Understanding Deep Learning via Analyzing Training Dynamics

by

Xiang Wang

Department of Computer Science
Duke University

Date: _____

Approved:

_____
Rong Ge, Supervisor

_____
Sayan Mukherjee

_____
Debmalya Panigrahi

_____
Jiaming Xu

Dissertation submitted in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in the Department of Computer Science
in the Graduate School of Duke University
2022

Abstract

Understanding Deep Learning via Analyzing Training Dynamics

by

Xiang Wang

Department of Computer Science
Duke University

Date: _____

Approved:

_____
Rong Ge, Supervisor

_____
Sayan Mukherjee

_____
Debmalya Panigrahi

_____
Jiaming Xu

An abstract of a dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science in the Graduate School of Duke University
2022

Abstract

Deep learning has achieved tremendous success in practice, yet the theoretical understanding lags behind. How does gradient descent successfully optimize the highly non-convex training objective, and how does it find a solution that also generalizes well to unseen data despite the model being over-parameterized? Answering these questions requires a characterization of the training dynamics of gradient descent. In this thesis, we first develop analysis techniques of training dynamics in tensor decompositions, and then showcase the explanation of two phenomena by analyzing gradient descent dynamics.

In the first part, we analyze the gradient descent dynamics in over-parameterized tensor decompositions. For non-orthogonal low-rank tensors, we show that gradient descent from a small initialization can identify the subspace that the ground-truth components lie in, and automatically exploit such structure to reduce the requirement on the over-parameterization. Then, for orthogonal tensors, we show gradient descent fits the ground truth components one by one from the larger components to the smaller components, similar to a tensor deflation process. Since tensor decomposition is closely related to the optimization of neural networks, we believe many techniques developed here will apply to neural networks as well.

In the second part, we explain two phenomena by analyzing the training dynamics of gradient descent. We first explain the representation learning process of non-contrastive self-supervised methods by analyzing the training dynamics on a linear

network. Our analysis reveals the role of weight decay in discarding the nuisance features and keeping the robust features. Then we show there will be a long plateau in both the loss and accuracy interpolation (between a random initialization with the minimizer it converges to) if different classes have different last-layer biases on a deep network. We also show how the last-layer biases for different classes can be different even on a perfectly balanced dataset by analyzing a simple model.

To my mother and father.

Contents

List of Tables

List of Figures

Acknowledgements

First and foremost, I would like to express my deepest gratitude to my advisor Rong Ge for all his advice, encouragement, and support. I continuously learn from his curiosity in exploring the unknown, passion for challenges, optimism under uncertainty, and comprehensive knowledge of theoretical machine learning. He has always been willing to discuss not only high-level ideas but also many technical details. He also gave me a lot of freedom in choosing research topics. I couldn't have wished for a better advisor.

I was very fortunate to work with great researchers during my internship at FAIR: Yuandong Tian, Simon S. Du, and Xinlei Chen. This experience broadened my horizon and brought a more practical perspective to my research. I also want to thank Fan Wu and Xiaohui Bei for advising my undergraduate research and leading me to the research path.

Many thanks to my committee members: Sayan Mukherjee, Debmalya Panigrahi, and Jiaming Xu for their thoughtful comments and valuable feedback on this thesis.

I want to express my gratitude to all my wonderful collaborators: Sanjeev Arora, Xinlei Chen, Muthu Chidambaram, Simon S. Du, Rong Ge, Yuzheng Hu, Wei Hu, Rohith Kuditipudi, Holden Lee, Jason D. Lee, Zhize Li, Zhiyuan Li, Tengyu Ma, Yunwei Ren, Yuandong Tian, Annie N. Wang, Zixuan Wang, Weiyao Wang, Chenwei Wu, Shuai Yuan, Yi Zhang, Mo Zhou, Xingyu Zhu. This thesis would not be possible without them.

