(l)}\hat{\mathbf{K}}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}-\mathcal{D}^{(l)}\mathbf{K}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\hat{\mathbf{b}}_j^{(l-1)})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})\right\|_\infty\\
&+\left\|\mathbb{E}_{(U,V)\sim\hat{\mathbf{A}}}(\mathcal{D}^{(l)}(\hat{\mathbf{b}}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}(\mathcal{D}^{(l)}(\mathbf{b}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\rho^{(l)}(\mathbf{V}))-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(\mathbf{U})^\top\rho^{(l)}(\mathbf{V}))\right\|_\infty\\
\leq&\left\|\mathcal{D}^{(l)}\hat{\mathbf{K}}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}-\mathcal{D}^{(l)}\mathbf{K}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\hat{\mathbf{b}}_j^{(l-1)})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}(\mathbf{a}^{(l)}(\hat{\mathbf{b}}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}(\mathcal{D}^{(l)}(\mathbf{b}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{A}}(\mathcal{D}^{(l)}(\mathbf{b}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}(\mathcal{D}^{(l)}(\mathbf{b}_i^{(l-1)}))^\top\rho^{(l)}(\mathbf{V})\right\|_\infty\\
&+\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\rho^{(l)}(\mathbf{V}))-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(\mathbf{U})^\top\rho^{(l)}(\mathbf{V}))\right\|_\infty
\end{aligned}$$

where we define

$$\hat{\mathbf{A}}=\begin{pmatrix}\mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ii}^{(l-1)}) & \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ij}^{(l-1)})\\ \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{ji}^{(l-1)}) & \mathcal{W}^{(l)}(\hat{\mathbf{K}}_{jj}^{(l-1)})\end{pmatrix} \text{ and } \mathbf{A}=\begin{pmatrix}\mathcal{W}^{(l)}(\mathbf{K}_{ii}^{(l-1)}) & \mathcal{W}^{(l)}(\mathbf{K}_{ij}^{(l-1)})\\ \mathcal{W}^{(l)}(\mathbf{K}_{ji}^{(l-1)}) & \mathcal{W}^{(l)}(\mathbf{K}_{jj}^{(l-1)})\end{pmatrix}$$

By definition, we have

$$\|\mathbf{A}-\hat{\mathbf{A}}\|_\infty \leq \mathfrak{W}\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty \text{ and}$$

$$\left\|\mathcal{D}^{(l)}\hat{\mathbf{K}}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}-\mathcal{D}^{(l)}\mathbf{K}_{ij}^{(l-1)}\mathcal{D}^{(l)\top}\right\|_\infty \leq A_{(l)}\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty.$$

We can also estimate other terms

$$\begin{aligned}
&\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\hat{\mathbf{b}}_j^{(l-1)})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(U)^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})\right\|_\infty\\
\leq&\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}\left(\hat{\mathbf{b}}_j^{(l-1)}-\mathbf{b}_j^{(l-1)}\right)\right\|_\infty\\
\leq&C_{(l)}A_{(l)}\sqrt{\mathfrak{W}\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l)}\|_\infty}\max_i\left\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\right\|_\infty\\
\leq&C_{(l)}A_{(l)}\sqrt{\mathfrak{W}_{(l)}M}\max_i\left\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\right\|_\infty,
\end{aligned}$$

$$\begin{aligned}
&\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(\mathbf{U})^\top\mathcal{D}^{(l)}(\mathbf{b}_j^{(l-1)})\right\|_\infty\\
\leq&A_{(l)}BC_{(l)}\|\mathbf{A}-\hat{\mathbf{A}}\|_\infty\\
\leq&A_{(l)}BC_{(l)}\mathfrak{W}\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty,
\end{aligned}$$

and

$$\begin{aligned}
&\left\|\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\hat{\mathbf{A}}}\rho^{(l)}(\mathbf{U})^\top\rho^{(l)}(\mathbf{V})-\mathbb{E}_{(\mathbf{U},\mathbf{V})\sim\mathbf{A}}\rho^{(l)}(U)^\top\rho^{(l)}(V)\right\|_\infty\\
\leq&\Lambda_{(l)}\|\mathbf{A}-\hat{\mathbf{A}}\|_\infty\\
\leq&\Lambda_{(l)}\mathfrak{W}\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty.
\end{aligned}$$

where we have used Lemma 3.30.

Putting these estimates together, we have

$$\begin{aligned}
&\max_{ij}\|\mathbb{E}^{(l)}\hat{\mathbf{K}}_{ij}^{(l)}-\mathbf{K}_{ij}^{(l)}\|_\infty\\
\leq&\left(A_{(l)}+\Lambda_{(l)}\mathfrak{W}+2C_{(l)}A_{(l)}B\mathfrak{W}\right)\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty\\
&+2C_{(l)}A_{(l)}\sqrt{\mathfrak{W}_{(l)}M}\max_i\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\|_\infty\\
\leq&\left(A_{(l)}+\Lambda_{(l)}\mathfrak{W}+2C_{(l)}A_{(l)}B\mathfrak{W}+2C_{(l)}A_{(l)}\sqrt{\mathfrak{W}_{(l)}M}\right)\\
&\cdot\left(\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty\vee\max_i\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\|_\infty\right)
\end{aligned}$$

and

$$\begin{aligned}
&\max_i\left\|\mathbb{E}^{(l)}\hat{\mathbf{b}}_{ij}^{(l)}-\mathbf{b}_{ij}^{(l)}\right\|_\infty\\
\leq&\Lambda_{(l)}\mathfrak{W}\max_{ij}\left\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\right\|_\infty+A_{(l)}\max_i\left\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\right\|_\infty\\
\leq&\left(A_{(l)}+\Lambda_{(l)}\mathfrak{W}\right)\left(\max_{ij}\|\hat{\mathbf{K}}_{ij}^{(l-1)}-\mathbf{K}_{ij}^{(l-1)}\|_\infty\vee\max_i\|\hat{\mathbf{b}}_{ij}^{(l-1)}-\mathbf{b}_{ij}^{(l-1)}\|_\infty\right).
\end{aligned}$$

These two bounds imply the theorem. □

3.12.2 From $\mathbf{K}^{(H-1)}$ to $\mathbf{K}^{(H)}$

Recall $\mathbf{K}^{(H)}$ defined in Equation (3.7), (3.8) and (3.9). Note the definition of $\mathbf{K}^{(H)}$ is qualitatively different from that of $\mathbf{K}^{(h)}$ for $h = 1, \ldots, H-1$ because $\mathbf{K}^{(H)}$ depends on $\mathbf{K}^{(H)}$ and $\sigma'(\cdot)$ instead of $\sigma(\cdot)$. Therefore, we take special care of $\mathbf{K}^{(H)}$. Further note $\mathbf{K}^{(H)}$ for our three architectures have the same form and only differ in scaling and dimension, so we will only prove the bound for the fully-connected architecture. The generalization to ResNet and convolutional ResNet is straightforward.

Lemma 3.22. *For $(i,j) \in [n] \times [n]$, define*

$$\hat{\mathbf{K}}_{ij}^{(H-1)} = \hat{\mathbf{K}}_{ij}^{(H-1)} \mathbb{E}_{\mathbf{w} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})} \left[\sigma'(\mathbf{w}^\top \mathbf{x}_i^{(H-1)}(0)) \sigma'(\mathbf{w}^\top \mathbf{x}_j^{(H-1)}(0)) \right].$$

and suppose $\left| \hat{\mathbf{K}}_{ij}^{(H-1)} - \mathbf{K}_{ij}^{(H-1)} \right| \leq \frac{c\lambda_0}{n^2}$ for some small constant $c > 0$. Then if $m = \Omega\left(\frac{n^2 \log(n/\delta)}{\lambda_0^2} \right)$, we have with probability at least $1 - \delta$ over $\{\mathbf{w}_r^{(H)}(0)\}_{r=1}^m$ and $\{a_r(0)\}_{r=1}^m$, $\left\| \mathbf{G}^{(H)}(0) - \mathbf{K}^{(H)} \right\|_{\text{op}} \leq \frac{\lambda_0}{4}$.

Proof of Lemma 3.22. We decompose

$$\mathbf{G}^{(H)}(0) - \mathbf{K}^{(H)} = \left(\mathbf{G}^{(H)}(0) - \hat{\mathbf{K}}^{(H)} \right) + \left(\hat{\mathbf{K}}^{(H)} - \mathbf{K}^{(H)} \right).$$

Recall $\mathbf{G}^{(H)}$ defined in Equation (3.12). Based on its expression, it is straightforward to use concentration inequality to show if $m = \Omega\left(\frac{n^2 \log(n/\delta)}{\lambda_0^2} \right)$, we have

$$\left\| \mathbf{G}^{(H)}(0) - \hat{\mathbf{K}}^{(H)} \right\|_{\text{op}} \leq \frac{\lambda_0}{8}.$$

For the other term. Recall $\mathbf{A}_{ij}^{(H)} = \begin{pmatrix} \mathbf{K}_{ii}^{(H-1)} & \mathbf{K}_{ij}^{(H-1)} \\ \mathbf{K}_{ji}^{(H-1)} & \mathbf{K}_{jj}^{(H-1)} \end{pmatrix}$ and let $\hat{\mathbf{A}}_{ij}^{(H)} = \begin{pmatrix} \hat{\mathbf{K}}_{ii}^{(H-1)} & \hat{\mathbf{K}}_{ij}^{(H-1)} \\ \hat{\mathbf{K}}_{ji}^{(H-1)} & \hat{\mathbf{K}}_{jj}^{(H-1)} \end{pmatrix}$.

According to Lemma 3.29 (viewing $\sigma'(\cdot)$ as the $\sigma(\cdot)$ in Lemma 3.29), we know

$$\left| \mathbb{E}_{(\mathbf{U}) \sim \hat{\mathbf{A}}_{ij}} \left[\sigma'(u)\sigma'(v) \right] - \mathbb{E}_{(u,v) \sim \mathbf{A}_{ij}} \left[\sigma'(u)\sigma'(v) \right] \right| \leq C \left| \hat{\mathbf{A}}_{ij} - \mathbf{A}_{ij} \right|$$

for some constant $C > 0$. Since c is small enough, we directly have

$$\left\| \hat{\mathbf{K}}^{(H)} - \mathbf{K}^{(H)} \right\|_{op} \leq \frac{\lambda_0}{8}$$

□

Remark 3.8. *Combing Theorem 3.4, Lemma 3.22 and standard matrix perturbation bound directly have Lemma 3.2. Similarly we can prove Lemma 3.9 and Lemma 3.16.*

3.13 Full Rankness of $\mathbf{K}^{(h)}$

3.13.1 Full Rankness of $\mathbf{K}^{(h)}$ for the Fully-connected Neural Network

In this section we show as long as no two input vectors are parallel, then $\mathbf{K}^{(H)}$ defined in Equation (3.8) is strictly positive definite.

Proposition 3.1. *Assume $\sigma(\cdot)$ satisfies Condition 3.2 and for any $i, j \in [n], i \neq j$, $\mathbf{x}_i \nparallel \mathbf{x}_j$. Then we have $\lambda_{\min}\left(\mathbf{K}^{(H)}\right) > 0$ where $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ is defined in Equation* (3.7)*.*

Proof of Proposition 3.1. By our assumption on the data point and using Lemma 3.23 we know $\mathbf{K}^{(1)}$ is strictly positive definite.

By letting $\mathbf{Z} = \mathbf{D}^{1/2}\mathbf{U}^\top$, where $\mathbf{U}\mathbf{D}\mathbf{U}^\top = \mathbf{K}^h$. We then use Lemma 3.23 inductively for $(H-2)$ times to conclude $\mathbf{K}^{(H-1)}$ is strictly positive definite. Lastly we use Lemma 3.24 to finish the proof. □

Lemma 3.23. *Assume $\sigma(\cdot)$ is analytic and not a polynomial function. Consider data $Z = \{\mathbf{z}_i\}_{i\in[n]}$ of n non-parallel points (meaning $\mathbf{z}_i \notin$ span$(\mathbf{z}_j)$ for all $i \neq j$). Define*

$$\mathbf{G}(Z)_{ij} = \mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},\mathbf{I})}[\sigma(\mathbf{w}^\top \mathbf{z}_i)\sigma(\mathbf{w}^\top \mathbf{z}_j)].$$

Then $\lambda_{\min}(\mathbf{G}(Z)) > 0$.

Proof of Lemma 3.23. The feature map induced by the kernel $\mathbf{G}$ is given by $\phi_{\mathbf{z}}(\mathbf{w}) = \sigma(\mathbf{w}^\top \mathbf{z})\mathbf{z}$. To show that $\mathbf{G}(Z)$ is strictly positive definite, we need to show $\phi_{\mathbf{z}_1}(\mathbf{w}), \ldots, \phi_{\mathbf{z}_n}(\mathbf{w})$ are linearly independent functions. Assume that there are a_i such that

$$0 = \sum_i a_i \phi_{\mathbf{z}_i} = \sum_i a_i \sigma(\mathbf{w}^\top \mathbf{z}_i)\mathbf{z}_i.$$

We wish to show that $a_i = 0$. Differentiating the above equation $(n-2)$ times with respect to $\mathbf{w}$, we have

$$0 = \sum_i \left(a_i \sigma^{(n-1)}(\mathbf{w}^\top \mathbf{z}_i)\right) \mathbf{z}_i^{\otimes(n-1)}.$$

Using Lemma 3.31, we know $\left\{\mathbf{z}_i^{\otimes(n-1)}\right\}_{i=1}^n$ are linearly independent. Therefore, we must have $a_i \sigma^{(n-1)}(\mathbf{w}^\top \mathbf{z}_i) = 0$ for all i. Now choosing a $\mathbf{w}$ such that $\sigma^{(n-1)}\left(\mathbf{w}^\top \mathbf{z}_i\right) \neq 0$ for all $i \in [n]$ (such $\mathbf{w}$ exists because of our assumption on σ), we have $a_i = 0$ for all $i \in [n]$. □

Lemma 3.24. *Assume $\sigma(\cdot)$ is analytic and not a polynomial function. Consider data $Z = \{\mathbf{z}_i\}_{i\in[n]}$ of n non-parallel points (meaning $\mathbf{z}_i \notin$ span$(\mathbf{z}_j)$ for all $i \neq j$). Define*

$$\mathbf{G}(Z)_{ij} = \mathbb{E}_{\mathbf{w}\sim N(\mathbf{0},\mathbf{I})}[\sigma'(\mathbf{w}^\top \mathbf{z}_i)\sigma'(\mathbf{w}^\top \mathbf{z}_j)(\mathbf{z}_i^\top \mathbf{z}_j)].$$

Then $\lambda_{\min}(\mathbf{G}(Z)) > 0$.

Proof of Lemma 3.24. The feature map induced by the kernel $\mathbf{G}$ is given by $\phi_{\mathbf{z}}(\mathbf{w}) = \sigma'(\mathbf{w}^\top \mathbf{z})\mathbf{z}$. To show that $\mathbf{G}(Z)$ is strictly positive definite, we need to show $\phi_{\mathbf{z}_1}(\mathbf{w}), \ldots, \phi_{\mathbf{z}_n}(\mathbf{w})$ are linearly independent functions. Assume that there are a_i such that

$$0 = \sum_i a_i \phi_{\mathbf{z}_i} = \sum_i a_i \sigma'(\mathbf{w}^\top \mathbf{z}_i)\mathbf{z}_i.$$

We wish to show that $a_i = 0$. Differentiating the above equation $(n-2)$ times with respect to $\mathbf{w}$, we have

$$0 = \sum_i \left(a_i \sigma^{(n)}(\mathbf{w}^\top \mathbf{z}_i)\right) \mathbf{z}_i^{\otimes(n-1)}.$$

Using Lemma 3.31, we know $\left\{\mathbf{z}_i^{\otimes(n-1)}\right\}_{i=1}^n$ are linearly independent. Therefore, we must have $a_i \sigma^n(\mathbf{w}^\top \mathbf{z}_i) = 0$ for all i. Now choosing a $\mathbf{w}$ such that $\sigma^{(n)}\left(\mathbf{w}^\top \mathbf{z}_i\right) \neq 0$ for all $i \in [n]$ (such $\mathbf{w}$ exists because of our assumption on σ), we have $a_i = 0$ for all $i \in [n]$. □

3.13.2 Full Rankness of $\mathbf{K}^{(h)}$ for ResNet

In this section we show as long as no two input vectors are parallel, then $\mathbf{K}^{(H)}$ defined in Equation (3.8) is strictly positive definite. Furthermore, $\lambda_{\min}\left(\mathbf{K}^{(H)}\right)$ does not depend inverse exponentially in H.

Proposition 3.2. *Assume $\sigma(\cdot)$ satisfies Condition 3.2 and for any $i, j \in [n], i \neq j$, $\mathbf{x}_i \nparallel \mathbf{x}_j$. Recall that in Equation* (3.8)*, we define*

$$\mathbf{K}_{ij}^{(H)} = c_H \mathbf{K}_{ij}^{(H-1)} \cdot \mathbb{E}_{(u,v)^\top \sim N\left(\mathbf{0}, \begin{pmatrix} \mathbf{K}_{ii}^{(H-1)} & \mathbf{K}_{ij}^{(H-1)} \\ \mathbf{K}_{ji}^{(H-1)} & \mathbf{K}_{jj}^{(H-1)} \end{pmatrix}\right)} \left[\sigma'(u)\sigma'(v)\right],$$

where $c_H \sim \frac{1}{H^2}$. Then we have $\lambda_{\min}(\mathbf{K}^{(H)}) \geq c_H \kappa$, where κ is a constant that only depends on the activation σ and the input data. In particular, κ does not depend on the depth.

Proof of Proposition 3.2. First note $\mathbf{K}_{ii}^{(H-1)} \in [1/c_{x,0}^2, c_{x,0}^2]$ for all H, so it is in a bounded range that does not depend on the depth (c.f. Lemma 3.8). Define a function

$$\mathbf{G} : \mathbb{R}^{n \times n} \to \mathbb{R}^{n \times n}$$

such that $\mathbf{G}(\mathbf{K})_{ij} = \mathbf{K}_{ij} \mathbb{E}_{(u,v)^\top \sim N\left(\mathbf{0}, \begin{pmatrix} \mathbf{K}_{ii} & \mathbf{K}_{ij} \\ \mathbf{K}_{ji} & \mathbf{K}_{jj} \end{pmatrix}\right)} \left[\sigma'(u)\sigma'(v)\right]$. Now define a scalar function

$$g(\lambda) = \min_{\mathbf{K}:\mathbf{K}\succ 0, \frac{1}{c_{x,0}^2} \leq \mathbf{K}_{ii} \leq c_{x,0}, \lambda(\mathbf{K}) \geq \lambda} \lambda_{\min}(\mathbf{G}(\mathbf{K}))$$

with

$$\lambda(\mathbf{K}) = \min_{ij} \begin{pmatrix} \mathbf{K}_{ii} & \mathbf{K}_{ij} \\ \mathbf{K}_{ji} & \mathbf{K}_{jj} \end{pmatrix}.$$

By Lemma 3.25, we know $\lambda(\mathbf{K}^{(H-1)}) \geq c_H \lambda\left(\mathbf{K}^{(0)}\right)$.

Next, let $\mathbf{U}\mathbf{D}\mathbf{U}^\top = \mathbf{K}^{(H-1)}$ be the eigen-decomposition of $\mathbf{K}$, and $\mathbf{Z} = \mathbf{D}^{1/2}\mathbf{U}^\top$ be the feature embedding into $\mathbb{R}^n$. Since $\begin{pmatrix} \mathbf{z}_i^\top \mathbf{z}_i & \mathbf{z}_i^\top \mathbf{z}_j \\ \mathbf{z}_j^\top \mathbf{z}_i & \mathbf{z}_j^\top \mathbf{z}_j \end{pmatrix}$ is full rank, then $\mathbf{z}_i \notin \text{span}(\mathbf{z}_j)$. Then using Lemma 3.24 , we know $g(\lambda\left(\mathbf{K}^{(0)}\right)) > 0$. Thus we have established that $\lambda_{\min}(\mathbf{K}^{(H)}) \geq c_H g(\lambda\left(\mathbf{K}^{(0)}\right))$, where $g(\lambda\left(\mathbf{K}^{(0)}\right))$ only depends on the input data and activation σ. In particular, it is independent of the depth. □

Lemma 3.25. *If $\mathcal{D}^{(h)}$ is the identity mapping defined in Section 3.12, then*

$$\lambda\left(\mathbf{K}^{(H)}\right) \geq \min_{(i,j)\in[n]\times[n]} \lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(0)} & \mathbf{K}_{ij}^{(0)} \\ \mathbf{K}_{ji}^{(0)} & \mathbf{K}_{jj}^{(0)} \end{pmatrix}.$$

Proof of Lemma 3.25. First recall

$$(\mathbf{U}, \mathbf{V}) \sim N\left(\mathbf{0}, \begin{pmatrix} \mathcal{W}^{(h)}(\mathbf{K}_{ii}^{(h-1)}) & \mathcal{W}^{(h)}(\mathbf{K}_{ij}^{(h-1)}) \\ \mathcal{W}^{(h)}(\mathbf{K}_{ji}^{(h-1)}) & \mathcal{W}^{(h)}(\mathbf{K}_{jj}^{(h-1)}) \end{pmatrix}\right)$$

Then we compute

$$\begin{aligned}
&\mathbf{K}_{ij}^{(h)} - \mathbf{b}_i^{(h)}\mathbf{b}_j^{(h)\top} \\
&= \mathcal{D}^{(h)}\mathbf{K}_{ij}^{(h-1)}\mathcal{D}^{(h)\top} + \mathbb{E}_{(\mathbf{U},\mathbf{V})}\Big(\rho(\mathbf{U})\mathcal{D}^{(h)}(\mathbf{b}_j^{(h-1)})^\top + (\mathcal{D}^{(h)}(\mathbf{b}_i^{(h-1)}))\rho(\mathbf{V})^\top + \rho(\mathbf{U})\rho(\mathbf{V})^\top)\Big) \\
&- \left(\mathcal{D}^{(h)}(\mathbf{b}_i^{(h-1)}) + \mathbb{E}_{\mathbf{U}}\rho^{(h)}(\mathbf{U})\right)\left(\mathcal{D}^{(h)}(\mathbf{b}_j^{(h-1)}) + \mathbb{E}_{\mathbf{V}}\rho^{(h)}(\mathbf{V}))\right)^\top \\
&= \mathcal{D}^{(h)}\left(\mathbf{K}_{ij}^{(h-1)} - \mathbf{b}_i^{(h-1)}\mathbf{b}_j^{(h-1)\top}\right)\mathcal{D}^{(h)\top} + \mathbb{E}_{(\mathbf{U},\mathbf{V})}\Big(\rho(\mathbf{U})\rho(\mathbf{V})^\top)\Big) - \left(\mathbb{E}_{\mathbf{U}}\rho^{(h)}(\mathbf{U})\right)\left(\mathbb{E}_{\mathbf{V}}\rho^{(h)}(\mathbf{V}))\right)^\top
\end{aligned}$$

For ResNet, $\mathcal{D}^{(h)}$ is the identity mapping so we have

$$\begin{aligned}
&\mathbf{K}_{ij}^{(h)} - \mathbf{b}_i^{(h)}\mathbf{b}_j^{(h)\top} \\
&= \mathbf{K}_{ij}^{(h-1)} - \mathbf{b}_i^{(h-1)}\mathbf{b}_j^{(h-1)\top} + \mathbb{E}_{(\mathbf{U},\mathbf{V})}\Big(\rho(\mathbf{U})\rho(\mathbf{V})^\top)\Big) - \left(\mathbb{E}_{\mathbf{U}}\rho^{(h)}(\mathbf{U})\right)\left(\mathbb{E}_{\mathbf{V}}\rho^{(h)}(\mathbf{V}))\right)^\top.
\end{aligned}$$

To proceed, we calculate

$$\begin{aligned}
&\begin{pmatrix} \mathbf{K}_{ii}^{(h)} & \mathbf{K}_{ij}^{(h)} \\ \mathbf{K}_{ji}^{(h)} & \mathbf{K}_{jj}^{(h)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(h)} \\ \mathbf{b}_j^{(h)} \end{pmatrix}\begin{pmatrix} \mathbf{b}_i^{(h)\top}, \mathbf{b}_j^{(h)\top} \end{pmatrix} \\
&= \begin{pmatrix} \mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(h-1)} \\ \mathbf{b}_j^{(h-1)} \end{pmatrix}\begin{pmatrix} \mathbf{b}_i^{(h-1)\top}, \mathbf{b}_j^{(h-1)\top} \end{pmatrix} \\
&\quad + \left(\mathbb{E}_{\mathbf{U},\mathbf{V}}\begin{pmatrix} \rho^{(h)}(\mathbf{U})\rho^{(h)}(\mathbf{U})^\top & \rho^{(h)}(\mathbf{U})\rho^{(h)}(\mathbf{V})^\top \\ \rho^{(h)}(\mathbf{V})\rho^{(h)}(\mathbf{U})^\top & \rho^{(h)}(\mathbf{V})\rho^{(h)}(\mathbf{V})^\top \end{pmatrix} - \mathbb{E}_{\mathbf{U},\mathbf{V}}\begin{pmatrix} \rho(\mathbf{U}) \\ \rho(\mathbf{V}) \end{pmatrix}\mathbb{E}_{\mathbf{U},\mathbf{V}}\begin{pmatrix} \rho(\mathbf{U})^\top, \rho(\mathbf{V})^\top \end{pmatrix}\right) \\
&\geq \begin{pmatrix} \mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(h-1)} \\ \mathbf{b}_j^{(h-1)} \end{pmatrix}\begin{pmatrix} \mathbf{b}_i^{(h-1)\top}, \mathbf{b}_j^{(h-1)\top} \end{pmatrix}
\end{aligned}$$

As a result, we have

$$
\begin{aligned}
&\lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(h)} & \mathbf{K}_{ij}^{(h)} \\ \mathbf{K}_{ji}^{(h)} & \mathbf{K}_{jj}^{(h)} \end{pmatrix} \\
\geq &\lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(h)} & \mathbf{K}_{ij}^{(h)} \\ \mathbf{K}_{ji}^{(h)} & \mathbf{K}_{jj}^{(h)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(h)} \\ \mathbf{b}_j^{(h)} \end{pmatrix} \begin{pmatrix} \mathbf{b}_i^{(h)\top}, \mathbf{b}_j^{(h)\top} \end{pmatrix} \\
\geq &\min \lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(h-1)} & \mathbf{K}_{ij}^{(h-1)} \\ \mathbf{K}_{ji}^{(h-1)} & \mathbf{K}_{jj}^{(h-1)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(h-1)} \\ \mathbf{b}_j^{(h-1)} \end{pmatrix} \begin{pmatrix} \mathbf{b}_i^{(h-1)\top}, \mathbf{b}_j^{(h-1)\top} \end{pmatrix} \\
\geq &\cdots \\
\geq &\lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(0)} & \mathbf{K}_{ij}^{(0)} \\ \mathbf{K}_{ji}^{(0)} & \mathbf{K}_{jj}^{(0)} \end{pmatrix} - \begin{pmatrix} \mathbf{b}_i^{(0)} \\ \mathbf{b}_j^{(0)} \end{pmatrix} \begin{pmatrix} \mathbf{b}_i^{(0)\top}, \mathbf{b}_j^{(0)\top} \end{pmatrix} \\
= &\lambda_{\min}\begin{pmatrix} \mathbf{K}_{ii}^{(0)} & \mathbf{K}_{ij}^{(0)} \\ \mathbf{K}_{ji}^{(0)} & \mathbf{K}_{jj}^{(0)} \end{pmatrix}.
\end{aligned}
\tag{3.19}
$$

We now prove the theorem. □

3.14 Useful Technical Lemmas

Lemma 3.26. *Given a set of matrices* $\{\mathbf{A}_i, \mathbf{B}_i : i \in [n]\}$*, if* $\|\mathbf{A}_i\|_2 \leq M_i$*,* $\|\mathbf{B}_i\|_2 \leq M_i$ *and* $\|\mathbf{A}_i - \mathbf{B}_i\|_F \leq \alpha_i M_i$*, we have*

$$
\left\| \prod_{i=1}^{n} \mathbf{A}_i - \prod_{i=1}^{n} \mathbf{B}_i \right\|_F \leq \left(\sum_{i=1}^{n} \alpha_i \right) \prod_{i=1}^{n} M_i.
$$

Proof of Lemma 3.26.

$$
\begin{aligned}
&\left\| \prod_{i=1}^{n} \mathbf{A}_i - \prod_{i=1}^{n} \mathbf{B}_i \right\|_F \\
= &\left\| \sum_{i=1}^{n} \left(\prod_{j=1}^{i-1} \mathbf{A}_j \right) (\mathbf{A}_i - \mathbf{B}_i) \left(\prod_{k=i+1}^{n} \mathbf{B}_k \right) \right\|_F \\
\leq &\sum_{i=1}^{n} \left\| \left(\prod_{j=1}^{i-1} \mathbf{A}_j \right) (\mathbf{A}_i - \mathbf{B}_i) \left(\prod_{k=i+1}^{n} \mathbf{B}_k \right) \right\|_F \\
\leq &\left(\sum_{i=1}^{n} \alpha_i \right) \prod_{i=1}^{n} M_i.
\end{aligned}
$$

□

Lemma 3.27. *Given a matrix* $\mathbf{W} \in \mathbb{R}^{m \times cm}$ *with* $\mathbf{W}_{i,j} \sim N(0,1)$*, where* c *is a constant. We have with probability at least* $1 - \exp\left(-\frac{\left(c_{w,0}-\sqrt{c}-1\right)^2 m}{2}\right)$

$$\|\mathbf{W}\|_2 \leq c_{w,0}\sqrt{m},$$

where $c_{w,0} > \sqrt{c} + 1$ *is a constant.*

Proof of Lemma 3.27. The lemma is a consequence of well-known deviations bounds concerning the singular values of Gaussian random matrices [76]

$$P\left(\lambda_{\max}\left(\mathbf{W}\right) > \sqrt{m} + \sqrt{cm} + t\right) \leq e^{t^2/2}.$$

Choosing $t = \left(c_{w,0} - \sqrt{c} - 1\right)\sqrt{m}$, we prove the lemma. □

Lemma 3.28. *Assume* $\sigma(\cdot)$ *satisfies Condition 3.1. For* $a, b \in \mathbb{R}$ *with* $\frac{1}{c} < \min(a,b)$, $\max(a,b) < c$ *for some constant* $c > 0$*, we have*

$$\left|\mathbb{E}_{z\sim N(0,a)}[\sigma(z)] - \mathbb{E}_{z\sim N(0,b)}[\sigma(z)]\right| \leq C\left|a-b\right|.$$

for some constant $C > 0$ *that depends only on* c *and the constants in Condition 3.1.*

Proof of Lemma 3.28. We compute for any $\min(a,b) \leq \alpha \leq \max(a,b)$

$$\left|\frac{d\mathbb{E}_{z\sim N(0,\alpha)}[\sigma\left(z\right)]}{d\alpha}\right| = \left|\frac{d\mathbb{E}_{z\sim N(0,1)}[\sigma\left(\alpha z\right)]}{d\alpha}\right| = \left|\mathbb{E}_{z\sim N(0,1)}[z\sigma'(\alpha z)]\right| \leq C.$$

Applying Taylor's Theorem we finish the proof. □

Lemma 3.29. *Assume* $\sigma(\cdot)$ *satisfies Condition 3.1. Suppose that there exists some constant* $c > 0$ *such that* $\mathbf{A} = \begin{bmatrix} a_1^2 & \rho a_1 b_1 \\ \rho_1 a_1 b_1 & b_1^2 \end{bmatrix}$, $\frac{1}{c} \leq \min(a_1, b_1)$, $\max(a_1, b_1) \leq c$, $\mathbf{B} = \begin{bmatrix} a_2^2 & \rho_2 a_2 b_2 \\ \rho a_2 b_2 & b_2^2 \end{bmatrix}$, $\frac{1}{c} \leq \min(a_2, b_2)$, $\max(a_2, b_2) \leq c$

and $\mathbf{A}, \mathbf{B} \succ 0$*. Define* $F(\mathbf{A}) = \mathbb{E}_{(u,v)\sim N(\mathbf{0},\mathbf{A})}\sigma(u)\sigma(v)$*. Then, we have*

$$|F(\mathbf{A}) - F(\mathbf{B})| \leq C\|\mathbf{A}-\mathbf{B}\|_F \leq 2C\|\mathbf{A}-\mathbf{B}\|_\infty.$$

for some constant $C > 0$ *that depends only on* c *and the constants in Condition 3.1.*

Proof. Let $\mathbf{A}' = \begin{bmatrix} a^2 & \rho ab \\ \rho ab & b^2 \end{bmatrix} \succ 0$ with $\min(a_1, a_2) \leq a \leq \max(a_1, a_2)$, $\min(b_1, b_2) \leq b \leq \max(b_1, b_2)$ and $\min(\rho_1, \rho_2) \leq \rho \leq \max(\rho_1, \rho 2)$. We can express

$$F(\mathbf{A}') = \mathbb{E}_{(z_1,z_2)\sim\mathcal{N}(0,C)}\sigma(az_1)\sigma(bz_2) \text{ with } \mathbf{C} = \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}.$$

Recall $\mathcal{L}^2 = \{f : \int f(z)e^{-z^2/2}dz < \infty\}$ is the Gaussian function space. We compute

$$\frac{dF}{da} = \mathbb{E}[\sigma'(az_1)\sigma(bz_2)z_1]$$

$$\begin{aligned}\Big|\frac{dF}{da}\Big| &\le \|\sigma'(az_1)z_1\|_{L^2}\|\sigma(bz_2)\|_{L^2} && (\|f\|_{L^2} := (\mathbb{E}f(z)^2)^{1/2}, \text{Cauchy})\\ &< \infty && \text{(by Condition 3.1)}\end{aligned}$$

By the same argument, we have

$$\Big|\frac{dF}{db}\Big| < \infty$$

Next, let $\sigma_a(z) := \sigma(az)$ with Hermite expansion $\sigma_a(z) = \sum_{i=0}^{\infty} \alpha_i h_i(z)$ and similarly $\sigma_b(z) = \sum_i \beta_i h_i(z)$. Using the orthonormality that $\mathbb{E}[h_i(z)h_j(z)] = 1_{i=j}$,

$$F(A) = \sum_{i=0}^{\infty} \alpha_i\beta_i\rho^i.$$

Differentiating, we have

$$\begin{aligned}\Big|\frac{dF}{d\rho}\Big| &= \Big|\sum_{i=1}^{\infty} \alpha_i\beta_i i\rho^{i-1}\Big|\\ &< \big(\sum_{i=1}^{\infty} \alpha_i^2 i\big)^{1/2}\big(\sum_{i=1}^{\infty} \beta_i^2 i\big)^{1/2} && (\ \rho = 1 \text{ and Cauchy})\\ &< \infty && (\ \text{Condition 3.1})\end{aligned}$$

Note by Condition 3.1 we know there exists B_ρ, B_a and B_b such that $\big|\frac{dF}{d\rho}\big| \le B_\rho$,$\big|\frac{dF}{da}\big| \le B_a$, and $\big|\frac{dF}{db}\big| \le B_b$.

Next, we bound $\nabla_{\mathbf{A}'} F(\mathbf{A}')$. We see that

$$\begin{aligned}\Big|\frac{dF}{dA'_{11}}\Big| &\le \Big|\frac{dF}{da}\Big|\Big|\frac{da}{dA'_{11}}\Big|\\ &\le B_a\frac{1}{2\sqrt{A'_{11}}} && (\text{since } a = \sqrt{A'_{11}})\\ &\le \frac{1}{2}B_a/c\\ \Big|\frac{dF}{dA'_{11}}\Big| &\le \frac{1}{2}B_b/c && (\text{analogous argument asa bove.})\end{aligned}$$

Using the change of variables, let

$$g(A'_{11}, A'_{22}, A'_{12}) = [\sqrt{A'_{11}}, \sqrt{A'_{22}}, A'_{12}/\sqrt{A'_{11}A'_{22}}] = [a, b, \rho].$$

By chain rule, we know

$$\frac{\partial F}{\partial A'_{12}} = \frac{\partial F}{\partial a}\frac{\partial a}{\partial A'_{12}} + \frac{\partial F}{\partial b}\frac{\partial b}{\partial A'_{12}} + \frac{\partial F}{\partial \rho}\frac{\partial \rho}{\partial A'_{12}} = \frac{\partial F}{\partial \rho}\frac{\partial \rho}{\partial A'_{12}}.$$

We can easily verify that $|\frac{\partial \rho}{\partial A'_{12}}| \leq 1/c^2$, and so we have

$$\left|\frac{\partial F}{\partial A'_{12}}\right| \leq \frac{B_\rho}{c^2}.$$

Similarly, we have

$$\left|\frac{\partial F}{\partial A'_{11}}\right| \leq \frac{B_a}{c^2}$$
$$\left|\frac{\partial F}{\partial A'_{22}}\right| \leq \frac{B_b}{c^2}$$

Define $B_\sigma = \max(B_a, B_b, B_\rho)$. This establishes $\|\nabla F(\mathbf{A}')\|_F \leq 2B_\sigma/c^2 \leq C$ for some constant $C > 0$. Thus by Taylor's Theorem, we have

$$|F(\mathbf{A}) - F(\mathbf{B})| \leq C\|\mathbf{A} - \mathbf{B}\|_F \leq 2C\|\mathbf{A} - \mathbf{B}\|_\infty.$$

□

With Lemma 3.28 and 3.29, we can prove the following useful lemma.

Lemma 3.30. *Suppose $\sigma(\cdot)$ satisfies Condition 3.1 For a positive definite matrix $\mathbf{A} \in \mathbb{R}^{2p\times 2p}$, define*

$$\mathbf{F}(\mathbf{A}) = \mathbb{E}_{\mathbf{U}\sim N(0,\mathbf{A})}\left[\sigma(\mathbf{U})\,\sigma(\mathbf{U})^\top\right],$$
$$\mathbf{G}(\mathbf{A}) = \mathbb{E}_{\mathbf{U}\sim N(\mathbf{0},\mathbf{A})}\left[\sigma(\mathbf{U})\right].$$

Then for any two positive definite matrices $\mathbf{A}, \mathbf{B}$ with $\frac{1}{c} \leq \mathbf{A}_{ii}, \mathbf{B}_{ii} \leq c$ for some constant $c > 0$, we have

$$\|\mathbf{G}(\mathbf{A}) - \mathbf{G}(\mathbf{B})\|_\infty \vee \|\mathbf{F}(\mathbf{A}) - \mathbf{F}(\mathbf{B})\|_\infty \leq C\,\|\mathbf{A} - \mathbf{B}\|_\infty$$

for some constant $C > 0$.

Proof of Lemma 3.30. The result follows by applying Lemma 3.28 to all coordiniates and applying Lemma 3.29 to all 2×2 submatrices. □

Lemma 3.31. *If $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^d$ satisfy that $\|\mathbf{v}_i\|_2 = 1$ and non-parallel (meaning $\mathbf{v}_i \notin$ span$(\mathbf{v}_j)$ for $i \neq j$), then the matrix $\left[vec\left(\mathbf{v}_1^{\otimes n}\right), \ldots, vec\left(\mathbf{v}_n^{\otimes n}\right)\right] \in \mathbb{R}^{d^n\times n}$ has rank-n.*

Proof of Lemma 3.31. We prove by induction. For $n = 2$, $v_1v_1^\top, v_2v_2^\top$ are linearly independent under the non-parallel assumption. By induction suppose $\{\text{vec}\left(\mathbf{v}_1^{\otimes n-1}\right), \ldots, \text{vec}\left(\mathbf{v}_{n-1}^{\otimes n-1}\right)\}$ are linearly independent. Suppose the conclusion does not hold, then there exists $\alpha_1, \ldots, \alpha_n \in \mathbb{R}$ not identically 0, such that

$$\sum_{i=1}^n \alpha_i \text{vec}\left(\mathbf{v}_i^{\otimes n}\right) = 0,$$

which implies for $p = 1, \ldots, d$

$$\sum_{i=1}^{n} (\alpha_i \mathbf{v}_{i,p}) \text{vec}\left(\mathbf{v}_i^{\otimes(n-1)}\right) = 0.$$

Note by induction hypothesis any size $(n-1)$ subset of $\left\{\text{vec}\left(\mathbf{v}_1^{\otimes(n-1)}\right), \ldots, \text{vec}\left(\mathbf{v}_n^{\otimes(n-1)}\right)\right\}$ is linearly independent. This implies if $\alpha_i \mathbf{v}_{i,p} = 0$ for some $i \in [n]$ and $p \in [d]$, then we must have $\alpha_j \mathbf{v}_{j,p} = 0$ for all $j \in [n]$. Combining this observation with the assumption that every $\mathbf{v}_i$ is non-zero, there must exist $p \in [d]$ such that $\mathbf{v}_{i,p} \neq 0$ for all $i \in [n]$. Without loss of generality, we assume $\mathbf{v}_{i,1} \neq 0$ for all $i \in [n]$.

Next, note if there exists $\alpha_i = 0$, then we have $\alpha_j = 0$ for all $j \in [n]$ because $\mathbf{v}_{j,p} \neq 0$ for all $j \in [n]$ and the linear independence induction hypothesis. Therefore from now on we assume $\alpha_i \neq 0$ for all $i \in [n]$.

For any $p \in [d]$, we have

$$\sum_{i=1}^{n} (\alpha_i \mathbf{v}_{i,p}) \text{vec}\left(\mathbf{v}_i^{\otimes(n-1)}\right) = 0 \text{ and } \sum_{i=1}^{n} (\alpha_i \mathbf{v}_{i,1}) \text{vec}\left(\mathbf{v}_i^{\otimes(n-1)}\right) = 0.$$

By multiplying the second equation by $\frac{\mathbf{v}_{1,p}}{\mathbf{v}_{1,1}}$ and subtracting,

$$\sum_{i=2}^{n} (\alpha_i \mathbf{v}_{i,p} - \alpha_i \frac{\mathbf{v}_{1,p}}{\mathbf{v}_{1,1}} \mathbf{v}_{i,1}) \text{vec}\left(\mathbf{v}_i^{\otimes(n-1)}\right) = 0.$$

Using the linear independence induction hypothesis, we know for $i = 2, \ldots, n$:

$$\frac{\mathbf{v}_{i,p}}{\mathbf{v}_{1,1}} = \frac{\mathbf{v}_{1,p}}{\mathbf{v}_{1,1}}.$$

Therefore we know

$$\frac{\mathbf{v}_{1,p}}{\mathbf{v}_{1,1}} = \cdots = \frac{\mathbf{v}_{n,p}}{\mathbf{v}_{n,1}}.$$

Thus there exists $c_2, \ldots, c_d \in \mathbb{R}^d$ such that

$$\mathbf{v}_{i,p} = c_p \mathbf{v}_{i,1} \text{ for all } i \in [n].$$

Note this implies all $\mathbf{v}_i$, $i \in [n]$ are on the same line. This contradicts with the non-parallel assumption. □

Chapter 4

Auto-balancing Property of Gradient Descent for Optimizing Deep Homogeneous Models

4.1 Introduction

In this chapter we study the structural properties of gradient descent for optimizing deep homogeneous models, which are ubiquitous in modern machine learning. The aim of this chapter is not about global convergence but about what quantities can the optimization algorithm preserves. Comparing to previous chapters, the analysis in this chapter does not require over-parameterization. We motive the problem by considering a feed-forward deep neural network that defines a prediction function

$$\mathbf{x} \mapsto f(\mathbf{x}; \mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(H)}) = \mathbf{W}^{(H)} \sigma\left(\mathbf{W}^{(H-1)} \cdots \mathbf{W}^{(2)} \sigma\left(\mathbf{W}^{(1)} \mathbf{x}\right) \cdots\right),$$

where $\mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(H)}$ are weight matrices in N layers, and $\sigma(\cdot)$ is a point-wise *homogeneous* activation functions such as ReLU, Leaky-ReLU and linear activation functions. A simple observation is that this model is *homogeneous*: if we multiply a layer by a positive scalar c and divide another layer by c, the prediction function remains the same, e.g. $f(\mathbf{x}; c\mathbf{W}^{(1)}, \ldots, \frac{1}{c}\mathbf{W}^{(H)}) = f(\mathbf{x}; \mathbf{W}^{(1)}, \ldots, \mathbf{W}^{(N)})$.

A direct consequence of homogeneity is that a solution can produce small function value while being unbounded, because one can always multiply one layer by a huge number and divide another layer by that number. Theoretically, this possible unbalancedness poses significant difficulty in analyzing first order optimization methods like gradient descent/stochastic gradient descent (GD/SGD), because when parameters are not a priori constrained to a compact set via either coerciveness[1] of the loss or an explicit constraint, GD and SGD are not even guaranteed to converge [48, Proposition 4.11]. In the context of deep learning, [65] determined that the primary barrier to providing algorithmic results is in that the sequence of parameter iterates is possibly unbounded.

[1] A function f is coercive if $\|\mathbf{x}\| \to \infty$ implies $f(\mathbf{x}) \to \infty$.

Now we take a closer look at asymmetric matrix factorization, which is a simple two-layer homogeneous model. Consider the following formulation for factorizing a low-rank matrix:

$$\min_{\mathbf{U}\in\mathbb{R}^{d_1\times r},\mathbf{V}\in\mathbb{R}^{d_2\times r}} L(\mathbf{U},\mathbf{V}) = \frac{1}{2}\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F^2, \tag{4.1}$$

where $\mathbf{M}^* \in \mathbb{R}^{d_1\times d_2}$ is a matrix we want to factorize. We observe that due to the homogeneity of f, it is not smooth[2] even in the neighborhood of a globally optimum point. To see this, we compute the gradient of L:

$$\frac{\partial L(\mathbf{U},\mathbf{V})}{\partial \mathbf{U}} = \left(\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right)\mathbf{V}, \qquad \frac{\partial L(\mathbf{U},\mathbf{V})}{\partial \mathbf{V}} = \left(\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right)^\top \mathbf{U}. \tag{4.2}$$

Notice that the gradient of L is not homogeneous anymore. Further, consider a globally optimal solution $(\mathbf{U},\mathbf{V})$ such that $\|\mathbf{U}\|_F$ is of order ϵ and $\|\mathbf{V}\|_F$ is of order $1/\epsilon$ (ϵ being very small). A small perturbation on $\mathbf{U}$ can lead to dramatic change to the gradient of $\mathbf{U}$. This phenomenon can happen for all homogeneous functions when the layers are unbalanced. The lack of nice geometric properties of homogeneous functions due to unbalancedness makes first-order optimization methods difficult to analyze.

A common theoretical workaround is to artificially modify the natural objective function as in (4.1) in order to prove convergence. In [38, 75], a regularization term for balancing the two layers is added to (4.1):

$$\min_{\mathbf{U}\in\mathbb{R}^{d_1\times r},\mathbf{V}\in\mathbb{R}^{d_2\times r}} \frac{1}{2}\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}\right\|_F^2 + \frac{1}{8}\left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F^2. \tag{4.3}$$

For problem (4.3), the regularizer removes the homogeneity issue and the optimal solution becomes unique (up to rotation). Ge et al. [38] showed that the modified objective (4.3) satisfies (i) every local minimum is a global minimum, (ii) all saddle points are strict[3], and (iii) the objective is smooth. These imply that (noisy) GD finds a global minimum [36, 48, 60].

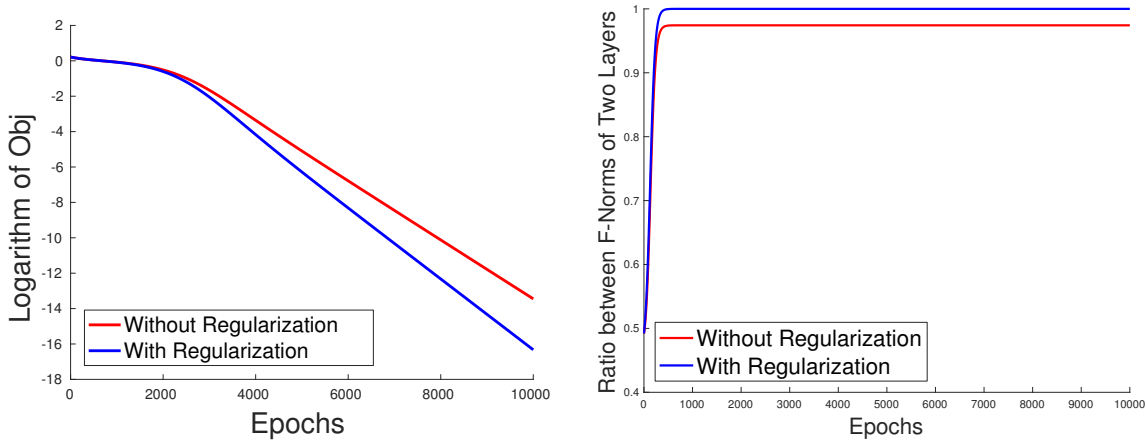
On the other hand, empirically, removing the homogeneity is not necessary. We use GD with random initialization to solve the optimization problem (4.1). Figure 4.1a shows that even *without* regularization term like in the modified objective (4.3) GD with random initialization converges to a global minimum and the convergence rate is also competitive. A more interesting phenomenon is shown in Figure 4.1b in which we track the Frobenius norms of $\mathbf{U}$ and $\mathbf{V}$ in all iterations. The plot shows that the ratio between norms remains a constant in all iterations. Thus the unbalancedness does not occur at all! In many practical applications, many models also admit the homogeneous property (like deep neural networks) and first order methods often converge to a balanced solution. A natural question arises:

Why does GD balance multiple layers and converge in learning homogeneous functions?

In this chapter, we take an important step towards answering this question. Our key finding is that the gradient descent algorithm provides an implicit regularization on the target homogeneous function. First, we show that on the gradient flow (gradient descent with infinitesimal

[2] A function is said to be smooth if its gradient is β-Lipschitz continuous for some finite $\beta > 0$.

[3] A saddle point of a function f is strict if the Hessian at that point has a negative eigenvalue.



(a) Comparison of convergence rates of GD for objective functions (4.1) and (4.3).

(b) Comparison of quantity $\|\mathbf{U}\|_F^2 / \|\mathbf{V}\|_F^2$ when running GD for objective functions (4.1) and (4.3).

Figure 4.1: Experiments on the matrix factorization problem with objective functions (4.1) and (4.3). Red lines correspond to running GD on the objective function (4.1), and blue lines correspond to running GD on the objective function (4.3).

step size) trajectory induced by any differentiable loss function, for a large class of homogeneous models, including fully connected and convolutional neural networks with linear, ReLU and Leaky ReLU activations, the differences between squared norms across layers remain invariant. Thus, as long as at the beginning the differences are small, they remain small at all time. Note that small differences arise in commonly used initialization schemes such as Gaussian initialization or Xavier/Kaiming initialization schemes [40, 44]. Our result thus explains why using ReLU activation is a better choice than sigmoid from the optimization point view. For linear activation, we prove an even stronger invariance for gradient flow: we show that $\mathbf{W}^{(h)}(\mathbf{W}^{(h)})^\top - (\mathbf{W}^{(h+1)})^\top \mathbf{W}^{(h+1)}$ stays invariant over time, where $\mathbf{W}^{(h)}$ and $\mathbf{W}^{(h+1)}$ are weight matrices in consecutive layers with linear activation in between.

Next, we go beyond gradient flow and consider gradient descent with positive step size. We focus on the asymmetric matrix factorization problem (4.1). Our invariance result for linear activation indicates that $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ stays unchanged for gradient flow. For gradient descent, $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ can change over iterations. Nevertheless we show that if the step size decreases like $\eta_t = O\left(t^{-\left(\frac{1}{2}+\delta\right)}\right)$ $(0 < \delta \leq \frac{1}{2})$, $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ will remain small in all iterations. In the set where $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ is small, the loss is coercive, and gradient descent thus ensures that all the iterates are bounded. Using these properties, we then show that gradient descent converges to a globally optimal solution. Furthermore, for rank-1 asymmetric matrix factorization, we give a finer analysis and show that randomly initialized gradient descent with *constant* step size converges to the global minimum at a globally linear rate.

4.1.1 Notations

In this chapter, for a matrix $\mathbf{A}$, we use $\mathbf{A}[i,j]$ to denote its (i,j)-th entry, and use $\mathbf{A}[i,:]$ and $\mathbf{A}[:,j]$ to denote its i-th row and j-th column, respectively (both as column vectors).

4.2 The Auto-Balancing Properties in Deep Neural Networks

In this section we study the implicit regularization imposed by gradient descent with infinitesimal step size (gradient flow) in training deep neural networks. In Section 4.2.1 we consider fully connected neural networks, and our main result (Theorem 4.1) shows that gradient flow automatically balances the incoming and outgoing weights at every neuron. This directly implies that the weights between different layers are balanced (Corollary 4.1). For linear activation, we derive a stronger auto-balancing property (Theorem 4.2). In Section 4.2.2 we generalize our result from fully connected neural networks to convolutional neural networks. In Section 4.2.3 we present the proof of Theorem 4.1. The proofs of other theorems in this section follow similar ideas and are deferred to Appendix 4.6.

4.2.1 Fully Connected Neural Networks

To formally state our results, we define a fully connected neural network with H layers in the following way. Let $\mathbf{W}^{(h)} \in \mathbb{R}^{n_h \times n_{h-1}}$ be the weight matrix in the h-th layer, and define $\theta = (\mathbf{W}^{(h)})_{h=1}^{N}$ as a shorthand of the collection of all the weights. Then the function $f_\theta : \mathbb{R}^d \to \mathbb{R}^p$ ($d = n_0, p = n_N$) computed by this network can be defined recursively: $f_\theta^{(1)}(\mathbf{x}) = \mathbf{W}^{(1)}\mathbf{x}$, $f_{\mathbf{w}}^{(h)}(\mathbf{x}) = \mathbf{W}^{(h)}\sigma\left(f_{\mathbf{w}}^{(h-1)}(\mathbf{x})\right)$ ($h = 2, \ldots, H$), and $f_{\mathbf{w}}(\mathbf{x}) = f_{\mathbf{w}}^{(N)}(\mathbf{x})$, where each $\sigma(\cdot)$ is an activation function that acts coordinate-wise on vectors. We assume that $\sigma(\cdot)$ is *homogeneous*, namely, $\sigma(x) = \sigma'(x)\, x$ for all x and all elements of the sub-differential $\sigma'(\cdot)$ when $\sigma(\cdot)$ is non-differentiable at x.

Let $\ell : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ be a differentiable loss function. Given a training dataset $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m \subset \mathbb{R}^d \times \mathbb{R}^p$, the training loss as a function of the network parameters $\mathbf{w}$ is defined as

$$L(\theta) = \frac{1}{m}\sum_{i=1}^{m} \ell\left(f_\theta(\mathbf{x}_i), \mathbf{y}_i\right). \tag{4.4}$$

We consider gradient descent with infinitesimal step size (also known as gradient flow) applied on $L(\theta)$, which is captured by the differential inclusion:

$$\frac{\mathrm{d}\mathbf{W}^{(h)}}{\mathrm{d}t} \in -\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}}, \qquad h = 1, \ldots, N, \tag{4.5}$$

where t is a continuous time index, and $\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}}$ is the Clarke sub-differential [15]. If curves $\mathbf{W}^{(h)} = \mathbf{W}^{(h)}(t)$ ($h \in [H]$) evolve with time according to (4.5) they are said to be a solution of the gradient flow differential inclusion.

Our main result in this section is the following invariance imposed by gradient flow.

Theorem 4.1 (Balanced incoming and outgoing weights at every neuron)**.** *For any $h \in [H-1]$ and $i \in [n_h]$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\|\mathbf{W}^{(h)}[i,:]\|^2 - \|\mathbf{W}^{(h+1)}[:,i]\|^2\right) = 0. \tag{4.6}$$

Note that $\mathbf{W}^{(h)}[i,:]$ is a vector consisting of network weights coming into the i-th neuron in the h-th hidden layer, and $\mathbf{W}^{(h+1)}[:,i]$ is the vector of weights going out from the same neuron. Therefore, Theorem 4.1 shows that gradient flow exactly preserves the difference between the squared ℓ_2-norms of incoming weights and outgoing weights at any neuron.

Taking sum of (4.6) over $i \in [n_h]$, we obtain the following corollary which says gradient flow preserves the difference between the squares of Frobenius norms of weight matrices.

Corollary 4.1 (Balanced weights across layers)**.** *For any $h \in [N-1]$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\|\mathbf{W}^{(h)}\|_F^2 - \|\mathbf{W}^{(h+1)}\|_F^2\right) = 0.$$

Corollary 4.1 explains why in practice, trained multi-layer models usually have similar magnitudes on all the layers: if we use a small initialization, $\|\mathbf{W}^{(h)}\|_F^2 - \|\mathbf{W}^{(h+1)}\|_F^2$ is very small at the beginning, and Corollary 4.1 implies this difference remains small at all time. This finding also partially explains why gradient descent converges. Although the objective function like (4.4) may not be smooth over the entire parameter space, given that $\|\mathbf{W}^{(h)}\|_F^2 - \|\mathbf{W}^{(h+1)}\|_F^2$ is small for all h, the objective function may have smoothness. Under this condition, standard theory shows that gradient descent converges. We believe this finding serves as a key building block for understanding first order methods for training deep neural networks.

For linear activation, we have the following stronger invariance than Theorem 4.1:

Theorem 4.2 (Stronger balancedness property for linear activation)**.** *If for some $h \in [N-1]$ we have $\sigma(x) = x$, then*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{W}^{(h)}(\mathbf{W}^{(h)})^\top - (\mathbf{W}^{(h+1)})^\top \mathbf{W}^{(h+1)}\right) = \mathbf{0}.$$

While Theorem 4.1 shows the invariance in a node-wise manner, Theorem 4.2 shows for linear activation, we can derive a layer-wise invariance. Inspired by this strong invariance, in Section 4.3 we prove gradient descent with positive step sizes preserves this invariance approximately for matrix factorization.

4.2.2 Convolutional Neural Networks

Now we show that the conservation property in Corollary 4.1 can be generalized to convolutional neural networks. In fact, we can allow *arbitrary sparsity pattern and weight sharing structure* within a layer; convolutional layers are a special case.

Neural networks with sparse connections and shared weights. We use the same notation as in Section 4.2.1, with the difference that some weights in a layer can be *missing* or *shared*. Formally, the weight matrix $\mathbf{W}^{(h)} \in \mathbb{R}^{n_h \times n_{h-1}}$ in layer h ($h \in [H]$) can be described by a vector $\mathbf{v}^{(h)} \in \mathbb{R}^{d_h}$ and a function $g_h : [n_h] \times [n_{h-1}] \to [d_h] \cup \{0\}$. Here $\mathbf{v}^{(h)}$ consists of the actual *free*

parameters in this layer and d_h is the number of free parameters (e.g. if there are k convolutional filters in layer h each with size r, we have $d_h = r \cdot k$). The map g_h represents the sparsity and weight sharing pattern:

$$\mathbf{W}^{(h)}[i,j] = \begin{cases} 0, & g_h(i,j) = 0, \\ \mathbf{v}^{(h)}[k], & g_h(i,j) = k > 0. \end{cases}$$

Denote by $\mathbf{v} = \left(\mathbf{v}^{(h)}\right)_{h=1}^{N}$ the collection of all the parameters in this network, and we consider gradient flow to learn the parameters:

$$\frac{\mathrm{d}\mathbf{v}^{(h)}}{\mathrm{d}t} \in -\frac{\partial L(\mathbf{v})}{\partial \mathbf{v}^{(h)}}, \qquad h = 1, \ldots, N.$$

The following theorem generalizes Corollary 4.1 to neural networks with sparse connections and shared weights:

Theorem 4.3. *For any $h \in [N-1]$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\|\mathbf{v}^{(h)}\|^2 - \|\mathbf{v}^{(h+1)}\|^2\right) = 0.$$

Therefore, for a neural network with arbitrary sparsity pattern and weight sharing structure, gradient flow still balances the magnitudes of all layers.

4.2.3 Proof of Theorem 4.1

The proofs of all theorems in this section are similar. They are based on the use of the chain rule (i.e. back-propagation) and the property of homogeneous activations. Below we provide the proof of Theorem 4.1 and defer the proofs of other theorems to Appendix 4.6.

Proof of Theorem 4.1. First we note that we can without loss of generality assume L is the loss associated with one data sample $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d \times \mathbb{R}^p$, i.e., $L(\mathbf{w}) = \ell(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})$. In fact, for $L(\mathbf{w}) = \frac{1}{m}\sum_{k=1}^{m} L_k(\mathbf{w})$ where $L_k(\mathbf{w}) = \ell\left(f_{\mathbf{w}}(\mathbf{x}_k), \mathbf{y}_k\right)$, for any single weight $\mathbf{W}^{(h)}[i,j]$ in the network we can compute $\frac{\mathrm{d}}{\mathrm{d}t}(\mathbf{W}^{(h)}[i,j])^2 = 2\mathbf{W}^{(h)}[i,j] \cdot \frac{\mathrm{d}\mathbf{W}^{(h)}[i,j]}{\mathrm{d}t} = -2\mathbf{W}^{(h)}[i,j] \cdot \frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}[i,j]} = -2\mathbf{W}^{(h)}[i,j] \cdot \frac{1}{m}\sum_{k=1}^{m} \frac{\partial L_k(\mathbf{w})}{\partial \mathbf{W}^{(h)}[i,j]}$, using the sharp chain rule of differential inclusions for tame functions [17, 19]. Thus, if we can prove the theorem for every individual loss L_k, we can prove the theorem for L by taking average over $k \in [m]$.

Therefore in the rest of proof we assume $L(\mathbf{w}) = \ell(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})$. For convenience, we denote $\mathbf{x}^{(h)} = f_{\mathbf{w}}^{(h)}(\mathbf{x})$ ($h \in [N]$), which is the input to the h-th hidden layer of neurons for $h \in [N-1]$ and is the output of the network for $h = N$. We also denote $\mathbf{x}^{(0)} = \mathbf{x}$.

Now we prove (4.6). Since $\mathbf{W}^{(h+1)}[k,i]$ ($k \in [n_{h+1}]$) can only affect $L(\mathbf{w})$ through $\mathbf{x}^{(h+1)}[k]$, we have for $k \in [n_{h+1}]$,

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h+1)}[k,i]} = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \frac{\partial \mathbf{x}^{(h+1)}[k]}{\partial \mathbf{W}^{(h+1)}[k,i]} = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \sigma\left(\mathbf{x}^{(h)}[i]\right),$$

which can be rewritten as

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h+1)}[:,i]} = \sigma\left(\mathbf{x}^{(h)}[i]\right) \cdot \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}.$$

It follows that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{W}^{(h+1)}[:,i]\|^2 &= 2\left\langle \mathbf{W}^{(h+1)}[:,i], \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h+1)}[:,i]\right\rangle = -2\left\langle \mathbf{W}^{(h+1)}[:,i], \frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h+1)}[:,i]}\right\rangle \\ &= -2\sigma\left(\mathbf{x}^{(h)}[i]\right) \cdot \left\langle \mathbf{W}^{(h+1)}[:,i], \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\right\rangle.\end{aligned} \tag{4.7}$$

On the other hand, $\mathbf{W}^{(h)}[i,:]$ only affects $L(\mathbf{w})$ through $\mathbf{x}^{(h)}[i]$. Using the chain rule, we get

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}[i,:]} = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h)}[i]} \cdot \sigma\left(\mathbf{x}^{(h-1)}\right) = \left\langle \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{W}^{(h+1)}[:,i]\right\rangle \cdot \sigma'\left(\mathbf{x}^{(h)}[i]\right) \cdot \sigma\left(\mathbf{x}^{(h-1)}\right),$$

where $\sigma'(\cdot)$ is interpreted as a set-valued mapping whenever it is applied at a non-differentiable point.[4] It follows that[5]

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{W}^{(h)}[i,:]\|_2^2 \\ =&2\left\langle \mathbf{W}^{(h)}[i,:], \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h)}[i,:]\right\rangle \\ =&-2\left\langle \mathbf{W}^{(h)}[i,:], \frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}[i,:]}\right\rangle \\ =&-2\left\langle \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{W}^{(h+1)}[:,i]\right\rangle \cdot \sigma'\left(\mathbf{x}^{(h)}[i]\right) \cdot \left\langle \mathbf{W}^{(h)}[i,:], \sigma\left(\mathbf{x}^{(h-1)}\right)\right\rangle \\ =&-2\left\langle \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{W}^{(h+1)}[:,i]\right\rangle \cdot \sigma'\left(\mathbf{x}^{(h)}[i]\right) \cdot \mathbf{x}^{(h)}[i] \\ =&-2\left\langle \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{W}^{(h+1)}[:,i]\right\rangle \cdot \sigma'\left(\mathbf{x}^{(h)}[i]\right).\end{aligned}$$

Comparing the above expression to (4.7), we finish the proof. □

4.3 Gradient Descent Converges to Global Minimum for Asymmetric Matrix Factorization

In this section we constrain ourselves to the asymmetric matrix factorization problem and analyze the gradient descent algorithm with random initialization. Our analysis is inspired by the auto-balancing properties presented in Section 4.2. We extend these properties from gradient flow to gradient descent with positive step size.

[4]More precisely, the equalities should be an inclusion whenever there is a sub-differential, but as we see in the next display the ambiguity in the choice of sub-differential does not affect later calculations.

[5]This holds for any choice of element of the sub-differential, since $\sigma'(x)\,x = \sigma(x)$ holds at $x = 0$ for any choice of sub-differential.

Formally, we study the following non-convex optimization problem:

$$\min_{\mathbf{U}\in\mathbb{R}^{d_1\times r},\mathbf{V}\in\mathbb{R}^{d_2\times r}} f(\mathbf{U},\mathbf{V}) = \frac{1}{2}\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F^2, \tag{4.8}$$

where $\mathbf{M}^* \in \mathbb{R}^{d_1\times d_2}$ has rank r. Note that we do not have any explicit regularization in (4.8). The gradient descent dynamics for (4.8) have the following form:

$$\mathbf{U}_{t+1} = \mathbf{U}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)\mathbf{V}_t, \qquad \mathbf{V}_{t+1} = \mathbf{V}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)^\top\mathbf{U}_t. \tag{4.9}$$

4.3.1 The General Rank-r Case

First we consider the general case of $r \geq 1$. Our main theorem below says that if we use a random small initialization $(\mathbf{U}_0, \mathbf{V}_0)$, and set step sizes η_t to be appropriately small, then gradient descent (4.9) will converge to a solution close to the global minimum of (4.8). To our knowledge, this is the first result showing that gradient descent with random initialization directly solves the un-regularized asymmetric matrix factorization problem (4.8).

Theorem 4.4. *Let $0 < \epsilon < \|\mathbf{M}^*\|_F$. Suppose we initialize the entries in $\mathbf{U}_0$ and $\mathbf{V}_0$ i.i.d. from $\mathcal{N}(0, \frac{\epsilon}{\mathrm{poly}(d)})$ ($d = \max\{d_1, d_2\}$), and run (4.9) with step sizes $\eta_t = \frac{\sqrt{\epsilon/r}}{100(t+1)\|\mathbf{M}^*\|_F^{3/2}}$ ($t = 0, 1, \ldots$).*[6] *Then with high probability over the initialization, $\lim_{t\to\infty}(\mathbf{U}_t, \mathbf{V}_t) = (\bar{\mathbf{U}}, \bar{\mathbf{V}})$ exists and satisfies $\left\|\bar{\mathbf{U}}\bar{\mathbf{V}}^\top - \mathbf{M}^*\right\|_F \leq \epsilon$.*

Proof sketch of Theorem 4.4. First let us imagine that we are using infinitesimal step size in GD. Then according to Theorem 4.2 (viewing problem (4.8) as learning a two-layer linear network where the inputs are all the standard unit vectors in $\mathbb{R}^{d_2}$), we know that $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ will stay invariant throughout the algorithm. Hence when $\mathbf{U}$ and $\mathbf{V}$ are initialized to be small, $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$ will stay small forever. Combined with the fact that the objective $L(\mathbf{U}, \mathbf{V})$ is decreasing over time (which means $\mathbf{U}\mathbf{V}^\top$ cannot be too far from $\mathbf{M}^*$), we can show that $\mathbf{U}$ and $\mathbf{V}$ will always stay bounded.

Now we are using positive step sizes η_t, so we no longer have the invariance of $\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}$. Nevertheless, by a careful analysis of the updates, we can still prove that $\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t$ is small, the objective $f(\mathbf{U}_t, \mathbf{V}_t)$ decreases, and $\mathbf{U}_t$ and $\mathbf{V}_t$ stay bounded. Formally, we have the following lemma:

Lemma 4.1. *With high probability over the initialization $(\mathbf{U}_0, \mathbf{V}_0)$, for all t we have:*

(i) Balancedness: $\left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F \leq \epsilon$;

(ii) Decreasing objective: $f(\mathbf{U}_t, \mathbf{V}_t) \leq f(\mathbf{U}_{t-1}, \mathbf{V}_{t-1}) \leq \cdots \leq f(\mathbf{U}_0, \mathbf{V}_0) \leq 2\|\mathbf{M}^*\|_F^2$;

(iii) Boundedness: $\|\mathbf{U}_t\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F, \|\mathbf{V}_t\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F$.

Now that we know the GD algorithm automatically constrains $(\mathbf{U}_t, \mathbf{V}_t)$ in a bounded region, we can use the smoothness of f in this region and a standard analysis of GD to show that $(\mathbf{U}_t, \mathbf{V}_t)$ converges to a stationary point $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ of f (Lemma 4.4). Furthermore, using the results of [48, 60] we know that $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ is almost surely not a strict saddle point. Then the following lemma

[6] The dependency of η_t on t can be $\eta_t = \Theta\left(t^{-(1/2+\delta)}\right)$ for any constant $\delta \in (0, 1/2]$.

implies that $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ has to be close to a global optimum since we know $\left\|\bar{\mathbf{U}}^\top\bar{\mathbf{U}} - \bar{\mathbf{V}}^\top\bar{\mathbf{V}}\right\|_F \le \epsilon$ from Lemma 4.1 (i). This would complete the proof of Theorem 4.4.

Lemma 4.2. *Suppose $(\mathbf{U}, \mathbf{V})$ is a stationary point of f such that $\left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F \le \epsilon$. Then either $\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F \le \epsilon$, or $(\mathbf{U}, \mathbf{V})$ is a strict saddle point of f.*

The full proof of Theorem 4.4 and the proofs of Lemmas 4.1 and 4.2 are given in Appendix 4.7.

4.3.2 The Rank-1 Case

We have shown in Theorem 4.4 that GD with small and diminishing step sizes converges to a global minimum for matrix factorization. Empirically, it is observed that a constant step size $\eta_t \equiv \eta$ is enough for GD to converge quickly to global minimum. Therefore, some natural questions are how to prove convergence of GD with a constant step size, how fast it converges, and how the discretization affects the invariance we derived in Section 4.2.

While these questions remain challenging for the general rank-r matrix factorization, we resolve them for the case of $r = 1$. Our main finding is that with constant step size, the norms of two layers are always within a constant factor of each other (although we may no longer have the stronger balancedness property as in Lemma 4.1), and we utilize this property to prove the *linear convergence* of GD to a global minimum.

When $r = 1$, the asymmetric matrix factorization problem and its GD dynamics become

$$\min_{\mathbf{u}\in\mathbb{R}^{d_1}, \mathbf{v}\in\mathbb{R}^{d_2}} \frac{1}{2}\left\|\mathbf{u}\mathbf{v}^\top - \mathbf{M}^*\right\|_F^2$$

and

$$\mathbf{u}_{t+1} = \mathbf{u}_t - \eta(\mathbf{u}_t\mathbf{v}_t^\top - \mathbf{M}^*)\mathbf{v}_t, \qquad \mathbf{v}_{t+1} = \mathbf{v}_t - \eta\left(\mathbf{v}_t\mathbf{u}_t^\top - \mathbf{M}^{*\top}\right)\mathbf{u}_t.$$

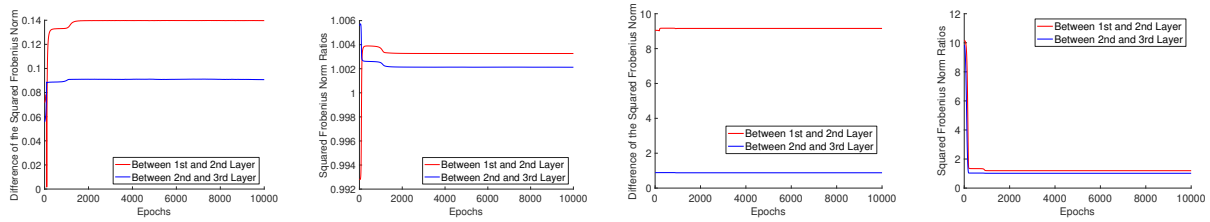
Here we assume $\mathbf{M}^*$ has rank 1, i.e., it can be factorized as $\mathbf{M}^* = \sigma_1\mathbf{u}^*\mathbf{v}^{*\top}$ where $\mathbf{u}^*$ and $\mathbf{v}^*$ are unit vectors and $\sigma_1 > 0$. Our main theoretical result is the following.

Theorem 4.5 (Approximate balancedness and linear convergence of GD for rank-1 matrix factorization)**.** *Suppose $\mathbf{u}_0 \sim \mathcal{N}(\mathbf{0}, \delta\mathbf{I})$, $\mathbf{v}_0 \sim \mathcal{N}(\mathbf{0}, \delta\mathbf{I})$ with $\delta = c_{init}\sqrt{\frac{\sigma_1}{d}}$ ($d = \max\{d_1, d_2\}$) for some sufficiently small constant $c_{init} > 0$, and $\eta = \frac{c_{step}}{\sigma_1}$ for some sufficiently small constant $c_{step} > 0$. Then with constant probability over the initialization, for all t we have $c_0 \le \frac{\left|\mathbf{u}_t^\top\mathbf{u}^*\right|}{\left|\mathbf{v}_t^\top\mathbf{v}^*\right|} \le C_0$ for some universal constants $c_0, C_0 > 0$. Furthermore, for any $0 < \epsilon < 1$, after $t = O\left(\log\frac{d}{\epsilon}\right)$ iterations, we have $\left\|\mathbf{u}_t\mathbf{v}_t^\top - \mathbf{M}^*\right\|_F \le \epsilon\sigma_1$.*

Theorem 4.5 shows for $\mathbf{u}_t$ and $\mathbf{v}_t$, their strengths in the signal space, $\left|\mathbf{u}_t^\top\mathbf{u}^*\right|$ and $\left|\mathbf{v}_t^\top\mathbf{v}^*\right|$, are of the same order. This approximate balancedness helps us prove the linear convergence of GD. We refer readers to Appendix 4.8 for the proof of Theorem 4.5.

4.4 Empirical Verifications

We perform experiments to verify the auto-balancing properties of gradient descent in neural networks with ReLU activation. Our results below show that for GD with small step size and



(a) Balanced initialization, squared norm differences.

(b) Balanced initialization, squared norm ratios.

(c) Unbalanced Initialization, squared norm differences.

(d) Unbalanced initialization, squared norm ratios.

Figure 4.2: Balancedness of a 3-layer neural network.

small initialization: (1) the difference between the squared Frobenius norms of any two layers remains small in all iterations, and (2) the ratio between the squared Frobenius norms of any two layers becomes close to 1. Notice that our theorems in Section 4.2 hold for gradient flow (step size $\to 0$) but in practice we can only choose a (small) positive step size, so we cannot hope the difference between the squared Frobenius norms to remain exactly the same but can only hope to observe that the differences remain small.

We consider a 3-layer fully connected network of the form $f(\mathbf{x}) = \mathbf{W}_3\phi(\mathbf{W}_2\phi(\mathbf{W}_1\mathbf{x}))$ where $\mathbf{x} \in \mathbb{R}^{1,000}$ is the input, $\mathbf{W}_1 \in \mathbb{R}^{100\times 1,000}$, $\mathbf{W}_2 \in \mathbb{R}^{100\times 100}$, $\mathbf{W}_3 \in \mathbb{R}^{10\times 100}$, and $\sigma(\cdot)$ is ReLU activation. We use 1,000 data points and the quadratic loss function, and run GD. We first test a balanced initialization: $\mathbf{W}_1[i,j] \sim N(0, \frac{10^{-4}}{100})$, $\mathbf{W}_2[i,j] \sim N(0, \frac{10^{-4}}{10})$ and $\mathbf{W}_3[i,j] \sim N(0, 10^{-4})$, which ensures $\|\mathbf{W}_1\|_F^2 \approx \|\mathbf{W}_2\|_F^2 \approx \|\mathbf{W}_3\|_F^2$. After 10,000 iterations we have $\|\mathbf{W}_1\|_F^2 = 42.90$, $\|\mathbf{W}_2\|_F^2 = 43.76$ and $\|\mathbf{W}_3\|_F^2 = 43.68$. Figure 4.2a shows that in all iterations $|\|\mathbf{W}_1\|_F^2 - \|\mathbf{W}_2\|_F^2|$ and $|\|\mathbf{W}_2\|_F^2 - \|\mathbf{W}_3\|_F^2|$ are bounded by 0.14 which is much smaller than the magnitude of each $\|\mathbf{W}_h\|_F^2$. Figures 4.2b shows that the ratios between norms approach 1. We then test an unbalanced initialization: $\mathbf{W}_1[i,j] \sim N(0, 10^{-4})$, $\mathbf{W}_2[i,j] \sim N(0, 10^{-4})$ and $\mathbf{W}_3[i,j] \sim N(0, 10^{-4})$. After 10,000 iterations we have $\|\mathbf{W}_1\|_F^2 = 55.50$, $\|\mathbf{W}_2\|_F^2 = 45.65$ and $\|\mathbf{W}_3\|_F^2 = 45.46$. Figure 4.2c shows that $|\|\mathbf{W}_1\|_F^2 - \|\mathbf{W}_2\|_F^2|$ and $|\|\mathbf{W}_2\|_F^2 - \|\mathbf{W}_3\|_F^2|$ are bounded by 9 (and indeed change very little throughout the process), and Figures 4.2d shows that the ratios become close to 1 after about 1,000 iterations.

4.5 Conclusion and Future Work

In this chapter we take a step towards characterizing the invariance imposed by first order algorithms. We show that gradient flow automatically balances the magnitudes of all layers in a deep neural network with homogeneous activations. For the concrete model of asymmetric matrix factorization, we further use the balancedness property to show that gradient descent converges to global minimum. We believe our findings on the invariance in deep models could serve as a fundamental building block for understanding optimization in deep learning. Below we list some future directions.

Other first-order methods. In this chapter, we focus on the invariance induced by gradient descent. In practice, different acceleration and adaptive methods are also used. A natural future direction is how to characterize the invariance properties of these algorithms.

From gradient flow to gradient descent: a generic analysis? As discussed in Section 4.3, while strong invariance properties hold for gradient flow, in practice one uses gradient descent with positive step sizes and the invariance may only hold approximately because positive step sizes discretize the dynamics. We use specialized techniques for analyzing asymmetric matrix factorization. It would be very interesting to develop a generic approach to analyze the discretization. Recent findings on the connection between optimization and ordinary differential equations [66, 69] might be useful for this purpose.

Appendix: Omitted Proofs

4.6 Proofs for Section 4.2

Proof of Theorem 4.2. Same as the proof of Theorem 4.1, we assume without loss of generality that $L(\mathbf{w}) = \ell(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})$ for some $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d \times \mathbb{R}^p$. We also denote $\mathbf{x}^{(h)} = f_{\mathbf{w}}^{(h)}(\mathbf{x})$ $(\forall h \in [H])$ and $\mathbf{x}^{(0)} = \mathbf{x}$.

Now we suppose $\sigma(x) = x$. Denote $\mathbf{u} = \sigma\left(\mathbf{x}^{(h-1)}\right)$. Then we have $\mathbf{x}^{(h+1)} = \mathbf{W}^{(h+1)}\mathbf{x}^{(h)} = \mathbf{W}^{(h+1)}\mathbf{W}^{(h)}\mathbf{u}$. Using the chain rule, we can directly compute

$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h)}} = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h)}}\mathbf{u}^\top = (\mathbf{W}^{(h+1)})^\top \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\mathbf{u}^\top,$$
$$\frac{\partial L(\mathbf{w})}{\partial \mathbf{W}^{(h+1)}} = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}(\mathbf{x}^{(h)})^\top = \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}(\mathbf{W}^{(h)}\mathbf{u})^\top.$$

Then we have

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{W}^{(h)}(\mathbf{W}^{(h)})^\top\right) &= \mathbf{W}^{(h)}\left(\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h)}\right)^\top + \left(\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h)}\right)(\mathbf{W}^{(h)})^\top \\
&= \mathbf{W}^{(h)}\mathbf{u}\left(\frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\right)^\top \mathbf{W}^{(h+1)} + (\mathbf{W}^{(h+1)})^\top \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\mathbf{u}^\top(\mathbf{W}^{(h)})^\top, \\
\frac{\mathrm{d}}{\mathrm{d}t}\left((\mathbf{W}^{(h+1)})^\top\mathbf{W}^{(h+1)}\right) &= (\mathbf{W}^{(h+1)})^\top\left(\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h+1)}\right) + \left(\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{W}^{(h+1)}\right)^\top \mathbf{W}^{(h+1)} \\
&= (\mathbf{W}^{(h+1)})^\top \frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\mathbf{u}^\top(\mathbf{W}^{(h)})^\top + \mathbf{W}^{(h)}\mathbf{u}\left(\frac{\partial L(\mathbf{w})}{\partial \mathbf{x}^{(h+1)}}\right)^\top \mathbf{W}^{(h+1)}.
\end{aligned}$$

Comparing the above two equations we know $\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{W}^{(h)}(\mathbf{W}^{(h)})^\top - (\mathbf{W}^{(h+1)})^\top\mathbf{W}^{(h+1)}\right) = \mathbf{0}$. □

Proof of Theorem 4.3. Same as the proof of Theorem 4.1, we assume without loss of generality that $L(\mathbf{v}) = L(\mathbf{w}) = \ell(f_{\mathbf{w}}(\mathbf{x}), \mathbf{y})$ for $(\mathbf{x}, \mathbf{y}) \in \mathbb{R}^d \times \mathbb{R}^p$, and denote $\mathbf{x}^{(h)} = f_{\mathbf{w}}^{(h)}(\mathbf{x})$ $(\forall h \in [H])$ and $\mathbf{x}^{(0)} = \mathbf{x}$.

Using the chain rule, we have

$$\frac{\partial L(\mathbf{v})}{\partial \mathbf{v}^{(h+1)}[l]} = \sum_{(k,i):g_{h+1}(k,i)=l} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \sigma\left(\mathbf{x}^{(h)}[i]\right), \qquad l \in [d_{h+1}].$$

Then we have using the sharp chain rule,

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}^{(h+1)}\|_2^2 &= 2\left\langle \mathbf{v}^{(h+1)}, \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}^{(h+1)} \right\rangle = -2\left\langle \mathbf{v}^{(h+1)}, \frac{\partial L(\mathbf{v})}{\partial \mathbf{v}^{(h+1)}} \right\rangle \\
&= -2\sum_{l} \sum_{(k,i):g_{h+1}(k,i)=l} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \mathbf{v}^{(h+1)}[l] \cdot \sigma\left(\mathbf{x}^{(h)}[i]\right) \\
&= -2\sum_{(k,i)} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \mathbf{W}^{(h+1)}[k,i] \cdot \sigma\left(\mathbf{x}^{(h)}[i]\right) \qquad (4.10) \\
&= -2\sum_{k} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \mathbf{x}^{(h+1)}[k] \\
&= -2\left\langle \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{x}^{(h+1)} \right\rangle.
\end{aligned}$$

Substituting h with $h-1$ in (4.10) gives $\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}^{(h)}\|_2^2 = -2\left\langle \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h)}}, \mathbf{x}^{(h)} \right\rangle$, which further implies

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\|\mathbf{v}^{(h)}\|_2^2 &= -2\left\langle \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h)}}, \mathbf{x}^{(h)} \right\rangle = -2\sum_{i} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h)}[i]} \cdot \mathbf{x}^{(h)}[i] \\
&= -2\sum_{i}\sum_{k} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \mathbf{W}^{(h+1)}[k,i] \cdot \sigma'\left(\mathbf{x}^{(h)}[i]\right) \cdot \mathbf{x}^{(h)}[i] \\
&= -2\sum_{k} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \sum_{i} \mathbf{W}^{(h+1)}[k,i] \cdot \sigma\left(\mathbf{x}^{(h)}[i]\right) \qquad (4.11) \\
&= -2\sum_{k} \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}[k]} \cdot \mathbf{x}^{(h+1)}[k] \\
&= -2\left\langle \frac{\partial L(\mathbf{v})}{\partial \mathbf{x}^{(h+1)}}, \mathbf{x}^{(h+1)} \right\rangle.
\end{aligned}$$

The proof is finished by combining (4.10) and (4.11). □

4.7 Proof for Rank-r Matrix Factorization (Theorem 4.4)

In this section we give the full proof of Theorem 4.4.

First we recall the gradient of our objective function $f(\mathbf{U},\mathbf{V}) = \frac{1}{2}\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F^2$:

$$\frac{\partial f(\mathbf{U},\mathbf{V})}{\partial \mathbf{U}} = (\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*)\mathbf{V}, \qquad \frac{\partial f(\mathbf{U},\mathbf{V})}{\partial \mathbf{V}} = (\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*)^\top \mathbf{U}.$$

We also need to calculate the Hessian $\nabla^2 f(\mathbf{U}, \mathbf{V})$. The Hessian can be viewed as a matrix that operates on vectorized matrices of dimension $(d_1 + d_2) \times r$ (i.e., the same shape as $\begin{pmatrix}\mathbf{U}\\ \mathbf{V}\end{pmatrix}$). Then, for any $\mathbf{W} \in \mathbb{R}^{(d_1+d_2)\times r}$, the Hessian $\nabla^2 f(\mathbf{W})$ defines a quadratic form

$$[\nabla^2 f(\mathbf{W})](\mathbf{A}, \mathbf{B}) = \sum_{i,j,k,l} \frac{\partial^2 f(\mathbf{W})}{\partial \mathbf{W}[i,j] \partial \mathbf{W}[k,l]} \mathbf{A}[i,j]\mathbf{B}[k,l], \qquad \forall \mathbf{A}, \mathbf{B} \in \mathbb{R}^{(d_1+d_2)\times r}.$$

With this notation, we can express the Hessian $\nabla^2 f(\mathbf{U}, \mathbf{V})$ as follows:

$$[\nabla^2 f(\mathbf{U}, \mathbf{V})](\mathbf{\Delta}, \mathbf{\Delta}) = 2\left\langle \mathbf{U}\mathbf{V}^\top - \mathbf{M}^*, \mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top \right\rangle + \left\| \mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{\Delta}_\mathbf{U}\mathbf{V}^\top \right\|_F^2, \\ \forall \mathbf{\Delta} = \begin{pmatrix}\mathbf{\Delta}_\mathbf{U}\\ \mathbf{\Delta}_\mathbf{V}\end{pmatrix}, \mathbf{\Delta}_\mathbf{U} \in \mathbb{R}^{d_1\times r}, \mathbf{\Delta}_\mathbf{V} \in \mathbb{R}^{d_2\times r}. \tag{4.12}$$

Now we use the expression of the Hessian to prove that $f(\mathbf{U}, \mathbf{V})$ is locally smooth when both arguments $\mathbf{U}$ and $\mathbf{V}$ are bounded.

Lemma 4.3 (Smoothness over a bounded set)**.** *For any $c > 0$, constrained on the set $\mathcal{S} = \{(\mathbf{U}, \mathbf{V}) : \mathbf{U} \in \mathbb{R}^{d_1\times r}, \mathbf{V} \in \mathbb{R}^{d_2\times r}, \|\mathbf{U}\|_F^2 \le c\|\mathbf{M}^*\|_F, \|\mathbf{V}\|_F^2 \le c\|\mathbf{M}^*\|_F\}$, the function f is $((6c+2)\|\mathbf{M}^*\|_F)$-smooth.*

Proof. We prove smoothness by giving an upper bound on $\lambda_{\max}(\nabla^2 f(\mathbf{U}, \mathbf{V}))$ for any $(\mathbf{U}, \mathbf{V}) \in \mathcal{S}$.