Thank you to all the staff at Computer Science Department, especially Marilyn Butler and Elizabeth Labriola for their administrative work. I also thank my friends in the department: Muthu Chidambaram, Jiyao Hu, Zhengjie Miao, Kangning Wang, Chenwei Wu, Yuxi Yang, Ming Yang, Shuai Yuan, Hanrui Zhang, Mo Zhou for all the fond memories.

Finally, I want to thank my mother and father for their unconditional love and support. They are always the strongest backing through all the ups and downs. I also thank my girlfriend Jiefang Li for bringing happiness to my life.

1

Introduction

In recent years, deep learning has achieved tremendous success in many applications, including computer vision (Krizhevsky et al., 2012), natural language processing (Sutskever et al., 2014), and reinforcement learning (Mnih et al., 2015). However, almost all the innovations in deep learning require a huge amount of trial-and-error work, due to a lack of theoretical understanding of the underlying mechanisms. Building a theoretical foundation not only will allow us to understand the strengths and weaknesses of current methods, but also will guide us to develop even better network architectures and training algorithms.

Applying deep learning to a task typically consists of three components: a network architecture, a training objective, and an optimizer. We select a neural network architecture based on the characteristics of the task. For example, one can decide the number of layers, the number of neurons in each layer, and how the neurons in different layers are connected in a neural network. At the same time, we construct a training objective/loss using a set of training samples. This training objective is a function of network parameters and it measures the performance of the network on the training set. Finally, we apply some optimization method in the hope of finding

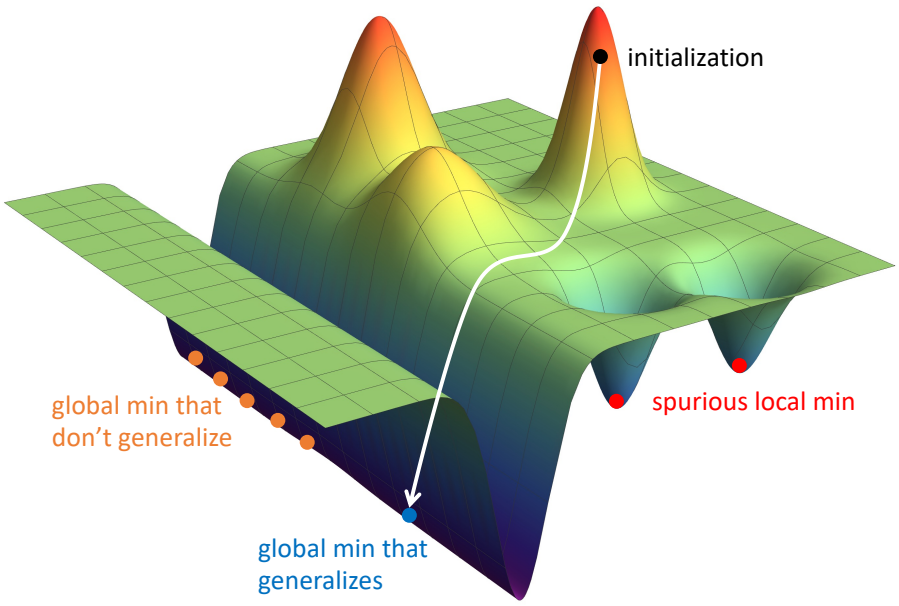

FIGURE 1.1: An illustration for the optimization landscape of over-parameterized neural networks. Gradient descent converges a global minimum that also generalizes, despite the existence of spurious local minimum and many global minima that do not generalize.

network parameters that can minimize the training objective. Most optimization methods used in practice are local search algorithms as varaints of gradient descent.

The empirical success of deep learning is very surprising from a theoretical perspective. The training loss is highly non-convex in the network parameters and has many local minima. It's known that for some matrix problems (Ge et al., 2016; Bhojanapalli et al., 2016) and simple neural networks (Kawaguchi, 2016; Ge et al., 2018a), all the local minima are global, therefore converging to any of them can minimize the loss. But for most neural networks and commonly used objective functions, spurious local minima do exist (Safran and Shamir, 2018). Presumably, gradient descent can get stuck at such bad local minimum, yet in practice, it usually succeeds in minimizing the training loss (see Figure 1.1). So how do local search algorithms such as (stochastic) gradient descent avoid the spurious local minima and converge to a global minimum? Furthermore, neural networks are usually over-parameterized

in practice with more parameters than the number of training samples. With the optimization problem being under-determined, there are many global minima of the training objective, most of which do not generalize to unseen data (Gunasekar et al., 2017; Soudry et al., 2018) (see Figure 1.1). So how does gradient descent converge to those particular minima that also generalize?