For any $(\mathbf{U}, \mathbf{V}) \in \mathcal{S}$ and any $\mathbf{\Delta} = \begin{pmatrix}\mathbf{\Delta}_\mathbf{U}\\ \mathbf{\Delta}_\mathbf{V}\end{pmatrix}$ ($\mathbf{\Delta}_\mathbf{U} \in \mathbb{R}^{d_1\times r}, \mathbf{\Delta}_\mathbf{V} \in \mathbb{R}^{d_2\times r}$), from (4.12) we have

$$\begin{aligned}
&[\nabla^2 f(\mathbf{U}, \mathbf{V})](\mathbf{\Delta}, \mathbf{\Delta}) \\
\le& 2\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F \left\|\mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top\right\|_F + \left\|\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{\Delta}_\mathbf{U}\mathbf{V}^\top\right\|_F^2 \\
\le& 2\left(\|\mathbf{U}\|_F \left\|\mathbf{V}^\top\right\|_F + \|\mathbf{M}^*\|_F\right)\|\mathbf{\Delta}_\mathbf{U}\|_F \left\|\mathbf{\Delta}_\mathbf{V}^\top\right\|_F + \left(\|\mathbf{U}\|_F \left\|\mathbf{\Delta}_\mathbf{V}^\top\right\|_F + \|\mathbf{\Delta}_\mathbf{U}\|_F \left\|\mathbf{V}^\top\right\|_F\right)^2 \\
\le& 2\left(c\|\mathbf{M}^*\|_F + \|\mathbf{M}^*\|_F\right)\|\mathbf{\Delta}\|_F^2 + \left(2\sqrt{c\|\mathbf{M}^*\|_F} \cdot \|\mathbf{\Delta}\|_F\right)^2 \\
=& (6c+2)\|\mathbf{M}^*\|_F \|\mathbf{\Delta}\|_F^2.
\end{aligned}$$

This implies $\lambda_{\max}(\nabla^2 f(\mathbf{U}, \mathbf{V})) \le (6c+2)\|\mathbf{M}^*\|_F$. □

4.7.1 Proof of Lemma 4.1

Recall the following three properties we want to prove in Lemma 4.1, which we call $\mathcal{A}(t)$, $\mathcal{B}(t)$ and $\mathcal{C}(t)$, respectively:

$$\begin{aligned}
\mathcal{A}(t):&\quad \left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F \le \epsilon, \\
\mathcal{B}(t):&\quad f(\mathbf{U}_t, \mathbf{V}_t) \le f(\mathbf{U}_{t-1}, \mathbf{V}_{t-1}) \le \cdots \le f(\mathbf{U}_0, \mathbf{V}_0) \le 2\|\mathbf{M}^*\|_F^2, \\
\mathcal{C}(t):&\quad \|\mathbf{U}_t\|_F^2 \le 5\sqrt{r}\|\mathbf{M}^*\|_F, \|\mathbf{V}_t\|_F^2 \le 5\sqrt{r}\|\mathbf{M}^*\|_F.
\end{aligned}$$

We use induction to prove these statements. For $t = 0$, we can make the Gaussian variance in the initialization sufficiently small such that with high probability we have

$$\|\mathbf{U}_0\|_F^2 \le \epsilon, \qquad \|\mathbf{V}_0\|_F^2 \le \epsilon, \qquad \left\|\mathbf{U}_0^\top \mathbf{U}_0 - \mathbf{V}_0^\top \mathbf{V}_0\right\|_F \le \frac{\epsilon}{2}.$$

From now on we assume they are all satisfied. Then $\mathcal{A}(0)$ is already satisfied, $\mathcal{C}(0)$ is satisfied because $\epsilon < \|\mathbf{M}^*\|_F$, and $\mathcal{B}(0)$ can be verified by $f(\mathbf{U}_0, \mathbf{V}_0) = \frac{1}{2}\left\|\mathbf{U}_0\mathbf{V}_0^\top - \mathbf{M}^*\right\|_F^2 \le \left\|\mathbf{U}_0\mathbf{V}_0^\top\right\|_F^2 + \|\mathbf{M}^*\|_F^2 \le \|\mathbf{U}_0\|_F^2 \left\|\mathbf{V}_0^\top\right\|_F^2 + \|\mathbf{M}^*\|_F^2 \le \epsilon^2 + \|\mathbf{M}^*\|_F^2 \le 2\|\mathbf{M}^*\|_F^2$.

To prove $\mathcal{A}(t)$, $\mathcal{B}(t)$ and $\mathcal{C}(t)$ for all t, we prove the following three claims. Since we have $\mathcal{A}(0)$, $\mathcal{B}(0)$ and $\mathcal{C}(0)$, if the following claims are all true, the proof will be completed by induction.

(i) $\mathcal{B}(0), \ldots, \mathcal{B}(t), \mathcal{C}(0), \ldots, \mathcal{C}(t) \implies \mathcal{A}(t+1)$;

(ii) $\mathcal{B}(0), \ldots, \mathcal{B}(t), \mathcal{C}(t) \implies \mathcal{B}(t+1)$;

(iii) $\mathcal{A}(t), \mathcal{B}(t) \implies \mathcal{C}(t)$.

Claim 4.1. $\mathcal{B}(0), \ldots, \mathcal{B}(t), \mathcal{C}(0), \ldots, \mathcal{C}(t) \implies \mathcal{A}(t+1)$.

Proof. Using the update rule (4.9) we can calculate

$$\begin{aligned}
&\mathbf{U}_{t+1}^\top \mathbf{U}_{t+1} - \mathbf{V}_{t+1}^\top \mathbf{V}_{t+1} \\
=& \left(\mathbf{U}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)\mathbf{V}_t\right)^\top \left(\mathbf{U}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)\mathbf{V}_t\right) \\
&- \left(\mathbf{V}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)^\top\mathbf{U}_t\right)^\top \left(\mathbf{V}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)^\top\mathbf{U}_t\right) \\
=& \mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t + \eta_t^2\left(\mathbf{V}_t^\top\mathbf{R}_t^\top\mathbf{R}_t\mathbf{V}_t - \mathbf{U}_t^\top\mathbf{R}_t^\top\mathbf{R}_t\mathbf{U}_t\right),
\end{aligned}$$

where $\mathbf{R}_t = \mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*$. Then we have

$$\begin{aligned}
&\left\|\mathbf{U}_{t+1}^\top\mathbf{U}_{t+1} - \mathbf{V}_{t+1}^\top\mathbf{V}_{t+1}\right\|_F \\
\le& \left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F + \eta_t^2\left(\left\|\mathbf{V}_t^\top\mathbf{R}_t^\top\mathbf{R}_t\mathbf{V}_t\right\|_F + \left\|\mathbf{U}_t^\top\mathbf{R}_t^\top\mathbf{R}_t\mathbf{U}_t\right\|_F\right) \\
\le& \left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F + \eta_t^2\left(\|\mathbf{V}_t\|_F^2\|\mathbf{R}_t\|_F^2 + \|\mathbf{U}_t\|_F^2\|\mathbf{R}_t\|_F^2\right) \qquad (4.13) \\
=& \left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F + 2\eta_t^2\left(\|\mathbf{V}_t\|_F^2 + \|\mathbf{U}_t\|_F^2\right) f(\mathbf{U}_t, \mathbf{V}_t) \\
\le& \left\|\mathbf{U}_t^\top\mathbf{U}_t - \mathbf{V}_t^\top\mathbf{V}_t\right\|_F + 2\eta_t^2 \cdot 10\sqrt{r}\|\mathbf{M}^*\|_F \cdot 2\|\mathbf{M}^*\|_F^2,
\end{aligned}$$

where the last line is due to $\mathcal{B}(t)$ and $\mathcal{C}(t)$.

Since we have $\mathcal{B}(t')$ and $\mathcal{C}(t')$ for all $t' \le t$, (4.13) is still true when substituting t with any $t' \le t$. Summing all of them and noting $\left\|\mathbf{U}_0^\top\mathbf{U}_0 - \mathbf{V}_0^\top\mathbf{V}_0\right\|_F \le \frac{\epsilon}{2}$, we get

$$\begin{aligned}
&\left\|\mathbf{U}_{t+1}^\top\mathbf{U}_{t+1} - \mathbf{V}_{t+1}^\top\mathbf{V}_{t+1}\right\|_F \\
\le& \left\|\mathbf{U}_0^\top\mathbf{U}_0 - \mathbf{V}_0^\top\mathbf{V}_0\right\|_F + 40\sqrt{r}\|\mathbf{M}^*\|_F^3 \sum_{i=0}^{t} \eta_i^2 \\
\le& \frac{\epsilon}{2} + 40\sqrt{r}\|\mathbf{M}^*\|_F^3 \sum_{i=0}^{t} \frac{1}{(i+1)^2} \cdot \frac{\epsilon/r}{100^2\|\mathbf{M}^*\|_F^3}
\end{aligned}$$

$$\leq \epsilon.$$

Therefore we have proved $\mathcal{A}(t+1)$. □

Claim 4.2. $\mathcal{B}(0), \ldots, \mathcal{B}(t), \mathcal{C}(t) \implies \mathcal{B}(t+1)$.

Proof. Note that we only need to show $f(\mathbf{U}_{t+1}, \mathbf{V}_{t+1}) \leq f(\mathbf{U}_t, \mathbf{V}_t)$. We prove this using the standard analysis of gradient descent, for which we need the smoothness of the objective function f (Lemma 4.3). We first need to bound $\|\mathbf{U}_t\|_F$, $\|\mathbf{V}_t\|_F$, $\|\mathbf{U}_{t+1}\|_F$ and $\|\mathbf{V}_{t+1}\|_F$. We know from $\mathcal{C}(t)$ that $\|\mathbf{U}_t\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F$ and $\|\mathbf{V}_t\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F$. We can also bound $\|\mathbf{U}_{t+1}\|_F^2$ and $\|\mathbf{V}_{t+1}\|_F^2$ easily from the GD update rule:

$$\begin{aligned}
&\|\mathbf{U}_{t+1}\|_F^2 \\
&= \left\|\mathbf{U}_t - \eta_t(\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*)\mathbf{V}_t\right\|_F^2 \\
&\leq 2\|\mathbf{U}_t\|_F^2 + 2\eta_t^2\left\|\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*\right\|_F^2\|\mathbf{V}_t\|_F^2 \\
&\leq 2\cdot 5\sqrt{r}\|\mathbf{M}^*\|_F + 2\eta_t^2\cdot 2f(\mathbf{U}_t,\mathbf{V}_t)\cdot 5\sqrt{r}\|\mathbf{M}^*\|_F \\
&\leq 10\sqrt{r}\|\mathbf{M}^*\|_F + 2\cdot\frac{\epsilon/r}{100^2(t+1)^2\|\mathbf{M}^*\|_F^3}\cdot 4\|\mathbf{M}^*\|_F^2\cdot 5\sqrt{r}\|\mathbf{M}^*\|_F && \text{(using } \mathcal{B}(t)\text{)} \\
&\leq 10\sqrt{r}\|\mathbf{M}^*\|_F + \frac{\epsilon}{100} \\
&\leq 11\sqrt{r}\|\mathbf{M}^*\|_F. && \text{(using } \epsilon < \|\mathbf{M}^*\|_F\text{)}
\end{aligned}$$

Let $\beta = (66\sqrt{r}+2)\|\mathbf{M}^*\|_F$. From Lemma 4.3, f is β-smooth over $\mathcal{S} = \{(\mathbf{U},\mathbf{V}) : \|\mathbf{U}\|_F^2 \leq 11\sqrt{r}\|\mathbf{M}^*\|_F, \|\mathbf{V}\|_F^2 \leq 11\sqrt{r}\|\mathbf{M}^*\|_F\}$. Also note that $\eta_t < \frac{1}{\beta}$ by our choice. Then using smoothness we have

$$\begin{aligned}
&f(\mathbf{U}_{t+1}, \mathbf{V}_{t+1}) \\
&\leq f(\mathbf{U}_t,\mathbf{V}_t) + \left\langle \nabla f(\mathbf{U}_t,\mathbf{V}_t), \begin{pmatrix}\mathbf{U}_{t+1}\\ \mathbf{V}_{t+1}\end{pmatrix} - \begin{pmatrix}\mathbf{U}_t\\ \mathbf{V}_t\end{pmatrix}\right\rangle + \frac{\beta}{2}\left\|\begin{pmatrix}\mathbf{U}_{t+1}\\ \mathbf{V}_{t+1}\end{pmatrix} - \begin{pmatrix}\mathbf{U}_t\\ \mathbf{V}_t\end{pmatrix}\right\|_F^2 \\
&= f(\mathbf{U}_t,\mathbf{V}_t) - \eta_t\|\nabla f(\mathbf{U}_t,\mathbf{V}_t)\|_F^2 + \frac{\beta}{2}\eta_t^2\|\nabla f(\mathbf{U}_t,\mathbf{V}_t)\|_F^2 \\
&\leq f(\mathbf{U}_t,\mathbf{V}_t) - \frac{\eta_t}{2}\|\nabla f(\mathbf{U}_t,\mathbf{V}_t)\|_F^2.
\end{aligned} \tag{4.14}$$

Therefore we have shown $\mathcal{B}(t+1)$. □

Claim 4.3. $\mathcal{A}(t), \mathcal{B}(t) \implies \mathcal{C}(t)$.

Proof. From $\mathcal{B}(t)$ we know $\frac{1}{2}\left\|\mathbf{U}_t\mathbf{V}_t^\top - \mathbf{M}^*\right\|_F^2 \leq 2\|\mathbf{M}^*\|_F^2$ which implies $\left\|\mathbf{U}_t\mathbf{V}_t^\top\right\|_F \leq 3\|\mathbf{M}^*\|_F$. Therefore it suffices to prove

$$\left\|\mathbf{U}\mathbf{V}^\top\right\|_F \leq 3\|\mathbf{M}^*\|_F, \left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F \leq \epsilon \implies \|\mathbf{U}\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F, \|\mathbf{V}\|_F^2 \leq 5\sqrt{r}\|\mathbf{M}^*\|_F. \tag{4.15}$$

Now we prove (4.15). Consider the SVD $\mathbf{U} = \mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^\top$, where $\mathbf{\Phi} \in \mathbb{R}^{d_1 \times d_1}$ and $\mathbf{\Psi} \in \mathbb{R}^{r \times r}$ are orthogonal matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{d_1 \times r}$ is a diagonal matrix. Let $\sigma_i = \mathbf{\Sigma}[i,i]$ ($i \in [r]$) which are all the singular values of $\mathbf{U}$. Define $\widetilde{\mathbf{V}} = \mathbf{V}\mathbf{\Psi}$. Then we have

$$3\left\|\mathbf{M}^*\right\|_F \geq \left\|\mathbf{U}\mathbf{V}^\top\right\|_F = \left\|\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^\top\mathbf{\Psi}\widetilde{\mathbf{V}}^\top\right\|_F = \left\|\mathbf{\Sigma}\widetilde{\mathbf{V}}^\top\right\|_F = \sqrt{\sum_{i=1}^r \sigma_i^2 \left\|\widetilde{\mathbf{V}}[:,i]\right\|^2}$$

and

$$\begin{aligned}\epsilon &\geq \left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F = \left\|\mathbf{\Psi}\mathbf{\Sigma}^\top\mathbf{\Phi}^\top\mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^\top - \mathbf{\Psi}\widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}}\mathbf{\Psi}^\top\right\|_F = \left\|\mathbf{\Sigma}^\top\mathbf{\Sigma} - \widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}}\right\|_F \\ &\geq \sqrt{\sum_{i=1}^r \left(\sigma_i^2 - \left\|\widetilde{\mathbf{V}}[:,i]\right\|^2\right)^2}.\end{aligned}$$

Using the above two inequalities we get

$$\begin{aligned}\sum_{i=1}^r \sigma_i^4 &\leq \sum_{i=1}^r \left(\sigma_i^4 + \left\|\widetilde{\mathbf{V}}[:,i]\right\|^4\right) = \sum_{i=1}^r \left(\sigma_i^2 - \left\|\widetilde{\mathbf{V}}[:,i]\right\|^2\right)^2 + 2\sum_{i=1}^r \sigma_i^2 \left\|\widetilde{\mathbf{V}}[:,i]\right\|^2 \\ &\leq \epsilon^2 + 2\left(3\left\|\mathbf{M}^*\right\|_F\right)^2 \leq 19\left\|\mathbf{M}^*\right\|_F^2.\end{aligned}$$

Then by the Cauchy-Schwarz inequality we have

$$\left\|\mathbf{U}\right\|_F^2 = \sum_{i=1}^r \sigma_i^2 \leq \sqrt{r\sum_{i=1}^r \sigma_i^4} \leq \sqrt{r \cdot 19\left\|\mathbf{M}^*\right\|_F^2} \leq 5\sqrt{r}\left\|\mathbf{M}^*\right\|_F.$$

Similarly, we also have $\left\|\mathbf{V}\right\|_F^2 \leq 5\sqrt{r}\left\|\mathbf{M}^*\right\|_F$. Therefore we have proved (4.15). □

4.7.2 Convergence to a Stationary Point

With the balancedness and boundedness properties in Lemma 4.1, it is then standard to show that $(\mathbf{U}_t, \mathbf{V}_t)$ converges to a stationary point of f.

Lemma 4.4. *Under the setting of Theorem 4.4, with high probability* $\lim_{t\to\infty}(\mathbf{U}_t, \mathbf{V}_t) = (\bar{\mathbf{U}}, \bar{\mathbf{V}})$ *exists, and* $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ *is a stationary point of* f*. Furthermore,* $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ *satisfies* $\left\|\bar{\mathbf{U}}^\top\bar{\mathbf{U}} - \bar{\mathbf{V}}^\top\bar{\mathbf{V}}\right\| \leq \epsilon$.

Proof. We assume the three properties in Lemma 4.1 hold, which happens with high probability. Then from (4.14) we have

$$\begin{aligned}f(\mathbf{U}_{t+1}, \mathbf{V}_{t+1}) &\leq f(\mathbf{U}_t, \mathbf{V}_t) - \frac{\eta_t}{2}\left\|\nabla f(\mathbf{U}_t, \mathbf{V}_t)\right\|_F^2 \\ &= f(\mathbf{U}_t, \mathbf{V}_t) - \frac{1}{2}\left\|\nabla f(\mathbf{U}_t, \mathbf{V}_t)\right\|_F \left\|\begin{pmatrix}\mathbf{U}_{t+1}\\ \mathbf{V}_{t+1}\end{pmatrix} - \begin{pmatrix}\mathbf{U}_t\\ \mathbf{V}_t\end{pmatrix}\right\|_F.\end{aligned} \tag{4.16}$$

Under the above descent condition, the result of [1] says that the iterates either diverge to infinity or converge to a fixed point. According to Lemma 4.1, $\{(\mathbf{U}_t, \mathbf{V}_t)\}_{t=1}^{\infty}$ are all bounded, so they have to converge to a fixed point $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ as $t \to \infty$.

Next, from (4.16) we know that $\sum_{t=1}^{\infty} \frac{\eta_t}{2} \|\nabla f(\mathbf{U}_t, \mathbf{V}_t)\|_F^2 \leq f(\mathbf{U}_0, \mathbf{V}_0)$ is bounded. Notice that η_t scales like $1/t$. So we must have $\liminf_{t\to\infty} \|\nabla f(\mathbf{U}_t, \mathbf{V}_t)\|_F = 0$. Then according to the smoothness of f in a bounded region (Lemma 4.3) we conclude $\nabla f(\bar{\mathbf{U}}, \bar{\mathbf{V}}) = \mathbf{0}$, i.e., $(\bar{\mathbf{U}}, \bar{\mathbf{V}})$ is a stationary point.

The second part of the lemma is evident according to Lemma 4.1 (i). □

4.7.3 Proof of Lemma 4.2

The main idea in the proof is similar to [38]. We want to find a direction $\mathbf{\Delta}$ such that either $[\nabla^2 f(\mathbf{U}, \mathbf{V})](\mathbf{\Delta}, \mathbf{\Delta})$ is negative or $(\mathbf{U}, \mathbf{V})$ is close to a global minimum. We show that this is possible when $\left\|\mathbf{U}^\top \mathbf{U} - \mathbf{V}^\top \mathbf{V}\right\|_F \leq \epsilon$.

First we define some notation. Take the SVD $\mathbf{M}^* = \mathbf{\Phi}^* \mathbf{\Sigma}^* \mathbf{\Psi}^{*\top}$, where $\mathbf{\Phi}^* \in \mathbb{R}^{d_1 \times r}$ and $\mathbf{\Psi}^* \in \mathbb{R}^{d_2 \times r}$ have orthonormal columns and $\mathbf{\Sigma}^* \in \mathbb{R}^{r\times r}$ is diagonal. Denote $\mathbf{U}^* = \mathbf{\Phi}^* (\mathbf{\Sigma}^*)^{1/2}$ and $\mathbf{V}^* = \mathbf{\Psi}^* (\mathbf{\Sigma}^*)^{1/2}$. Then we have $\mathbf{U}^* \mathbf{V}^{*\top} = \mathbf{M}^*$ (i.e., $(\mathbf{U}^*, \mathbf{V}^*)$ is a global minimum) and $\mathbf{U}^{*\top} \mathbf{U}^* = \mathbf{V}^{*\top} \mathbf{V}^*$.

Let $\mathbf{M} = \mathbf{U}\mathbf{V}^\top$, $\mathbf{W} = \begin{pmatrix} \mathbf{U} \\ \mathbf{V} \end{pmatrix}$ and $\mathbf{W}^* = \begin{pmatrix} \mathbf{U}^* \\ \mathbf{V}^* \end{pmatrix}$. Define

$$\mathbf{R} = \operatorname{argmin}_{\mathbf{R}' \in \mathbb{R}^{r\times r},\, \text{orthogonal}} \left\|\mathbf{W} - \mathbf{W}^* \mathbf{R}'\right\|_F$$

and

$$\mathbf{\Delta} = \mathbf{W} - \mathbf{W}^* \mathbf{R}.$$

We will show that $\mathbf{\Delta}$ is the desired direction. Recall (4.12):

$$[\nabla^2 f(\mathbf{U}, \mathbf{V})](\mathbf{\Delta}, \mathbf{\Delta}) = 2 \left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{\Delta}_\mathbf{U} \mathbf{\Delta}_\mathbf{V}^\top \right\rangle + \left\|\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{\Delta}_\mathbf{U} \mathbf{V}^\top\right\|_F^2, \tag{4.17}$$

where $\mathbf{\Delta} = \begin{pmatrix} \mathbf{\Delta}_\mathbf{U} \\ \mathbf{\Delta}_\mathbf{V} \end{pmatrix}$, $\mathbf{\Delta}_\mathbf{U} \in \mathbb{R}^{d_1 \times r}$, $\mathbf{\Delta}_\mathbf{V} \in \mathbb{R}^{d_2 \times r}$. We consider the two terms in (4.17) separately.

For the first term in (4.17), we have:

Claim 4.4. $\left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{\Delta}_\mathbf{U} \mathbf{\Delta}_\mathbf{V}^\top \right\rangle = -\left\|\mathbf{M} - \mathbf{M}^*\right\|_F^2$.

Proof. Since $(\mathbf{U}, \mathbf{V})$ is a stationary point of f, we have the first-order optimality condition:

$$\frac{\partial f(\mathbf{U}, \mathbf{V})}{\partial \mathbf{U}} = (\mathbf{M} - \mathbf{M}^*)\mathbf{V} = \mathbf{0}, \qquad \frac{\partial f(\mathbf{U}, \mathbf{V})}{\partial \mathbf{V}} = (\mathbf{M} - \mathbf{M}^*)^\top \mathbf{U} = \mathbf{0}. \tag{4.18}$$

Note that $\mathbf{\Delta}_\mathbf{U} = \mathbf{U} - \mathbf{U}^* \mathbf{R}$ and $\mathbf{\Delta}_\mathbf{V} = \mathbf{V} - \mathbf{V}^* \mathbf{R}$. We have

$$\begin{aligned}
&\left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{\Delta}_\mathbf{U} \mathbf{\Delta}_\mathbf{V}^\top \right\rangle \\
&= \left\langle \mathbf{M} - \mathbf{M}^*, (\mathbf{U} - \mathbf{U}^*\mathbf{R})(\mathbf{V} - \mathbf{V}^*\mathbf{R})^\top \right\rangle \\
&= \left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{M} - \mathbf{U}^*\mathbf{R}\mathbf{V}^\top - \mathbf{U}\mathbf{R}^\top \mathbf{V}^{*\top} + \mathbf{M}^* \right\rangle \\
&= \left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{M}^* \right\rangle
\end{aligned}$$

$$
\begin{aligned}
&= \langle \mathbf{M} - \mathbf{M}^*, \mathbf{M}^* - \mathbf{M} \rangle \\
&= - \|\mathbf{M} - \mathbf{M}^*\|_F^2 ,
\end{aligned}
$$

where we have used the following consequences of (4.18):

$$
\begin{aligned}
&\langle \mathbf{M} - \mathbf{M}^*, \mathbf{M} \rangle = \left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{U}\mathbf{V}^\top \right\rangle = 0, \\
&\left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{U}^*\mathbf{R}\mathbf{V}^\top \right\rangle = 0, \\
&\left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{U}\mathbf{R}^\top\mathbf{V}^{*\top} \right\rangle = 0.
\end{aligned}
$$

□

The second term in (4.17) has the following upper bound:

Claim 4.5. $\|\mathbf{U}\mathbf{\Delta}_\mathbf{V} + \mathbf{\Delta}_\mathbf{U}\mathbf{V}\|_F^2 \le \|\mathbf{M} - \mathbf{M}^*\|_F^2 + \frac{1}{2}\epsilon^2$.

Proof. We make use of the following identities, all of which can be directly verified by plugging in definitions:

$$
\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{\Delta}_\mathbf{U}\mathbf{V}^\top = \mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{M} - \mathbf{M}^*, \tag{4.19}
$$

$$
\left\|\mathbf{\Delta}\mathbf{\Delta}^\top\right\|_F^2 = 4\left\|\mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top\right\|_F^2 + \left\|\mathbf{\Delta}_\mathbf{U}^\top\mathbf{\Delta}_\mathbf{U} - \mathbf{\Delta}_\mathbf{V}^\top\mathbf{\Delta}_\mathbf{V}\right\|_F^2 , \tag{4.20}
$$

$$
\begin{aligned}
\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{W}^*\mathbf{W}^{*\top}\right\|_F^2 = 4\|\mathbf{M} - \mathbf{M}^*\|_F^2 - 2\left\|\mathbf{U}^\top\mathbf{U}^* - \mathbf{V}^\top\mathbf{V}^*\right\|_F^2 \\
+ \left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F^2 + \left\|\mathbf{U}^{*\top}\mathbf{U}^* - \mathbf{V}^{*\top}\mathbf{V}^*\right\|_F^2 .
\end{aligned} \tag{4.21}
$$

We also need the following inequality, which is [38, Lemma 6]:

$$
\left\|\mathbf{\Delta}\mathbf{\Delta}^\top\right\|_F^2 \le 2\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{W}^*\mathbf{W}^{*\top}\right\|_F^2 . \tag{4.22}
$$

Now we can prove the desired bound as follows:

$$
\begin{aligned}
&\|\mathbf{U}\mathbf{\Delta}_\mathbf{V} + \mathbf{\Delta}_\mathbf{U}\mathbf{V}\|_F^2 \\
&= \left\|\mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top + \mathbf{M} - \mathbf{M}^*\right\|_F^2 && ((4.19)) \\
&= \left\|\mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top\right\|_F^2 + 2\left\langle \mathbf{M} - \mathbf{M}^*, \mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top \right\rangle + \|\mathbf{M} - \mathbf{M}^*\|_F^2 \\
&= \left\|\mathbf{\Delta}_\mathbf{U}\mathbf{\Delta}_\mathbf{V}^\top\right\|_F^2 - \|\mathbf{M} - \mathbf{M}^*\|_F^2 && (\text{Claim 4.4}) \\
&\le \frac{1}{4}\left\|\mathbf{\Delta}\mathbf{\Delta}^\top\right\|_F^2 - \|\mathbf{M} - \mathbf{M}^*\|_F^2 && ((4.20)) \\
&\le \frac{1}{2}\left\|\mathbf{W}\mathbf{W}^\top - \mathbf{W}^*\mathbf{W}^{*\top}\right\|_F^2 - \|\mathbf{M} - \mathbf{M}^*\|_F^2 && ((4.22)) \\
&= 2\|\mathbf{M} - \mathbf{M}^*\|_F^2 - \left\|\mathbf{U}^\top\mathbf{U}^* - \mathbf{V}^\top\mathbf{V}^*\right\|_F^2 + \frac{1}{2}\left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\|_F^2 \\
&\quad + \frac{1}{2}\left\|\mathbf{U}^{*\top}\mathbf{U}^* - \mathbf{V}^{*\top}\mathbf{V}^*\right\|_F^2 - \|\mathbf{M} - \mathbf{M}^*\|_F^2 && ((4.21)) \\
&\le \|\mathbf{M} - \mathbf{M}^*\|_F^2 + \frac{1}{2}\epsilon^2,
\end{aligned}
$$

where in the last line we have used $\mathbf{U}^{*\top}\mathbf{U}^* = \mathbf{V}^{*\top}\mathbf{V}^*$ and $\left\|\mathbf{U}^\top\mathbf{U} - \mathbf{V}^\top\mathbf{V}\right\| \le \epsilon$. □

Using Claims 4.4 and 4.5, we obtain an upper bound on (4.17):

$$[\nabla^2 f(\mathbf{U},\mathbf{V})](\mathbf{\Delta},\mathbf{\Delta}) \le -\left\|\mathbf{M}-\mathbf{M}^*\right\|_F^2 + \frac{1}{2}\epsilon^2.$$

Therefore, we have either $\left\|\mathbf{U}\mathbf{V}^\top - \mathbf{M}^*\right\|_F = \left\|\mathbf{M}-\mathbf{M}^*\right\|_F \le \epsilon$ or $[\nabla^2 f(\mathbf{U},\mathbf{V})](\mathbf{\Delta},\mathbf{\Delta}) \le -\frac{1}{2}\epsilon^2 < 0$. In the latter case, $(\mathbf{U},\mathbf{V})$ is a strict saddle point of f. This completes the proof of Lemma 4.2.

4.7.4 Finishing the Proof of Theorem 4.4

Theorem 4.4 is a direct corollary of Lemma 4.4, Lemma 4.2, and the fact that gradient descent does not converge to a strict saddle point almost surely [48, 60].

4.8 Proof for Rank-1 Matrix Factorization (Theorem 4.5)

In this section we prove Theorem 4.5.

Proof of Theorem 4.5. We define the following four key quantities:

$$\alpha_t = \mathbf{u}_t^\top \mathbf{u}^*, \qquad \alpha_{t,\perp} = \left\|\mathbf{U}_\perp^* \mathbf{u}_t\right\|_2, \qquad \beta_t = \mathbf{v}_t^\top \mathbf{v}^*, \qquad \beta_{t,\perp} = \left\|\mathbf{V}_\perp^* \mathbf{v}_t\right\|_2,$$

where $\mathbf{U}_\perp^* = \mathbf{I} - \mathbf{u}^*\mathbf{u}^{*\top}$ and $\mathbf{V}_\perp^* = \mathbf{I} - \mathbf{v}^*\mathbf{v}^{*\top}$ are the projection matrices onto the orthogonal complement spaces of $\mathbf{u}^*$ and $\mathbf{v}^*$, respectively. Notice that $\|\mathbf{u}_t\|_2^2 = \alpha_t^2 + \alpha_{t,\perp}^2$ and $\|\mathbf{v}_t\|_2^2 = \beta_t^2 + \beta_{t,\perp}^2$. It turns out that we can write down the explicit formulas for the dynamics of these quantities:

$$\begin{aligned}
\alpha_{t+1} &= \left(1-\eta\left(\beta_t^2+\beta_{t,\perp}^2\right)\right)\alpha_t + \eta\sigma_1\beta_t, \qquad & \beta_{t+1} &= \left(1-\eta\left(\alpha_t^2+\alpha_{t,\perp}^2\right)\right)\beta_t + \eta_1\sigma_1\alpha_t, \\
\alpha_{t+1,\perp} &= \left(1-\eta\left(\beta_t^2+\beta_{t,\perp}^2\right)\right)\alpha_{t,\perp}, & \beta_{t+1,\perp} &= \left(1-\eta\left(\alpha_t^2+\alpha_{t,\perp}^2\right)\right)\beta_{t,\perp}.
\end{aligned} \tag{4.23}$$

To facilitate the analysis, we also define:

$$\begin{aligned}
h_t =& \alpha_t\beta_t - \sigma_1, \\
\xi_t =& \alpha_{t,\perp}^2 + \beta_{t,\perp}^2.
\end{aligned}$$

Then our goal is to show $\xi_t \to 0$ and $h_t \to 0$ as $t \to \infty$. We calculate the dynamics of h_t and ξ_t:

$$\begin{aligned}
h_{t+1} &= \left(1-\eta\left(\alpha_t^2+\beta_t^2\right)+\eta^2\left(\alpha_t\beta_t h_t + \alpha_t^2\beta_{t,\perp}^2 + \beta_t^2\alpha_{t,\perp}^2 + \alpha_{t,\perp}^2\beta_{t,\perp}^2\right)\right)h_t - \eta\alpha_t\beta_t\xi_t + \eta^2\sigma_1\alpha_{t,\perp}^2\beta_{t,\perp}^2, \\
\xi_{t+1} &= \left(1-\eta\left(\beta_t^2+\beta_{t,\perp}^2\right)\right)^2\alpha_{t,\perp}^2 + \left(1-\eta\left(\alpha_t^2+\alpha_{t,\perp}^2\right)\right)^2\beta_{t,\perp}^2.
\end{aligned} \tag{4.24}$$

According to our initialization scheme, with high probability we have

$$|\alpha_0|, |\beta_0| \in \left[0.1c_{init}\sqrt{\frac{\sigma_1}{d}}, 10c_{init}\sqrt{\frac{\sigma_1}{d}}\right] \text{ and } |\alpha_{0,\perp}|, |\beta_{0,\perp}| \le 10c_{init}\sqrt{\sigma_1}.$$

We assume that these conditions are satisfied. We also assume that the signal at the beginning is positive: $\alpha_0\beta_0 > 0$, which holds with probability $1/2$. Without loss of generality we assume $\alpha_0, \beta_0 > 0$.[7]

We divide the dynamics into two stages.

Lemma 4.5 (Stage 1: escaping from saddle point $(\mathbf{0}, \mathbf{0})$)**.** *Let* $T_1 = \min\left\{t \in \mathbb{N} : \alpha_t^2 + \beta_t^2 \geq \frac{1}{2}\sigma_1\right\}$. *Then for* $t = 0, 1, \ldots, T_1 - 1$*, the followings hold:*

(i) Positive signal strengths: $\alpha_t, \beta_t > 0$*;*
(ii) Small magnitudes in complement space: $\xi_t \leq \xi_0 \leq 100c_{init}^2\sigma_1$*;*
(iii) Growth of magnitude in signal space: $\left(1 + \frac{c_{step}}{3}\right)(\alpha_t+\beta_t) \leq \alpha_{t+1} + \beta_{t+1} \leq (1 + c_{step})(\alpha_t+\beta_t)$*;*
(iv) Bounded ratio between two layers: $|\alpha_t - \beta_t| \leq \frac{99}{101}(\alpha_t + \beta_t)$.

Furthermore, we have $T_1 = O(\log d)$.

In this stage, the strengths in the complement spaces remain small ($\xi_t \leq \xi_0$) and the strength in the signal space is growing exponentially ($\alpha_{t+1} + \beta_{t+1} \geq \left(1 + \frac{c_{step}}{3}\right)(\alpha_t + \beta_t)$). Furthermore, $|\alpha_t - \beta_t| \leq \frac{99}{101}(\alpha_t + \beta_t)$ implies $\frac{\alpha_t}{\beta_t} \in [\frac{1}{100}, 100]$, which means the signal strengths in the two layers are of the same order.

Then we enter stage 2, which is essentially a local convergence phase. The following lemma characterizes the behaviors of the strengths in the signal and noise spaces in this stage.

Lemma 4.6 (Stage 2: convergence to global minimum)**.** *Let* T_1 *be as defined in Lemma 4.5. Then there exists a universal constant* $c_1 > 0$ *such that the followings hold for all* $t \geq T_1$*:*

(a) Non-vanishing signal strengths in both layers: $\alpha_t, \beta_t \geq \sqrt{c_1\sigma_1}$*;*
(b) Bounded signal strengths: $\alpha_t\beta_t \leq \sigma_1$*, i.e.,* $h_t \leq 0$*;*
(c) Shrinking magnitudes in complement spaces: $\xi_t \leq (1 - c_1c_{step})^{t-T_1}\xi_0 \leq (1 - c_1c_{step})^{t-T_1} \cdot 100c_{init}^2\sigma_1$*;*
(d) Convergence in signal space: $|h_{t+1}| \leq (1 - c_1c_{step})|h_t| + c_{step}\xi_t$.

Note that properties (a) and (b) in Lemma 4.6 imply $c_0 \leq \frac{\alpha_t}{\beta_t} \leq C_0$ for all $t \geq T_1$, where $c_0, C_0 > 0$ are universal constants. Property (c) implies that for all $t \geq T_1 + T_2$ where $T_2 = \Theta(\log \frac{1}{\epsilon})$, we have $\xi_t = O(\epsilon\sigma_1)$. Then property (d) tells us that after another $T_3 = \Theta(\log \frac{1}{\epsilon})$ iterations, we can ensure $|h_t| = O(\epsilon\sigma_1)$ for all $t \geq T_1 + T_2 + T_3$. These imply $\left\|\mathbf{u}_t\mathbf{v}_t^\top - \mathbf{M}^*\right\|_F = O(\epsilon\sigma_1)$ after $t = T_1 + T_2 + T_3 = O(\log \frac{d}{\epsilon})$ iterations, completing the proof of Theorem 4.5. □

Now we prove Lemmas 4.5 and 4.6.

Proof of Lemma 4.5. We use induction to prove the following statements for $t = 0, 1, \ldots, T_1 - 1$:

$$\begin{aligned}
\mathcal{D}(t): &\quad \alpha_t, \beta_t > 0, \\
\mathcal{E}(t): &\quad \xi_t \leq \xi_0 \leq 100c_{init}^2\sigma_1, \\
\mathcal{F}(t): &\quad \left(1 + \frac{c_{step}}{3}\right)(\alpha_t + \beta_t) \leq \alpha_{t+1} + \beta_{t+1} \leq (1 + c_{step})(\alpha_t + \beta_t),
\end{aligned}$$

[7] If $\alpha_0, \beta_0 < 0$, we can simply flip the signs of $\mathbf{u}^*$ and $\mathbf{v}^*$.

$$\mathcal{G}(t): \qquad |\alpha_t - \beta_t| \le \frac{99}{101}(\alpha_t + \beta_t),$$
$$\mathcal{H}(t): \qquad \|\mathbf{u}_t\|^2 + \|\mathbf{v}_t\|^2 \le \sigma_1.$$

- Base cases.
  We know that $\mathcal{D}(0)$, $\mathcal{E}(0)$ and $\mathcal{G}(0)$ hold from our assumptions on the initialization.
- $\mathcal{D}(t), \mathcal{E}(t) \Longrightarrow \mathcal{F}(t)$ $(\forall t \le T_1 - 1)$.
  From (4.23) we have

$$\begin{aligned}
\alpha_{t+1} + \beta_{t+1} &= (1 + \eta\sigma_1)(\alpha_t + \beta_t) - \eta\left(\alpha_t^2 + \alpha_{t,\perp}^2\right)\beta_t - \eta\left(\beta_t^2 + \beta_{t,\perp}^2\right)\alpha_t \\
&\ge \left(1 + \eta\sigma_1 - \eta\left(\alpha_t^2 + \beta_t^2 + \xi_t\right)\right)(\alpha_t + \beta_t) \\
&\ge \left(1 + \eta\sigma_1 - \eta\left(\frac{\sigma_1}{2} + 100c_{init}^2\sigma_1\right)\right)(\alpha_t + \beta_t) \\
&\ge \left(1 + \frac{\eta\sigma_1}{3}\right)(\alpha_t + \beta_t) \\
&= \left(1 + \frac{c_{step}}{3}\right)(\alpha_t + \beta_t),
\end{aligned}$$

  where in the second inequality we have used the definition of T_1, and the last inequality is true when c_{init} is sufficiently small.
  On the other hand we have

$$\begin{aligned}
\alpha_{t+1} + \beta_{t+1} &= (1 + \eta\sigma_1)(\alpha_t + \beta_t) - \eta\left(\alpha_t^2 + \alpha_{t,\perp}^2\right)\beta_t - \eta\left(\beta_t^2 + \beta_{t,\perp}^2\right)\alpha_t \\
&\le (1 + \eta\sigma_1)(\alpha_t + \beta_t) \\
&= (1 + c_{step})(\alpha_t + \beta_t).
\end{aligned}$$

- $\mathcal{E}(t) \Longrightarrow \mathcal{H}(t)$ $(\forall t \le T_1 - 1)$.
  We have

$$\|\mathbf{u}_t\|^2 + \|\mathbf{v}_t\|^2 = \alpha_t^2 + \beta_t^2 + \xi_t \le \frac{1}{2}\sigma_1 + 100c_{init}^2\sigma_1 \le \sigma_1.$$

- $\mathcal{D}(t), \mathcal{H}(t) \Longrightarrow \mathcal{D}(t+1)$ $(\forall t \le T_1 - 1)$.
  From (4.23) we have

$$\alpha_{t+1} = \left(1 - \eta\|\mathbf{v}_t\|^2\right)\alpha_t + \eta\sigma_1\beta_t \ge (1 - \eta\sigma_1)\alpha_t = (1 - c_{step})\alpha_t > 0.$$

  Similarly we have $\beta_{t+1} > 0$. Note that c_{step} is chosen to be sufficiently small.
- $\mathcal{H}(t) \Longrightarrow \mathcal{E}(t+1)$ $(\forall t \le T_1 - 1)$.
  Recall from (4.24):

$$\xi_{t+1} = \left(1 - \eta\|\mathbf{v}_t\|^2\right)^2\alpha_{t,\perp}^2 + \left(1 - \eta\|\mathbf{u}_t\|^2\right)^2\beta_{t,\perp}^2.$$

  Since $\eta\|\mathbf{v}_t\|^2 \le \eta(\|\mathbf{u}_t\|^2 + \|\mathbf{v}_t\|^2) \le \eta\sigma_1 = c_{step} \le 1$ and $\eta\|\mathbf{u}_t\|^2 \le 1$, we have

$$\xi_{t+1} \le \alpha_{t,\perp}^2 + \beta_{t,\perp}^2 = \xi_t.$$

- $\mathcal{D}(t), \mathcal{E}(t), \mathcal{F}(t), \mathcal{G}(t) \Longrightarrow \mathcal{G}(t+1)$ $(\forall t \le T_1 - 1)$.
  From (4.23) we have

$$\begin{aligned}\alpha_{t+1} - \beta_{t+1} &= (1 - \eta\sigma_1)(\alpha_t - \beta_t) - \eta(\beta_t^2 + \beta_{t,\perp}^2)\alpha_t + \eta(\alpha_t^2 + \alpha_{t,\perp}^2)\beta_t \\ &= (1 - \eta\sigma_1 + \eta\alpha_t\beta_t)(\alpha_t - \beta_t) - \eta\beta_{t,\perp}^2\alpha_t + \eta\alpha_{t,\perp}^2\beta_t.\end{aligned}$$

  From $\alpha_t^2 + \beta_t^2 < \frac{1}{2}\sigma_1$ we know $\alpha_t\beta_t < \frac{1}{4}\sigma_1$. Thus

$$\begin{aligned}|\alpha_{t+1} - \beta_{t+1}| &\le (1 - \eta\sigma_1 + \eta\alpha_t\beta_t)\,|\alpha_t - \beta_t| + \eta\beta_{t,\perp}^2\alpha_t + \eta\alpha_{t,\perp}^2\beta_t \\ &\le \left(1 - \frac{3}{4}\eta\sigma_1\right)|\alpha_t - \beta_t| + \eta\xi_t(\alpha_t + \beta_t) \\ &\le \left(1 - \frac{3}{4}\eta\sigma_1\right)\cdot\frac{99}{101}(\alpha_t + \beta_t) + \eta\cdot 100c_{init}^2\sigma_1(\alpha_t + \beta_t) \\ &\le \left(1 - \eta\sigma_1\left(\frac{3}{4} - 100c_{init}^2\cdot\frac{101}{99}\right)\right)\cdot\frac{99}{101}(\alpha_t + \beta_t) \\ &\le \frac{99}{101}(\alpha_t + \beta_t) \\ &\le \frac{99}{101}(\alpha_{t+1} + \beta_{t+1}).\end{aligned}$$

  Lastly we upper bound T_1. Note that for all $t < T_1$ we have $\alpha_t + \beta_t \le \sqrt{2(\alpha_t^2 + \beta_t^2)} < \sqrt{2\cdot\frac{1}{2}\sigma_1} = \sqrt{\sigma_1}$. From $\mathcal{F}(t)$ we know that $\alpha_t + \beta_t$ is increasing exponentially. Therefore, we must have $T_1 = O\left(\log\frac{\sqrt{\sigma_1}}{\alpha_0+\beta_0}\right) = O\left(\log\frac{\sqrt{\sigma_1}}{\sqrt{\sigma_1/d}}\right) = O(\log d)$. □

Proof of Lemma 4.6. By the definition of T_1 we know $\alpha_{T_1}^2 + \beta_{T_1}^2 \ge \frac{1}{2}\sigma_1$. In the proof of Lemma 4.5, we have shown $\alpha_{T_1}, \beta_{T_1} > 0$ and $|\alpha_{T_1} - \beta_{T_1}| \le \frac{99}{101}(\alpha_{T_1} + \beta_{T_1})$. These imply $\min\{\alpha_{T_1}, \beta_{T_1}\} \ge 2\sqrt{c_1\sigma_1}$ for some small universal constant $c_1 > 0$.

We use induction to prove the following statements for all $t \ge T_1$:

$$\begin{aligned}\mathcal{I}(t):&\quad \alpha_t \ge \alpha_{T_1}\cdot\prod_{i=T_1}^{t-1}\left(1 - \eta\xi_0\left(1 - c_1c_{step}\right)^{i-T_1}\right), \\ &\quad \beta_t \ge \beta_{T_1}\cdot\prod_{i=T_1}^{t-1}\left(1 - \eta\xi_0\left(1 - c_1c_{step}\right)^{i-T_1}\right), \\ \mathcal{J}(t):&\quad \alpha_t, \beta_t \ge \sqrt{c_1\sigma_1}, \\ \mathcal{K}(t):&\quad \alpha_t\beta_t \le \sigma_1, \text{ i.e., } h_t \le 0, \\ \mathcal{L}(t):&\quad \xi_t \le (1 - c_1c_{step})^{t-T_1}\xi_0 \le (1 - c_1c_{step})^{t-T_1}\cdot 100c_{init}^2\sigma_1, \\ \mathcal{M}(t):&\quad |h_{t+1}| \le (1 - c_1c_{step})|h_t| + c_{step}\xi_t.\end{aligned}$$

- Base cases.

$\mathcal{I}(T_1)$ is obvious. We know that $\mathcal{J}(T_1)$ is true by the definition of c_1. $\mathcal{K}(T_1)$ can be shown as follows:

$$\begin{aligned}
\alpha_{T_1}\beta_{T_1} &\le \frac{1}{4}\left(\alpha_{T_1}+\beta_{T_1}\right)^2 \\
&\le \frac{1}{4}(1+c_{step})^2\left(\alpha_{T_1-1}+\beta_{T_1-1}\right)^2 && \text{(by Lemma 4.5 (iii))} \\
&\le \frac{1}{4}(1+c_{step})^2\cdot 2\left(\alpha_{T_1-1}^2+\beta_{T_1-1}^2\right) \\
&\le \frac{1}{4}(1+c_{step})^2\cdot 2\cdot\frac{1}{2}\sigma_1 && \text{(by the definition of } T_1) \\
&\le \sigma_1. && \text{(choosing } c_{step} \text{ to be small)}
\end{aligned}$$

$\mathcal{L}(T_1)$ reduces to $\xi_{T_1} \le \xi_0$, which was shown in the proof of Lemma 4.5.

- $\mathcal{I}(t) \Longrightarrow \mathcal{J}(t)$ $(\forall t \ge T_1)$.
  Notice that we have $\eta\xi_0 \le \frac{c_{step}}{\sigma_1}\cdot 100c_{init}^2\sigma_1 = 100c_{step}c_{init}^2 < \frac{1}{2}$ since c_{step} and c_{init} are sufficiently small. Then we have

$$\begin{aligned}
\alpha_t &\ge \alpha_{T_1}\cdot\prod_{i=T_1}^{t-1}\left(1-\eta\xi_0\left(1-c_1c_{step}\right)^{i-T_1}\right) \\
&\ge \alpha_{T_1}\cdot\prod_{i=0}^{\infty}\left(1-\eta\xi_0\left(1-c_1c_{step}\right)^{i}\right) \\
&\ge \alpha_{T_1}\cdot\prod_{i=0}^{\infty}\exp\left(-2\eta\xi_0\left(1-c_1c_{step}\right)^{i}\right) && (1-x\ge e^{-2x}, \forall 0\le x\le 1/2) \\
&= \alpha_{T_1}\cdot\exp\left(-\frac{2\eta\xi_0}{c_1c_{step}}\right) \\
&\ge \alpha_{T_1}\cdot\exp\left(-\frac{200c_{step}c_{init}^2}{c_1c_{step}}\right) \\
&\ge 2\sqrt{c_1\sigma_1}\cdot\exp\left(-\frac{200c_{init}^2}{c_1}\right) \\
&\ge \sqrt{c_1\sigma_1}. && \text{(choosing } c_{init} \text{ to be small)}
\end{aligned}$$

  Similarly we have $\beta_t \ge \sqrt{c_1\sigma_1}$.

- $\mathcal{I}(t), \mathcal{J}(t), \mathcal{K}(t), \mathcal{L}(t) \Longrightarrow \mathcal{I}(t+1)$ $(\forall t \ge T_1)$.
  From (4.23) we have

$$\begin{aligned}
\alpha_{t+1} &= \left(1-\eta\left(\beta_t^2+\beta_{t,\perp}^2\right)\right)\alpha_t+\eta\sigma_1\beta_t \\
&= \left(1-\eta\beta_{t,\perp}^2\right)\alpha_t-\eta h_t\beta_t \\
&\ge \left(1-\eta\beta_{t,\perp}^2\right)\alpha_t && (h_t\le 0, \beta_t>0) \\
&\ge \left(1-\eta\xi_t\right)\alpha_t \\
&\ge \left(1-\eta\xi_0(1-c_1c_{step})^{t-T_1}\right)\alpha_t && (\mathcal{L}(t))
\end{aligned}$$

$$\geq \alpha_{T_1} \cdot \prod_{i=T_1}^{t} \left(1 - \eta\xi_0 \left(1 - c_1 c_{step}\right)^{i-T_1}\right). \qquad (\mathcal{I}(t))$$

Similarly we have $\beta_{t+1} \geq \beta_{T_1} \cdot \prod_{i=T_1}^{t} \left(1 - \eta\xi_0 \left(1 - c_1 c_{step}\right)^{i-T_1}\right)$.

- $\mathcal{J}(t), \mathcal{K}(t), \mathcal{L}(t) \Longrightarrow \mathcal{K}(t+1)$ $(\forall t \geq T_1)$.
  From (4.24) we have

$$\begin{aligned} h_{t+1} &= \left(1 - \eta\left(\alpha_t^2 + \beta_t^2\right) + \eta^2\left(\alpha_t\beta_t h_t + \alpha_t^2\beta_{t,\perp}^2 + \beta_t^2\alpha_{t,\perp}^2 + \alpha_{t,\perp}^2\beta_{t,\perp}^2\right)\right) h_t - \eta\alpha_t\beta_t\xi_t \\ &\quad + \eta^2\sigma_1\alpha_{t,\perp}^2\beta_{t,\perp}^2 \\ &\leq \left(1 - \eta\left(\alpha_t^2 + \beta_t^2\right)\right) h_t + \eta^2\alpha_t\beta_t h_t^2 - \eta\alpha_t\beta_t\xi_t + \eta^2\sigma_1\alpha_{t,\perp}^2\beta_{t,\perp}^2, \end{aligned} \qquad (4.25)$$

  where we have used $h_t \leq 0$. Since $\alpha_t, \beta_t \geq \sqrt{c_1\sigma_1}$ and $\alpha_t\beta_t \leq \sigma_1$, we have $\alpha_t, \beta_t = \Theta(\sqrt{\sigma_1})$. Furthermore, we can choose c_{step} and c_{init} small enough such that $\eta\xi_0 \leq 4c_1$ which implies

$$\eta^2\sigma_1\alpha_{t,\perp}^2\beta_{t,\perp}^2 \leq \eta^2\sigma_1 \cdot \frac{1}{4}\xi_t^2 \leq \frac{1}{4}\eta\xi_t \cdot \eta\sigma_1\xi_0 \leq \eta\xi_t \cdot c_1\sigma_1 \leq \eta\xi_t \cdot \alpha_t\beta_t.$$

  Therefore (4.25) implies

$$\begin{aligned} h_{t+1} &\leq \left(1 - \eta \cdot O(\sigma_1)\right) h_t + \eta^2\alpha_t\beta_t h_t^2 \\ &= \left(1 - \eta \cdot O(\sigma_1) + \eta^2\alpha_t\beta_t h_t\right) h_t \\ &\leq \left(1 - \eta \cdot O(\sigma_1) - \eta^2\sigma_1^2\right) h_t \qquad (0 < \alpha_t\beta_t \leq \sigma_1) \\ &= \left(1 - O(c_{step}) - c_{step}^2\right) h_t \\ &\leq 0, \end{aligned}$$

  where the last step is true when c_{step} is sufficiently small.
- $\mathcal{J}(t), \mathcal{K}(t), \mathcal{L}(t) \Longrightarrow \mathcal{L}(t+1)$ $(\forall t \geq T_1)$.
  From $\alpha_t, \beta_t \geq \sqrt{c_1\sigma_1}$ and $\alpha_t\beta_t \leq \sigma_1$ we have $\alpha_t, \beta_t = \Theta(\sqrt{\sigma_1})$. Also we have $\xi_t \leq \xi_0$. Thus we can make sure $\eta(\alpha_t^2 + \alpha_{t,\perp}^2) < 1$ and $\eta(\beta_t^2 + \beta_{t,\perp}^2) < 1$. Then from (4.24) we have

$$\begin{aligned} \xi_{t+1} &= \left(1 - \eta\left(\beta_t^2 + \beta_{t,\perp}^2\right)\right)^2 \alpha_{t,\perp}^2 + \left(1 - \eta\left(\alpha_t^2 + \alpha_{t,\perp}^2\right)\right)^2 \beta_{t,\perp}^2 \\ &\leq \left(1 - \eta\beta_t^2\right)^2 \alpha_{t,\perp}^2 + \left(1 - \eta\alpha_t^2\right) \beta_{t,\perp}^2 \\ &\leq \left(1 - \eta c_1\sigma_1\right) \xi_t \\ &= \left(1 - c_1 c_{step}\right) \xi_t. \end{aligned}$$

- We have shown $\mathcal{I}(t), \mathcal{J}(t), \mathcal{K}(t)$ and $\mathcal{L}(t)$ for all $t \geq T_1$. Now we use them to prove $\mathcal{M}(t)$ for all $t \geq T_1$:

$$\begin{aligned} |h_{t+1}| &= \left(1 - \eta\left(\alpha_t^2 + \beta_t^2\right) + \eta^2\left(\alpha_t\beta_t h_t + \alpha_t^2\beta_{t,\perp}^2 + \beta_t^2\alpha_{t,\perp}^2 + \alpha_{t,\perp}^2\beta_{t,\perp}^2\right)\right) |h_t| + \eta\alpha_t\beta_t\xi_t \\ &\quad - \eta^2\sigma_1\alpha_{t,\perp}^2\beta_{t,\perp}^2 \end{aligned}$$

$$
\begin{aligned}
&\leq \left(1 - \frac{1}{2}\eta\left(\alpha_t^2 + \beta_t^2\right)\right)|h_t| + \eta\alpha_t\beta_t\xi_t \\
&\leq \left(1 - \frac{1}{2}\eta \cdot 2c_1\sigma_1\right)|h_t| + \eta\sigma_1\xi_t \\
&= \left(1 - c_1 c_{step}\right)|h_t| + c_{step}\xi_t.
\end{aligned}
$$

Here we have used $\eta \leq \frac{\alpha_t^2+\beta_t^2}{2\left|\alpha_t\beta_t h_t+\alpha_t^2\beta_{t,\perp}^2+\beta_t^2\alpha_{t,\perp}^2+\alpha_{t,\perp}^2\beta_{t,\perp}^2\right|}$, which is clearly true when c_{step} is small enough.

Therefore, we have finished the proof of Lemma 4.6. □

Part II

Parameter Estimation in Convolutional Neural Networks via Gradient Descent

Chapter 5

Learning a Two-layer Convolutional Neural Network via Gradient Descent

5.1 Introduction

In the previous part, we only consider the optimization aspect of gradient descent for neural networks. It is not clear whether the neural network learned by gradient descent can have good generalization ability. Note in general generalization is not possible unless one puts some assumptions. In this part, we assume there exists an underlying convolutional neural network with good generalization ability and our goal is to recover it. A line of research [10, 51, 64, 67, 73] assumed the input distribution is Gaussian and showed that gradient descent with random or $\mathbf{0}$ initialization is able to train a recover a neural network of the form $f(\mathbf{x}, \{\mathbf{w}_j\}) = \sum_j a_j \sigma(\mathbf{w}_j^T \mathbf{x})$ with ReLU activation $\sigma(z) = \max(z, 0)$ where $\mathbf{x}$ is the input, $\mathbf{w}_j$ is the weight vector and a_j is the output weight. 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:

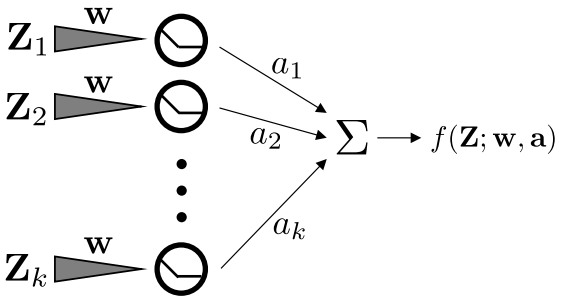
Does randomly initialized gradient descent recover neural networks with multiple layers?

In this chapter, 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 $\mathbf{a}$. 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 q: $\mathbf{Z} \in \mathbb{R}^{q \times k}$ 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:

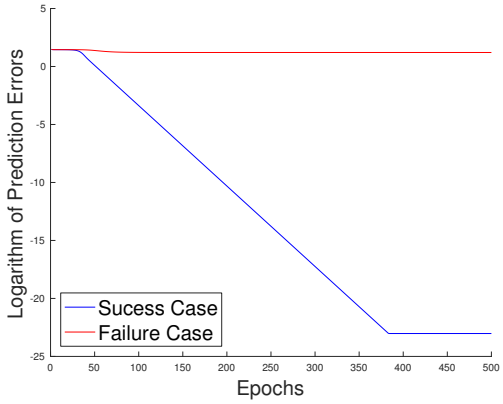
$$f(\mathbf{Z}, \mathbf{w}, \mathbf{a}) = \sum_{i=1}^{k} a_i \sigma\left(\mathbf{w}^\top \mathbf{Z}_i\right).$$

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:

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



(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 5.1a 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 $\sim$ 50 iterations the convergence is slow. After that gradient descent converges at a fast linear rate.

Figure 5.1: Network architecture that we consider in this chapter and convergence of gradient descent for learning the parameters of this network.

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 [10, 85]. Following [10, 51, 64, 73, 84, 85], in this chapter, 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.

An important 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 [59], 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 and Kingma [63]. As reported in [63], 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.

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**
  $\mathbf{v}^{t+1} \leftarrow \mathbf{v}^t - \eta \frac{\partial \ell\left(\mathbf{v}^t, \mathbf{a}^t\right)}{\partial \mathbf{v}^t}$,
  $\mathbf{a}^{t+1} \leftarrow \mathbf{a}^t - \eta \frac{\partial \ell\left(\mathbf{v}^t, \mathbf{a}^t\right)}{\partial \mathbf{a}^t}$.
end for

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

$$f(\mathbf{Z}, \mathbf{v}, \mathbf{a}) = \sum_{i=1}^{k} a_i \frac{\sigma\left(\mathbf{Z}_i^\top \mathbf{v}\right)}{\|\mathbf{v}\|_2}. \tag{5.1}$$

The loss function is

$$\ell(\mathbf{v}, \mathbf{a}) = \frac{1}{2}\mathbb{E}_{\mathbf{Z}}\left[\left(f(\mathbf{Z}, \mathbf{v}, \mathbf{a}) - f(\mathbf{Z}, \mathbf{v}^*, \mathbf{a}^*)\right)^2\right]. \tag{5.2}$$

In this chapter 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. This chapter have three contributions to the literature. 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 (5.2)) *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 [8, 37, 38, 39, 46, 49], our result provides a concrete counter-example and highlights a conceptually surprising phenomenon:

Randomly initialized local search can find a global 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 5.1b), 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 5.1b). The

[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 chapter, with a little abuse of the notation, we use $\mathbf{v}^*$ to denote the equivalent class of optimal solutions.

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:

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

for some absolute constant $c > 0$.

While it is hard to characterize $\mathbf{B}(\mathbf{v}^*, \mathbf{a}^*)$, we find that the set $\tilde{\mathbf{B}}(\mathbf{v}^*, \mathbf{a}^*) \equiv \{(\mathbf{v}^0, \mathbf{a}^0) : (\mathbf{v}^0)^\top \mathbf{v}^* \geq 0, (\mathbf{a}^0)^\top \mathbf{a}^* \geq 0, \left|\mathbf{1}^\top \mathbf{a}^0\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|\}$ 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:

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

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).

5.2 Preliminaries

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. 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 chapter, 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 5.7 for details.

Theorem 5.1. *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*

$$\begin{aligned}\ell(\mathbf{v}, \mathbf{a}) =& \frac{1}{2}\left[\frac{(\pi-1)\|\mathbf{w}^*\|_2^2}{2\pi}\|\mathbf{a}^*\|_2^2 + \frac{(\pi-1)}{2\pi}\|\mathbf{a}\|_2^2\right. \\ &\left. - \frac{2(g(\phi)-1)\|\mathbf{w}^*\|_2}{2\pi}\mathbf{a}^\top\mathbf{a}^* + \frac{\|\mathbf{w}^*\|_2^2}{2\pi}\left(\mathbf{1}^\top\mathbf{a}^*\right)^2 + \frac{1}{2\pi}\left(\mathbf{1}^\top\mathbf{a}\right)^2 - 2\|\mathbf{w}^*\|_2 \mathbf{1}^\top\mathbf{a}\cdot\mathbf{1}^\top\mathbf{a}^*\right]\end{aligned} \tag{5.3}$$

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 5.2. *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*

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

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 [10, 13]. 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.

5.3 Main Result

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

Theorem 5.3. *Suppose the initialization satisfies* $(\mathbf{a}^0)^\top\mathbf{a}^* > 0$*,* $\left|\mathbf{1}^\top\mathbf{a}^0\right| \le \left|\mathbf{1}^\top\mathbf{a}^*\right|$*,* $\phi^0 < \pi/2$ *and step size satisfies*

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

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* $\left(\mathbf{a}^{T_1}\right)^\top\mathbf{a}^*\|\mathbf{w}^*\|_2 = \Theta\left(\|\mathbf{a}^*\|_2^2\|\mathbf{w}^*\|_2^2\right)$ *where*

$$\beta^0 = \min\left\{\left(\mathbf{a}^0\right)^\top\mathbf{a}^*\|\mathbf{w}^*\|_2, (g(\phi^0) - 1)\|\mathbf{a}^*\|_2^2\|\mathbf{a}^*\|_2^2\right\}.$$

(Phase II: Fast Rate) *Suppose at the* T_1*-th iteration,* $\phi^{T_1} = \Theta(1)$ *and* $\left(\mathbf{a}^{T_1}\right)^\top\mathbf{a}^*\|\mathbf{w}^*\|_2 = \Theta\left(\|\mathbf{a}^*\|_2^2\|\mathbf{w}^*\|_2^2\right)$*, then there exists* $T_2 = \widetilde{O}(\left(\frac{1}{\eta\|\mathbf{w}^*\|_2^2\|\mathbf{a}^*\|_2^2} + \frac{1}{\eta}\right)\log\left(\frac{1}{\epsilon}\right))^2$ *such that*

$$\ell\left(\mathbf{v}^{T_1+T_2}, \mathbf{a}^{T_1+T_2}\right) \le \epsilon\|\mathbf{w}^*\|_2^2\|\mathbf{a}^*\|_2^2.$$

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

Theorem 5.3 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 5.3 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 5.3 and a large enough initial signal.

Theorem 5.4. *Let* $\mathbf{v} \sim \mathrm{unif}\left(\mathcal{S}^{p-1}\right)$ *and* $\mathbf{a} \sim \mathrm{unif}\left(\mathcal{B}\left(\mathbf{0}, \frac{\left|\mathbf{1}^\top\mathbf{a}^*\right|\|\mathbf{w}^*\|_2}{\sqrt{k}}\right)\right)$*, 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* $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\|_2 = \Theta\left(\frac{\left|\mathbf{1}^\top\mathbf{a}^*\right|\|\mathbf{a}^*\|_2\|\mathbf{w}^*\|_2^2}{k}\right)$*, and* $\phi^0 = \Theta\left(\frac{1}{\sqrt{p}}\right)$.

Theorem 5.4 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 $O(1/\mathrm{poly}(k, \left\|\mathbf{w}^*\right\|_2 \left\|\mathbf{a}\right\|_2))$ 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 5.3 and Theorem 5.4 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 5.4 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 [42] for example.

Remark 5.1. *For the second layer we use* $O\left(\frac{1}{\sqrt{k}}\right)$ *type initialization, verifying common initialization techniques [40, 43, 47].*

Remark 5.2. *The Gaussian input assumption is not necessarily true in practice, although this is a common assumption appeared in the previous papers [10, 51, 64, 73, 79, 84, 85] and also considered plausible in [14]. Our result can be easily generalized to rotation invariant distributions. However, extending to more general distributional assumption, e.g., structural conditions used in [31] remains a challenging open problem.*

Remark 5.3. *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.*

5.3.1 Gradient Descent Can Converge to the Spurious Local Minimum

Theorem 5.4 shows that among $\left\{\left(\mathbf{v}, \mathbf{a}\right), \left(\mathbf{v}, -\mathbf{a}\right), \left(-\mathbf{v}, \mathbf{a}\right), \left(-\mathbf{v}, -\mathbf{a}\right)\right\}$, there is a pair that enables gradient descent to converge to the global minimum. Perhaps surprisingly, the next the-

orem shows that under some conditions of the underlying truth, there is also a pair that makes gradient descent converge to the spurious local minimum.

Theorem 5.5. *Without loss of generality, we let* $\|\mathbf{w}^*\|_2 = 1$. *Suppose* $\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 < \frac{1}{\mathrm{poly}(q)} \|\mathbf{a}^*\|_2^2$ *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* $(\mathbf{v}^0, \mathbf{a}^0) \in \left\{(\mathbf{v}, \mathbf{a}), (\mathbf{v}, -\mathbf{a}), (-\mathbf{v}, \mathbf{a}), (-\mathbf{v}, -\mathbf{a})\right\}$ *that* $\left(\mathbf{a}^0\right)^\top \mathbf{a}^* < 0$, $\left|\mathbf{1}^\top \mathbf{a}^0\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$, $g\left(\phi^0\right) \leq \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2} + 1$. *If* $(\mathbf{v}^0, \mathbf{a}^0)$ *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 5.3 which requires no assumption on the underlying truth $\mathbf{a}^*$, Theorem 5.5 assumes $\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 < \frac{1}{\mathrm{poly}(q)} \|\mathbf{a}^*\|_2^2$. This technical condition comes from the proof which requires invariance $g(\phi^t) \leq \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$ for all iterations. To ensure there exists $(\mathbf{v}^0, \mathbf{a}^0)$ which makes $g(\phi^0) \leq \frac{-2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$, we need $\frac{\left(\mathbf{1}^\top \mathbf{a}^*\right)^2}{\|\mathbf{a}^*\|_2^2}$ relatively small. See Section 5.11 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 5.5.

5.4 Proof Sketch

In Section 5.4.1, we give qualitative high level intuition on why the initial conditions are sufficient for gradient descent to converge to the global minimum. In Section 5.4.2, we explain why the gradient descent has two phases.

5.4.1 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 5.2.

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*

$$\theta\left(\mathbf{v}, \mathbf{w}^*\right) = 0, \mathbf{a} = \|\mathbf{w}^*\|_2 \mathbf{a}^* \text{ or } \theta\left(\mathbf{v}, \mathbf{w}^*\right) = \pi,$$
$$\mathbf{a} = \left(\mathbf{1}\mathbf{1}^\top + (\pi - 1)\mathbf{I}\right)^{-1}\left(\mathbf{1}\mathbf{1}^\top - \mathbf{I}\right)\|\mathbf{w}^*\|_2 \mathbf{a}^*.$$

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 $\mathbf{v}$: $\frac{\partial \ell(\mathbf{v},\mathbf{a})}{\partial \mathbf{v}} = -\frac{1}{2\pi\|\mathbf{v}\|_2}\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{a}^\top\mathbf{a}^*\left(\pi - \phi\right)\mathbf{w}^*$. Notice that $\phi \leq \pi$ 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 an important 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: The Angle between $\mathbf{v}$ and $\mathbf{w}^*$ always decreases.)**.** *If* $\left(\mathbf{a}^t\right)^\top\mathbf{a}^* > 0$*, then* $\phi^{t+1} \leq \phi^t$.

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

$$\begin{aligned}&\left(\mathbf{a}^{t+1}\right)^\top\mathbf{a}^* \\ &= \left(1 - \frac{\eta(\pi-1)}{2\pi}\right)\left(\mathbf{a}^t\right)^\top\mathbf{a}^* + \frac{\eta(g(\phi^t)-1)}{2\pi}\left\|\mathbf{a}^t\right\|_2^2 + \frac{\eta}{2\pi}\left(\left(\mathbf{1}^\top\mathbf{a}^*\right)^2 - \left(\mathbf{1}^\top\mathbf{a}^t\right)\left(\mathbf{1}^\top\mathbf{a}^*\right)\right).\end{aligned} \tag{5.4}$$

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 (5.4) 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* $\left(\mathbf{a}^t\right)^\top\mathbf{a}^* > 0$*,* $0 \leq \mathbf{1}^\top\mathbf{a}^* \cdot \mathbf{1}^\top\mathbf{a}^t \leq \left(\mathbf{1}^\top\mathbf{a}^*\right)^2$*,* $0 < \phi^t < \pi/2$ *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

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

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

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

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 5.4 shows these three conditions are true with constant probability using random initialization.

5.4.2 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 5.4.1 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 chapter, 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 5.3. Let* $\beta^0 = \min\left\{(\mathbf{a}^0)^\top \mathbf{a}^*, (g(\phi^0)-1)\left\|\mathbf{a}^*\right\|_2^2\right\}\left\|\mathbf{w}^*\right\|_2^2$. *If the step size satisfies* $\eta = O\left(\min\left\{\frac{\beta^0\cos\phi^0}{\left(\left\|\mathbf{a}^*\right\|_2^2+\left(\mathbf{1}^\top\mathbf{a}^*\right)^2\right)\left\|\mathbf{w}^*\right\|_2^2}, \frac{\cos\phi^0}{\left(\left\|\mathbf{a}^*\right\|_2^2+\left(\mathbf{1}^\top\mathbf{a}^*\right)^2\right)\left\|\mathbf{w}^*\right\|_2^2}, \frac{1}{k}\right\}\right)$, *we have*

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

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

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 5.9 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 $\cos\phi^t = \Omega\left(1\right)$. Once $\cos\phi^t = \Omega\left(1\right)$, Lemma 5.2 shows, after $O\left(\frac{1}{\eta}\right)$ iterations, $\left(\mathbf{a}^t\right)\mathbf{a}^*\left\|\mathbf{w}^*\right\| = \Omega\left(\left\|\mathbf{w}^*\right\|_2^2\left\|\mathbf{a}^*\right\|_2^2\right)$. Combining the facts $\left\|\mathbf{v}^t\right\|_2 \le 2$ (Lemma 5.9) and $\phi^t < \pi/2$, we have $\cos\phi^t\lambda^t = \Omega\left(\left\|\mathbf{w}^*\right\|_2^2\left\|\mathbf{a}^*\right\|_2^2\right)$. Now we enter phase II.

In phase II, Lemma 5.5 shows

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

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

Once we have this, we can use Lemma 5.10 to show $\left|\mathbf{1}^\top\mathbf{a}^* - \mathbf{1}^\top\mathbf{a}\right| \le O\left(\epsilon\left\|\mathbf{a}^*\right\|_2\right)$ after $\widetilde{O}(\frac{1}{\eta k}\log\left(\frac{1}{\epsilon}\right))$ iterations. Next, using Lemma 5.11, we can show after $\widetilde{O}\left(\frac{1}{\eta}\log\frac{1}{\epsilon}\right)$ iterations, $\left\|\mathbf{a}-\mathbf{a}^*\right\|_2 = O\left(\epsilon\left\|\mathbf{a}^*\right\|_2\right)$. Lastly, Lemma 5.12 shows if $\left\|\mathbf{a}-\mathbf{a}^*\right\|_2 = O\left(\epsilon\left\|\mathbf{a}^*\right\|_2\right)$ and $\phi = O\left(\epsilon\right)$ we have we have $\ell\left(\mathbf{v},\mathbf{a}\right) = O\left(\epsilon\left\|\mathbf{a}^*\right\|_2^2\right)$.

5.5 Experiments

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

5.5.1 Multi-phase Phenomenon

In Figure 5.2, we set $k = 20$, $p = 25$ and we consider 4 key quantities in proving Theorem 5.3, namely, angle between $\mathbf{v}$ and $\mathbf{w}^*$ (c.f. Lemma 5.5), $\left\|\mathbf{a}-\mathbf{a}^*\right\|$ (c.f. Lemma 5.11), $\left|\mathbf{1}^\top\mathbf{a}-\mathbf{1}^\top\mathbf{a}^*\right|$ (c.f. Lemma 5.10) and prediction error (c.f. Lemma 5.12).

When we achieve the global minimum, all these quantities are 0. At the beginning (first $\sim$ 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 $\sim$ 200 iterations, all quantities decrease at a slow rate. This phenomenon is explained to the Phase I stage in Theorem 5.3. The rate is slow because the initial signal is small.

After $\sim$ 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 5.3).

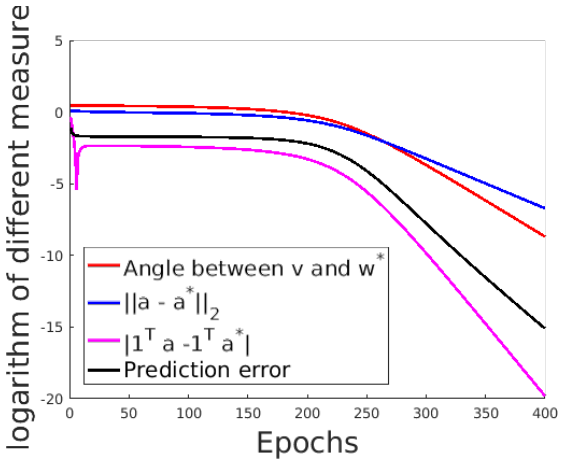


Figure 5.2: Convergence of different measures we considered in proving Theorem 5.3. In the first $\sim$ 200 iterations, all quantities drop slowly. After that, these quantities converge at much faster linear rates.

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 5.1: 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.

5.5.2 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 5.4. 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 5.5, 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 5.1 empirically verifies the importance of this assumption. For 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.

5.6 Conclusion and Future Work

In this chapter we proved the 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.

One interesting direction is to generalize our result to deeper 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 [84, 85]. We believe the insights from this chapter, especially the invariance principles in Section 5.4.1 are helpful to understand the behaviors of gradient-based algorithms in these settings.

Appendix: Omitted Proofs

5.7 Proofs of Section 5.2

Proof of Theorem 5.1. We first expand the loss function directly.

$$
\begin{aligned}
&\ell\left(\mathbf{v}, \mathbf{a}\right) \\
=&\mathbb{E}\left[\frac{1}{2}\left(y-\mathbf{a}^{\top} \sigma\left(\mathbf{Z}\right) \mathbf{w}\right)^{2}\right] \\
=&\left(\mathbf{a}^{*}\right)^{\top} \mathbb{E}\left[\sigma\left(\mathbf{Z} \mathbf{w}^{*}\right) \sigma\left(\mathbf{Z} \mathbf{w}^{*}\right)^{\top}\right] \mathbf{a}^{*}+\mathbf{a}^{\top} \mathbb{E}\left[\sigma\left(\mathbf{Z} \mathbf{w}\right) \sigma\left(\mathbf{Z} \mathbf{w}\right)^{\top}\right] \mathbf{a}-2 \mathbf{a}^{\top} \mathbb{E}\left[\sigma\left(\mathbf{Z} \mathbf{w}\right) \sigma\left(\mathbf{Z} \mathbf{w}^{*}\right)^{\top}\right] \mathbf{a}^{*} \\
=&\left(\mathbf{a}^{*}\right)^{\top} \mathbf{A}\left(\mathbf{w}^{*}\right) \mathbf{a}^{*}+\mathbf{a}^{\top} \mathbf{A}\left(\mathbf{w}\right) \mathbf{a}-2 \mathbf{a}^{\top} \mathbf{B}\left(\mathbf{w}, \mathbf{w}^{*}\right) \mathbf{w}^{*}.
\end{aligned}
$$

where for simplicity, we denote

$$
\mathbf{A}(\mathbf{w}) =\mathbb{E}\left[\sigma\left(\mathbf{Z} \mathbf{w}\right) \sigma\left(\mathbf{Z} \mathbf{w}\right)^{\top}\right] \tag{5.5}
$$

$$
\mathbf{B}\left(\mathbf{w}, \mathbf{w}^{*}\right) =\mathbb{E}\left[\sigma\left(\mathbf{Z} \mathbf{w}\right) \sigma\left(\mathbf{Z} \mathbf{w}^{*}\right)^{\top}\right]. \tag{5.6}
$$

For $i \neq j$, using the second identity of Lemma 5.6, we can compute

$$
\mathbf{A}(\mathbf{w})_{ij}=\mathbb{E}\left[\sigma\left(\mathbf{Z}_{i}^{\top} \mathbf{w}\right)\right] \mathbb{E}\left[\sigma\left(\mathbf{Z}_{j}^{\top} \mathbf{w}\right)\right]=\frac{1}{2 \pi}\|\mathbf{w}\|_{2}^{2}
$$

For $i=j$, using the second moment formula of half-Gaussian distribution we can compute

$$
\mathbf{A}\left(\mathbf{w}\right)_{ii}=\frac{1}{2}\|\mathbf{w}\|_{2}^{2}.
$$

Therefore

$$\mathbf{A}(\mathbf{w}) = \frac{1}{2\pi} \|\mathbf{w}\|_2^2 \left(\mathbf{1}\mathbf{1}^\top + (\pi - 1)\,\mathbf{I}\right).$$

Now let us compute $\mathbf{B}(\mathbf{w}, \mathbf{w}_*)$. For $i \neq j$, similar to $\mathbf{A}(\mathbf{w})_{ij}$, using the independence property of Gaussian, we have

$$\mathbf{B}(\mathbf{w}, \mathbf{w}_*)_{ij} = \frac{1}{2\pi} \|\mathbf{w}\|_2 \|\mathbf{w}^*\|_2.$$

Next, using the fourth identity of Lemma 5.6, we have

$$\mathbf{B}(\mathbf{w}, \mathbf{w}^*)_{ii} = \frac{1}{2\pi} (\cos\phi\,(\pi - \phi) + \sin\phi) \|\mathbf{w}\|_2 \|\mathbf{w}^*\|_2.$$

Therefore, we can also write $\mathbf{B}(\mathbf{w}, \mathbf{w}^*)$ in a compact form

$$\mathbf{B}(\mathbf{w}, \mathbf{w}^*) = \frac{1}{2\pi} \|\mathbf{w}\|_2 \|\mathbf{w}^*\|_2 \left(\mathbf{1}\mathbf{1}^\top + (\cos\phi\,(\pi - \phi) + \sin\phi - 1)\,\mathbf{I}\right).$$

Plugging in the formulas of $\mathbf{A}(\mathbf{w})$ and $\mathbf{B}(\mathbf{w}, \mathbf{w}^*)$ and $\mathbf{w} = \frac{\mathbf{v}}{\|\mathbf{v}\|_2}$, we obtain the desired result. □

Proof of Theorem 5.2. We first compute the expect gradient for $\mathbf{v}$. From[63], we know

$$\frac{\partial \ell(\mathbf{v}, \mathbf{a})}{\partial \mathbf{v}} = \frac{1}{\|\mathbf{v}\|_2} \left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right) \frac{\partial \ell(\mathbf{w}, \mathbf{a})}{\partial \mathbf{w}}.$$

Recall the gradient formula,

$$\begin{aligned}
&\frac{\partial \ell(\mathbf{Z}, \mathbf{w}, \mathbf{a})}{\partial \mathbf{w}} \\
&= \left(\sum_{i=1}^k a_i^* \sigma(\mathbf{Z}_i \mathbf{w}) - \sum_{i=1}^k a_i^* \sigma(\mathbf{Z}\mathbf{w}^*)\right) \left(\sum_{i=1}^k a_i \mathbf{Z}_i \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w}\right\}\right) \\
&= \left(\sum_{i=1}^k a_i^2 \mathbf{Z}_i \mathbf{Z}_i^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0\right\} + \sum_{i \neq j} a_i a_j \mathbf{Z}_i \mathbf{Z}_j^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_j^\top \mathbf{w} \geq 0\right\}\right) \mathbf{w} \qquad (5.7) \\
&- \left(\sum_{i=1}^k a_i a_i^* \mathbf{Z}_i \mathbf{Z}_i^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_i^\top \mathbf{w}^* \geq 0\right\} + \sum_{i \neq j} a_i a_j^* \mathbf{Z}_i \mathbf{Z}_j^* \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_j^\top \mathbf{w}^* \geq 0\right\}\right) \mathbf{w}^*. \qquad (5.8)
\end{aligned}$$

Now we calculate expectation of Equation (5.7) and (5.8) separately. For (5.7), by first two formulas of Lemma 5.6, we have

$$\begin{aligned}
&\left(\sum_{i=1}^k a_i^2 \mathbf{Z}_i \mathbf{Z}_i^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0\right\} + \sum_{i \neq j} a_i a_j \mathbf{Z}_i \mathbf{Z}_j^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_j^\top \mathbf{w} \geq 0\right\}\right) \mathbf{w} \\
&= \sum_{i=1}^k a_i^2 \cdot \frac{\mathbf{w}}{2} + \sum_{i \neq j} a_i a_j \frac{\mathbf{w}}{2\pi}.
\end{aligned}$$

For (5.8), we use the second and third formula in Lemma 5.6 to obtain

$$\begin{aligned}
&\left(\sum_{i=1}^{k} a_i a_i^* \mathbf{Z}_i \mathbf{Z}_i^\top \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_i^\top \mathbf{w}^* \geq 0\right\} + \sum_{i \neq j} a_i a_j^* \mathbf{Z}_i \mathbf{Z}_j^* \mathbb{I}\left\{\mathbf{Z}_i^\top \mathbf{w} \geq 0, \mathbf{Z}_j^\top \mathbf{w}^* \geq 0\right\}\right) \mathbf{w}^* \\
=&\mathbf{a}^\top \mathbf{a}^* \left(\frac{1}{\pi}\left(\pi - \phi\right) \mathbf{w}^* + \frac{1}{\pi} \sin\phi \frac{\left\|\mathbf{w}^*\right\|_2}{\left\|\mathbf{w}\right\|_2} \mathbf{w}\right) + \sum_{i \neq j} a_i a_j^* \frac{1}{2\pi} \frac{\left\|\mathbf{w}^*\right\|_2}{\left\|\mathbf{w}\right\|_2} \mathbf{w}.
\end{aligned}$$

In summary, aggregating them together we have

$$\begin{aligned}
&\mathbb{E}_{\mathbf{Z}}\left[\frac{\partial \ell\left(\mathbf{Z}, \mathbf{w}, \mathbf{a}\right)}{\partial \mathbf{w}}\right] \\
=&\frac{1}{2\pi} \mathbf{a}^\top \mathbf{a}^* \left(\pi - \phi\right) \mathbf{w}^* + \left(\frac{\left\|\mathbf{a}\right\|_2^2}{2} + \frac{\sum_{i \neq j} a_i a_j}{2\pi} + \frac{\mathbf{a}^\top \mathbf{a}^* \sin\phi}{2\pi} \frac{\left\|\mathbf{w}^*\right\|_2}{\left\|\mathbf{w}\right\|_2} + \frac{\sum_{i \neq j} a_j a_j^*}{2\pi} \frac{\left\|\mathbf{w}_*\right\|_2}{\left\|\mathbf{w}\right\|_2}\right) \mathbf{w}.
\end{aligned}$$

As a sanity check, this formula matches Equation (16) of [10] when $\mathbf{a} = \mathbf{a}^* = \mathbf{1}$.

Next, we calculate the expected gradient of $\mathbf{a}$. Recall the gradient formula of $\mathbf{a}$

$$\begin{aligned}
\frac{\partial \ell(\mathbf{Z}, \mathbf{w}, \mathbf{a})}{\mathbf{a}} =& \left(\mathbf{a}^\top \sigma\left(\mathbf{Z}\mathbf{w}\right) - \left(\mathbf{a}^*\right)^\top \sigma\left(\mathbf{Z}\mathbf{w}^*\right)\right) \sigma\left(\mathbf{Z}\mathbf{w}\right) \\
=& \sigma\left(\mathbf{Z}\mathbf{w}\right) \sigma\left(\mathbf{Z}\mathbf{w}\right)^\top \mathbf{a} - \sigma\left(\mathbf{Z}\mathbf{w}\right) \sigma\left(\mathbf{Z}\mathbf{w}^*\right)^\top \mathbf{a}^*
\end{aligned}$$

Taking expectation we have

$$\frac{\partial \ell\left(\mathbf{w}, \mathbf{a}\right)}{\partial \mathbf{a}} = \mathbf{A}\left(\mathbf{w}\right) \mathbf{a} - \mathbf{B}\left(\mathbf{w}, \mathbf{w}^*\right) \mathbf{a}^*$$

where $\mathbf{A}\left(\mathbf{w}\right)$ and $\mathbf{B}\left(\mathbf{w}, \mathbf{w}^*\right)$ are defined in Equation (5.5) and (5.6). Plugging in the formulas for $\mathbf{A}\left(\mathbf{w}\right)$ and $\mathbf{B}\left(\mathbf{w}, \mathbf{w}^*\right)$ derived in the proof of Theorem 5.1 we obtained the desired result.