To understand the optimization and generalization in over-parameterized neural networks, we need to explain how gradient descent avoids all the bad local minima and the global minima that fail to generalize. These theoretical challenges call for an optimization theory that can characterize the training dynamics of gradient descent. In the first part of this thesis, we analyze the gradient descent dynamics on a simpler non-convex problem: over-parameterized tensor decompositions. Since tensor decomposition is closely related to the the optimization of two-layer neural networks, we expect many techniques developed for tensor decompositions will also be useful for neural networks. We will give an overview of Part 1 in Section 1.1.

Besides the mysteries in optimization and generalization, understanding many other phenomena in deep learning also requires the analysis of training dynamics. For example, in non-contrastive self-supervised learning, useful representations are learned by minimizing the distance between representations of different augmentations of the same image (Grill et al., 2020; Chen and He, 2020). For such a training objective, a trivial but useless solution is to map all the images to the same constant representation. To explain how gradient descent avoids such trivial solutions and finds meaningful representations, we need a characterization of gradient descent dynamics. In the second part of this thesis, we explain two such phenomena by analyzing the training dynamics of gradient descent. Since the neural networks that we are able to analyze are usually much simpler than the modern neural networks used in practice, we also conduct empirical experiments to verify that our theoretical insights can indeed transfer to realistic settings. We will outline the contribution of

Part 2 in Section 1.2.

1.1 Analyzing Gradient Descent Dynamics on Tensor Decomposition

The first part of this thesis consists of two works that analyze the gradient descent dynamics of over-parameterized tensor decomposition problems.

In a tensor decomposition problem, we are given a rank-r and order-l tensor[1] $T^* := \sum_{i=1}^{r} c_i^* \left[u_i^*\right]^{\otimes l}$ with $c_i^* \in \mathbb{R}$ and $u_i^* \in \mathbb{R}^d$ and our goal is to decompose it into a sum of rank-1 tensors[2]. Finding the decomposition with smallest possible rank r is known to be NP-Hard (Hillar and Lim, 2013). The problem becomes easier if we relax the goal to finding a decomposition with m components where m can be larger than r. A natural approach is to optimize the following objective using gradient descent

$$\min_{u_i \in \mathbb{R}^d, c_i \in \mathbb{R}} \left\| \sum_{i=1}^{m} c_i u_i^{\otimes l} - \sum_{i=1}^{r} c_i^* \left[u_i^*\right]^{\otimes l} \right\|_F^2 .$$

Empirically, gradient descent on the above objective will get stuck at a bad local minimum if $m = r$, but will find a global minimum if we over-parameterize the model with a larger m. So from a theoretical perspective, how much over-parameterization do we need to guarantee the convergence to a global minimum? In a lazy-training regime (similar to the NTK regime for neural networks), one needs m to be at least $\Omega(d^{l-1})$. According to the mean-field theory, gradient descent from a small initialization can escape the lazy-training regime. Yet simply applying the general mean-field analysis without leveraging the particular problem structure would require m to be exponentially large in d. Is it possible to improve this dependency by carefully char-

[1] We use $\otimes$ as the tensor product (outer-product). For a vector $v \in \mathbb{R}^d$, we define $v^{\otimes l}$ as a tensor in $\left(\mathbb{R}^d\right)^{\otimes l}$ and $v^{\otimes l}_{i_1,\cdots,i_l} = v_{i_1} v_{i_2} \cdots v_{i_l}$.