□

Lemma 5.6 (Useful Identities)**.** *Given* $\mathbf{w}$*,* $\mathbf{w}^*$ *with angle* ϕ *and* $\mathbf{Z}$ *is a Gaussian random vector, then*

$$\begin{aligned}
\mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] \mathbf{w} =& \frac{1}{2} \mathbf{w} \\
\mathbb{E}\left[\mathbf{z} \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] =& \frac{1}{\sqrt{2\pi}} \frac{\mathbf{w}}{\left\|\mathbf{w}\right\|_2} \\
\mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0, \mathbf{z}^\top \mathbf{w}_* \geq 0\right\}\right] \mathbf{w}_* =& \frac{1}{2\pi}\left(\pi - \phi\right) \mathbf{w}^* + \frac{1}{2\pi} \sin\phi \frac{\left\|\mathbf{w}^*\right\|_2}{\left\|\mathbf{w}\right\|_2} \mathbf{w} \\
\mathbb{E}\left[\sigma\left(\mathbf{z}^\top \mathbf{w}\right) \sigma\left(\mathbf{z}^\top \mathbf{w}_*\right)\right] =& \frac{1}{2\pi}\left(\cos\phi\left(\pi - \phi\right) + \sin\phi\right) \left\|\mathbf{w}\right\|_2 \left\|\mathbf{w}^*\right\|_2
\end{aligned}$$

Proof. Consider an orthonormal basis of $\mathbb{R}^{d \times d}$: $\left\{\mathbf{e}_i \mathbf{e}_j^\top\right\}$ with $\mathbf{e}_1 \parallel \mathbf{w}$. Then for $i \neq j$, we know

$$\left\langle \mathbf{e}_i \mathbf{e}_j, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] \right\rangle = 0$$

by the independence properties of Gaussian random vector. For $i = j = 1$,

$$\langle \mathbf{e}_i \mathbf{e}_j^\top, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right]\rangle = \mathbb{E}\left[\left(\mathbf{z}^\top \mathbf{w}\right)^2 \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] = \frac{1}{2}$$

where the last step is by the property of half-Gaussian. For $i = j \neq j$, $\langle \mathbf{e}_i \mathbf{e}_j^\top, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right]\rangle = 1$ by standard Gaussian second moment formula. Therefore, $\mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] \mathbf{w} = \frac{1}{2}\mathbf{w}$. $\mathbb{E}\left[\mathbf{z}\mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0\right\}\right] = \frac{1}{\sqrt{2\pi}}\mathbf{w}$ can be proved by mean formula of half-normal distribution. To prove the third identity, consider an orthonormal basis of $\mathbb{R}^{d\times d}$: $\left\{\mathbf{e}_i \mathbf{e}_j^\top\right\}$ with $\mathbf{e}_1 \parallel \mathbf{w}_*$ and $\mathbf{w}$ lies in the plane spanned by $\mathbf{e}_1$ and $\mathbf{e}_2$. Using the polar representation of 2D Gaussian random variables (r is the radius and θ is the angle with $\mathrm{d}P_r = r\exp(-r^2/2)$ and $\mathrm{d}P_\theta = \frac{1}{2\pi}$):

$$\begin{aligned}
\langle \mathbf{e}_1 \mathbf{e}_1^\top, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0, \mathbf{z}^\top \mathbf{w}_* \geq 0\right\}\right]\rangle =& \frac{1}{2\pi}\int_0^\infty r^3 \exp\left(-r^2/2\right)\mathrm{d}r \cdot \int_{-\pi/2+\phi}^{\pi/2} \cos^2\theta \mathrm{d}\theta \\
=& \frac{1}{2\pi}\left(\pi - \phi + \sin\phi\cos\phi\right), \\
\langle \mathbf{e}_1 \mathbf{e}_2^\top, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0, \mathbf{z}^\top \mathbf{w}_* \geq 0\right\}\right]\rangle =& \frac{1}{2\pi}\int_0^\infty r^3 \exp\left(-r^2/2\right)\mathrm{d}r \cdot \int_{-\pi/2+\phi}^{\pi/2} \sin\theta\cos\theta \mathrm{d}\theta \\
=& \frac{1}{2\pi}\left(\sin^2\phi\right), \\
\langle \mathbf{e}_2 \mathbf{e}_2^\top, \mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0, \mathbf{z}^\top \mathbf{w}_* \geq 0\right\}\right]\rangle =& \frac{1}{2\pi}\int_0^\infty r^3 \exp\left(-r^2/2\right)\mathrm{d}r \cdot \int_{-\pi/2+\phi}^{\pi/2} \sin^2\theta \mathrm{d}\theta \\
=& \frac{1}{2\pi}\left(\pi - \phi - \sin\phi\cos\phi\right).
\end{aligned}$$

Also note that $\mathbf{e}_2 = \frac{\bar{\mathbf{w}} - \cos\phi \mathbf{e}_1}{\sin\phi}$. Therefore

$$\begin{aligned}
\mathbb{E}\left[\mathbf{z}\mathbf{z}^\top \mathbb{I}\left\{\mathbf{z}^\top \mathbf{w} \geq 0, \mathbf{z}^\top \mathbf{w}_* \geq 0\right\}\right] \mathbf{w}_* =& \frac{1}{2\pi}\left(\pi - \phi + \sin\phi\cos\phi\right)\mathbf{w}^* + \frac{1}{2\pi}\sin^2\phi \cdot \frac{\bar{\mathbf{w}} - \cos\phi\mathbf{e}_1}{\sin\phi}\left\|\mathbf{w}^*\right\|_2 \\
=& \frac{1}{2\pi}\left(\pi - \phi\right)\mathbf{w}^* + \frac{1}{2\pi}\sin\phi\frac{\left\|\mathbf{w}^*\right\|_2}{\left\|\mathbf{w}\right\|_2}\mathbf{w}.
\end{aligned}$$

For the fourth identity, focusing on the plane spanned by $\mathbf{w}$ and $\mathbf{w}_*$, using the polar decomposition, we have

$$\begin{aligned}
&\mathbb{E}\left[\sigma\left(\mathbf{z}^\top \mathbf{w}\right)\sigma\left(\mathbf{z}^\top \mathbf{w}_*\right)\right] \\
=& \frac{1}{2\pi}\int_0^\infty r^3 \exp\left(-r^2/2\right)\mathrm{d}r \cdot \int_{-\pi/2+\phi}^{\pi/2} \left(\cos\theta\cos\phi + \sin\theta\sin\phi\right)\cos\theta \mathrm{d}\theta \left\|\mathbf{w}\right\|_2 \left\|\mathbf{w}^*\right\|_2 \\
=& \frac{1}{2\pi}\left(\cos\phi\left(\pi - \phi + \sin\phi\cos\phi\right) + \sin^3\phi\right)\left\|\mathbf{w}\right\|_2 \left\|\mathbf{w}^*\right\|_2.
\end{aligned}$$

□

5.8 Proofs of Qualitative Convergence Results

Proof of Lemma 5.1. When Algorithm 1 converges, since $\mathbf{a}^\top \mathbf{a}^* \neq 0$ and $\|\mathbf{v}\|_2 < \infty$, using the gradient formula in Theorem 5.2, we know that either $\pi - \phi = 0$ or $\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{w}^* = \mathbf{0}$. For the second case, since $\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}$ is a projection matrix on the complement space of $\mathbf{v}$, $\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{w}^* = \mathbf{0}$ is equivalent to $\theta(\mathbf{v}, \mathbf{w}^*) = 0$. Once the angle between $\mathbf{v}$ and $\mathbf{w}^*$ is fixed, using the gradient formula for $\mathbf{a}$ we have the desired formulas for saddle points. □

Proof of Lemma 5.2. By the gradient formula of $\mathbf{w}$, if $\mathbf{a}^\top \mathbf{a}^* > 0$, the gradient is of the form $c\left(\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}\right)\mathbf{w}^*$ where $c > 0$. Thus because $\mathbf{I} - \frac{\mathbf{v}\mathbf{v}^\top}{\|\mathbf{v}\|_2^2}$ is the projection matrix onto the complement space of $\mathbf{v}$, the gradient update always makes the angle smaller. □

5.9 Proofs of Quantitative Convergence Results

5.9.1 Useful Technical Lemmas

We first prove the lemma about the convergence of ϕ^t.

Proof of Lemma 5.5. We consider the dynamics of $\sin^2\phi^t$.

$$
\begin{aligned}
&\sin^2\phi^{t+1} \\
=&1 - \frac{\left(\left(\mathbf{v}^{t+1}\right)^\top \mathbf{w}^*\right)^2}{\|\mathbf{v}^{t+1}\|_2^2 \|\mathbf{w}^*\|_2^2} \\
=&1 - \frac{\left(\left(\mathbf{v}^{\mathbf{t}} - \eta\frac{\partial \ell}{\partial \mathbf{v}^{\mathbf{t}}}\right)^\top \mathbf{w}^*\right)^2}{\left(\|\mathbf{v}^t\|_2^2 + \eta^2\left(\frac{\partial \ell}{\partial \mathbf{v}^{\mathbf{t}}}\right)^2\right)\|\mathbf{w}^*\|_2^2} \\
=&1 - \frac{\left(\left(\mathbf{v}^t\right)^\top \mathbf{v} + \eta\frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}^*\left(\pi - \phi^t\right)}{2\pi\|\mathbf{v}\|_2} \cdot \sin^2\phi^t \|\mathbf{w}\|_2^2\right)^2}{\|\mathbf{v}^t\|_2^2 \|\mathbf{w}^*\|_2^2 + \eta^2\left(\frac{(\mathbf{a}^t)^\top \mathbf{a}^*(\pi-\phi^t)}{2\pi}\right)^2 \frac{\sin^2\phi^t \|\mathbf{w}^*\|_2^4}{\|\mathbf{v}^t\|_2^2}} \\
\leq&1 - \frac{\|\mathbf{v}^t\|_2^2 \|\mathbf{w}^*\|_2^2 \cos^2\phi^t + 2\eta\|\mathbf{w}^*\|_2^3 \cdot \frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}(\pi-\phi)}{2\pi} \cdot \sin^2\phi^t \cos\phi^t}{\|\mathbf{v}^t\|_2^2 \|\mathbf{w}^*\|_2^2 + \eta^2\left(\frac{(\mathbf{a}^t)^\top \mathbf{a}^*(\pi-\phi^t)}{2\pi}\right)^2 \frac{\sin^2\phi^t \|\mathbf{w}^*\|_2^4}{\|\mathbf{v}^t\|_2^2}} \\
=&\frac{\sin^2\phi^t - 2\eta\frac{\|\mathbf{w}^*\|_2}{\|\mathbf{v}^t\|_2^2} \cdot \frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}(\pi-\phi)}{2\pi} \cdot \sin^2\phi^t \cos\phi^t + \eta^2\left(\frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}^*(\pi-\phi)}{2\pi}\right)^2 \sin^2\phi^t \left(\frac{\|\mathbf{w}^*\|_2}{\|\mathbf{v}^t\|_2^2}\right)^2}{1 + \eta^2\left(\frac{(\mathbf{a}^t)^\top \mathbf{a}^*(\pi-\phi)}{2\pi}\right)^2 \sin^2\phi^t \left(\frac{\|\mathbf{w}^*\|_2}{\|\mathbf{v}^t\|_2^2}\right)^2} \\
\leq&\sin^2\phi^t - 2\eta\frac{\|\mathbf{w}^*\|_2}{\|\mathbf{v}^t\|_2^2} \cdot \frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}\left(\pi - \phi\right)}{2\pi} \cdot \sin^2\phi^t \cos\phi^t + \eta^2\left(\frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}^*\left(\pi - \phi\right)}{2\pi}\right)^2 \sin^2\phi^t \left(\frac{\|\mathbf{w}^*\|_2}{\|\mathbf{v}^t\|_2^2}\right)^2
\end{aligned}
$$

where in the first inequality we dropped term proportional to $O(\eta^4)$ because it is negative, in the last equality, we divided numerator and denominator by $\|\mathbf{v}^t\|_2^2 \|\mathbf{w}^*\|_2^2$ and the last inequality we dropped the denominator because it is bigger than 1. Therefore, recall $\lambda^t = \frac{\|\mathbf{w}^*\|_2 \left(\left(\mathbf{a}^t\right)^\top \mathbf{a}^*\right)\left(\pi - \phi^t\right)}{2\pi \|\mathbf{v}^t\|_2^2}$ and we have

$$\sin^2 \phi^{t+1} \leq \left(1 - 2\eta \cos\phi^t \lambda^t + \eta^2 \left(\lambda^t\right)^2\right) \sin^2 \phi^t. \tag{5.9}$$

To this end, we need to make sure $\eta \leq \frac{\cos\phi^t}{\lambda^t}$. Note that since $\|\mathbf{v}^t\|_2^2$ is monotonically increasing, it is lower bounded by 1. Next notice $\phi^t \leq \pi/2$. Finally, from Lemma 5.8, we know $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* \leq \left(\|\mathbf{a}^*\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right) \|\mathbf{w}\|_2^2$. Combining these, we have an upper bound

$$\lambda^t \leq \frac{\left(\|\mathbf{a}^*\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right) \|\mathbf{w}^*\|_2^2}{4}.$$

Plugging this back to Equation (5.9) and use our assumption on η, we have

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

□

Lemma 5.7. $\left(\mathbf{a}^{t+1}\right)^\top \mathbf{a}^* \geq \min\left\{\left(\mathbf{a}^t\right)^\top \mathbf{a}^* + \eta\left(\frac{g(\phi^t)-1}{\pi-1}\|\mathbf{a}^*\|_2^2 - \left(\mathbf{a}^t\right)^\top \mathbf{a}^*\right), \frac{g(\phi^t)-1}{\pi-1}\|\mathbf{a}^*\|_2^2\right\}$

Proof. Recall the dynamics of $\left(\mathbf{a}^t\right)^\top \mathbf{a}^*$.

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

where the inequality is due to Lemma 5.4. If $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* \geq \frac{g(\phi^t)-1}{\pi-1}\|\mathbf{a}^*\|_2^2$,

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

If $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* \leq \frac{g(\phi^t)-1}{\pi-1}\|\mathbf{a}^*\|_2^2$, simple algebra shows $\left(\mathbf{a}^{t+1}\right)^\top \mathbf{a}^*$ increases by at least

$$\eta\left(\frac{g(\phi^t)-1}{\pi-1}\|\mathbf{a}^*\|_2^2 - \left(\mathbf{a}^t\right)^\top \mathbf{a}^*\right).$$

□

A simple corollary is $\mathbf{a}^\top \mathbf{a}^*$ is uniformly lower bounded.

Corollary 5.1. *For all* $t = 1, 2, \ldots$, $(\mathbf{a}^t)^\top \mathbf{a}^* \geq \min\left\{(\mathbf{a}^0)^\top \mathbf{a}^*, \frac{g(\phi^0)-1}{\pi-1}\|\mathbf{a}^*\|_2^2\right\}$.

This lemma also gives an upper bound of number of iterations to make $\mathbf{a}^\top\mathbf{a}^* = \Theta\left(\|\mathbf{a}^*\|_2^2\right)$.

Corollary 5.2. *If* $g(\phi) - 1 = \Omega(1)$*, then after* $\frac{1}{\eta}$ *iterations,* $\mathbf{a}^\top\mathbf{a}^* = \Theta\left(\|\mathbf{a}^*\|_2^2\right)$.

Proof. Note if $g(\phi) - 1 = \Omega(1)$ and $\mathbf{a}^\top\mathbf{a}^* \leq \frac{1}{2} \cdot \frac{g(\phi)}{\pi-1}\|\mathbf{a}^*\|_2^2$, each iteration $\mathbf{a}^\top\mathbf{a}^*$ increases by $\eta\frac{g(\phi)}{\pi-1}\|\mathbf{a}^*\|_2^2$. □

We also need an upper bound of $(\mathbf{a}^t)^\top \mathbf{a}^*$.

Lemma 5.8. *For* $t = 0, 1, \ldots$, $(\mathbf{a}^t)^\top \mathbf{a}^* \leq \left(\|\mathbf{a}^*\|_2^2 + \left(\mathbf{1}^\top\mathbf{a}^*\right)^2\right)\|\mathbf{w}^*\|_2^2$.

Proof. Without loss of generality, assume $\|\mathbf{w}^*\|_2 = 1$. Again, recall the dynamics of $(\mathbf{a}^t)^\top \mathbf{a}^*$.

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

Now we prove by induction, suppose the conclusion holds at iteration t, $(\mathbf{a}^t)^\top \mathbf{a}^* \leq \|\mathbf{a}^*\|_2^2 + \left(\mathbf{1}^\top\mathbf{a}^*\right)^2$. Plugging in we have the desired result. □

5.9.2 Convergence of Phase I

In this section we prove the convergence of Phase I.

Proof of Convergence of Phase I. Lemma 5.9 implies after $O\left(\frac{1}{\cos\phi^0\beta^0}\right)$ iterations, $\cos\phi^t = \Omega(1)$, which implies $\frac{g(\phi^t)-1}{\pi-1} = \Omega(1)$. Using Corollary 5.2, we know after $O\left(\frac{1}{\eta}\right)$ iterations we have $(\mathbf{a}^t)^\top \mathbf{a}^* \|\mathbf{w}^*\| = \Omega\left(\|\mathbf{w}^*\|_2^2\|\mathbf{a}^*\|_2^2\right)$. □

The main ingredient of the proof of phase I is the follow lemma where we use a joint induction argument to show the convergence of ϕ^t and a uniform upper bound of $\|\mathbf{v}^t\|_2$.

Lemma 5.9. *Let* $\beta^0 = \min\left\{(\mathbf{a}^0)^\top \mathbf{a}^*, (g(\phi^0)-1)\|\mathbf{a}^*\|_2^2\right\}\|\mathbf{w}^*\|_2^2$*. If the step size satisfies* $\eta \leq \min\left\{\frac{\beta^*\cos\phi^0}{8\left(\|\mathbf{a}^*\|_2^2+\left(\mathbf{1}^\top\mathbf{a}^*\right)^2\right)\|\mathbf{w}^*\|_2^2}, \frac{\cos\phi^0}{\left(\|\mathbf{a}^*\|_2^2+\left(\mathbf{1}^\top\mathbf{a}^*\right)^2\right)\|\mathbf{w}^*\|_2^2}, \frac{2\pi}{k+\pi-1}\right\}$*, we have for* $t = 0, 1, \ldots$

$$\sin^2\phi^t \leq \left(1 - \eta \cdot \frac{\cos\phi^0\beta^0}{8}\right)^t \text{ and } \left\|\mathbf{v}^t\right\|_2 \leq 2.$$

Proof. We prove by induction. The initialization ensure when $t = 0$, the conclusion is correct. Now we consider the dynamics of $\|\mathbf{v}^t\|_2^2$. Note because the gradient of $\mathbf{v}$ is orthogonal to $\mathbf{v}$ [63], we have a simple dynamic of $\|\mathbf{v}^t\|_2^2$.

$$\left\|\mathbf{v}^t\right\|_2^2 = \left\|\mathbf{v}^{t-1}\right\|_2^2 + \eta^2\left\|\frac{\partial\ell(\mathbf{v},\mathbf{a})}{\partial\mathbf{v}}\right\|_2^2$$

$$
\begin{aligned}
&= \left\|\mathbf{v}^{t-1}\right\|_2^2 + \eta^2 \left(\frac{\left(\mathbf{a}^t\right)^\top \mathbf{a}^* \left(\pi - \phi^{t-1}\right)}{2\pi}\right)^2 \frac{\sin^2\phi^t \left\|\mathbf{w}^*\right\|_2^2}{\left\|\mathbf{v}^t\right\|_2^2} \\
&\leq \left\|\mathbf{v}^{t-1}\right\|_2^2 + \eta^2 \left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right) \left\|\mathbf{w}^*\right\|_2^2 \sin^2\phi^{t-1} \\
&= 1 + \eta^2 \left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right) \left\|\mathbf{w}^*\right\|_2^2 \sum_{i=1}^{t-1} \sin^2\phi^i \\
&\leq 1 + \eta^2 \left(\left\|\mathbf{a}^*\right\|_2^2 + \left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right) \left\|\mathbf{w}^*\right\|_2^2 \frac{8}{\eta \cos\phi^0 \beta^0} \\
&\leq 2
\end{aligned}
$$

where the first inequality is by Lemma 5.8 and the second inequality we use our induction hypothesis. Recall $\lambda^t = \frac{\left\|\mathbf{w}^*\right\|_2 \left(\left(\mathbf{a}^t\right)^\top \mathbf{a}^*\right)\left(\pi - \phi^t\right)}{2\pi \left\|\mathbf{v}^t\right\|_2^2}$. The uniform upper bound of $\left\|\mathbf{v}\right\|_2$ and the fact that $\phi^t \leq \pi/2$ imply a lower bound $\lambda^t \geq \frac{\beta^0}{8}$. Plugging in Lemma 5.5, we have

$$
\sin^2\phi^{t+1} \leq \left(1 - \eta \frac{\cos\phi^0 \beta^0}{8}\right) \sin^2\phi^t \leq \left(1 - \eta \frac{\cos\phi^0 \beta^0}{8}\right)^{t+1}.
$$

We finish our joint induction proof. □

5.9.3 Analysis of Phase II

In this section we prove the convergence of phase II and necessary auxiliary lemmas.

Proof of Convergence of Phase II. At the beginning of Phase II, $\left(\mathbf{a}^{T_1}\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\| = \Omega\left(\left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2\right)$ and $g(\phi^{T_1}) - 1 = \Omega(1)$. Therefore, Lemma 5.7 implies for all $t = T_1, T_1+1, \ldots$, $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* \left\|\mathbf{w}^*\right\| = \Omega\left(\left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2\right)$. Combining with the fact that $\left\|\mathbf{v}\right\|_2 \leq 2$ (c.f. Lemma 5.9), we obtain a lower bound $\lambda_t \geq \Omega\left(\left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2\right)$ We also know that $\cos\phi^{T_1} = \Omega(1)$ and $\cos\phi^t$ is monotinically increasing (c.f. Lemma 5.2), so for all $t = T_1, T_1 + 1, \ldots$, $\cos\phi^t = \Omega(1)$. Plugging in these two lower bounds into Theorem 5.5, we have

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

for some absolute constant C. Thus, after $O\left(\frac{1}{\eta \left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations, we have $\sin^2\phi^t \leq \min\left\{\epsilon^{10}, \left(\epsilon \frac{\left\|\mathbf{a}^*\right\|_2}{\left|\mathbf{1}^\top \mathbf{a}^*\right|}\right)^{10}\right\}$, which implies $\pi - g(\phi^t) \leq \min\left\{\epsilon, \epsilon \frac{\left\|\mathbf{a}^*\right\|_2}{\left|\mathbf{1}^\top \mathbf{a}^*\right|}\right\}$.

Now using Lemma 5.10,Lemma 5.11 and Lemma 5.12, we have after $\widetilde{O}\left(\frac{1}{\eta k} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations $\ell(\mathbf{v}, \mathbf{a}) \leq C_1 \epsilon \left\|\mathbf{a}^*\right\|_2^2 \left\|\mathbf{w}^*\right\|_2^2$ for some absolute constant C_1. Rescaling ϵ properly we obtain the desired result. □

Technical Lemmas for Analyzing Phase II

In this section we provide some technical lemmas for analyzing Phase II. Because of the positive homogeneity property, without loss of generality, we assume $\left\|\mathbf{w}^*\right\|_2 = 1$.

Lemma 5.10. *If* $\pi - g(\phi^0) \leq \epsilon \frac{\|\mathbf{a}^*\|_2}{\left|\mathbf{1}^\top \mathbf{a}^*\right|}$, *after* $T = O\left(\frac{1}{\eta k} \log\left(\frac{\left|\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^0\right|}{\epsilon \|\mathbf{a}^*\|_2}\right)\right)$ *iterations,*

$$\left|\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^T\right| \leq 2\epsilon \left\|\mathbf{a}^*\right\|_2.$$

Proof. Recall the dynamics of $\mathbf{1}^\top \mathbf{a}^t$.

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

Assume $\mathbf{1}^\top \mathbf{a}^* > 0$ (the other case is similar). By Lemma 5.4 we know $\mathbf{1}^\top \mathbf{a}^t < \mathbf{1}^\top \mathbf{a}^*$ for all t. Consider

$$\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^{t+1} = \left(1 - \frac{\eta\left(k + \pi - 1\right)}{2\pi}\right)\left(\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^*\right) + \frac{\eta\left(\pi - g(\phi^t)\right)}{2\pi}\mathbf{1}^\top \mathbf{a}^*.$$

Therefore we have

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

After $T = O\left(\frac{1}{\eta k} \log\left(\frac{\left|\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^0\right|}{\epsilon \|\mathbf{a}^*\|_2}\right)\right)$ iterations, we have $\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^t - \frac{\left(\pi - g\left(\phi^t\right)\right)\mathbf{1}^\top \mathbf{a}^*}{k + \pi - 1} \leq \epsilon \left\|\mathbf{a}^*\right\|_2$, which implies$\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^t \leq 2\epsilon \left\|\mathbf{a}^*\right\|_2$. □

Lemma 5.11. *If* $\pi - g(\phi^0) \leq \epsilon \frac{\|\mathbf{a}^*\|_2}{\left|\mathbf{1}^\top \mathbf{a}^*\right|}$ *and* $\left|\mathbf{1}^\top \mathbf{a}^* - \mathbf{1}^\top \mathbf{a}^0\right| \leq \frac{\epsilon}{k} \left\|\mathbf{a}^*\right\|_2$, *then after*

$$T = O\left(\frac{1}{\eta} \log\left(\frac{\left\|\mathbf{a}^* - \mathbf{a}^0\right\|_2}{\epsilon \left\|\mathbf{a}^*\right\|_2}\right)\right)$$

iterations, $\left\|\mathbf{a}^* - \mathbf{a}^0\right\|_2 \leq C\epsilon \left\|\mathbf{a}^*\right\|_2$ *for some absolute constant* C.

Proof. We first consider the inner product

$$\begin{aligned}&\left\langle \frac{\partial \ell\left(\mathbf{v}^t, \mathbf{a}^t\right)}{\mathbf{a}^t}, \mathbf{a}^t - \mathbf{a}^* \right\rangle \\ =&\frac{\pi - 1}{2\pi}\left\|\mathbf{a}^t - \mathbf{a}^*\right\|_2^2 - \frac{g(\phi^t) - \pi}{2\pi}\left(\mathbf{a}^*\right)^\top \left(\mathbf{a}^t - \mathbf{a}^*\right) + \left(\mathbf{a}^t - \mathbf{a}^*\right)\mathbf{1}\mathbf{1}^\top \left(\mathbf{a}^\top - \mathbf{a}^*\right)\end{aligned}$$

$$\geq \frac{\pi-1}{2\pi}\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2^2-\frac{g(\phi^t)-\pi}{2\pi}\left\|\mathbf{a}^*\right\|_2\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2.$$

Next we consider the squared norm of gradient

$$\begin{aligned}\left\|\frac{\partial \ell\left(\mathbf{v},\mathbf{a}\right)}{\partial \mathbf{a}}\right\|_2^2=&\frac{1}{4\pi^2}\left\|(\pi-1)\left(\mathbf{a}^t-\mathbf{a}^*\right)+\left(\pi-g(\phi^t)\right)\mathbf{a}^*+\mathbf{1}\mathbf{1}^\top\left(\mathbf{a}^t-\mathbf{a}^*\right)\right\|_2^2\\ \leq&\frac{3}{4\pi^2}\left((\pi-1)^2\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2^2+\left(\pi-g(\phi^t)\right)^2\left\|\mathbf{a}^*\right\|_2^2+k^2\left(\mathbf{1}^\top\mathbf{a}^t-\mathbf{1}^\top\mathbf{a}^*\right)^2\right).\end{aligned}$$

Suppose $\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2\leq\epsilon\left\|\mathbf{a}^*\right\|_2$, then

$$\begin{aligned}\langle\frac{\partial \ell\left(\mathbf{v}^t,\mathbf{a}^t\right)}{\mathbf{a}^t},\mathbf{a}^t-\mathbf{a}^*\rangle\geq&\frac{\pi-1}{2\pi}\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2^2-\frac{\epsilon^2}{2\pi}\left\|\mathbf{a}^*\right\|_2^2\\ \left\|\frac{\partial \ell\left(\mathbf{v},\mathbf{a}\right)}{\partial \mathbf{a}}\right\|_2^2\leq&3\epsilon^2\left\|\mathbf{a}^*\right\|_2^2.\end{aligned}$$

Therefore we have

$$\begin{aligned}&\left\|\mathbf{a}^{t+1}-\mathbf{a}^*\right\|_2^2\leq\left(1-\frac{\eta\left(\pi-1\right)}{2\pi}\right)\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2^2+4\eta\epsilon^2\left\|\mathbf{a}\right\|^2\\ \Rightarrow&\left\|\mathbf{a}^{t+1}-\mathbf{a}^*\right\|_2^2-\frac{8\left(\pi-1\right)\epsilon^2\left\|\mathbf{a}^*\right\|_2^2}{\pi-1}\leq\left(1-\frac{\eta\left(\pi-1\right)}{2\pi}\right)\left(\left\|\mathbf{a}^t-\mathbf{a}^*\right\|_2^2-\frac{8\left(\pi-1\right)\epsilon^2\left\|\mathbf{a}^*\right\|_2^2}{\pi-1}\right).\end{aligned}$$

Thus after $O\left(\frac{1}{\eta}\left(\frac{1}{\epsilon}\right)\right)$ iterations, we must have $\left\|\mathbf{a}^{t+1}-\mathbf{a}^*\right\|_2^2\leq C\epsilon\left\|\mathbf{a}^*\right\|_2$ for some large absolute constant C. Rescaling ϵ, we obtain the desired result. □

Lemma 5.12. *If $\pi-g(\phi)\leq\epsilon$ and $\left\|\mathbf{a}-\mathbf{a}^*\left\|\mathbf{w}^*\right\|_2\right\|\leq\epsilon\left\|\mathbf{a}^*\right\|_2\left\|\mathbf{w}^*\right\|_2$, then the population loss satisfies $\ell\left(\mathbf{v},\mathbf{a}\right)\leq C\epsilon\left\|\mathbf{a}^*\right\|_2^2\left\|\mathbf{w}^*\right\|_2^2$ for some constant $C>0$.*

Proof. The result follows by plugging in the assumptions in Theorem 5.1. □

5.10 Proofs of Initialization Scheme

Proof of Theorem 5.4. The proof of the first part of Theorem 5.4 just uses the symmetry of unit sphere and ball and the second part is a direct application of Lemma 2.5 of [42]. Lastly, since $\mathbf{a}^0\sim\mathcal{B}\left(\mathbf{0}\frac{\left|\mathbf{1}^\top\mathbf{a}^*\right|}{\sqrt{k}}\right)$, we have $\mathbf{1}^\top\mathbf{a}^0\leq\left\|\mathbf{a}^0\right\|_1\leq\sqrt{k}\left\|\mathbf{a}^0\right\|_2\leq\left|\mathbf{1}^\top\mathbf{a}^*\right|\left\|\mathbf{w}^*\right\|_2$ where the second inequality is due to Hölder's inequality. □

5.11 Proofs of Converging to Spurious Local Minimum

Proof of Theorem 5.5. The main idea is similar to Theorem 5.3 but here we show $\mathbf{w}\to-\mathbf{w}^*$ (without loss of generality, we assume $\left\|\mathbf{w}^*\right\|_2=1$). Different from Theorem 5.3, here we need to

prove the invariance $\mathbf{a}^\top \mathbf{a}^* < 0$, which implies our desired result. We prove by induction, suppose $\left(\mathbf{a}^t\right)^\top \mathbf{a}^* > 0$, $\left|\mathbf{1}^\top \mathbf{a}^t\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$, $g\left(\phi^0\right) \leq \frac{-2\left(\mathbf{1}^\top \mathbf{a}\right)^2}{\|\mathbf{a}^*\|_2^2}+1$ and $\eta < \frac{k+\pi-1}{2\pi}$. Note $\left|\mathbf{1}^\top \mathbf{a}^t\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$ are satisfied by Lemma 5.4 and $g\left(\phi^0\right) \leq \frac{-2\left(\mathbf{1}^\top \mathbf{a}\right)^2}{\|\mathbf{a}^*\|_2^2} + 1$ by our initialization condition and induction hypothesis that implies ϕ^t is increasing. Recall the dynamics of $\left(\mathbf{a}^t\right)^\top \mathbf{a}^*$.

$$\begin{aligned}\left(\mathbf{a}^{t+1}\right)^\top \mathbf{a}^* =& \left(1 - \frac{\eta\left(\pi-1\right)}{2\pi}\right)\left(\mathbf{a}^t\right)^\top \mathbf{a}^* + \frac{\eta\left(g\left(\phi^t\right)-1\right)}{2\pi}\left\|\mathbf{a}^*\right\|_2^2 + \frac{\eta}{2\pi}\left(\left(\mathbf{1}^\top \mathbf{a}^*\right)^2 - \left(\mathbf{1}^\top \mathbf{a}^t\right)\left(\mathbf{1}^\top \mathbf{a}^*\right)\right) \\ \leq& \frac{\eta\left(\left(g(\phi^t)-1\right)\left\|\mathbf{a}^*\right\|_2 + 2\left(\mathbf{1}^\top \mathbf{a}^*\right)^2\right)}{2\pi} < 0\end{aligned}$$

where the first inequality we used our induction hypothesis on inner product between $\mathbf{a}^t$ and $\mathbf{a}^*$ and $\left|\mathbf{1}^\top \mathbf{a}^t\right| \leq \left|\mathbf{1}^\top \mathbf{a}^*\right|$ and the second inequality is by induction hypothesis on ϕ^t. Thus when gradient descent algorithm converges, according Lemma 5.1,

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

Plugging these into Theorem 5.1, with some routine algebra, we show $\ell\left(\mathbf{v}, \mathbf{a}\right) = \Omega\left(\left\|\mathbf{w}^*\right\|_2^2 \left\|\mathbf{a}^*\right\|_2^2\right)$. □

Chapter 6

Learning a Convolutional Filter via Gradient Descent

6.1 Introduction

The results in the previous chapter requires that the input distribution is Gaussian and there is no overlap between patches. These assumptions may not hold in many real world data. In this chapter, we consider the convolutional filter recovery problem. We show under fairly general conditions on the input distribution, we can recover a planted convolutional filter. In this chapter we also go beyond the gradient descent and study a more general setting where one only gets access to a noisy version of the gradient.

The setup is similar to the previous chapter. Formally, we consider a simple architecture: a convolution layer, followed by a ReLU activation function, and then average pooling. We 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 k: $\mathbf{Z} \in \mathbb{R}^{q \times k}$ where the i-th column is the i-th patch generated by some known function $\mathbf{Z}_i = \mathbf{Z}_i(\mathbf{x})$. For a filter with size 2 and stride 1, $\mathbf{Z}_i(\mathbf{x})$ is the i-th and $(i+1)$-th pixels. Since for convolutional filters, we only need to focus on the patches instead of the input, in the following definitions and theorems, we will refer $\mathbf{Z}$ as input and let $\mathcal{Z}$ as the distribution of $\mathbf{Z}$: ($\sigma(x) = \max(x, 0)$ is the ReLU activation function)

$$f(\mathbf{w}, \mathbf{Z}) = \frac{1}{k} \sum_{i=1}^{k} \sigma\left(\mathbf{w}^\top \mathbf{Z}_i\right). \tag{6.1}$$

See Figure 6.1 (a) for a graphical illustration. Such architectures have been used as the first layer of many works in computer vision [52, 56]. We address the realizable case, where training data are generated from (6.1) with some unknown teacher parameter $\mathbf{w}_*$ under input distribution $\mathcal{Z}$. Consider the ℓ_2 loss $\ell(\mathbf{w}, \mathbf{Z}) = \frac{1}{2}\left(f(\mathbf{w}, \mathbf{Z}) - f(\mathbf{w}_*, \mathbf{Z})\right)^2$. We learn by (stochastic) gradient descent, i.e.,

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta_t g(\mathbf{w}_t) \tag{6.2}$$

where η_t is the step size which may change over time and $g(\mathbf{w}_t)$ is a random function where its expectation equals to the population gradient $\mathbb{E}\left[g(\mathbf{w})\right] = \mathbb{E}_{\mathbf{Z} \sim \mathcal{Z}}\left[\nabla \ell(\mathbf{w}, \mathbf{Z})\right]$. We assume the

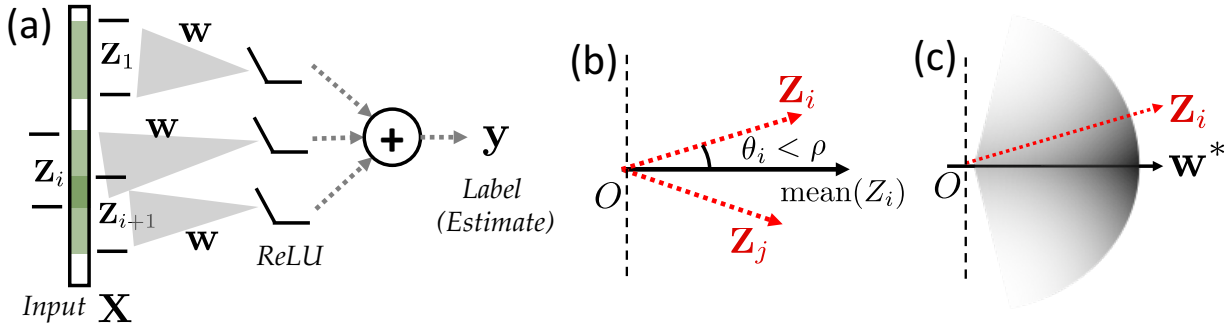


Figure 6.1: **(a)** Architecture of the network we are considering. Given input X, we extract its patches $\{\mathbf{Z}_i\}$ and send them to a shared weight vector $\mathbf{w}$. The outputs are then sent to ReLU and then summed to yield the final label (and its estimation). **(b)-(c)** Two conditions we proposed for convergence. We want the data to be (b) highly correlated and (c) concentrated more on the direction aligned with the ground truth vector $\mathbf{w}^*$.

gradient function is uniformly bounded, i.e., There exists $B > 0$ such that $\|g(\mathbf{w})\|_2 \leq B$. This condition is satisfied as long as patches, $\mathbf{w}$ and noise are all bounded. The goal of our analysis is to understand the conditions where $\mathbf{w} \to \mathbf{w}_*$, if $\mathbf{w}$ is optimized under (stochastic) gradient descent.

In this setup, our main contributions are as follows:

- **Learnability of Filters:** We show if the input patches are highly correlated (Section 6.3), i.e., $\theta(\mathbf{Z}_i, \mathbf{Z}_j) \leq \rho$ for some small $\rho > 0$, then gradient descent and stochastic gradient descent with random initialization recovers the filter in polynomial time.[1] Furthermore, strong correlations imply faster convergence.
- **Distribution-Aware Convergence Rate**. We formally establish the connection between the smoothness of the input distribution and the convergence rate for filter weights recovery where the smoothness in our chapter is defined as the ratio between the largest and the least eigenvalues of the second moment of the activation region (Section 6.2). We show that a smoother input distribution leads to faster convergence, and Gaussian distribution is a special case that leads to the tightest bound. This theoretical finding also justifies the two-stage learning rate strategy proposed by [44, 72] if the step size is allowed to change over time.

6.2 Warm Up: Analyzing One-Layer One-Neuron Model

Before diving into the convolutional filter, we first analyze the special case for $k = 1$, which is equivalent to the one-layer one-neuron architecture. The analysis in this simple case will give us insights for the fully general case. For the ease of presentation, we define following two events

[1]Note since in this chapter we focus on continuous distribution over $\mathbf{Z}$, our results do not conflict with previous negative results[9, 10] whose constructions rely on discrete distributions.

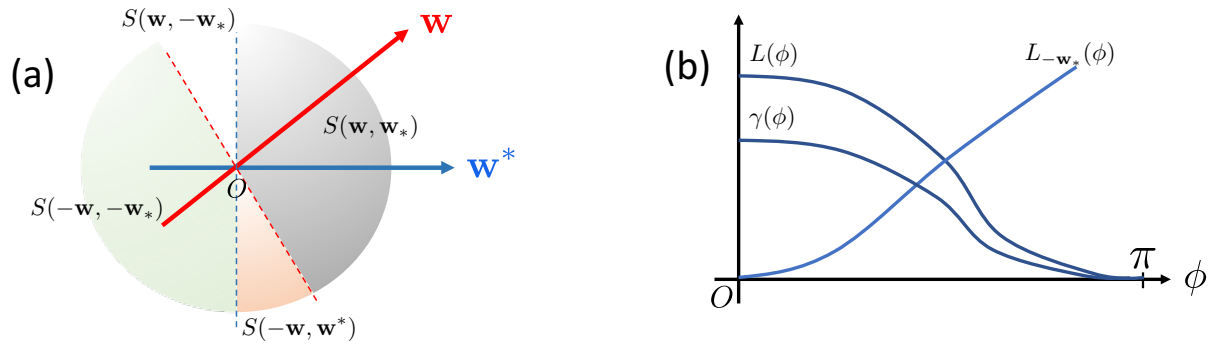


Figure 6.2: **(a)** The four regions considered in our analysis. **(b)** Illustration of $L(\phi), \gamma(\phi)$ and $L_{-\mathbf{w}_*}(\phi)$ defined in Definition 6.1 and Assumption 6.1.

and corresponding second moments

$$S(\mathbf{w}, \mathbf{w}_*) = \left\{\mathbf{Z} : \mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}, \quad S(\mathbf{w}, -\mathbf{w}_*) = \left\{\mathbf{Z} : \mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right\}, \tag{6.3}$$
$$\mathbf{A}_{\mathbf{w},\mathbf{w}_*} = \mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{S(\mathbf{w}, \mathbf{w}_*)\right\}\right], \quad \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} = \mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{S(\mathbf{w}, -\mathbf{w}_*)\right\}\right].$$

where $\mathbb{I}\{\cdot\}$ is the indicator function. Intuitively, $S(\mathbf{w}, \mathbf{w}_*)$ is the joint activation region of $\mathbf{w}$ and $\mathbf{w}_*$ and $S(\mathbf{w}, -\mathbf{w}_*)$ is the joint activation region of $\mathbf{w}$ and $-\mathbf{w}_*$. See Figure 6.2 (a) for the graphical illustration. With some simple algebra we can derive the population gradient.

$$\mathbb{E}\left[\nabla \ell(\mathbf{w}, \mathbf{Z})\right] = \mathbf{A}_{\mathbf{w},\mathbf{w}_*}(\mathbf{w} - \mathbf{w}_*) + \mathbf{A}_{\mathbf{w},-\mathbf{w}_*}\mathbf{w}.$$

One key observation is we can write the inner product $\langle \nabla_{\mathbf{w}} \ell(\mathbf{w}), \mathbf{w} - \mathbf{w}_* \rangle$ as the sum of two non-negative terms (c.f. Lemma 6.1). This observation directly leads to the following Theorem 6.1.

Theorem 6.1. *Suppose for any* $\mathbf{w}_1, \mathbf{w}_2$ *with* $\theta(\mathbf{w}_1, \mathbf{w}_2) < \pi$, $\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{S(\mathbf{w}, \mathbf{w}_*)\right\}\right] \succ 0$ *and the initialization* $\mathbf{w}_0$ *satisfies* $\ell(\mathbf{w}_0) < \ell(\mathbf{0})$ *then gradient descent algorithm recovers* $\mathbf{w}_*$.

The first assumption is about the non-degeneracy of input distribution. For $\theta(\mathbf{w}_1, \mathbf{w}_2) < \pi$, one case that the assumption fails is that the input distribution is supported on a low-dimensional space, or degenerated. The second assumption on the initialization is to ensure that gradient descent does not converge to $\mathbf{w} = \mathbf{0}$, at which the gradient is undefined. This is a general convergence theorem that holds for a wide class of input distribution and initialization points. In particular, it includes Theorem 6 of [73] as a special case. If the input distribution is degenerate, i.e., there are holes in the input space, the gradient descent may stuck around saddle points and we believe more data are needed to facilitate the optimization procedure This is also consistent with empirical evidence in which more data are helpful for optimization.

6.2.1 Convergence Rate of One-Layer One-Neuron Model

In the previous section we showed if the distribution is regular and the weights are initialized appropriately, gradient descent recovers the true weights when it converges. In practice we also

want to know how many iterations are needed. To characterize the convergence rate, we need some quantitative assumptions. We note that different set of assumptions will lead to a different rate and ours is only one possible choice. In this chapter, we use the following quantities.

Definition 6.1 (The Largest/Smallest eigenvalue Values of the Second Moment on Intersection of two Half Spaces)**.** *For $\phi \in [0, \pi]$, define*

$$\gamma(\phi) = \min_{\mathbf{w}:\angle \mathbf{w},\mathbf{w}_* = \phi} \lambda_{\min}\left(\mathbf{A}_{\mathbf{w},\mathbf{w}_*}\right), \quad L(\phi) = \max_{\mathbf{w}:\angle \mathbf{w},\mathbf{w}_* = \phi} \lambda_{\max}\left(\mathbf{A}_{\mathbf{w},\mathbf{w}_*}\right),$$

These two conditions quantitatively characterize the angular smoothness of the input distribution. For a given angle ϕ, if the difference between $\gamma(\phi)$ and $L(\phi)$ is large then there is one direction has large probability mass and one direction has small probability mass, meaning the input distribution is not smooth. On the other hand, if $\gamma(\phi)$ and $L(\phi)$ are close, then all directions have similar probability mass, which means the input distribution is smooth. The smoothest input distributions are rotationally invariant distributions (e.g. standard Gaussian) which have $\gamma(\phi) = L(\phi)$. For analogy, we can think of $L(\phi)$ as Lipschitz constant of the gradient and $\gamma(\phi)$ as the strong convexity parameter in the optimization literature but here we also allow they change with the angle. Also observe that when $\phi = \pi$, $\gamma(\phi) = L(\phi) = 0$ because the intersection has measure 0 and both $\gamma(\phi)$ and $L(\phi)$ are monotonically decreasing.

Our next assumption is on the growth of $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}$. Note that when $\theta(\mathbf{w}, \mathbf{w}_*) = 0$, then $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*} = \mathbf{0}$ because the intersection between $\mathbf{w}$ and $-\mathbf{w}_*$ has 0 measure. Also, $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}$ grows as the angle between $\mathbf{w}$ and $\mathbf{w}_*$ becomes larger.

In the following, we assume the operator norm of $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}$ increases smoothly with respect to the angle. The intuition is that as long as input distribution bounded probability density with respect to the angle, the operator norm of $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}$ is bounded. We show in Theorem 6.7 that $\beta = 1$ for rotational invariant distribution and in Theorem 6.8 that $\beta = p$ for standard Gaussian distribution.

Assumption 6.1. *We assume there exists $\beta > 0$ that for $0 \le \phi \le \pi/2$,*

$$L_{-w_*}(\phi) \triangleq \max_{\mathbf{w},\theta(\mathbf{w},\mathbf{w}_*)\le\phi} \lambda_{\max}\left(\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}\right) \le \beta\phi.$$

Now we are ready to state the convergence rate.

Theorem 6.2. *Suppose the initialization $\mathbf{w}_0$ satisfies $\|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2$. Denote $\phi_t = \arcsin\left(\frac{\|\mathbf{w_t} - \mathbf{w}_*\|_2}{\|\mathbf{w}_*\|_2}\right)$ then if step size is set as $0 \le \eta_t \le \min_{0\le\phi\le\phi_t} \frac{\gamma(\phi)}{2(L(\phi)+4\beta)^2}$, we have for $t = 1, 2, \ldots$*

$$\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 \le \left(1 - \frac{\eta_t \gamma\left(\phi_t\right)}{2}\right) \|\mathbf{w}_t - \mathbf{w}_*\|_2^2.$$

Note both $\gamma(\phi)$ and $L(\phi)$ increases as ϕ decreases so we can choose a constant step size $\eta_t = \Theta\left(\frac{\gamma(\phi_0)}{(L(0)+\beta)^2}\right)$. This theorem implies that we can find the ϵ-close solution of $\mathbf{w}_*$ in $O\left(\frac{(L(0)+\beta)^2}{\gamma^2(\phi_0)} \log\left(\frac{1}{\epsilon}\right)\right)$ iterations. It also suggests a direct relation between the smoothness of the distribution and the convergence rate. For smooth distribution where $\gamma(\phi)$ and $L(\phi)$ are close and β is small then $\frac{(L(0)+\beta)^2}{\gamma^2(\phi_0)}$ is relatively small and we need fewer iterations. On the other hand,

if $L(\phi)$ or β is much larger than $\gamma(\phi)$, we will need more iterations. We verify this intuition in Section 6.4.

If we are able to choose the step sizes adaptively $\eta_t = \Theta\left(\frac{\gamma(\phi_t)}{(L(\phi_t)+\beta)^2}\right)$, like using methods proposed by Lin and Xiao [53], we may improve the computational complexity to

$$O\left(\max_{\phi\leq\phi_0}\frac{\left(L(\phi)+\beta\right)^2}{\gamma^2\left(\phi\right)}\log\left(\frac{1}{\epsilon}\right)\right).$$

This justifies the use of two-stage learning rate strategy proposed by He et al. [44], Szegedy et al. [72] where at the beginning we need to choose learning to be small because $\frac{\gamma(\phi_0)}{2(L(\phi_0)+2\beta)^2}$ is small and later we can choose a large learning rate because as the angle between $\mathbf{w}_t$ and $\mathbf{w}_*$ becomes smaller, $\frac{\gamma(\phi_t)}{2(L(\phi_t)+2\beta)^2}$ becomes bigger.

The theorem requires the initialization satisfying $\|\mathbf{w}_0-\mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2$, which can be achieved by random initialization with constant success probability. See Section 6.3.2 for a detailed discussion.

6.3 Main Results for Learning a Convolutional Filter

In this section we generalize ideas from the previous section to analyze the convolutional filter. First, for given $\mathbf{w}$ and $\mathbf{w}_*$ we define four events that divide the input space of each patch $\mathbf{Z}_i$. Each event corresponds to a different activation region induced by $\mathbf{w}$ and $\mathbf{w}_*$, similar to (6.3).

$$S(\mathbf{w},\mathbf{w}_*)_i=\left\{\mathbf{Z}_i:\mathbf{w}^\top\mathbf{Z}_i\geq 0,\mathbf{w}_*^\top\mathbf{Z}_i\geq 0\right\},\quad S(\mathbf{w},-\mathbf{w}_*)_i=\left\{\mathbf{Z}_i:\mathbf{w}^\top\mathbf{Z}_i\geq 0,\mathbf{w}_*^\top\mathbf{Z}_i\leq 0\right\},$$
$$S(-\mathbf{w},-\mathbf{w}_*)_i=\left\{\mathbf{Z}_i:\mathbf{w}^\top\mathbf{Z}_i\leq 0,\mathbf{w}_*^\top\mathbf{Z}_i\leq 0\right\},\quad S(-\mathbf{w},\mathbf{w}_*)_i=\left\{\mathbf{Z}_i:\mathbf{w}^\top\mathbf{Z}_i\leq 0,\mathbf{w}_*^\top\mathbf{Z}_i\geq 0\right\}.$$

Please check Figure 6.2 (a) again for illustration. For the ease of presentation we also define the average over all patches in each region

$$\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}=\frac{1}{k}\sum_{i=1}^k\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\},\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}=\frac{1}{k}\sum_{i=1}^k\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i\right\},$$
$$\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}=\frac{1}{k}\sum_{i=1}^k\mathbf{Z}_i\mathbb{I}\left\{S(-\mathbf{w},\mathbf{w}_*)_i\right\}.$$

Next, we generalize the smoothness conditions analogue to Definition 6.1 and Assumption 6.1. Here the smoothness is defined over the average of patches.

Assumption 6.2. *For $\phi\in[0,\pi]$, define*

$$\gamma(\phi)=\min_{\mathbf{w}:\theta(\mathbf{w},\mathbf{w}_*)=\phi}\lambda_{\min}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}^\top\right]\right),$$
$$L(\phi)=\max_{\mathbf{w}:\theta(\mathbf{w},\mathbf{w}_*)=\phi}\lambda_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}^\top\right]\right). \tag{6.4}$$

We assume for all $0\leq\phi\leq\pi/2$, $\max_{\mathbf{w}:\theta(\mathbf{w},\mathbf{w}_*)=\phi}\lambda_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}^\top\right]\right)\leq\beta\phi$ for some $\beta>0$.

The main difference between the simple one-layer one-neuron network and the convolution filter is two patches may appear in different regions. For a given sample, there may exists patch $\mathbf{Z}_i$ and $\mathbf{Z}_j$ such that $\mathbf{Z}_i \in S(\mathbf{w}, \mathbf{w}_*)_i$ and $\mathbf{Z}_j \in S(\mathbf{w}, -\mathbf{w}_*)_j$ and their interaction plays an important role in the convergence of (stochastic) gradient descent. Here we assume the second moment of this interaction, i.e., cross-covariance, also grows smoothly with respect to the angle.

Assumption 6.3. *We assume there exists $L_{cross} > 0$ such that*

$$\max_{\mathbf{w}:\theta(\mathbf{w},\mathbf{w}_*)\leq\phi} \lambda_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}^\top\right]\right) + \lambda_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}^\top\right]\right) + \lambda_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}^\top\right]\right) \leq L_{cross}\phi.$$

First note if $\phi = 0$, then $\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}$ and $\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}$ has measure 0 and this assumption models the growth of cross-covariance. Next note this L_{cross} represents the closeness of patches. If $\mathbf{Z}_i$ and $\mathbf{Z}_j$ are very similar, then the joint probability density of $\mathbf{Z}_i \in S(\mathbf{w}, \mathbf{w}_*)_i$ and $\mathbf{Z}_j \in S(\mathbf{w}, -\mathbf{w}_*)_j$ is small which implies L_{cross} is small. In the extreme setting, $\mathbf{Z}_1 = \ldots = \mathbf{Z}_k$, we have $L_{\text{cross}} = 0$ because in this case the events $\{\mathbf{Z}_i \in S(\mathbf{w}, \mathbf{w}_*)_i\} \cap \{\mathbf{Z}_j \in S(\mathbf{w}, -\mathbf{w}_*)_j\}$, $\{\mathbf{Z}_i \in S(\mathbf{w}, \mathbf{w}_*)_i\} \cap \{\mathbf{Z}_j \in S(-\mathbf{w}, \mathbf{w}_*)_j\}$ and $\{\mathbf{Z}_i \in S(\mathbf{w}, -\mathbf{w}_*)_i\} \cap \{\mathbf{Z}_j \in S(-\mathbf{w}, \mathbf{w}_*)_j\}$ all have measure 0.

Now we are ready to present our result on learning a convolutional filter by gradient descent.

Theorem 6.3. *If the initialization satisfies $\|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2$ and denote $\phi_t = \arcsin\left(\frac{\|\mathbf{w}_t - \mathbf{w}_*\|_2}{\|\mathbf{w}_*\|_2}\right)$ which satisfies $\gamma(\phi_0) > 6L_{\text{cross}}$. Then if we choose $\eta_t \leq \min_{0\leq\phi\leq\phi_t} \frac{\gamma(\phi)-6L_{\text{cross}}}{2(L(\phi)+10L_{\text{cross}}+4\beta)^2}$, we have for $t = 1, 2, \ldots$ and $\phi_t \triangleq \arcsin\left(\frac{\|\mathbf{w}_t - \mathbf{w}_*\|_2}{\|\mathbf{w}_*\|_2}\right)$*

$$\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 \leq \left(1 - \frac{\eta(\gamma(\phi_t) - 6L_{\text{cross}})}{2}\right)\|\mathbf{w}_t - \mathbf{w}_*\|_2^2$$

Our theorem suggests if the initialization satisfies $\gamma(\phi_0) > 6L_{\text{cross}}$, we obtain linear convergence rate. In Section 6.3.1, we give a concrete example showing closeness of patches implies large $\gamma(\phi)$ and small L_{cross}. Similar to Theorem 6.2, if the step size is chosen so that $\eta_t = \Theta\left(\frac{\gamma(\phi_0)-6L_{\text{cross}}}{\left(L_{S(\mathbf{w},\mathbf{w}_*)}(0)+10L_{\text{cross}}+4\beta\right)^2}\right)$, in $O\left(\left(\frac{\gamma(\phi_0)-6L_{\text{cross}}}{L_{S(\mathbf{w},\mathbf{w}_*)}(0)+10L_{\text{cross}}+4\beta}\right)^2 \log\left(\frac{1}{\epsilon}\right)\right)$ iterations, we can find the ϵ-close solution of $\mathbf{w}_*$ and the proof is also similar to that of Theorem 6.3.

In practice,we never get a true population gradient but only stochastic gradient $g(\mathbf{w})$ (c.f. Equation 6.2). The following theorem shows SGD also recovers the underlying filter.

Theorem 6.4. *Let $\phi_* = \text{argmax}_\phi \gamma(\phi) \geq 6L_{\text{cross}}$. Denote $r_0 = \|\mathbf{w}_0 - \mathbf{w}_*\|_2$, $\phi_0 = \arcsin\left(\frac{r_0}{\|\mathbf{w}_*\|_2}\right)$ and $\phi_1 = \frac{\phi_*+\phi_0}{2}$. For ϵ sufficiently small, if $\eta_t = \Theta\left(\frac{\epsilon^2(\gamma(\phi_1)-6L_{\text{cross}})^2\|\mathbf{w}_*\|_2^2}{B^2}\right)$, then we have in $T = O\left(\frac{B^2}{\epsilon^2(\gamma(\phi_1)-6L_{\text{cross}})^2\|\mathbf{w}_*\|_2^2}\log\left(\frac{\|\mathbf{w}_0-\mathbf{w}_*\|}{\epsilon\delta\|\mathbf{w}_*\|_2}\right)\right)$ iterations, with probability at least $1-\delta$ we have $\|\mathbf{w}_T - \mathbf{w}_*\| \leq \epsilon\|\mathbf{w}_*\|_2$.*

Unlike the vanilla gradient descent case, here the convergence rate depends on ϕ_1 instead of ϕ_0. This is because of the randomness in SGD and we need a more robust initialization. We choose ϕ_1 to be the average of ϕ_0 and ϕ_* for the ease of presentation. As will be apparent in the proof we only require ϕ_0 not very close to ϕ_*. The proof relies on constructing a martingale and use Azuma-Hoeffding inequality and this idea has been previously used by Ge et al. [36].

6.3.1 What distribution is easy for SGD to learn a convolutional filter?

Different from One-Layer One-Neuron model, here we also requires the Lipschitz constant for closeness L_{cross} to be relatively small and $\gamma(\phi_0)$ to be relatively large. A natural question is: What input distributions satisfy this condition?

Here we give an example. We show if (1) patches are close to each other (2) the input distribution has small probability mass around the decision boundary then the assumption in Theorem 6.3 is satisfied. See Figure 6.1 (b)-(c) for the graphical illustrations.

Theorem 6.5. *Denote $\mathbf{Z}_{avg} = \frac{1}{k}\sum_{i=1}^{k}\mathbf{Z}_i$. Suppose all patches have unit norm [2] and for all for all i, $\theta(\mathbf{Z}_i, \mathbf{Z}_{avg}) \leq \rho$. Further assume there exists $L \geq 0$ such that for any $\phi \leq \rho$ and for all $\mathbf{Z}_i$*

$$\mathbf{P}\left[\theta(\mathbf{Z}_i, \mathbf{w}_*) \in \left[\frac{\pi}{2} - \phi, \frac{\pi}{2} + \phi\right]\right] \leq \mu\phi, \quad \mathbf{P}\left[\theta(\mathbf{Z}_i, \mathbf{w}_*) \in -\left[\frac{\pi}{2} - \phi, -\frac{\pi}{2} + \phi\right]\right] \leq \mu\phi,$$

then we have

$$\gamma(\phi_0) \geq \gamma_{avg}(\phi_0) - 4(1 - \cos\rho) \text{ and } L_{\text{cross}} \leq 3\mu.$$

where $\gamma_{avg}(\phi_0) = \sigma_{\min}\left(\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_0^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\right)$, analogue to Definition 6.1.

Several comments are in sequel. We view ρ as a quantitative measure of the closeness between different patches, i.e., ρ small means they are similar.

This lower bound is monotonically decreasing as a function of ρ and note when $\rho = 0$, $\sigma_{\min}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}^\top\right]\right) = \gamma_{avg}(\phi_0)$ which recovers Definition 6.1.

For the upper bond on L_{cross}, μ represents the upper bound of the probability density around the decision boundary. For example if $\mathbf{P}\left[\theta(\mathbf{Z}_i, \mathbf{w}_*) \in \left[\frac{\pi}{2} - \phi, \frac{\pi}{2} + \phi\right]\right] \propto \phi^2$, then for ϕ in a small neighborhood around $\pi/2$, say radius ϵ, we have $\mathbf{P}\left[\theta(\mathbf{Z}_i, \mathbf{w}_*) \in \left[\frac{\pi}{2} - \phi, \frac{\pi}{2} + \phi\right]\right] \lesssim \epsilon\phi$. This assumption is usually satisfied in real world examples like images because the image patches are not usually close to the decision boundary. For example, in computer vision, the local image patches often form clusters and is not evenly distributed over the appearance space. Therefore, if we use linear classifier to separate their cluster centers from the rest of the clusters, near the decision boundary the probability mass should be very low.

6.3.2 The Power of Random Initialization

For one-layer one-neuron model, we need initialization $\|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2$ and for the convolution filter, we need a stronger initialization $\|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2 \cos(\phi_*)$. The following theorem shows with uniformly random initialization we have constant probability to obtain a good initialization. Note with this theorem at hand, we can boost the success probability to arbitrary close to 1 by random restarts. The proof is similar to [73].

Theorem 6.6. *If we uniformly sample $\mathbf{w}_0$ from a p-dimensional ball with radius $\alpha\|\mathbf{w}_*\|$ so that $\alpha \leq \sqrt{\frac{1}{2\pi p}}$, then with probability at least $\frac{1}{2} - \sqrt{\frac{\pi p}{2}}\alpha$, we have $\|\mathbf{w}_0 - \mathbf{w}_*\|_2 \leq \sqrt{1 - \alpha^2}\|\mathbf{w}_*\|$.*

To apply this general initialization theorem to our convolution filter case, we can choose $\alpha = \cos\phi_*$. Therefore, with some simple algebra we have the following corollary.

[2] This is condition can be relaxed to the norm and the angle of each patch are independent and the norm of each pair is independent of others.

Corollary 6.1. *Suppose* $\cos(\phi_*) < \frac{1}{\sqrt{8\pi p}}$*, then if* $\mathbf{w}_0$ *is uniformly sampled from a ball with center* $\mathbf{0}$ *and radius* $\|\mathbf{w}_*\|\cos(\phi_*)$*, we have with probability at least* $\frac{1}{2} - \cos(\phi_*)\sqrt{\frac{\pi p}{2}} > \frac{1}{4}$.

The assumption of this corollary is satisfied if the patches are close to each other as discussed in the previous section.

6.4 Experiments

In this section we use simulations to verify our theoretical findings. We first test how the smoothness affect the convergence rate in one-layer one-neuron model described in Section 6.2 To construct input distribution with different $L(\phi)$, $\gamma(\phi)$ and β (c.f. Definition 6.1 and Assumption 6.1), we fix the patch to have unit norm and use a mixture of truncated Gaussian distribution to model on the angle around $\mathbf{w}_*$ and around the $-\mathbf{w}_*$ Specifically, the probability density of $\angle\mathbf{Z}, \mathbf{w}_*$ is sampled from $\frac{1}{2}N(0,\sigma)\mathbb{I}_{[-\pi/2,\pi/2]} + \frac{1}{2}N(-\pi,\sigma)\mathbb{I}_{[-\pi/2,\pi/2]}$. Note by definitions of $L(\phi)$ and $\gamma(\phi)$ if $\sigma \to 0$ the probability mass is centered around $\mathbf{w}_*$, so the distribution is very spiky and $L(\phi)/\gamma(\phi)$ and β will be large. On the other hand, if $\sigma \to \infty$, then input distribution is close to the rotation invariant distribution and $L(\phi)/\gamma(\phi)$ and β will be small. Figure 6.3a verifies our prediction where we fix the initialization and step size.

Next we test how the closeness of patches affect the convergence rate in the convolution setting. We first generate a single patch $\widetilde{\mathbf{Z}}$ using the above model with $\sigma = 1$, then generate each unit norm $\mathbf{Z}_i$ whose angle with $\widetilde{\mathbf{Z}}$, $\angle\mathbf{Z}_i, \widetilde{\mathbf{Z}}$ is sampled from $\angle\mathbf{Z}_i, \widetilde{\mathbf{Z}} \sim N(0,\sigma_2)\mathbb{I}_{[-\pi,\pi)}$. Figure 6.3b shows as variance between patches becomes smaller, we obtain faster convergence rate, which coincides with Theorem 6.3.

We also test whether SGD can learn a filter on real world data. Here we choose MNIST data and generate labels using two filters. One is random filter where each entry is sampled from a standard Gaussian distribution (Figure 6.4a) and the other is a Gabor filter (Figure 6.4b). Figure 6.3a and Figure 6.3c show convergence rates of SGD with different initializations. Here, better initializations give faster rates, which coincides our theory. Note that here we report the relative loss, logarithm of squared error divided by the square of mean of data points instead of the difference between learned filter and true filter because we found SGD often cannot converge to the exact filter but rather a filter with near zero loss. We believe this is because the data are approximately lying in a low dimensional manifold in which the learned filter and the true filter are equivalent. Lastly, we visualize the true filters and the learned filters in Figure 6.4 and we can see that the they have similar patterns.

6.5 Conclusions and Future Work

In this chapter we provide the recovery guarantee of (stochastic) gradient descent algorithm with random initialization for learning a convolution filter when the input distribution is not Gaussian. Our analyses only used the definition of ReLU and some mild structural assumptions on the input distribution. Here we list some future directions.

A possible direction is to consider the agnostic setting, where the label is not equal to the output of a neural network. This will lead to different dynamics of (stochastic) gradient descent

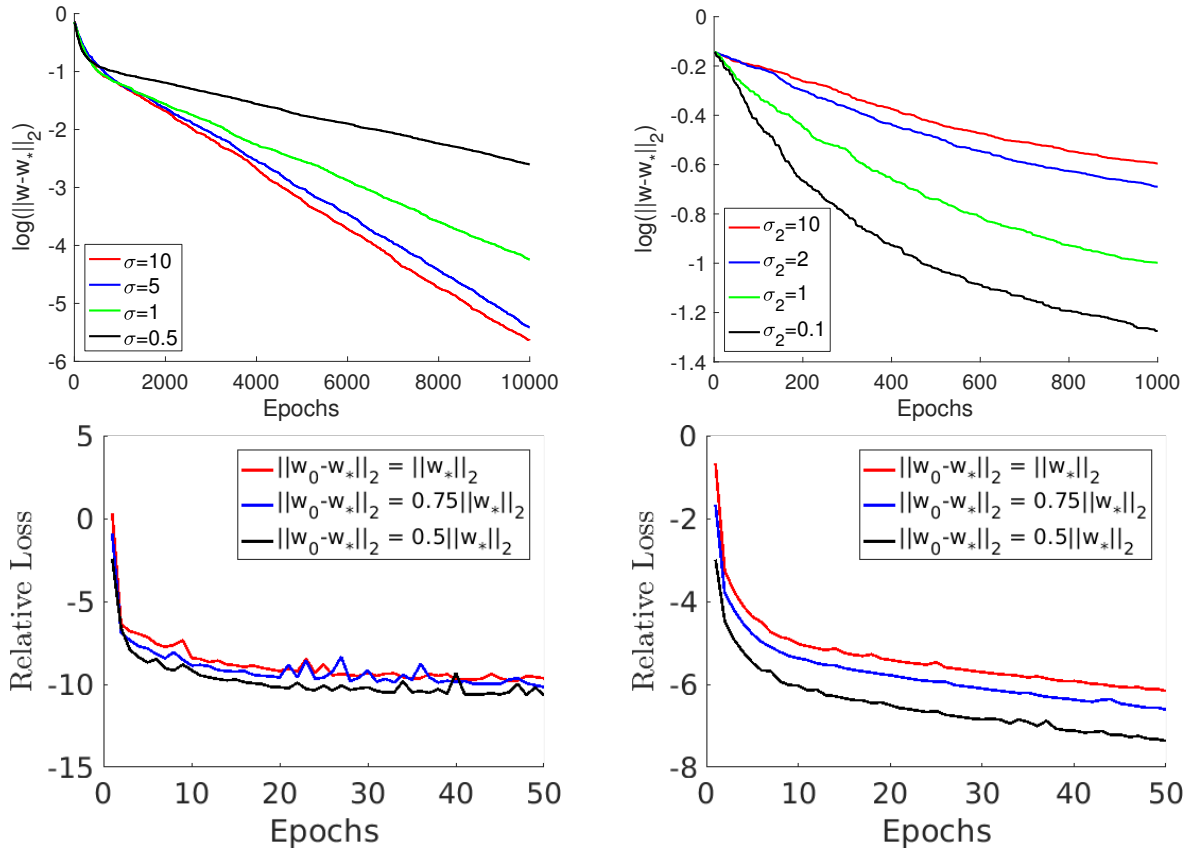
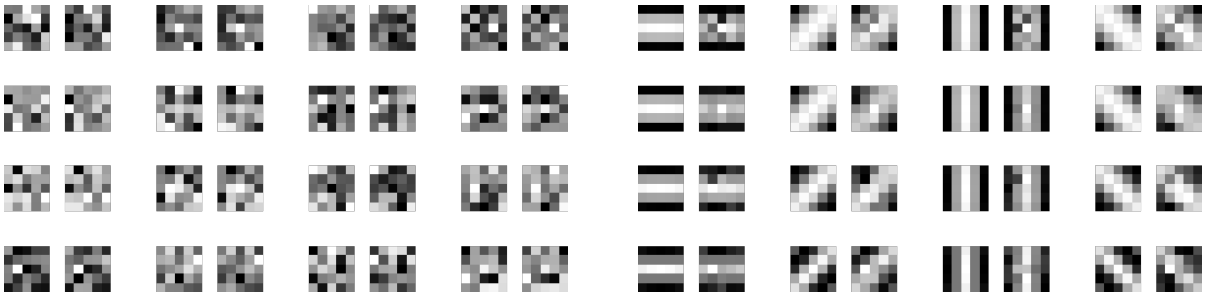


Figure 6.3: Convergence rates of SGD **(a)** with different smoothness where larger σ is smoother; **(b)** with different closeness of patches where smaller σ_2 is closer; (c) for a learning a random filter with different initialization on MNIST data; **(d)** for a learning a Gabor filter with different initialization on MNIST data.



(a) Random generated target filters.

(b) Gabor filters.

Figure 6.4: Visualization of true and learned filters. For each pair, the left one is the underlying truth and the right is the filter learned by SGD.

and we may need to analyze the robustness of the optimization procedures. This problem is also related to the expressiveness of the neural network [62] where if the underlying function is not equal bot is close to a neural network. We believe our analysis can be extend to this setting.

Appendix: Omitted Proofs

6.6 Proofs and Additional Theorems

6.6.1 Proofs of the Theorem in Section 6.2

Lemma 6.1.

$$\langle \nabla_{\mathbf{w}} \ell(\mathbf{w}), \mathbf{w} - \mathbf{w}_* \rangle = (\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},\mathbf{w}_*} (\mathbf{w} - \mathbf{w}_*) + (\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w}. \tag{6.5}$$

and both terms are non-negative.

Proof. Since $\mathbf{A}_{\mathbf{w},\mathbf{w}_*} \succeq 0$ and $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \succeq 0$ (positive-semidefinite), both the first term and one part of the second term $\mathbf{w}^\top \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w}$ are non-negative. The other part of the second term is

$$-\mathbf{w}_*^\top \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w} = -\mathbb{E}\left[\left(\mathbf{w}_*^\top \mathbf{Z}\right) \left(\mathbf{w}^\top \mathbf{Z}\right) \mathbb{I}\left\{ \mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0 \right\} \right] \geq 0.$$

□

Proof of Theorem 6.1. The assumption on the input distribution ensures when $\theta(\mathbf{w}, \mathbf{w}_*) \neq \pi$, $\mathbf{A}_{\mathbf{w},\mathbf{w}_*} \succ \mathbf{0}$ and when $\theta(\mathbf{w}, \mathbf{w}_*) \neq 0$, $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \succ \mathbf{0}$. Now when gradient descent converges we have $\nabla_{\mathbf{w}} \ell(\mathbf{w}) = \mathbf{0}$. We have the following theorem. By assumption, since $\ell(\mathbf{w}) < \ell(\mathbf{0})$ and gradient descent only decreases function value, we will not converge to $\mathbf{w} = \mathbf{0}$. Note that at any critical points, $\langle \nabla_{\mathbf{w}} \ell(\mathbf{w}), \mathbf{w} - \mathbf{w}_* \rangle = 0$, from Lemma 6.1, we have:

$$(\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},\mathbf{w}_*} (\mathbf{w} - \mathbf{w}_*) = 0 \tag{6.6}$$

$$(\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w} = 0. \tag{6.7}$$

Suppose we are converging to a critical point $\mathbf{w} \neq \mathbf{w}_*$. There are two cases:

- If $\theta(\mathbf{w}, \mathbf{w}_*) \neq \pi$, then we have$(\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},\mathbf{w}_*} (\mathbf{w} - \mathbf{w}_*) > 0$, which contradicts with Eqn. 6.6.
- If $\theta(\mathbf{w}, \mathbf{w}_*) = \pi$, without loss of generality, let $\mathbf{w} = -\alpha \mathbf{w}_*$ for some $\alpha > 0$. By the assumption we know $\mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \succ \mathbf{0}$. Now the second equation becomes $(\mathbf{w} - \mathbf{w}_*)^\top \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w} = (1+\gamma) \mathbf{w}_* \mathbf{A}_{\mathbf{w},-\mathbf{w}_*} \mathbf{w}_* > 0$, which contradicts with Eqn. 6.7.

Therefore we have $\mathbf{w} = \mathbf{w}_*$. □

Proof of Theorem 6.2. Our proof relies on the following simple but crucial observation: if $\|\mathbf{w} - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2$, then

$$\theta(\mathbf{w}, \mathbf{w}_*) \leq \arcsin\left(\frac{\|\mathbf{w} - \mathbf{w}_*\|_2}{\|\mathbf{w}_*\|_2} \right).$$

We denote $\theta\left(\mathbf{w}_t, \mathbf{w}_*\right) = \theta_t$ and by the observation we have $\theta_t \leq \phi_t$. Recall the gradient descent dynamics,

$$\begin{aligned}\mathbf{w}_{t+1} &= \mathbf{w}_t - \eta \nabla_{\mathbf{w}_t} \ell(\mathbf{w}_t) \\ &= \mathbf{w}_t - \eta \left(\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right) - \mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t\right).\end{aligned}$$

Consider the squared distance to the optimal weight

$$\begin{aligned}&\left\|\mathbf{w}_{t+1} - \mathbf{w}_*\right\|_2^2 \\ =& \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2 \\ &- \eta \left(\mathbf{w_t} - \mathbf{w}_*\right)^\top \left(\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right) - \mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t\right) \\ &+ \eta^2 \left\|\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right) - \mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t\right\|_2^2.\end{aligned}$$

By our analysis in the previous section, the second term is smaller than

$$-\eta \left(\mathbf{w_t} - \mathbf{w}_*\right)^\top \mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right) \leq -\eta\gamma(\theta_t) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2$$

where we have used our assumption on the angle. For the third term, we expand it as

$$\begin{aligned}&\left\|\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right) - \mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t\right\|_2^2 \\ =& \left\|\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right)\right\|_2^2 \\ &- 2\left(\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{\mathbf{w}_t^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \geq 0\right\}\right]\left(\mathbf{w}_t - \mathbf{w}_*\right)\right)^\top \mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t \\ &+ \left\|\mathbb{E}\left[\mathbf{w}^\top \mathbf{Z} \geq 0, \mathbf{w}_*^\top \mathbf{Z} \leq 0\right] \mathbf{w}_t\right\|_2^2 \\ \leq& L^2(\theta_t) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2 + 2L(\theta_t) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2 \cdot 2\beta \frac{\left\|\mathbf{w}_t - \mathbf{w}_*\right\|}{\left\|\mathbf{w}_*\right\|_2} + \left(2\beta \frac{\left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2}{\left\|\mathbf{w}_*\right\|_2}\right)^2 \left\|\mathbf{w}_t\right\|_2^2 \\ \leq& L^2(\theta_t) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2 + 2L(\theta_t) \left\|\mathbf{w}_t - \mathbf{w}_*\right\| \cdot 2\beta \frac{\left\|\mathbf{w}_t - \mathbf{w}_*\right\|}{\left\|\mathbf{w}_*\right\|_2} \cdot 2\left\|\mathbf{w}_*\right\|_2 + \left(2\beta \frac{\left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2}{\left\|\mathbf{w}_*\right\|_2}\right)^2 \cdot 4\left\|\mathbf{w}_*\right\|_2^2 \\ \leq& \left(L^2(\theta_t) + 8L(\theta_t)\beta + 16\beta^2\right) \left\|\mathbf{w} - \mathbf{w}_*\right\|_2^2.\end{aligned}$$

Therefore, in summary,

$$\begin{aligned}\left\|\mathbf{w}_{t+1} - \mathbf{w}_*\right\|_2^2 &\leq \left(1 - \eta\gamma(\theta_t) + \eta^2 \left(L(\theta_t) + 4\beta\right)^2\right) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2 \\ &\leq \left(1 - \frac{\eta\gamma(\theta_t)}{2}\right) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2 \\ &\leq \left(1 - \frac{\eta\gamma(\phi_t)}{2}\right) \left\|\mathbf{w}_t - \mathbf{w}_*\right\|_2^2\end{aligned}$$

where the first inequality is by our assumption of the step size and second is because $\theta_t \leq \phi_t$ and $\gamma(\cdot)$ is monotonically decreasing. □

Theorem 6.7 (Rotational Invariant Distribution)**.** *For any unit norm rotational invariant input distribution, we have* $\beta = 1$.

Proof of Theorem 6.7. Without loss of generality, we only need to focus on the plane spanned by $\mathbf{w}$ and $\mathbf{w}_*$ and suppose $\mathbf{w}_* = (1,0)^\top$. Then

$$\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top \mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)\right\}\right] = \int_{-\pi/2}^{-\pi/2+\phi} \begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix}(\cos\theta,\sin\theta)\mathrm{d}\theta = \frac{1}{2}\begin{pmatrix}\phi-\sin\phi\cos\phi & -\sin^2\phi\\ -\sin^2\phi & \phi+\sin\phi\cos\phi\end{pmatrix}.$$

It has two eigenvalues

$$\lambda_1(\phi) = \frac{\phi+\sin\phi}{2} \text{ and } \lambda_2(\phi) = \frac{\phi-\sin\phi}{2}.$$

Therefore, $\max_{\mathbf{w},\theta(\mathbf{w},\mathbf{w}_*)\le\phi}\lambda_{\max}\left(\mathbf{A}_{\mathbf{w},-\mathbf{w}_*}\right) = \frac{\phi+\sin\phi}{2} \le \phi$ for $0\le\phi\le\pi$. □

Theorem 6.8. *If* $\mathbf{Z}\sim N(0,\mathbf{I})$*, then* $\beta\le p$

Proof. Note in previous theorem we can integrate angle and radius separately then multiply them together. For Gaussian distribution, we have $\mathbb{E}\left[\|\mathbf{Z}\|_2^2\right]\le p$. The result follows. □

6.6.2 Proofs of Theorems in Section 6.3

Proof of Theorem 6.3. The proof is very similar to Theorem 6.2. Notation-wise, for two events S_1, S_2 we use S_1S_2 as a shorthand for $S_1\cap S_2$ and S_1+S_2 as a shorthand for $S_1\cup S_2$. Denote $\theta_t = \theta\left(\mathbf{w}_t,\mathbf{w}_*\right)$. First note with some routine algebra, we can write the gradient as

$$\begin{aligned}
&\nabla_{\mathbf{w}_t}\ell\left(\mathbf{w}_t\right)\\
=&\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j^\top\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right](\mathbf{w}-\mathbf{w}_*)\\
&+\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j^\top\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right]\mathbf{w}\\
&+\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j^\top\mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\mathbf{w}\\
&-\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j^\top\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\mathbf{w}_*
\end{aligned}$$

We first examine the inner product between the gradient and $\mathbf{w}-\mathbf{w}_*$.

$$\begin{aligned}
&\langle\nabla_{\mathbf{w}_t}\ell(\mathbf{w}),\mathbf{w}-\mathbf{w}_*\rangle\\
&=(\mathbf{w}-\mathbf{w}_*)^\top\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right](\mathbf{w}-\mathbf{w}_*)\\
&+(\mathbf{w}-\mathbf{w}_*)^\top\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\mathbf{w}\\
&-(\mathbf{w}-\mathbf{w}_*)^\top\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)}\mathbf{Z}_i\mathbf{Z}_j\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\mathbf{w}_*
\end{aligned}$$

$$
\begin{aligned}
&\geq (\mathbf{w}-\mathbf{w}_*)^\top \mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right](\mathbf{w}-\mathbf{w}_*)\\
&+ (\mathbf{w}-\mathbf{w}_*)^\top \mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\mathbf{w}\\
&- (\mathbf{w}-\mathbf{w}_*)^\top \mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j + S(\mathbf{w},-\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\mathbf{w}_*\\
&\geq \gamma(\theta_t)\left\|\mathbf{w}-\mathbf{w}_*\right\|_2^2\\
&- \left\|\mathbf{w}-\mathbf{w}_*\right\|_2\left\|\mathbf{w}\right\|_2\left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\right\|_{op}\\
&- \left\|\mathbf{w}-\mathbf{w}_*\right\|_2\left\|\mathbf{w}_*\right\|_2\left(\left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op}\right.\\
&\left.+ \left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op} + \left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op}\right)\\
&\geq \gamma(\theta_t)\left\|\mathbf{w}-\mathbf{w}_*\right\|_2^2\\
&- 2\left\|\mathbf{w}-\mathbf{w}_*\right\|_2\left\|\mathbf{w}_*\right\|_2\left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\right\|_{op}\\
&- \left\|\mathbf{w}-\mathbf{w}_*\right\|_2\left\|\mathbf{w}_*\right\|_2\left(\left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op}\right.\\
&\left.+ \left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i S(\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op} + \left\|\mathbb{E}\left[\sum_{(i,j)=(1,1)}^{(d,d)} \mathbf{Z}_i\mathbf{Z}_j \mathbb{I}\left\{S(\mathbf{w},-\mathbf{w}_*)_i S(-\mathbf{w},\mathbf{w}_*)_j\right\}\right]\right\|_{op}\right)\\
&\geq \gamma\left(\theta_t\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2 - 3L_{\text{cross}}\phi_t\left\|\mathbf{w}_*\right\|_2\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2\\
&\geq \gamma\left(\theta_t\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2 - 6L_{\text{cross}}\frac{\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2}{\left\|\mathbf{w}_*\right\|_2}\cdot\left\|\mathbf{w}_*\right\|_2\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2\\
&\geq \left(\gamma\left(\theta_t\right) - 6L_{\text{cross}}\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2
\end{aligned}
$$

where the first inequality we used the definitions of the regions; the second inequality we used the definition of operator norm; the third inequality we used the fact $\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2 \leq \left\|\mathbf{w}_*\right\|_2$; the fourth inequality we used the definition of L_{cross} and the fifth inequality we used $\phi \leq 2\sin\phi$ for any $0 \leq \phi \leq \pi/2$. Next we can upper bound the norm of the gradient using similar argument

$$
\begin{aligned}
\left\|\nabla_{\mathbf{w}_t}\ell(\mathbf{w}_t)\right\|_2 \leq& L\left(\theta_t\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2 + 10L_{\text{cross}}\left\|\mathbf{w}_t-\mathbf{w}_*\right\| + 2\beta\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2\\
=& \left(L(\theta_t) + 10L_{\text{cross}} + 4\beta\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2.
\end{aligned}
$$

Therefore, using the dynamics of gradient descent, putting the above two bounds together, we have

$$
\left\|\mathbf{w}_{t+1}-\mathbf{w}_*\right\|_2^2 \leq \left(1-\eta\left(\gamma(\theta_t) - 6L_{\text{cross}}\right) + \eta^2(L(\theta_t) + 10L_{\text{cross}} + 4\beta)^2\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2
$$

$$\leq \left(1 - \frac{\eta(\gamma(\theta_t) - 6L_{\text{cross}})}{2}\right) \|\mathbf{w}_t - \mathbf{w}_*\|_2^2$$
$$\leq \left(1 - \frac{\eta(\gamma(\phi_t) - 6L_{\text{cross}})}{2}\right) \|\mathbf{w}_t - \mathbf{w}_*\|_2^2$$

where the last step we have used our choice of η_t and $\theta_t \leq \phi_t$. □

The proof of Theorem 6.4 consists of two parts. First we show if η is chosen properly and T is not to big, then for all $1 \leq t \leq T$, with high probability the iterates stat in a neighborhood of $\mathbf{w}_*$. Next, conditioning on this, we derive the rate.

Lemma 6.2. *Denote $r_0 = \|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2 \sin\phi_*$. Given $0 < r_1 < \|\mathbf{w}_*\|_2 \sin\phi_*$, number of iterations $T \in \mathbb{Z}_{++}$ and failure probability δ, denote $\phi_1 = \arcsin\left(\frac{r_1}{\|\mathbf{w}_*\|_2}\right)$ then if the step size satisfies*

$$0 < 1 - \eta\gamma(\phi_1) + \eta^2(L(0) + 10L_{\text{cross}} + 4\beta)^2 < 1$$
$$\frac{\left(r_1^2 - r_0^2\right)^2}{T\left(1 + 2\eta\alpha T\right)\left(2\eta B\left(L(0) + 10L_{\text{cross}} + 4\beta\right) r_1 + \eta^2 B^2\right)^2} \geq \log\left(\frac{T}{\delta}\right)$$

with $\alpha = \gamma\left(\phi_1\right) - \eta\left(L(0) + 10L_{\text{cross}} + 4\beta\right)$. Then with probability at least $1 - \delta$, for all $t = 1, \ldots, T$, we have

$$\|\mathbf{w}_t - \mathbf{w}_*\| \leq r_1.$$

Proof of Lemma 6.2. Let $g(\mathbf{w}_t) = \mathbb{E}\left[\nabla_{\mathbf{w}_t}\ell\left(\mathbf{w}_t\right)\right] + \xi_t$. We denote $\mathcal{F}_t = \sigma\left\{\xi_1, \ldots, \xi_t\right\}$, the sigma-algebra generated by $\xi_1, \ldots, \xi_t$ and define the event

$$\mathcal{C}_t = \left\{\forall\tau \leq t, \|\mathbf{w}_\tau - \mathbf{w}_*\| \leq r_1\right\}.$$

Consider

$$\begin{aligned}
&\mathbb{E}\left[\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 \mathbb{I}_{\mathcal{C}_t} | \mathcal{F}_t\right] \\
=&\mathbb{E}\left[\|\mathbf{w}_t - \eta\nabla_{\mathbf{w}_t}\ell(\mathbf{w}_t) - \mathbf{w}_* - \eta\xi_t\|_2^2 \mathbb{I}_{\mathcal{C}_t} | \mathcal{F}_t\right] \\
\leq& \left(\left(1 - \eta\gamma(\phi_1) + \eta^2\left(L(0) + 10L_{\text{cross}} + 4\beta\right)^2\right)\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 + \eta^2 B^2\right)\mathbb{I}_{\mathcal{C}_t}
\end{aligned}$$

where the inequality follows by our analysis of gradient descent together with definition of $\mathcal{C}_t$ and $\mathbb{E}\left[\xi_t | \mathcal{F}_t\right] = 0$. Define

$$G_t = (1 - \eta\alpha)^{-t}\left(\|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \frac{\eta B^2}{\alpha}\right).$$

By our analysis above, we have

$$\mathbb{E}\left[G_{t+1}\mathbb{I}_{\mathcal{C}_t} | \mathcal{F}_t\right] \leq G_t\mathbb{I}_{\mathcal{C}_t} \leq G_t\mathbb{I}_{\mathcal{C}_{t-1}}$$

where the last inequality is because $\mathcal{C}_t$ is a subset of $\mathcal{C}_{t-1}$. Therefore, $G_t \mathbb{I}_{\mathcal{C}_{t-1}}$ is a super-martingale and we may apply Azuma-Hoeffding inequality. Before that, we need to bound the difference between $G_t \mathbb{I}_{\mathcal{C}_t}$ and its expectation. Note

$$\begin{aligned}
\left|G_t \mathbb{I}_{\mathcal{C}_{t-1}} - \mathbb{E}\left[G_t \mathbb{I}_{\mathcal{C}_{t-1}}\right] |\mathcal{F}_{t-1}\right| &= (1-\eta\alpha)^{-t} \left| \|\mathbf{w}_t - \mathbf{w}_*\|_2^2 - \mathbb{E}\left[\|\mathbf{w}_t - \mathbf{w}_*\|_2^2\right] |\mathcal{F}_{t-1}\right| \mathbb{I}_{\mathcal{C}_{t-1}} \\
&= (1-\eta\alpha)^{-t} \left| 2\eta \langle \xi_t, \mathbf{w}_t - \eta \nabla_{\mathbf{w}_t} \ell(\mathbf{w}_t) - \mathbf{w}_* \rangle - \eta^2 \mathbb{E}\left[\|\xi_t\|_2^2 |\mathcal{F}_{t-1}\right] \right| \mathbb{I}_{\mathcal{C}_{t-1}} \\
&\le (1-\eta\alpha)^{-t} \left(2\eta B \left(L(0) + 10 L_{\text{cross}} + 4\beta\right) \|\mathbf{w}_t - \mathbf{w}_*\|_2 + \eta^2 B^2\right) \mathbb{I}_{\mathcal{C}_{t-1}} \\
&\le (1-\eta\alpha)^{-t} \left(2\eta B \left(L(0) + 10 L_{\text{cross}} + 4\beta\right) r_1 + \eta^2 B^2\right) \\
&\triangleq d_t.
\end{aligned}$$

Therefore for all $t \le T$

$$\begin{aligned}
c_t^2 &\triangleq \sum_{\tau=1}^{t} d_\tau^2 \\
&= \sum_{\tau=1}^{t} (1-\eta\alpha)^{-2t} \left(2\eta B \left(L(0) + 10 L_{\text{cross}} + 4\beta\right) r_1 + \eta^2 B^2\right)^2 \\
&\le t (1-\eta\alpha)^{-2t} \left(2\eta B \left(L(0) + 10 L_{\text{cross}} + 4\beta\right) r_1 + \eta^2 B^2\right)^2 \\
&\le T (1 + 2\eta\alpha T) \left(2\eta B \left(L(0) + 10 L_{\text{cross}} + 4\beta\right) r_1 + \eta^2 B^2\right)^2
\end{aligned}$$

where the first inequality we used $1 - \eta\alpha < 1$, the second we used $t \le T$ and the third we used our assumption on η. Let us bound at $(t+1)$-th step, the iterate goes out of the region,

$$\begin{aligned}
\mathbf{P}\left[\mathcal{C}_t \cap \left\{\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2 > r_1\right\}\right] &= \mathbf{P}\left[\mathcal{C}_t \cap \left\{\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 > r_1^2\right\}\right] \\
&= \mathbf{P}\left[\mathcal{C}_t \cap \left\{\|\mathbf{w}_{t+1} - \mathbf{w}_*\|_2^2 > r_0^2 + (r_1^2 - r_0^2)\right\}\right] \\
&= \mathbf{P}\left[\mathcal{C}_t \cap \left\{G_{t+1} (1-\eta\alpha)^t + \frac{\eta B^2}{\alpha} \ge G_0 + \frac{\eta B^2}{\alpha} + r_1^2 - r_0^2\right\}\right] \\
&\le \mathbf{P}\left[\mathcal{C}_t \cap \left\{G_{t+1} - G_0 \ge r_1^2 - r_0^2\right\}\right] \\
&\le \exp\left\{-\frac{\left(r_1^2 - r_0^2\right)^2}{2c_t^2}\right\} \\
&\le \frac{\delta}{T}
\end{aligned}$$

where the second inequality we used Azuma-Hoeffding inequality, the last one we used our assumption of η. Therefore for all $0 \le t \le T$, we have with probability at least $1 - \delta$, $\mathcal{C}_t$ happens. □

Now we can derive the rate.