[2] A tensor $T \in (\mathbb{R}^d)^{\otimes l}$ is rank-1 if it can be written as $T = w \cdot v^{(1)} \otimes v^{(2)} \otimes \cdots \otimes v^{(l)}$ for some $w \in \mathbb{R}$ and $v^{(1)}, \cdots, v^{(l)} \in \mathbb{R}^d$, and the rank of a tensor is defined as the minimum integer k such that this tensor equals the sum of k rank-1 tensors.

acterizing the training dynamics on tensor decomposition problems? In Chapter 2, we prove that gradient descent from a small initialization can identify the subspace that the ground-truth components lie in, and automatically exploit such structure to reduce the number of necessary components. Our analysis only requires the number of components m to be $O(r^{2.5l} \log d)$, which is a significant improvement over the NTK requirement of $m = \Omega(d^{l-1})$ when $r \ll d$ and an exponential improvement over the existing mean-field analysis that requires $m = \exp(d)$.

Chapter 2 demonstrates that even identifying a simple property of the training dynamics (all u_i's lie on the space of u_i^*'s) can significantly reduce the requirement on the over-parameterization. In Chapter 3, we give a more precise characterization of the training dynamics when the ground truth tensor T^* is orthogonal (u_i^*'s are orthogonal). In particular, we prove that gradient descent fits the ground truth components one by one from the larger components to the smaller components, similar to a tensor deflation process. This also resembles the phenomenon in neural networks that gradient descent learns functions with increasing complexity during training (Nakkiran et al., 2019; Xu et al., 2019).

1.2 Explaining Deep Learning Phenomena via Dynamics Analysis

Understanding many phenomena in deep learning requires the analysis of training dynamics. In the second part of this thesis, we showcase the explaining of two phenomena in deep learning by combining the training dynamics analysis on simple neural networks and also empirical experiments in realistic settings.

1.2.1 Non-contrastive Self-supervised Learning

Self-supervised learning is a method to learn representations without using manual labels. As one popular approach, contrastive learning minimizes the distances between representations of two augmented views of the same data point (positive pairs) and

maximizes such distances between different data points (negative pairs) (He et al., 2020; Chen et al., 2020b). Intuitively, minimizing the distance between positive pairs encourages the learned representation to be invariant to data augmentations, and maximizing distances between negative pairs helps the representation to capture semantic meanings.

Recently, non-contrastive self-supervised learning (abbreviated as nc-SSL) was proposed to only minimize the distance between positive pairs (Grill et al., 2020; Chen and He, 2020). Presumably, nc-SSL can simply map all data to the same constant representation, which minimizes the training objective but does not capture any semantic meanings. However, in practice, nc-SSL learns non-trivial representation and achieves remarkable performance. So how does nc-SSL avoids the trivial solution in training and finds useful representations?

In the analysis of nc-SSL on a linear neural network, we prove that nc-SSL can learn a desirable projection matrix onto the features that are robust under data augmentations. Our analysis reveals an implicit threshold determined by the weight decay that discards the nuisance features with high augmentation variance and keeps the robust features with low augmentation variance. In our experiments, we verify this theoretical insight also applies to complicated networks on standard datasets. Motivated by our theory, we also propose a simpler and computationally more efficient algorithm than the previous counterpart.

1.2.2 Monotonic Linear Interpolation

To visualize and understand the optimization landscape of neural networks, one simple approach is to plot the loss and accuracy on the line connecting a random initialization with the minimizer it converges to. It has been observed that for some neural networks on some datasets, the interpolation curve is monotonic and approximately convex (which seemingly suggest an easy optimization landscape) (Goodfellow et al.,

2014), yet in some other settings both the loss and error remain high until close to the minimizer along the interpolation path (Frankle, 2020). So what causes this long plateau along the interpolation curve and do they imply the optimization is difficult?