Lemma 6.3. *Denote $r_0 = \|\mathbf{w}_0 - \mathbf{w}_*\|_2 < \|\mathbf{w}_*\|_2 \sin\phi_*$. Given $0 < r_1 < \|\mathbf{w}_*\|_2 \sin\phi_*$, number of iterations $T \in \mathbb{Z}_{++}$ and failure probability δ, denote $\phi_1 = \arcsin\left(\frac{r_1}{\|\mathbf{w}_*\|_2}\right)$ then if the step size satisfies*

$$0 < 1 - \eta\gamma(\phi_1) + \eta^2 (L(0) + 10 L_{\text{cross}} + 4\beta)^2 < 1$$

$$\frac{\left(r_1^2 - r_0^2\right)^2}{T\left(1+2\eta\alpha T\right)\left(2\eta B\left(L(0)+10L_{\text{cross}}+4\beta\right)r_1+\eta^2B^2\right)^2} \geq \log\left(\frac{T}{\delta}\right)$$

$$\eta T\left(\gamma(\phi_1)-\eta\left(L(0)+10L_{\text{cross}}+4\beta\right)^2\right) \geq \log\left(\frac{r_0^2}{\epsilon^2\left\|\mathbf{w}_*\right\|_2^2\delta}\right)$$

$$\epsilon^2\left(\gamma(\phi_1)-\eta\left(L(0)+10L_{\text{cross}}+4\beta\right)^2\right)\left\|\mathbf{w}_*\right\|_2^2 \geq \eta B^2$$

with $\alpha = \gamma\left(\phi_1\right)-\eta\left(L(0)+10L_{\text{cross}}+4\beta\right)$*, then we have with probability* $1-2\delta$*,*

$$\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2 \leq 2\epsilon\left\|\mathbf{w}_*\right\|_2.$$

Proof of Lemma 6.3. We use the same notations in the proof of Lemma 6.2. By the analysis of Lemma 6.2, we know

$$\mathbb{E}\left[\left\|\mathbf{w}_{t+1}-\mathbf{w}_*\right\|_2^2\mathbb{I}_{\mathcal{C}_t}|\mathcal{F}_t\right] \leq \left(\left(1-\eta\alpha\right)\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2+\eta^2B^2\right)\mathbb{I}_{\mathcal{C}_t}.$$

Therefore we have

$$\mathbb{E}\left[\left\|\mathbf{w}_t-\mathbf{w}_*\right\|_2^2\mathbb{I}_{\mathcal{C}_t}-\frac{\eta B^2}{\alpha}\right] \leq \left(1-\eta\alpha\right)^t\left(\left\|\mathbf{w}_0-\mathbf{w}_*\right\|_2^2-\frac{\eta B}{\alpha}\right).$$

Now we can bound the failure probability

$$\begin{aligned}
\mathbf{P}\left[\left\|\mathbf{w}_T-\mathbf{w}_*\right\|_2 \geq 2\epsilon\left\|\mathbf{w}_*\right\|_2\right] \leq & \mathbf{P}\left[\left\|\mathbf{w}_T-\mathbf{w}_*\right\|_2^2-\frac{\eta B^2}{\alpha} \geq \epsilon^2\left\|\mathbf{w}_*\right\|_2^2\right] \\
\leq & \mathbf{P}\left[\left\{\left\|\mathbf{w}_T-\mathbf{w}_*\right\|_2^2\mathbb{I}_{\mathcal{C}_t}-\frac{\eta B^2}{\alpha} \geq \epsilon^2\left\|\mathbf{w}\right\|_2^2\right\}\cup\mathcal{C}_t^c\right] \\
\leq & \mathbf{P}\left[\left\{\left\|\mathbf{w}_T-\mathbf{w}_*\right\|_2^2\mathbb{I}_{\mathcal{C}_t}-\frac{\eta B^2}{\alpha} \geq \epsilon^2\left\|\mathbf{w}\right\|_2^2\right\}\right]+\delta \\
\leq & \frac{\mathbb{E}\left[\left\|\mathbf{w}_T-\mathbf{w}_*\right\|_2^2\mathbb{I}_{\mathcal{C}_t}-\frac{\eta B^2}{\alpha}\right]}{\epsilon^2\left\|\mathbf{w}_*\right\|_2^2}+\delta \\
\leq & \frac{\left(1-\eta\alpha\right)^t\left(\left\|\mathbf{w}_0-\mathbf{w}_*\right\|_2^2-\frac{\eta B}{\alpha}\right)}{\epsilon^2\left\|\mathbf{w}_*\right\|_2^2}+\delta \\
\leq & 2\delta.
\end{aligned}$$

The first inequality we used the last assumption. The second inequality we used the probability of an event is upper bound by any superset of this event. The third one we used Lemma 6.2 and the union bound. The fourth one we used Markov's inequality. □

Now we can specify the T and η and derive the convergence rate of SGD for learning a convolution filter.

Proof of Theorem 6.4. With the choice of η and T, it is straightforward to check they satisfies conditions in Lemma 6.3. □

Proof of Theorem 6.5. We first prove the lower bound of $\gamma(\phi_0)$.

$$
\begin{aligned}
&\mathbb{E}\left[\left(\sum_{i=1}^{k}\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}\right)\left(\sum_{i=1}^{k}\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}\right)^\top\right]\\
=&\mathbb{E}\left[k\mathbf{Z}+\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\left(k\mathbf{Z}+\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right)^\top\right]\\
=&k^2\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top\right]+k\mathbb{E}\left[\mathbf{Z}\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right)^\top\right]\\
&+k\mathbb{E}\left[\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right)\mathbf{Z}^\top\right]\\
&+\mathbb{E}\left[\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}_1\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_1\right\}\right)\right)\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}_1\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_1\right\}\right)\right)^\top\right]\\
\succcurlyeq&k^2\mathbb{E}\left[\mathbf{Z}\mathbf{Z}^\top\right]+k\mathbb{E}\left[\mathbf{Z}\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right)^\top\right]\\
&+k\mathbb{E}\left[\left(\sum_{i=1}^{k}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right)\mathbf{Z}^\top\right]
\end{aligned}
$$

Note because $\mathbf{Z}_i$s have unit norm and by law of cosines $\left\|\mathbf{Z}\left(\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}-\mathbf{Z}\right)\right\|_{op}\leq 2(1-\cos\rho)$. Therefore,

$$
\sigma_{\min}\left(\mathbb{E}\left[\left(\sum_{i=1}^{d}\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}\right)\left(\sum_{i=1}^{d}\mathbf{Z}_i\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_i\right\}\right)^\top\right]\right)\geq k^2(\gamma_1(\phi_0)-4(1-\cos\rho)).
$$

Now we prove the upper bound of L_{cross}. Notice that

$$
\begin{aligned}
\left\|\mathbb{E}\left[\mathbf{Z}_i\mathbf{Z}_j^\top\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_iS(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\right\|_2\leq&\mathbb{E}\left[\left\|\mathbf{Z}_i\right\|_2\left\|\mathbf{Z}_j\right\|_2\mathbb{I}\left\{S(\mathbf{w},\mathbf{w}_*)_iS(\mathbf{w},-\mathbf{w}_*)_j\right\}\right]\\
=&\int_{S(\mathbf{w},-\mathbf{w}_*)_j}\left(\int_{S(\mathbf{w},\mathbf{w}_*)_i}\mathrm{d}\mathbf{P}\left(\mathbf{Z}_i|\mathbf{Z}_j\right)\right)\mathrm{d}\mathbf{P}\left(\theta_j\right).
\end{aligned}
$$

If $\phi\leq\psi$, then by our assumption, we have

$$
\int_{S(\mathbf{w},-\mathbf{w}_*)_j}\left(\int_{S(\mathbf{w},\mathbf{w}_*)_i}\mathrm{d}\mathbf{P}\left(\mathbf{Z}_i|\mathbf{Z}_j\right)\right)\mathrm{d}\mathbf{P}\left(\theta_j\right)\leq\int_{S(\mathbf{w},-\mathbf{w}_*)_j}\mathrm{d}\mathbf{P}\left(\mathbf{Z}_j\right)\leq L\phi.
$$

On the other hand, if $\phi\geq\gamma$, let θ_j be the angle between $\mathbf{w}_*$ and $\mathbf{Z}_j$, we have

$$
\int_{S(\mathbf{w},-\mathbf{w}_*)_j}\left(\int_{S(\mathbf{w},\mathbf{w}_*)_i}\mathrm{d}\mathbf{P}\left(\mathbf{Z}_i|\mathbf{Z}_j\right)\right)\mathrm{d}\mathbf{P}\left(\theta_j\right)\leq\int_{\frac{\pi}{2}}^{\frac{\pi}{2}+\gamma}\left(\int_{S(\mathbf{w},\mathbf{w}_*)_i}\mathrm{d}\mathbf{P}\left(\mathbf{Z}_i|\mathbf{Z}_j\right)\right)\mathrm{d}\mathbf{P}\left(\theta_j\right)
$$

$$\leq L\gamma$$
$$\leq L\phi.$$

Therefore, $\sigma_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}^{\top}\right]\right) \leq L\phi$. Using similar arguments we can show

$$\sigma_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},\mathbf{w}_*)}\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}\right]\right) \leq L\phi \text{ and } \sigma_{\max}\left(\mathbb{E}\left[\mathbf{Z}_{S(\mathbf{w},-\mathbf{w}_*)}\mathbf{Z}_{S(-\mathbf{w},\mathbf{w}_*)}\right]\right) \leq L\phi.$$

□

Proof of Theorem 6.6. We use the same argument by Tian [73]. Let r_{init} be the initialization radius. The failure probability is lower bounded

$$\frac{1}{2}\left(r_{init}\right) - \frac{\left(\frac{r_{init}^2}{2\|\mathbf{w}_*\|_2} + \frac{\|\mathbf{w}_*\|_2\cos(\phi_*)}{2}\right)\delta V_{k-1}\left(r_{init}\right)}{V_k\left(r_{init}\right)}.$$

Therefore, $r_{init} = \cos\left(\phi_*\right)\|\mathbf{w}_*\|_2$ maximizes this lower bound. Plugging this optimizer in and using formula for the volume of the Euclidean ball, the failure probability is lower bounded by

$$\frac{1}{2} - \cos\left(\phi_*\right)\frac{\pi\Gamma\left(p/2+1\right)}{\Gamma\left(p/2+1/2\right)} \geq \frac{1}{2} - \cos\left(\phi_*\right)\sqrt{\frac{\pi p}{2}}$$

where we used Gautschi's inequality for the last step. □

Part III

When Does Gradient Descent Fail?

Chapter 7

Gradient Descent Can Take Exponential Time to Escape Saddle Points

7.1 Introduction

In the previous chapters we showed gradient descent is a power optimization technique for machine learning problems. However it is also important to understand its limitations and modify gradient descent to make more generally applicable. Recall, for general smooth non-convex problems, gradient descent is only known to find a stationary point (i.e., a point where the gradient equals zero) in polynomial time [57].

A stationary point can be a local minimizer, saddle point, or local maximizer. In recent years, there has been an increasing focus on conditions under which it is possible to escape saddle points (more specifically, *strict* saddle points as in Definition 7.4) and converge to a local minimizer. Moreover, stronger statements can be made when the following two key properties hold: 1) all local minima are global minima, and 2) all saddle points are strict. These properties hold for a variety of machine learning problems, including tensor decomposition [36], dictionary learning [71], phase retrieval [70], matrix sensing [8, 61], matrix completion [37, 38], and matrix factorization [49]. For these problems, any algorithm that is capable of escaping strict saddle points will converge to a global minimizer from an arbitrary initialization point.

Recent work has analyzed variations of GD that include stochastic perturbations. It has been shown that when perturbations are incorporated into GD at each step the resulting algorithm can escape strict saddle points in polynomial time [36]. It has also been shown that episodic perturbations suffice; in particular, Jin et al. [45] analyzed an algorithm that occasionally adds a perturbation to GD (see Algorithm 2), and proved that not only does the algorithm escape saddle points in polynomial time, but additionally the number of iterations to escape saddle points is nearly dimension-independent[1]. These papers in essence provide sufficient conditions under which a variant of GD has favorable convergence properties for non-convex functions. This leaves open the question as to whether such perturbations are in fact necessary. If not, we might prefer to avoid the perturbations if possible, as they involve additional hyper-parameters. The current understanding of gradient descent is silent on this issue. The major existing result

[1] Assuming that the smoothness parameters (see Definition 7.1- 7.3) are all independent of dimension.

is provided by Lee et al. [48], who show that gradient descent, with any reasonable random initialization, will always escape strict saddle points *eventually*—but without any guarantee on the number of steps required. This motivates the following question:

Does randomly initialized gradient descent escape saddle points in polynomial time?

In this chapter, perhaps surprisingly, we give a strong *negative* answer to this question. We show that even under a fairly natural initialization scheme (e.g., uniform initialization over a unit cube, or Gaussian initialization) and for non-pathological functions satisfying smoothness properties considered in previous work, GD can take exponentially long time to escape saddle points and reach local minima, while perturbed GD (Algorithm 2) only needs polynomial time. This result shows that GD is fundamentally slower in escaping saddle points than its perturbed variant, and justifies the necessity of adding perturbations for efficient non-convex optimization.

The counter-example that supports this conclusion is a smooth function defined on $\mathbb{R}^d$, where GD with random initialization will visit the vicinity of d saddle points before reaching a local minimum. While perturbed GD takes a constant amount of time to escape each saddle point, GD will get closer and closer to the saddle points it encounters later, and thus take an increasing amount of time to escape. Eventually, GD requires time that is exponential in the number of saddle points it needs to escape, thus $e^{\Omega(d)}$ steps.

7.2 Preliminaries

Let $\mathbb{B}_x(r)$ denote the d-dimensional ℓ_2 ball centered at x with radius r, $[-1,1]^d$ denote the d-dimensional cube centered at 0 with side-length 2, and $B_\infty(x,R) = x + [-R,R]^d$ denote the d-dimensional cube centered at x with side-length $2R$.

Throughout the chapter we consider functions that satisfy the following smoothness assumptions.

Definition 7.1. *A function $f(\cdot)$ is B-bounded if for any $\mathbf{x} \in \mathbb{R}^d$:*

$$|f(\mathbf{x})| \le B.$$

Definition 7.2. *A differentiable function $f(\cdot)$ is ℓ-gradient Lipschitz if for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \le \ell \|\mathbf{x} - \mathbf{y}\|_2.$$

Definition 7.3. *A twice-differentiable function $f(\cdot)$ is ρ-Hessian Lipschitz if for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$:*

$$\left\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\right\|_{\mathrm{op}} \le \rho \|\mathbf{x} - \mathbf{y}\|_2.$$

Intuitively, definition 7.1 says function value is both upper and lower bounded; definition 7.2 and 7.3 state the gradients and Hessians of function can not change dramatically if two points are close by. Definition 7.2 is a standard asssumption in the optimization literature, and definition 7.3 is also commonly assumed when studying saddle points and local minima.

Our goal is to escape saddle points. The saddle points discussed in this chapter are assumed to be "strict" [36]:

Definition 7.4. *A saddle point $\mathbf{x}^*$ is called an α-strict saddle point if there exists some $\alpha > 0$ such that $\|\nabla f(\mathbf{x}^*)\|_2 = 0$ and $\lambda_{min}(\nabla^2 f(\mathbf{x}^*)) \le -\alpha$.*

Algorithm 2 Perturbed Gradient Descent [45]

Input: $\mathbf{x}^{(0)}$, step size η, perturbation radius r, time interval t_{thres}, gradient threshold g_{thres}.
$t_{\text{noise}} \leftarrow -t_{\text{thres}} - 1$.
for $t = 1, 2, \cdots$ **do**
  if $\|\nabla f(x^t)\|_2 \leq g_{\text{thres}}$ and $t - t_{\text{noise}} > t_{\text{thres}}$ **then**
    $\mathbf{x}^{(t)} \leftarrow \mathbf{x}^{(t)} + \xi^t,\ \xi^t \sim \text{unif}(\mathbb{B}_0(r)),\ t_{\text{noise}} \leftarrow t,$
  end if
  $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} - \eta \nabla f\left(\mathbf{x}^{(t)}\right).$
end for

That is, a strict saddle point must have an escaping direction so that the eigenvalue of the Hessian along that direction is strictly negative. It turns out that for many non-convex problems studied in machine learning, all saddle points are strict (see Section 7.1 for more details).

To escape strict saddle points and converge to local minima, we can equivalently study the approximation of second-order stationary points. For ρ-Hessian Lipschitz functions, such points are defined as follows by [58]:

Definition 7.5. *A point $\mathbf{x}$ is a called a second-order stationary point if $\|\nabla f(\mathbf{x})\|_2 = 0$ and $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq 0$. We also define its ϵ-version, that is, an ϵ-second-order stationary point for some $\epsilon > 0$, if point $\mathbf{x}$ satisfies $\|\nabla f(\mathbf{x})\|_2 \leq \epsilon$ and $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq -\sqrt{\rho\epsilon}$.*

Second-order stationary points must have a positive semi-definite Hessian in additional to a vanishing gradient. Note if all saddle points $\mathbf{x}^*$ are strict, then second-order stationary points are exactly equivalent to local minima.

In this chapter, we compare gradient descent:

$$\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} - \eta \nabla f\left(\mathbf{x}^{(t)}\right), \tag{7.1}$$

and one of its variants—the perturbed gradient descent algorithm (Algorithm 2) proposed by Jin et al. [45]. We focus on the case where the step size satisfies $\eta < 1/\ell$, which is commonly required for finding a minimum even in the convex setting [57].

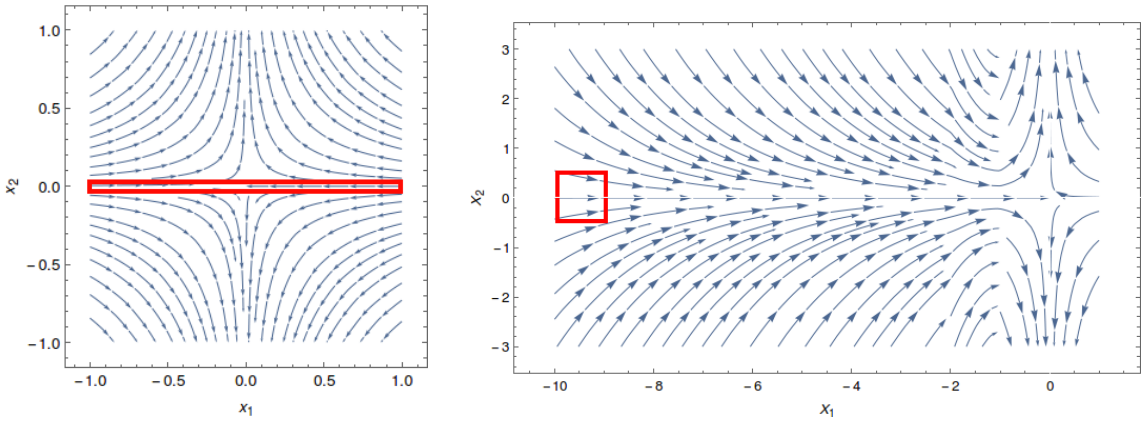
The following theorem shows that if GD with random initialization converges, then it will converge to a second-order stationary point almost surely.

Theorem 7.1 ([48])**.** *Suppose that f is ℓ-gradient Lipschitz, has continuous Hessian, and step size $\eta < \frac{1}{\ell}$. Furthermore, assume that gradient descent converges, meaning $\lim_{t\to\infty} \mathbf{x}^{(t)}$ exists, and the initialization distribution ν is absolutely continuous with respect to Lebesgue measure. Then $\lim_{t\to\infty} \mathbf{x}^{(t)} = \mathbf{x}^*$ with probability one, where $\mathbf{x}^*$ is a second-order stationary point.*

The assumption that gradient descent converges holds for many non-convex functions (including all the examples considered in this chapter). This assumption is used to avoid the case when $\left\|\mathbf{x}^{(t)}\right\|_2$ goes to infinity, so $\lim_{t\to\infty} \mathbf{x}^{(t)}$ is undefined.

Note the Theorem 7.1 only provides limiting behavior without specifying the convergence rate. On the other hand, if we are willing to add perturbations, the following theorem not only establishes convergence but also provides a sharp convergence rate:

Theorem 7.2 ([45])**.** *Suppose f is B-bounded, ℓ-gradient Lipschitz, ρ-Hessian Lipschitz. For any $\delta > 0$, $\epsilon \leq \frac{\ell^2}{\rho}$, there exists a proper choice of $\eta, r, t_{\text{thres}}, g_{\text{thres}}$ (depending on $B, \ell, \rho, \delta, \epsilon$)*



(a) Negative Gradient Field of $f(\mathbf{x}) = x_1^2 - x_2^2$.

(b) Negative Gradient Field for function defined in Equation (7.2).

Figure 7.1: If the initialization point is in red rectangle then it takes GD a long time to escape the neighborhood of saddle point $(0, 0)$.

such that Algorithm 2 will find an ϵ-second-order stationary point, with at least probability $1 - \delta$, in the following number of iterations:

$$O\left(\frac{\ell B}{\epsilon^2}\log^4\left(\frac{d\ell B}{\epsilon^2\delta}\right)\right).$$

This theorem states that with proper choice of hyperparameters, perturbed gradient descent can consistently escape strict saddle points and converge to second-order stationary point in a polynomial number of iterations.

7.3 Warmup: Examples with "Un-natural" Initialization

The convergence result of Theorem 7.1 raises the following question: can gradient descent find a second-order stationary point in a polynomial number of iterations? In this section, we discuss two very simple and intuitive counter-examples for which gradient descent with random initialization requires an exponential number of steps to escape strict saddle points. We will also explain that, however, these examples are unnatural and pathological in certain ways, thus unlikely to arise in practice. A more sophisticated counter-example with natural initialization and non-pathological behavior will be given in Section 7.4.

Initialize uniformly within an extremely thin band. Consider a two-dimensional function f with a strict saddle point at $(0, 0)$. Suppose that inside the neighborhood $U = [-1, 1]^2$ of the saddle point, function is locally quadratic $f(x_1, x_2) = x_1^2 - x_2^2$, For GD with $\eta = \frac{1}{4}$, the update equation can be written as

$$x_1^{(t+1)} = \frac{x_1^{(t)}}{2} \quad \text{and} \quad x_2^{(t+1)} = \frac{3x_2^{(t)}}{2}.$$

If we initialize uniformly within $[-1,1] \times [-(\frac{3}{2})^{-\exp(\frac{1}{\epsilon})}, (\frac{3}{2})^{-\exp(\frac{1}{\epsilon})}]$ then GD requires at least $\exp(\frac{1}{\epsilon})$ steps to get out of neighborhood U, and thereby escape the saddle point. See Figure 7.1a for illustration. Note that in this case the initialization region is *exponentially* thin (only of width $2 \cdot (\frac{3}{2})^{-\exp(\frac{1}{\epsilon})}$). We would seldom use such an initialization scheme in practice.

Initialize far away. Consider again a two-dimensional function with a strict saddle point at $(0,0)$. This time, instead of initializing in a extremely thin band, we construct a very long slope so that a relatively large initialization region necessarily converges to this extremely thin band. Specifically, consider a function in the domain $[-\infty, 1] \times [-1,1]$ that is defined as follows:

$$f(x_1, x_2) = \begin{cases} x_1^2 - x_2^2 & \text{if} \quad -1 < x_1 < 1 \\ -4x_1 + x_2^2 & \text{if} \quad x_1 < -2 \\ h(x_1, x_2) & \text{otherwise,} \end{cases} \tag{7.2}$$

where $h(x_1, x_2)$ is a smooth function connecting region $[-\infty, -2] \times [-1,1]$ and $[-1,1] \times [-1,1]$ while making f have continuous second derivatives and ensuring x_2 does not suddenly increase when $x_1 \in [-2,-1]$.[2] For GD with $\eta = \frac{1}{4}$, when $-1 < x_1 < 1$, the dynamics are

$$x_1^{(t+1)} = \frac{x_1^{(t)}}{2} \quad \text{and} \quad x_2^{(t+1)} = \frac{3x_2^{(t)}}{2},$$

and when $x_1 < -2$ the dynamics are

$$x_1^{(t+1)} = x_1^{(t)} + 1 \quad \text{and} \quad x_2^{(t+1)} = \frac{x_2^{(t)}}{2}.$$

Suppose we initialize uniformly within $[-R-1, -R+1] \times [-1,1]$, for R large. See Figure 7.1b for an illustration. Letting t denote the first time that $x_1^{(t)} \geq -1$, then approximately we have $t \approx R$ and so $x_2^{(t)} \approx x_2^{(0)} \cdot (\frac{1}{2})^R$. From the previous example, we know that if $(\frac{1}{2})^R \approx (\frac{3}{2})^{-\exp\frac{1}{\epsilon}}$, that is $R \approx \exp\frac{1}{\epsilon}$, then GD will need exponential time to escape from the neighborhood $U = [-1,1] \times [-1,1]$ of the saddle point $(0,0)$. In this case, we require an initialization region leading to a saddle point at distance R which is exponentially large. In practice, it is unlikely that we would initialize exponentially far away from the saddle points or optima.

7.4 Main Result

In the previous section we have shown that gradient descent takes exponential time to escape saddle points under "un-natural" initialization schemes. Is it possible for the same statement to hold even under "natural" initialization schemes and non-pathological functions? The following theorem confirms this:

Theorem 7.3 (Uniform initialization over a unit cube)**.** *Suppose the initialization point is uniformly sampled from $[-1,1]^d$. There exists a function f defined on $\mathbb{R}^d$ that is B-bounded, ℓ-gradient Lipschitz and ρ-Hessian Lipschitz with parameters B, ℓ, ρ at most $\mathrm{poly}(d)$ such that:*

[2] We can construct such a function using splines. See Appendix 7.7.

1. *with probability one, gradient descent with step size $\eta \leq 1/\ell$ will be $\Omega(1)$ distance away from any local minima for any $T \leq e^{\Omega(d)}$.*
2. *for any $\epsilon > 0$, with probability $1 - e^{-d}$, perturbed gradient descent (Algorithm 2) will find a point x such that $\|x - x^*\|_2 \leq \epsilon$ for some local minimum x^* in $\mathrm{poly}(d, \frac{1}{\epsilon})$ iterations.*

Remark: As will be apparent in the next section, in the example we constructed, there are 2^d symmetric local minima at locations $(\pm c \ldots, \pm c)$, where c is some constant. The saddle points are of the form $(\pm c, \ldots, \pm c, 0, \ldots, 0)$. Both algorithms will travel across d neighborhoods of saddle points before reaching a local minimum. For GD, the number of iterations to escape the i-th saddle point increases as κ^i (κ is a multiplicative factor larger than 1), and thus GD requires exponential time to escape d saddle points. On the other hand, PGD takes about the same number of iterations to escape each saddle point, and so escapes the d saddle points in polynomial time. Notice that $B, \ell, \rho = O(\mathrm{poly}(d))$, so this does not contradict Theorem 7.2.

We also note that in our construction, the local minimizers are outside the initialization region. We note this is common especially for unconstrained optimization problems, where the initialization is usually uniform on a rectangle or isotropic Gaussian. Due to isoperimetry, the initialization concentrates in a thin shell, but frequently the final point obtained by the optimization algorithm is not in this shell.

It turns out in our construction, the only second-order stationary points in the path are the final local minima. Therefore, we can also strengthen Theorem 7.3 to provide a negative result for approximating ϵ-second-order stationary points as well.

Corollary 7.1. *Under the same initialization as in Theorem 7.3, there exists a function f satisfying the requirements of Theorem 7.3 such that for some $\epsilon = 1/\mathrm{poly}(d)$, with probability one, gradient descent with step size $\eta \leq 1/\ell$ will not visit any ϵ-second-order stationary point in $T \leq e^{\Omega(d)}$.*

The corresponding positive result that PGD to find ϵ-second-order stationary point in polynomial time immediately follows from Theorem 7.2.

The next result shows that gradient descent does not fail due to the special choice of initializing uniformly in $[-1, 1]^d$. For a large class of initialization distributions ν, we can generalize Theorem 7.3 to show that gradient descent with random initialization ν requires exponential time, and perturbed gradient only requires polynomial time.

Corollary 7.2. *Let $B_\infty(\mathbf{z}, R) = \{\mathbf{z}\} + [-R, R]^d$ be the ℓ_∞ ball of radius R centered at $\mathbf{z}$. Then for any initialization distribution ν that satisfies $\nu(B_\infty(\mathbf{z}, R)) \geq 1 - \delta$ for any $\delta > 0$, the conclusion of Theorem 7.3 holds with probability at least $1 - \delta$.*

That is, as long as most of the mass of the initialization distribution ν lies in some ℓ_∞ ball, a similar conclusion to that of Theorem 7.3 holds with high probability. This result applies to random Gaussian initialization, $\nu = \mathcal{N}(0, \sigma^2 \mathbf{I})$, with mean 0 and covariance $\sigma^2 \mathbf{I}$, where $\nu(B_\infty(0, \sigma \log d)) \geq 1 - 1/\mathrm{poly}(d)$.

7.4.1 Proof Sketch

In this section we present a sketch of the proof of Theorem 7.3. The full proof is presented in the Appendix. Since the polynomial-time guarantee for PGD is straightforward to derive from Jin

et al. [45], we focus on showing that GD needs an exponential number of steps. We rely on the following key observation.

Key observation: escaping two saddle points sequentially. Consider, for $L > \gamma > 0$,

$$f(x_1, x_2) = \begin{cases} -\gamma x_1^2 + L x_2^2 & \text{if } x_1 \in [0,1], x_2 \in [0,1] \\ L(x_1-2)^2 - \gamma x_2^2 & \text{if } x_1 \in [1,3], x_2 \in [0,1] \\ L(x_1-2)^2 + L(x_2-2)^2 & \text{if } x_1 \in [1,3], x_2 \in [1,3] \end{cases}. \tag{7.3}$$

Note that this function is not continuous. In the next paragraph we will modify it to make it smooth and satisfy the assumptions of the Theorem but useful intuition is obtained using this discontinuous function. The function has an optimum at $(2,2)$ and saddle points at $(0,0)$ and $(2,0)$. We call $[0,1] \times [0,1]$ the neighborhood of $(0,0)$ and $[1,3] \times [0,1]$ the neighborhood of $(2,0)$. Suppose the initialization $\left(x^{(0)}, y^{(0)}\right)$ lies in $[0,1] \times [0,1]$. Define $t_1 = \min_{x_1^{(t)} \geq 1} t$ to be the time of first departure from the neighborhood of $(0,0)$ (thereby escaping the first saddle point). By the dynamics of gradient descent, we have

$$x_1^{(t_1)} = (1 + 2\eta\gamma)^{t_1} x_1^{(0)}, \quad x_2^{(t_1)} = (1 - 2\eta L)^{t_1} x_2^{(0)}.$$

Next we calculate the number of iterations such that $x_2 \geq 1$ and the algorithm thus leaves the neighborhood of the saddle point $(2,0)$ (thus escaping the second saddle point). Letting $t_2 = \min_{x_2^{(t)} \geq 1} t$, we have:

$$x_2^{(t_1)}(1 + 2\eta\gamma)^{t_2 - t_1} = (1 + 2\eta\gamma)^{t_2 - t_1}(1 - 2\eta L)^{t_1} x_2^{(0)} \geq 1.$$

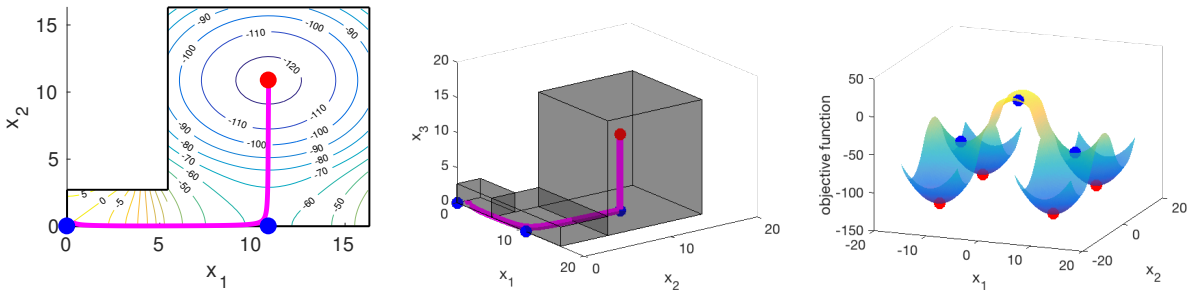
We can lower bound t_2 by

$$t_2 \geq \frac{2\eta(L+\gamma)t_1 + \log(\frac{1}{x_2^0})}{2\eta\gamma} \geq \frac{L+\gamma}{\gamma} t_1.$$

The key observation is that the number of steps to escape the second saddle point is $\frac{L+\gamma}{\gamma}$ times the number of steps to escape the first one.

Spline: connecting quadratic regions. To make our function smooth, we create buffer regions and use splines to interpolate the discontinuous parts of Equation (7.3). Formally, we consider the following function, for some fixed constant $\tau > 1$:

$$f(x_1, x_2) = \begin{cases} -\gamma x_1^2 + L x_2^2 & \text{if } x_1 \in [0,\tau], x_2 \in [0,\tau] \\ g(x_1, x_2) & \text{if } x_1 \in [\tau, 2\tau], x_2 \in [0,\tau] \\ L(x_1 - 4\tau)^2 - \gamma x_2^2 - \nu & \text{if } x_1 \in [2\tau, 6\tau], x_2 \in [0,\tau] \\ L(x_1 - 4\tau)^2 + g_1(x_2) - \nu & \text{if } x_1 \in [2\tau, 6\tau], x_2 \in [\tau, 2\tau] \\ L(x_1 - 4\tau)^2 + L(x_2 - 4\tau)^2 - 2\nu & \text{if } x_1 \in [2\tau, 6\tau], x_2 \in [2\tau, 6\tau], \end{cases} \tag{7.4}$$



(a) Contour plot of the objective function and tube defined in 2D.

(b) Trajectory of gradient descent in the tube for $d = 3$.

(c) Octopus defined in 2D.

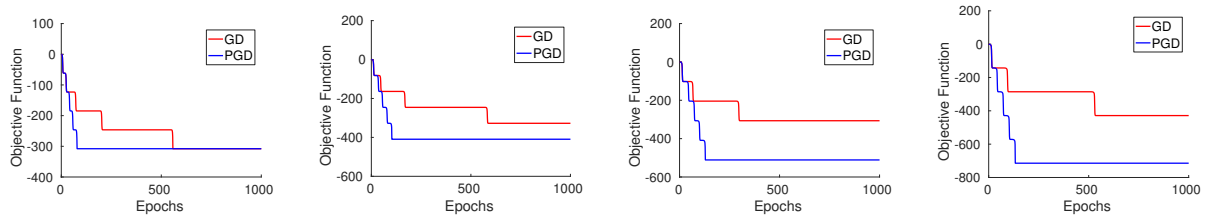
Figure 7.2: Graphical illustrations of our counter-example with $\tau = e$. The blue points are saddle points and the red point is the minimum. The pink line is the trajectory of gradient descent.

where g, g_1 are spline polynomials and $\nu > 0$ is a constant defined in Lemma 7.4. In this case, there are saddle points at $(0, 0)$, and $(4\tau, 0)$ and the optimum is at $(4\tau, 4\tau)$. Intuitively, $[\tau, 2\tau] \times [0, \tau]$ and $[2\tau, 6\tau] \times [\tau, 2\tau]$ are buffer regions where we use splines (g and g_1) to transition between regimes and make f a smooth function. Also in this region there is no stationary point and the smoothness assumptions are still satisfied in the theorem. Figure. 7.2a shows the surface and stationary points of this function. We call the union of the regions defined in Equation (7.4) a *tube*.

From two saddle points to d saddle points. We can readily adapt our construction of the tube to d dimensions, such that the function is smooth, the location of saddle points are $(0, \ldots, 0)$, $(4\tau, 0, \ldots, 0)$, ..., $(4\tau, \ldots, 4\tau, 0)$, and optimum is at $(4\tau, \ldots, 4\tau)$. Let t_i be the number of step to escape the neighborhood of the i-th saddle point. We generalize our key observation to this case and obtain $t_{i+1} \geq \frac{L+\gamma}{\gamma} \cdot t_i$ for all i. This gives $t_d \geq (\frac{L+\gamma}{\gamma})^d$ which is exponential time. Figure 7.2b shows the tube and trajectory of GD.

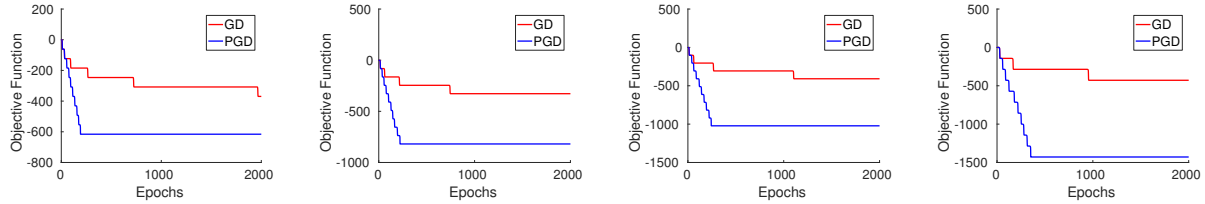
Mirroring trick: from tube to octopus. In the construction thus far, the saddle points are all on the boundary of tube. To avoid the difficulties of constrained non-convex optimization, we would like to make all saddle points be interior points of the domain. We use a simple mirroring trick; i.e., for every coordinate x_i we reflect f along its axis. See Figure 7.2c for an illustration in the case $d = 2$.

Extension: from octopus to $\mathbb{R}^d$. Up to now we have constructed a function defined on a closed subset of $\mathbb{R}^d$. The last step is to extend this function to the entire Euclidean space. Here we apply the classical Whitney Extension Theorem (Theorem 7.5) to finish our construction. We remark that the Whitney extension may lead to more stationary points. However, we will demonstrate in the proof that GD and PGD stay within the interior of "octopus" defined above, and hence cannot converge to any other stationary point.



(a) $L = 1, \gamma = 1$ (b) $L = 1.5, \gamma = 1$ (c) $L = 2, \gamma = 1$ (d) $L = 3, \gamma = 1$

Figure 7.3: Performance of GD and PGD on our counter-example with $d = 5$.



(a) $L = 1, \gamma = 1$ (b) $L = 1.5, \gamma = 1$ (c) $L = 2, \gamma = 1$ (d) $L = 3, \gamma = 1$

Figure 7.4: Performance of GD and PGD on our counter-example with $d = 10$

7.5 Experiments

In this section we use simulations to verify our theoretical findings. The objective function is defined in (7.14) and (7.15) in the Appendix. In Figures 7.3 and Figure 7.4, GD stands for gradient descent and PGD stands for Algorithm 2. For both GD and PGD we let the stepsize $\eta = \frac{1}{4L}$. For PGD, we choose $t_{\text{thres}} = 1$, $g_{\text{thres}} = \frac{\gamma e}{100}$ and $r = \frac{e}{100}$. In Figure 7.3 we fix dimension $d = 5$ and vary L as considered in Section 7.4.1; similarly in Figure 7.4 we choose $d = 10$ and vary L. First notice that in all experiments, PGD converges faster than GD as suggested by our theorems. Second, observe the "horizontal" segment in each plot represents the number of iterations to escape a saddle point. For GD the length of the segment grows at a fixed rate, which coincides with the result mentioned at the beginning for Section 7.4.1 (that the number of iterations to escape a saddle point increase at each time with a multiplicative factor $\frac{L+\gamma}{\gamma}$). This phenomenon is also verified in the figures by the fact that as the ratio $\frac{L+\gamma}{\gamma}$ becomes larger, the rate of growth of the number of iterations to escape increases. On the other hand, the number of iterations for PGD to escape is approximately constant ($\sim \frac{1}{\eta\gamma}$).

7.6 Conclusion and Future Work

In this chapter we established the failure of gradient descent to efficiently escape saddle points for general non-convex smooth functions. We showed that even under a very natural initialization scheme, gradient descent can require exponential time to converge to a local minimum whereas perturbed gradient descent converges in polynomial time. Our results demonstrate the necessity of adding perturbations for efficient non-convex optimization.

We expect that our results and constructions will naturally extend to a stochastic setting. In

particular, we expect that with random initialization, general stochastic gradient descent will need exponential time to escape saddle points in the worst case. However, if we add perturbations per iteration or the inherent randomness is non-degenerate in every direction (so the covariance of noise is lower bounded), then polynomial time is known to suffice [36].

One open problem is whether GD is inherently slow if the local optimum is inside the initialization region in contrast to the assumptions of initialization we used in Theorem 7.3 and Corollary 7.2. We believe that a similar construction in which GD goes through the neighborhoods of d saddle points will likely still apply, but more work is needed. Another interesting direction is to use our counter-example as a building block to prove a computational lower bound under an oracle model [57, 78].

Appendix: Omitted Proofs

7.7 Proofs for Results in Section 7.4

In this section, we provide proofs for Theorem 7.3 and Corollary 7.2. The proof for Corollary 7.1 easily follows from the same construction as in Theorem 7.3, so we omit it here. For Theorem 7.3, we will prove each claim individually.

7.7.1 Proof for Claim 1 of Theorem 7.3

Outline of the proof. Our construction of the function is based on the intuition in Section 7.4.1. Note the function f defined in (7.3) is 1) not continuous whereas we need a C^2 continuous function and 2) only defined on a subset of Euclidean space whereas we need a function defined on $\mathbb{R}^d$. To connect these quadratic functions, we use high-order polynomials based on spline theory. We connect d such quadratic functions and show that GD needs exponential time to converge if $\mathbf{x}^{(0)} \in [0,1]^d$. Next, to make all saddle points as interior point, we exploit symmetry and use a mirroring trick to create 2^d copies of the spline. This ensures that as long as the initialization is in $[-1,1]^d$, gradient descent requires exponential steps. Lastly, we use the classical Whitney extension theorem [77] to extend our function from a closed subset to $\mathbb{R}^d$.

Step 1: The tube. We fix four constants $L = e$, $\gamma = 1$, $\tau = e$ and $\nu = -g_1(2\tau) + 4L\tau^2$ where g_1 is defined in Lemma 7.4. We first construct a function f and a closed subset $D_0 \subset \mathbb{R}^d$ such that if $\mathbf{x}^{(0)}$ is initialized in $[0,1]^d$ then the gradient descent dynamics will get stuck around some saddle point for exponential time. Define the domain as:

$$D_0 = \bigcup_{i=1}^{d+1} \{x \in \mathbb{R}^d : 6\tau \geq x_1, \dots x_{i-1} \geq 2\tau, 2\tau \geq x_i \geq 0, \tau \geq x_{i+1} \dots, x_d \geq 0\}, \tag{7.5}$$

which $i = 1$ means $0 \leq x_1 \leq 2\tau$ and other coordinates are smaller than τ, and $i = d+1$ means that all coordinates are larger than 2τ. See Figure 7.5a for an illustration. Next we define the

objective function as follows. For a given $i = 1, \ldots, d-1$, if $6\tau \geq x_1, \ldots x_{i-1} \geq 2\tau, \tau \geq x_i \geq 0, \tau \geq x_{i+1} \ldots, x_d \geq 0$, we have

$$f(\mathbf{x}) = \sum_{j=1}^{i-1} L(x_j - 4\tau)^2 - \gamma x_i^2 + \sum_{j=i+1}^{d} L x_j^2 - (i-1)\nu \triangleq f_{i,1}(\mathbf{x}), \tag{7.6}$$

and if $6\tau \geq x_1, \ldots x_{i-1} \geq 2\tau, 2\tau \geq x_i \geq \tau, \tau \geq x_{i+1} \ldots, x_d \geq 0$, we have

$$f(\mathbf{x}) = \sum_{j=1}^{i-1} L(x_j - 4\tau)^2 + g(x_i, x_{i+1}) + \sum_{j=i+2}^{d} L x_j^2 - (i-1)\nu \triangleq f_{i,2}(\mathbf{x}), \tag{7.7}$$

where the constant ν and the bivariate function g are specified in Lemma 7.4 to ensure f is a C^2 function and satisfies the smoothness assumptions in Theorem 7.3. For $i = d$, we define the objective function as

$$f(\mathbf{x}) = \sum_{j=1}^{d-1} L(x_j - 4\tau)^2 - \gamma x_d^2 - (d-1)\nu \triangleq f_{d,1}(\mathbf{x}), \tag{7.8}$$

if $6\tau \geq x_1, \ldots x_{d-1} \geq 2\tau$ and $\tau \geq x_d \geq 0$ and

$$f(\mathbf{x}) = \sum_{j=1}^{d-1} L(x_j - 4\tau)^2 + g_1(x_d) - (d-1)\nu \triangleq f_{d,2}(\mathbf{x}) \tag{7.9}$$

if $6\tau \geq x_1, \ldots x_{d-1} \geq 2\tau$ and $2\tau \geq x_d \geq \tau$ where g_1 is defined in Lemma 7.4. Lastly, if $6\tau \geq x_1, \ldots x_d \geq 2\tau$, we define

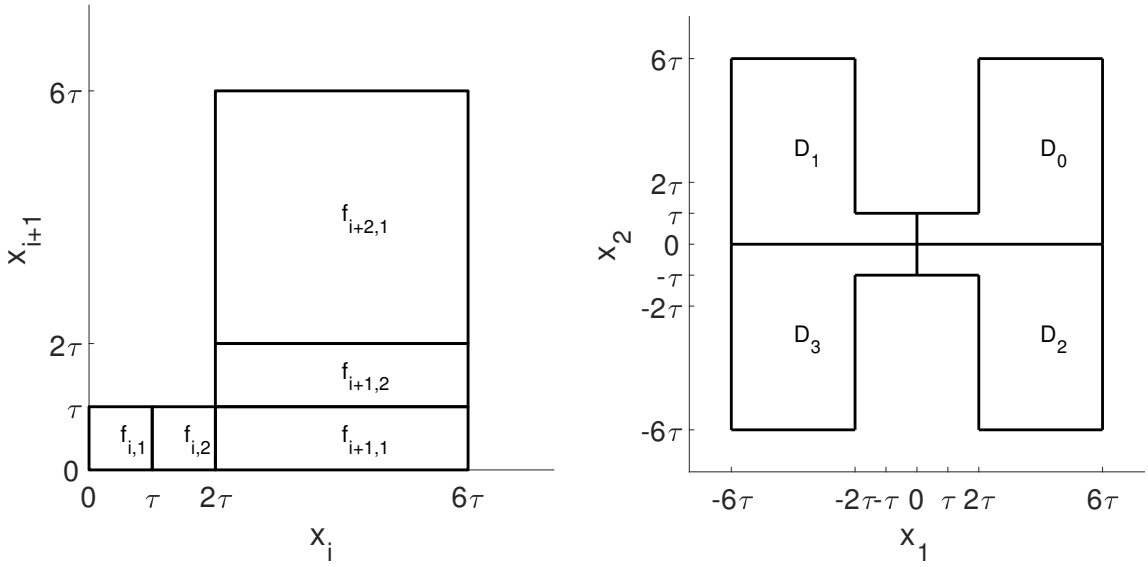
$$f(\mathbf{x}) = \sum_{j=1}^{d} L(x_j - 4\tau)^2 - d\nu \triangleq f_{d+1,1}(\mathbf{x}). \tag{7.10}$$

Figure. 7.5a shows an intersection surface (a slice along the x_i-x_{i+1} plane) of this construction.

Remark 7.1. *As will be apparent in Theorem 7.4, g and g_1 are polynomials with degrees bounded by five, which implies that for $\tau \leq x_i \leq 2\tau$ and $0 \leq x_{i+1} \leq \tau$ the function values and derivatives of $g(x_i, x_{i+1})$ and $g(x_i)$ are bounded by $\mathrm{poly}(L)$; in particular, $\rho = \mathrm{poly}(L)$.*

Remark 7.2. *In Theorem 7.4 we show that the norms of the gradients of g and g_1 gradients are strictly larger than zero by a constant ($\geq \gamma\tau$), which implies that for $\epsilon < \gamma\tau$, there is no ϵ-second-order stationary point in the connection region. Further note that in the domain of the function defined in Eq.* (7.6) *and* (7.8)*, the smallest eigenvalue of Hessian is -2γ. Therefore we know that if $\mathbf{x} \in D_0$ and $x_d \leq 2\tau$, then x cannot be an ϵ-second-order stationary point for $\epsilon \leq \frac{4\gamma^2}{\rho}$*

Now let us study the stationary points of this function. Technically, the differential is only defined on the interior of D_0. However in Steps 2 and 3, we provide a C^2 extension of f to all of $\mathbb{R}^d$, so the lemma below should be interpreted as characterizing the critical points of this extended function f in D_0. Using the analytic form of Eq. (7.6)- (7.10) and Remark 7.2, we can easily identify the stationary points of f.



(a) The intersection surface of the Tube defined in Equation (7.5) (7.6)and (7.7) for $2\tau \leq x_1, \ldots, x_{i-1} \leq 6\tau, 0 \leq x_{i+2} \leq \tau$.

(b) The "octopus"-like domain we defined in Equation (7.12) and (7.13) for $d = 2$.

Figure 7.5: Illustration of intersection surfaces used in our construction.

Lemma 7.1. *For $f : D_0 \to R$ defined in Eq.* (7.6) *to Eq.* (7.10)*, there is only one local optimum:*

$$\mathbf{x}^* = (4\tau, \ldots, 4\tau)^\top,$$

and d saddle points:

$$(0, \ldots, 0)^\top, (4\tau, 0, \ldots, 0)^\top, \ldots, (4\tau, \ldots, 4\tau, 0)^\top.$$

Next we analyze the convergence rate of gradient descent. The following lemma shows that it takes exponential time for GD to achieve $x_d \geq 2\tau$.

Lemma 7.2. *Let $\tau \geq e$ and $x^{(0)} \in [-1, 1]^d \cap D_0$. GD with $\eta \leq \frac{1}{2L}$ and any $T \leq \left(\frac{L+\gamma}{\gamma}\right)^{d-1}$ satisfies $x_d^{(T)} \leq 2\tau$.*

Proof. Define $T_0 = 0$ and for $k = 1, \ldots, d$, let $T_k = \min\{t | x_k^{(t)} \geq 2\tau\}$ be the first time the iterate escapes the neighborhood of the k-th saddle point. We also define T_k^τ as the number of iterations inside the region

$$\{x_1, \ldots, x_{k-1} \geq 2\tau, \tau \leq x_k \leq 2\tau, 0 \leq x_{k+\ell}, \ldots, x_d \leq \tau\}.$$

First we bound T_k^τ. Lemma 7.4 shows $\frac{\partial g(x_k, x_{k+1})}{\partial x_k} \leq -2\gamma\tau$ so after every gradient descent step, x_k is increased by at least $2\eta\gamma\tau$. Therefore we can upper bound T_k^τ by

$$T_k^\tau \leq \frac{2\tau - \tau}{2\eta\gamma\tau} = \frac{1}{2\eta\gamma}.$$

Note this bound holds for all k.

Next, we lower bound T_1. By definition, T_1 is the smallest number such that $x_1^{(T_1)} \geq 2\tau$ and using the definition of T_1^τ we know $x_1^{(T_1 - T_1^\tau)} \geq \tau$. By the gradient update equation, for $t = 1 \ldots, T_1 - T_1^\tau$, we have $x_1^t = (1 + 2\eta\gamma)^t x_1^0$. Thus we have:

$$\begin{aligned} x_1^{(0)} \left(1 + 2\eta\gamma\right)^{T_1 - T_1^\tau} &\geq \tau \\ \Rightarrow\ T_1 - T_1^\tau &\geq \frac{1}{2\eta\gamma} \log\left(\frac{\tau}{x_1^{(0)}}\right). \end{aligned}$$

Since $x_0^1 \leq 1$ and $\tau \geq e$, we know $\log(\frac{\tau}{x_1^0}) \geq 1$. Therefore $T_1 - T_1^\tau \geq \frac{1}{\eta\gamma} \geq T_1^\tau$.

Next we show iterates generated by GD stay in D_0. If $\mathbf{x}^{(t)}$ satisfies $6\tau \geq x_1, \ldots x_{k-1} \geq 2\tau, \tau \geq x_k \geq 0, \tau \geq x_{k+1} \ldots, x_d \geq 0$, then for $1 \leq j \leq k$,

$$x_j^{(t+1)} = \left(1 - \eta L\right) x_j^{(t)} - 4\eta L\tau \in \left[2\tau, 6\tau\right],$$

for $j = k$,

$$x_j^{(t+1)} = \left(1 + 2\eta\gamma\right) x_j^{(t)} \in \left[0, 2\tau\right],$$

and for $j \geq k + 1$

$$x_j^{(t+1)} = \left(1 - 2\eta L\right) x_j^{(t)} \in \left[0, \tau\right].$$

Similarly, if $x^{(t)}$ satisfies $6\tau \geq x_1, \ldots x_{k-1} \geq 2\tau, 2\tau \geq x_k \geq \tau, \tau \geq x_{k+1} \ldots, x_d \geq 0$, the above arguments still hold for $j \leq k - 1$ and $j \geq k + 2$. For $j = k$, note that

$$\begin{aligned} x_j^{(t+1)} &= x_j^{(t)} - \eta \frac{\partial g\left(x_j, x_{j+1}\right)}{\partial x_j} \\ &\leq x_j^{(t)} + 2\eta\gamma\tau \leq 6\tau, \end{aligned}$$

where in the first inequality we have used Lemma 7.4. For $j = k+1$, by the dynamics of gradient descent, at $(T_k - T_k^\tau)$-th iteration, $x_{k+1}^{(T_k - T_k^\tau)} = x_{k+1}^{(0)} \left(1 - 2\eta L\right)^{T_k - T_k^\tau}$. Note Lemma 7.4 shows in the region

$$\left\{x_1, \ldots, x_{k-1} \geq 2\tau, \tau \leq x_k \leq 2\tau, 0 \leq x_{k+1}, \ldots, x_d \leq \tau\right\},$$

we have

$$\frac{\partial f(x)}{\partial x_{k+1}} \geq -2\gamma x_{k+1}.$$

Putting this together we have the following upper bounds for $t = T_k - T_k^\tau + 1, \ldots, T_k$:

$$x_{k+1}^{(t)} \leq x_{k+1}^0 \left(1 - 2\eta L\right)^{\left(T_k - T_k^\tau\right)} \cdot \left(1 + 2\eta\gamma\right)^{t - \left(T_k - T_k^\tau\right)} \leq \tau, \tag{7.11}$$

which implies $\mathbf{x}^{(t)}$ is in D_0.

Next, let us calculate the relation between T_k and T_{k+1}. By our definition of T_k and T_k^τ, we have:

$$x_{k+1}^{(T_k)} \leq x_{k+1}^{(0)} \left(1 - 2\eta L\right)^{T_k - T_k^\tau} \cdot \left(1 + 2\eta\gamma\right)^{T_k^\tau}.$$

For T_{k+1}, with the same logic we used for lower bounding T_1, we have

$$\begin{aligned}
& x_{k+1}^{(T_{k+1}-T_{k+1}^{\tau})} \geq \tau \\
\Rightarrow\ & x_{k+1}^{(T_k)} (1+2\eta\gamma)^{T_{k+1}-T_{k+1}^{\tau}-T_k} \geq \tau \\
\Rightarrow\ & x_{k+1}^{(0)} (1-2\eta L)^{T_k-T_k^{\tau}} \cdot (1+2\eta\gamma)^{T_k^{\tau}} \cdot (1+2\eta\gamma)^{T_{k+1}-T_{k+1}^{\tau}-T_k} \geq \tau.
\end{aligned}$$

Taking logarithms on both sides and then using $\log(1-\theta) \leq -\theta$, $\log(1+\theta) \leq \theta$ for $0 \leq \theta \leq 1$, and $\eta \leq \frac{1}{2L}$, we have

$$\begin{aligned}
& 2\eta\gamma \left(T_{k+1} - T_{k+1}^{\tau} - (T_k - T_k^{\tau})\right) \geq \log\left(\frac{\tau}{x_{k+1}^0}\right) + 2\eta L \left(T_k - T_k^{\tau}\right) \\
\Rightarrow\ & T_{k+1} - T_{k+1}^{\tau} \geq \frac{L+\gamma}{\gamma} \left(T_k - T_k^{\tau}\right)
\end{aligned}$$

In last step, we used the initialization condition whereby $\log\left(\frac{\tau}{x_{k+1}^0}\right) \geq 1 \geq 0$. Since $T_1 - T_1^{\tau} \geq \frac{1}{2\eta\gamma}$, to enter the region $x_1, \ldots, x_d \geq 2\tau$ we need T_d iterations, which is lower bounded by

$$T_d \geq \frac{1}{2\eta\gamma} \cdot \left(\frac{L+\gamma}{\gamma}\right)^{d-1} \geq \left(\frac{L+\gamma}{\gamma}\right)^{d-1}.$$

□

Step 2: From the tube to the octopus. We have shown that if $x^0 \in [-1,1]^d \cap D_0$, then gradient descent needs exponential time to approximate a second order stationary point. To deal with initialization points in $[-1,1]^d - D_0$, we use a simple mirroring trick; i.e., for each coordinate x_i, we create a mirror domain of D_0 and a mirror function according to i-th axis and then take union of all resulting reflections. Therefore, we end up with an "octopus" which has 2^d copies of D_0 and $[-1,1]^d$ is a subset of this "octopus." Figure 7.5b shows the construction for $d = 2$.

The mirroring trick is used mainly to make saddle points be interior points of the region (octopus) and ensure that the positive result of PGD (claim 2) will hold.

We now formalize this mirroring trick. For $a = 0, \ldots, 2^d - 1$, let a_2 denote its binary representation. Denote $a_2(0)$ as the indices of a_2 with digit 0 and $a_2(1)$ as those that are 1. Now we define the domain

$$\begin{aligned}
D_a = \bigcup_{i=1}^{d} \big\{ & x \in \mathbb{R}^d : x_i \geq 0 \text{ if } i \in a_2(0), x_i \leq 0 \text{ otherwise }, \\
& 6\tau \geq |x_1| \ldots, |x_{i-1}| \geq 2\tau, |x_i| \leq 2\tau, |x_{i+1}| \ldots, |x_d| \leq \tau \big\}, \qquad (7.12)
\end{aligned}$$

$$D = \bigcup_{a=0}^{2^d-1} D_a. \qquad (7.13)$$

Note this is a closed subset of $\mathbb{R}^d$ and $[-1,1]^d \subset D$. Next we define the objective function. For $i = 1, \ldots, d-1$, if $6\tau \geq |x_1|, \ldots, |x_{i-1}| \geq 2\tau, |x_i| \leq \tau, |x_{i+1}| \ldots, |x_d| \leq \tau$:

$$f(\mathbf{x}) = \sum_{j \leq i-1, j \in a_2(0)} L(x_j - 4\tau)^2 + \sum_{j \leq i-1, j \in a_2(1)} L(x_j + 4\tau)^2 - \gamma x_i^2 + \sum_{j=i+1}^{d} L x_j^2 - (i-1)\nu, \tag{7.14}$$

and if $6\tau \geq |x_1|, \ldots, |x_{i-1}| \geq 2\tau, \tau \leq |x_i| \leq 2\tau, |x_{i+1}|, \ldots, |x_d| \leq \tau$:

$$f(\mathbf{x}) = \sum_{j \leq i-1, j \in a_2(0)} L(x_j - 4\tau)^2 + \sum_{j \leq i-1, j \in a_2(1)} L(x_j + 4\tau)^2 + G(x_i, x_{i+1}) + \sum_{j=i+2}^{d} L x_j^2 - (i-1)\nu, \tag{7.15}$$

where

$$G(x_i, x_{i+1}) = \begin{cases} g(x_i, x_{i+1}) \text{ if } i \in a_2(0) \\ g(-x_i, x_{i+1}) \text{ if } i \in a_2(1). \end{cases}$$

For $i = d$, if $6\tau \geq |x_1|, \ldots, |x_{i-1}| \geq 2\tau, |x_i| \leq \tau$:

$$f(\mathbf{x}) = \sum_{j \leq i-1, j \in a_2(0)} L(x_j - 4\tau)^2 + \sum_{j \leq i-1, j \in a_2(1)} L(x_j + 4\tau)^2 - \gamma x_i^2 - (i-1)\nu, \tag{7.16}$$

and if $6\tau \geq |x_1| \ldots, |x_{i-1}| \geq 2\tau, \tau \leq |x_i| \leq 2\tau$:

$$f(\mathbf{x}) = \sum_{j \leq i-1, j \in a_2(0)} L(x_j - 4\tau)^2 + \sum_{j \leq i-1, j \in a_2(1)} L(x_j + 4\tau)^2 + G_1(x_i) - (i-1)\nu, \tag{7.17}$$

where

$$G_1(x_i) = \begin{cases} g_1(x_i) \text{ if } i \in a_2(0) \\ g_1(-x_i) \text{ if } i \in a_2(1). \end{cases}$$

Lastly, if $6\tau \geq |x_1|, \ldots, |x_d| \geq 2\tau$:

$$f(\mathbf{x}) = \sum_{j \leq i-1, j \in a_2(0)} L(x_j - 4\tau)^2 + \sum_{j \leq i-1, j \in a_2(1)} L(x_j + 4\tau)^2 - d\nu. \tag{7.18}$$

Note that if a coordinate x_i satisfies $|x_i| \leq \tau$, the function defined in Eq. (7.14) to (7.17) is an even function (fix all x_j for $j \neq i$, $f(\ldots, x_i, \ldots) = f(\ldots, -x_i, \ldots)$) so f preserves the smoothness of f_0. By symmetry, mirroring the proof of Lemma 7.2 for D_a for $a = 1, \ldots, 2^d - 1$ we have the following lemma:

Lemma 7.3. *Choosing $\tau = e$, if $\mathbf{x}^{(0)} \in [-1,1]^d$ then for gradient descent with $\eta \leq \frac{1}{2L}$ and any $T \leq \left(\frac{L+\gamma}{\gamma}\right)^{d-1}$, we have $x_d^{(T)} \leq 2\tau$.*

Step 3: From the octopus to $\mathbb{R}^d$. It remains to extend f from D to $\mathbb{R}^d$. Here we use the classical Whitney extension theorem (Theorem 7.5) to obtain our final function F. Applying Theorem 7.5 to f we have that there exists a function F defined on $\mathbb{R}^d$ which agrees with f on D and the norms of its function values and derivatives of all orders are bounded by $O(\text{poly}(d))$. Note that this extension may introduce new stationary points. However, as we have shown previously, GD never leaves D so we can safely ignore these new stationary points. We have now proved the negative result regarding gradient descent.

7.7.2 Proof for Claim 2 of Theorem 7.3

To show that PGD approximates a local minimum in polynomial time, we first apply Theorem 7.2 which shows that PGD finds an ϵ-second-order stationary point. Remark 7.2 shows in D, every ϵ-second-order stationary point is ϵ close to a local minimum. Thus, it suffices to show iterates of PGD stay in D. We will prove the following two facts: 1) after adding noise, $\mathbf{x}$ is still in D, and 2) until the next time we add noise, $\mathbf{x}$ is in D.

For the first fact, using the choices of g_{thres} and r in Jin et al. [45] we can pick ϵ polynomially small enough so that $g_{\text{thres}} \leq \frac{\gamma\tau}{10}$ and $r \leq \frac{\tau}{20}$, which ensures there is no noise added when there exists a coordinate x_i with $\tau \leq x_i \leq 2\tau$. Without loss of generality, suppose that in the region

$$\{x_1, \ldots, x_{k-1} \geq 2\tau, 0 \leq x_k, \ldots, x_d \leq \tau\},$$

we have $\|\nabla f(\mathbf{x})\|_2 \leq g_{\text{thres}} \leq \frac{\gamma\tau}{10}$, which implies $|x_j - 4\tau| \leq \frac{\tau}{20}$for $j = 1, \ldots, k-1$, and $x_j \leq \frac{\tau}{20}$ for $j = k, \ldots, d$. Therefore, $\left|(\mathbf{x}+\xi)_j - 4\tau\right| \leq \frac{\tau}{10}$ for $j = 1, \ldots, k-1$ and

$$\left|(\mathbf{x}+\xi)_j\right| \leq \frac{\tau}{10} \tag{7.19}$$

for $j = k, \ldots, d$.

For the second fact suppose at the t'-th iteration we add noise. Now without loss of generality, suppose that after adding noise, $\mathbf{x}^{(t')} \geq 0$, and by the first fact $\mathbf{x}^{t'}$ is in the region

$$\left\{x_1, \ldots, x_{i-1} \geq 2\tau, 0 \leq x_i \leq \ldots, x_d \leq \frac{\tau}{10}\right\}.$$

Now we use the same argument as for proving GD stays in D. Suppose at t''-th iteration we add noise again. Then for $t' < t < t''$, we have that if $\mathbf{x}^{(t)}$ satisfies $6\tau \geq x_1, \ldots x_{k-1} \geq 2\tau, \tau \geq x_k \geq 0, \tau \geq x_{k+1} \ldots, x_d \geq 0$, then for $1 \leq j \leq k$,

$$x_j^{(t+1)} = (1 - \eta L) x_j^{(t)} - 4\eta L\tau \in [2\tau, 6\tau],$$

for $j = k$,

$$x_j^{(t+1)} = (1 + 2\eta\gamma) x_j^{(t)} \in [0, 2\tau],$$

and for $j \geq k+1$

$$x_j^{(t+1)} = (1 - 2\eta L) x_j^{(t)} \in [0, \tau].$$

Similarly, if $x^{(t)}$ satisfies $6\tau \geq x_1, \ldots x_{k-1} \geq 2\tau, 2\tau \geq x_k \geq \tau, \tau \geq x_{k+1} \ldots, x_d \geq 0$, the above arguments still hold for $j \leq k-1$ and $j \geq k+2$. For $j = k$, note that

$$\begin{aligned} x_j^{(t+1)} &= x_j^{(t)} - \eta \frac{\partial g\left(x_j, x_{j+1}\right)}{\partial x_j} \\ &\leq x_j^{(t)} + 4\eta L\tau \leq 6\tau, \end{aligned}$$

where the first inequality we have used Lemma 7.4.

For $j = k+1$, by the dynamics of gradient descent, at the $(T_k - T_k^\tau)$-th iteration, $x_{k+1}^{(T_k - T_k^\tau)} = x_{k+1}^{(t')} \left(1 - 2\eta L\right)^{T_k - T_k^\tau - t'}$. Note that Lemma 7.4 shows in the region

$$\left\{x_1, \ldots, x_{k-1} \geq 2\tau, \tau \leq x_k \leq 2\tau, 0 \leq x_{k+1}, \ldots, x_d \leq \tau\right\},$$

we have

$$\frac{\partial f(x)}{\partial x_{k+1}} \geq -2\gamma x_{k+1}.$$

Putting this together we obtain the following upper bound, for $t = T_k - T_k^\tau + 1, \ldots, T_k$:

$$x_{k+1}^{(t)} \leq x_{k+1}^{(t')} \left(1 - 2\eta L\right)^{(T_k - T_k^\tau - t')} \cdot \left(1 + 2\eta\gamma\right)^{t - \left(T_k - T_k^\tau\right)} \leq \tau,$$

where the last inequality is because $t - (T_k - T_k^\tau) \leq T_k^\tau \leq \frac{1}{2\eta\gamma}$. This implies $\mathbf{x}^{(t)}$ is in D_0. Our proof is complete.

7.7.3 Proof for Corollary 7.2

Define $g(\mathbf{x}) = f(\frac{\mathbf{x}-\mathbf{z}}{R})$ to be an affine transformation of f, $\nabla g(\mathbf{x}) = \frac{1}{R}\nabla f(\frac{\mathbf{x}-\mathbf{z}}{R})$, and $\nabla^2 g(\mathbf{x}) = \frac{1}{R^2}\nabla^2 f(\frac{\mathbf{x}-\mathbf{z}}{R})$. We see that $\ell_g = \frac{\ell_f}{R^2}$, $\rho_g = \frac{\rho_f}{R^3}$, and $B_g = B_f$, which are $poly(d)$.

Define the mapping $h(x) = \frac{\mathbf{x}-\mathbf{z}}{R}$, and the auxiliary sequence $\mathbf{y}_t = h(\mathbf{x}_t)$. We see that

$$\begin{aligned} \mathbf{x}^{(t+1)} &= \mathbf{x}^{(t)} - \eta \nabla g(\mathbf{x}^{(t)}) \\ h^{-1}(\mathbf{y}^{(t+1)}) &= h^{-1}(\mathbf{y}^{(t)}) - \frac{\eta}{R}\nabla f(\mathbf{y}^{(t)}) \\ \mathbf{y}^{(t+1)} &= h(R\mathbf{y}^{(t)} + z - \frac{\eta}{R}\nabla f(\mathbf{y}^{(t)})) \\ &= \mathbf{y}^{(t)} - \frac{\eta}{R^2}\nabla f(\mathbf{y}^{(t)}). \end{aligned}$$

Thus gradient descent with stepsize η on g is equivalent to gradient descent on f with stepsize $\frac{\eta}{R^2}$. The first conclusion follows from noting that with probability $1 - \delta$, the initial point $\mathbf{x}^{(0)}$ lies in $B_\infty(z, R)$, and then applying Theorem 7.3. The second conclusion follows from applying Theorem 7.2 in the same way as in the proof of Theorem 7.3.

7.8 Auxiliary Theorems

The following are basic facts from spline theory. See Equation (2.1) and (3.1) of [18]

Theorem 7.4. *Given data points $y_0 < y_1$, function values $f(y_0)$, $f(y_1)$ and derivatives $f'(y_0)$, $f'(y_1)$ with $f'(y_0) < 0$ the cubic Hermite interpolant is defined by*

$$p(y) = c_0 + c_1\delta_y + c_2\delta_y^2 + c_3\delta_y^3,$$

where

$$\begin{aligned} c_0 = f(y_0), c_1 = f'(y_0) \\ c_2 = \frac{3S - f'(y_1) - 2f'(y_0)}{y_1 - y_0} \\ c_3 = -\frac{2S - f'(y_1) - f'(y_0)}{(y_1 - y_0)^2} \end{aligned}$$

for $y \in [y_0, y_1]$, $\delta_y = y - y_0$ and slope $S = \frac{f(y_1)-f(y_0)}{y_1-y_0}$. $p(y)$ satisfies $p(y_0) = f(y_0)$, $p(y_1) = f(y_1)$, $p'(y_0) = f'(y_0)$ and $p'(y_1) = f'(y_1)$. Further, for $f(y_1) < f(y_0) < 0$, if

$$f'(y_1) \geq \frac{3\left(f(y_1) - f(y_0)\right)}{y_1 - y_0}$$

then we have $f(y_1) \leq p(y) \leq f(y_0)$ for $y \in [y_0, y_1]$.

We use these properties of splines to construct the bivariate function g and the univariate function g_1 in Section 7.7. The next lemma studies the properties of the connection functions $g(\cdot, \cdot)$ and $g_1(\cdot)$.

Lemma 7.4. *Define $g(x_i, x_{i+1}) = g_1(x_i) + g_2(x_i)x_{i+1}^2$. There exist polynomial functions g_1, g_2 and $\nu = -g_1(2\tau) + 4L\tau^2$ such that for any $i = 1, \cdots, d$, for $f_{i,1}$ and $f_{i,2}$ defined in Eq.* (7.6)-(7.10)*, $g(x_i, x_{i+1})$ ensures $f_{i,2}$ satisfies, if $x_i = \tau$, then*

$$\begin{aligned} f_{i,2}(\mathbf{x}) = f_{i,1}(\mathbf{x}), \\ \nabla f_{i,2}(\mathbf{x}) = \nabla f_{i,1}(\mathbf{x}), \\ \nabla^2 f_{i,2}(\mathbf{x}) = \nabla^2 f_{i,1}(\mathbf{x}), \end{aligned}$$

and if $x_i = 2\tau$ then

$$\begin{aligned} f_{i,2}(\mathbf{x}) = f_{i+1,1}(\mathbf{x}), \\ \nabla f_{i,2}(\mathbf{x}) = \nabla f_{i+1,1}(\mathbf{x}), \\ \nabla^2 f_{i,2}(\mathbf{x}) = \nabla^2 f_{i+1,1}(\mathbf{x}). \end{aligned}$$

Further, g satisfies for $\tau \leq x_i \leq 2\tau$ and $0 \leq x_{i+1} \leq \tau$

$$\begin{aligned} -4L\tau \leq \frac{\partial g(x_i, x_{i+1})}{\partial x_i} \leq -2\gamma\tau \\ \frac{\partial g(x_i, x_{i+1})}{\partial x_{i+1}} \geq -2\gamma x_{i+1}. \end{aligned}$$

and g_1 satisfies for $\tau \leq x_i \leq 2\tau$

$$-4L\tau \leq \frac{\partial g_1(x_i)}{\partial x_i} \leq -2\gamma\tau.$$

Proof. Let us first construct g_1. Since we know for a given $i \in [1, \ldots, d]$, if $x_i = \tau$, $\frac{\partial f_{i,1}}{\partial x_i} = -2\gamma\tau$, $\frac{\partial^2 f_{i,1}}{\partial x_i^2} = -2\gamma$ and if $x_i = 2\tau$, $\frac{\partial f_{i+1,1}}{\partial x_i} = -4L\tau$ and $\frac{\partial^2 f_{i+1,1}}{\partial x_i^2} = 2L$. Note for $L > \gamma$, $0 > -2\gamma\tau > -4L\tau$ and $2L > \frac{-4L\tau-(-2\gamma\tau)}{2\tau-\tau}$. Applying Theorem 7.4, we know there exists a cubic polynomial $p(x_i)$ such that

$$\begin{aligned} p(\tau) = -2\gamma\tau \quad &\text{and} \quad p(2\tau) = -4L\tau \\ p'(\tau) = -2\gamma \quad &\text{and} \quad p'(2\tau) = 2L, \end{aligned}$$

and $p(x_i) \leq -2\gamma\tau$ for $\tau \leq x_i \leq 2\tau$. Now define

$$g_1(x_i) = \left(\int p\right)(x_i) - \left(\int p\right)(\tau) - \gamma\tau^2.$$

where $\int p$ is the anti-derivative. Note by this definition g_1 satisfies the boundary condition at τ. Lastly we choose $\nu = -g_1(2\tau) + 4L\tau^2$. It can be verified that this construction satisfies all the boundary conditions.

Now we consider x_{i+1}. Note when if $x_i = \tau$, the only term in f that involves x_{i+1} is Lx_{i+1}^2 and when $x_i = 2\tau$, the only term in f that involves x_{i+1} is $-\gamma x_{i+1}^2$. Therefore we can construct g_2 directly:

$$g_2(x_i) = -\gamma - \frac{10(L+\gamma)(x_i - 2\tau)^3}{\tau^3} - \frac{15(L+\gamma)(x_i - 2\tau)^4}{\tau^4} - \frac{6(L+\gamma)(x_i - 2\tau)^5}{\tau^5}.$$

Note

$$g_2'(x_i) = -\frac{30(L+\gamma)(x_i - 2\tau)^2(x_i - \tau)^2}{\tau^5}.$$

After some algebra, we can show this function satisfies for $\tau \leq x_i \leq 2\tau$

$$\begin{aligned} g_2(x_i) &\geq -\gamma, \\ g_2'(x_i) &\leq 0, \\ g_2(\tau) = L, \quad g_2(2\tau) &= -\gamma \\ g_2'(\tau) = g_2'(2\tau) &= 0 \\ g_2''(\tau) = g_2''(2\tau) &= 0. \end{aligned}$$

Therefore it satisfies the boundary conditions related to x_{i+1}. Further note that at the boundary ($x_i = \tau$ or 2τ), the derivative and the second derivative are zero, so it will not contribute to the boundary conditions involving x_i. Now we can conclude that g and g_1 satisfy the requirements of the lemma. □

We use the following continuous extension theorem which is a sharpened result of the seminal Whitney extension theorem [77].

Theorem 7.5 (Theorem 1.3 of [11]). *Suppose $E \subseteq \mathbb{R}^d$. Let the $C^m(E)$ norm of a function $F : E \to \mathbb{R}$ be $\sup\{|\partial^\alpha| : \mathbf{x} \in E, |\alpha| \leq m\}$. If E is a closed subset in $\mathbb{R}^d$, then there exists a linear operator $T : C^m(E) \to C^m\left(R^d\right)$ such that if $f \in C^m(E)$ is mapped to $F \in C^m\left(\mathbb{R}^d\right)$, then $F|_E = f$ and F has derivatives of all orders on E^c. Furthermore, the operator norm $\|T\|_{op}$ is at most $Cd^{5m/2}$, where C depends only on m.*

Bibliography

[1] Pierre-Antoine Absil, Robert Mahony, and Benjamin Andrews. Convergence of the iterates of descent methods for analytic cost functions. *SIAM Journal on Optimization*, 16(2):531–547, 2005. 4.7.2

[2] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. *arXiv preprint arXiv:1811.04918*, 2018. 1

[3] Zeyuan Allen-Zhu, Yuanzhi Li, and Zhao Song. A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*, 2018. 5

[4] Sanjeev Arora, Nadav Cohen, and Elad Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International Conference on Machine Learning*, pages 244–253, 2018.

[5] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, and Ruosong Wang. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. *arXiv preprint arXiv:1901.08584*, 2019. 1.2.1, 1

[6] Sivaraman Balakrishnan, Simon S Du, Jerry Li, and Aarti Singh. Computationally efficient robust sparse estimation in high dimensions. In *Conference on Learning Theory*, pages 169–212, 2017. 1.2.1

[7] Maria-Florina Balcan, Simon Shaolei Du, Yining Wang, and Adams Wei Yu. An improved gap-dependency analysis of the noisy power method. In *Conference on Learning Theory*, pages 284–309, 2016. 1.2.1

[8] Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Global optimality of local search for low rank matrix recovery. In *Advances in Neural Information Processing Systems*, pages 3873–3881, 2016. 1, 5.1, 7.1

[9] Avrim Blum and Ronald L Rivest. Training a 3-node neural network is NP-complete. In *Advances in neural information processing systems*, pages 494–501, 1989. 1, 1

[10] 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*, pages 605–614. JMLR. org, 2017. 5.1, 5.1, 5.2, 5.2, 5.7, 1

[11] Alan Chang. The whitney extension theorem in high dimensions. *Revista Matemática Iberoamericana*, 33(2):623–632, 2017. 7.5

[12] 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*, pages 3040–3050, 2018. 2.1

[13] Youngmin Cho and Lawrence K Saul. Kernel methods for deep learning. In *Advances in neural information processing systems*, pages 342–350, 2009. 5.2

[14] Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pages 192–204, 2015. 5.2

[15] Francis H Clarke, Yuri S Ledyaev, Ronald J Stern, and Peter R Wolenski. *Nonsmooth analysis and control theory*, volume 178. Springer Science & Business Media, 2008. 4.2.1

[16] Amit Daniely. SGD learns the conjugate kernel class of the network. *arXiv preprint arXiv:1702.08503*, 2017. 1

[17] Damek Davis, Dmitriy Drusvyatskiy, Sham Kakade, and Jason D Lee. Stochastic subgradient method converges on tame functions. *Foundations of Computational Mathematics*, pages 1–36, 2018. 2, 4.2.3

[18] Randall L Dougherty, Alan S Edelman, and James M Hyman. Nonnegativity-, monotonicity-, or convexity-preserving cubic and quintic Hermite interpolation. *Mathematics of Computation*, 52(186):471–494, 1989. 7.8

[19] Dmitriy Drusvyatskiy, Alexander D Ioffe, and Adrian S Lewis. Curves of descent. *SIAM Journal on Control and Optimization*, 53(1):114–138, 2015. 4.2.3

[20] Simon Du, Jason Lee, Yuandong Tian, Aarti Singh, and Barnabas Poczos. Gradient descent learns one-hidden-layer CNN: Dont be afraid of spurious local minima. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1339–1348, 2018. 1.1

[21] Simon S Du and Surbhi Goel. Improved learning of one-hidden-layer convolutional neural networks with overlaps. *arXiv preprint arXiv:1805.07798*, 2018. 1.2.1

[22] Simon S Du and Wei Hu. Linear convergence of the primal-dual gradient method for convex-concave saddle point problems without strong convexity. *arXiv preprint arXiv:1802.01504*, 2018. 1.2.1

[23] Simon S Du and Wei Hu. Width provably matters in optimization for deep linear neural networks. *arXiv preprint arXiv:1901.08572*, 2019. 1.2.1

[24] Simon S Du and Jason D Lee. On the power of over-parametrization in neural networks with quadratic activation. In *International Conference on Machine Learning*, pages 1328–1337, 2018. 1.2.1

[25] Simon S Du, Jianshu Chen, Lihong Li, Lin Xiao, and Dengyong Zhou. Stochastic variance reduction methods for policy evaluation. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1049–1058. JMLR. org, 2017. 1.2.1

[26] Simon S Du, Chi Jin, Jason D Lee, Michael I Jordan, Aarti Singh, and Barnabas Poczos. Gradient descent can take exponential time to escape saddle points. In *Advances in Neural Information Processing Systems*, pages 1067–1077, 2017. 1.1

[27] Simon S Du, Jayanth Koushik, Aarti Singh, and Barnabás Póczos. Hypothesis transfer

learning via transformation functions. In *Advances in Neural Information Processing Systems*, pages 574–584, 2017. 1.2.1

[28] Simon S Du, Yining Wang, and Aarti Singh. On the power of truncated svd for general high-rank matrix estimation problems. In *Advances in Neural Information Processing Systems*, pages 445–455, 2017. 1.2.1

[29] Simon S Du, Wei Hu, and Jason D Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. In *Advances in Neural Information Processing Systems 31*, pages 382–393. 2018. 1.1

[30] Simon S Du, Jason D Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. *arXiv preprint arXiv:1811.03804*, 2018. 1.1

[31] Simon S. Du, Jason D. Lee, and Yuandong Tian. When is a convolutional filter easy to learn? In *International Conference on Learning Representations*, 2018. 1.1, 2.1, 5.2

[32] Simon S Du, Yining Wang, Sivaraman Balakrishnan, Pradeep Ravikumar, and Aarti Singh. Robust nonparametric regression under Huber's epsilon-contamination model. *arXiv preprint arXiv:1805.10406*, 2018. 1.2.1

[33] Simon S Du, Yining Wang, Xiyu Zhai, Sivaraman Balakrishnan, Ruslan R Salakhutdinov, and Aarti Singh. How many samples are needed to estimate a convolutional neural network? In *Advances in Neural Information Processing Systems*, pages 371–381, 2018. 1.2.1

[34] Simon S Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudík, and John Langford. Provably efficient rl with rich observations via latent state decoding. *arXiv preprint arXiv:1901.09018*, 2019. 1.2.1

[35] Simon S. Du, Xiyu Zhai, Barnabas Poczos, and Aarti Singh. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2019. 1.1

[36] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points − online stochastic gradient for tensor decomposition. In *Proceedings of The 28th Conference on Learning Theory*, pages 797–842, 2015. 1, 4.1, 6.3, 7.1, 7.2, 7.6

[37] Rong Ge, Jason D Lee, and Tengyu Ma. Matrix completion has no spurious local minimum. In *Advances in Neural Information Processing Systems*, pages 2973–2981, 2016. 1, 5.1, 7.1

[38] Rong Ge, Chi Jin, and Yi Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1233–1242, 2017. 1, 4.1, 4.1, 4.7.3, 4.7.3, 5.1, 7.1

[39] Rong Ge, Jason D Lee, and Tengyu Ma. Learning one-hidden-layer neural networks with landscape design. *arXiv preprint arXiv:1711.00501*, 2017. 5.1

[40] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010. 4.1, 5.1

[41] Moritz Hardt and Tengyu Ma. Identity matters in deep learning. *arXiv preprint arXiv:1611.04231*, 2016. 3.2.2, 3

[42] Moritz Hardt and Eric Price. The noisy power method: A meta algorithm with applications. In *Advances in Neural Information Processing Systems*, pages 2861–2869, 2014. 5.3, 5.10

[43] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 5.1

[44] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1.1, 3.1, 4.1, 6.1, 6.2.1

[45] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan. How to escape saddle points efficiently. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1724–1732, 2017. 1.1, 7.1, 2, 7.2, 7.2, 7.4.1, 7.7.2

[46] Kenji Kawaguchi. Deep learning without poor local minima. In *Advances In Neural Information Processing Systems*, pages 586–594, 2016. 5.1

[47] Yann LeCun, Léon Bottou, Genevieve B Orr, and Klaus-Robert Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–50. Springer, 1998. 5.1

[48] Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent only converges to minimizers. In *Conference on Learning Theory*, pages 1246–1257, 2016. 4.1, 4.1, 4.3.1, 4.7.4, 7.1, 7.1

[49] Xingguo Li, Zhaoran Wang, Junwei Lu, Raman Arora, Jarvis Haupt, Han Liu, and Tuo Zhao. Symmetry, saddle points, and global geometry of nonconvex matrix factorization. *arXiv preprint arXiv:1612.09296*, 2016. 1, 5.1, 7.1

[50] Yuanzhi Li and Yingyu Liang. Learning overparameterized neural networks via stochastic gradient descent on structured data. *arXiv preprint arXiv:1808.01204*, 2018. 1

[51] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. In *Advances in Neural Information Processing Systems*, pages 597–607, 2017. 2.1, 2.1, 5.1, 5.1, 5.2

[52] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013. 6.1

[53] Qihang Lin and Lin Xiao. An adaptive accelerated proximal gradient method and its homotopy continuation for sparse optimization. In *International Conference on Machine Learning*, pages 73–81, 2014. 6.2.1

[54] Paul Malliavin. Gaussian sobolev spaces and stochastic calculus of variations. In *Integration and Probability*, pages 229–252. Springer, 1995. 3.3

[55] Song Mei, Andrea Montanari, and Phan-Minh Nguyen. A mean field view of the landscape of two-layers neural networks. In *Proceedings of the National Academy of Sciences*, volume 115, pages E7665–E7671, 2018. 2.1

[56] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 565–571. IEEE, 2016. 6.1

[57] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*, volume 87. Springer Science & Business Media, 2013. 7.1, 7.2, 7.6

[58] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006. 7.2

[59] Behnam Neyshabur, Ruslan R Salakhutdinov, and Nati Srebro. Path-SGD: Path-normalized optimization in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 2422–2430, 2015. 5.1

[60] Ioannis Panageas and Georgios Piliouras. Gradient descent only converges to minimizers: Non-isolated critical points and invariant regions. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, 2017. 4.1, 4.3.1, 4.7.4

[61] Dohyung Park, Anastasios Kyrillidis, Constantine Carmanis, and Sujay Sanghavi. Non-square matrix sensing without spurious local minima via the Burer-Monteiro approach. In *Artificial Intelligence and Statistics*, pages 65–74, 2017. 1, 7.1

[62] Maithra Raghu, Ben Poole, Jon Kleinberg, Surya Ganguli, and Jascha Sohl Dickstein. On the expressive power of deep neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2847–2854. JMLR. org, 2017. 6.5

[63] Tim Salimans and Diederik P Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. In *Advances in Neural Information Processing Systems*, pages 901–909, 2016. 5.1, 5.7, 5.9.2

[64] Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. Weight sharing is crucial to succesful optimization. *arXiv preprint arXiv:1706.00687*, 2017. 5.1, 5.1, 5.2

[65] Ohad Shamir. Are ResNets provably better than linear predictors? In *Advances in Neural Information Processing Systems*, pages 505–514, 2018. 4.1

[66] Bin Shi, Simon S Du, Michael I Jordan, and Weijie J Su. Understanding the acceleration phenomenon via high-resolution differential equations. *arXiv preprint arXiv:1810.08907*, 2018. 4.5

[67] Mahdi Soltanolkotabi. Learning relus via gradient descent. In *Advances in Neural Information Processing Systems*, pages 2007–2017, 2017. 5.1

[68] Daniel Soudry and Yair Carmon. No bad local minima: Data independent training error guarantees for multilayer neural networks. *arXiv preprint arXiv:1605.08361*, 2016. 2.1

[69] Weijie Su, Stephen Boyd, and Emmanuel J Candès. A differential equation for modeling Nesterov's accelerated gradient method: Theory and insights. *Journal of Machine Learning Research*, 17(1):1–43, 2016. 4.5

[70] Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 2379–2383. IEEE, 2016. 1, 7.1

[71] Ju Sun, Qing Qu, and John Wright. Complete dictionary recovery over the sphere I: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2): 853–884, 2017. 1, 7.1

[72] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, pages 4278–4284, 2017. 6.1, 6.2.1

[73] 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*, pages 3404–3413, 2017. 5.1, 5.1, 5.2, 6.2, 6.3.2, 6.6.2

[74] Russell Tsuchida, Farbod Roosta-Khorasani, and Marcus Gallagher. Invariance of weight distributions in rectified MLPs. In *International Conference on Machine Learning*, pages 5002–5011, 2018. 2.2

[75] Stephen Tu, Ross Boczar, Max Simchowitz, Mahdi Soltanolkotabi, and Ben Recht. Low-rank solutions of linear matrix equations via procrustes flow. In *International Conference on Machine Learning*, pages 964–973, 2016. 4.1

[76] Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010. 3.14

[77] Hassler Whitney. Analytic extensions of differentiable functions defined in closed sets. *Transactions of the American Mathematical Society*, 36(1):63–89, 1934. 7.7.1, 7.8

[78] Blake E Woodworth and Nati Srebro. Tight complexity bounds for optimizing composite objectives. In *Advances in Neural Information Processing Systems*, pages 3639–3647, 2016. 7.6

[79] Bo Xie, Yingyu Liang, and Le Song. Diverse neural network learns true target functions. In *Artificial Intelligence and Statistics*, pages 1216–1224, 2017. 2.2, 5.2

[80] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 87.1–87.12, September 2016. 1.1

[81] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *Proceedings of the International Conference on Learning Representations (ICLR), 2017*, 2017. 1.1

[82] Hongyi Zhang, Yann N. Dauphin, and Tengyu Ma. Residual learning without normalization via better initialization. In *International Conference on Learning Representations*, 2019. 3.2.2

[83] Xiao Zhang, Simon Du, and Quanquan Gu. Fast and sample efficient inductive matrix completion via multi-phase procrustes flow. In *Proceedings of the 35th International Conference on Machine Learning*, pages 5756–5765, 2018. 1.2.1

[84] Kai Zhong, Zhao Song, and Inderjit S Dhillon. Learning non-overlapping convolutional neural networks with multiple kernels. *arXiv preprint arXiv:1711.03440*, 2017. 5.1, 5.2, 5.6

[85] Kai Zhong, Zhao Song, Prateek Jain, Peter L Bartlett, and Inderjit S Dhillon. Recovery guarantees for one-hidden-layer neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 4140–4149. JMLR. org, 2017. 5.1, 5.2, 5.6