In Chapter 5, we show that the shape of the interpolation curves can be easily manipulated by changing the bias terms, the network initialization scales and the network depth, which factors may not necessarily affect the difficulty of optimization. In particular, we prove that if a deep network has a small initialization, and its last-layer biases are different for different classes, then both the loss and error curves will have a long plateau. But why would the last-layer biases be different for different classes, especially when all the biases are initialized as zeros and all classes are balanced? By analyzing the training dynamics on a simple model, we show the bias associated with one class starts to decrease once this class is learned[3], and eventually the class that is learned last has the largest bias. In the experiments, we also show that the plateau in the loss/error interpolation curves can be expanded or shortened just by changing the bias terms, initialization scale and network depth, while these changes do not seem to affect the difficult of training

1.3 Related Works

Gradient descent can successfully optimize the non-convex training objective of neural networks. This seems to suggest that the loss landscape of neural networks has some benign properties. Early research in non-convex matrix problems (Sun et al., 2015, 2018; Ge et al., 2016; Bhojanapalli et al., 2016) has proved that all local minima are global and all saddle points have negative curvature. For non-convex functions satisfying these properties, gradient descent is guaranteed to converge to a global minimum (Ge et al., 2015a, 2019b; Jin et al., 2017, 2018). There have also been many landscape analyses for neural networks, but the results are mixed. Desirable

[3] One class is learned when most samples in this class get classified correctly with good confidence.

landscape properties have been established for deep linear networks (Kawaguchi, 2016; Hardt and Ma, 2016; Kawaguchi and Bengio, 2019; Zhou and Liang, 2017; Yun et al., 2018), over-parameterized neural networks (Freeman and Bruna, 2016; Soudry and Carmon, 2016; Venturi et al., 2018a; Li et al., 2018; Nguyen and Hein, 2017, 2018; Nguyen et al., 2018; Nguyen, 2019) and networks with modified training objectives or architectures (Liang et al., 2018; Kawaguchi and Kaelbling, 2020; Ge et al., 2018b), but spurious local minima do exist in many settings (Safran and Shamir, 2017; Venturi et al., 2018b; Ding et al., 2019). In deep neural networks, due to the existence of high-order saddle points (Kawaguchi, 2016), most landscape results do not imply that gradient descent converges to a global minimum.

Another way to prove convergence results is to directly analyze the training dynamics of gradient descent. Most early works assume the labels are generated by a ground truth network, and the goal is to recover the ground truth parameters by running gradient descent on a model with the same architecture. From a random initialization, it has been proved that gradient descent can learn a single ReLU neuron (Tian, 2017; Soltanolkotabi, 2017) or a single convolutional filter (Brutzkus and Globerson, 2017; Du et al., 2018c,a). Local convergence results from a tensor initialization have been proved in two-layer networks with multiple hidden neurons/filters (Zhong et al., 2017a,b; Zhang et al., 2019).

Neural networks in practice are often over-parameterized with more parameters than the number of samples or more than what's necessary for expressivity. Recently, there is a surge of research that proves the global convergence of gradient descent in over-parameterized neural networks, notably the Neural Tangent Kernel (NTK) theory and the mean-field theory. The NTK theory shows that starting from a relatively large initialization, when the network is polynomially wide, gradient descent will stay around the initialization (lazy-training regime) and its training dynamic is close to the dynamic of the kernel regression with NTK (Jacot et al., 2018; Allen-Zhu

et al., 2018a; Du et al., 2018a; Li and Liang, 2018; Du et al., 2019; Arora et al., 2019b; Zou et al., 2020; Oymak and Soltanolkotabi, 2020). However, the neural networks trained in practice usually go beyond the lazy-training regime and outperform the corresponding neural tangent kernels (Arora et al., 2019d; Chizat and Bach, 2018a). This gap between NTK and the neural network trained in the practical regime has also been established in theory (Wei et al., 2019; Allen-Zhu and Li, 2019, 2020a; Yehudai and Shamir, 2019; Ghorbani et al., 2019; Woodworth et al., 2020).

As an alternative optimization theory for over-parameterized neural networks, the mean-field theory was first developed for two-layer neural networks (Mei et al., 2018; Chizat and Bach, 2018b; Wei et al., 2018; Rotskoff and Vanden-Eijnden, 2018a; Sirignano and Spiliopoulos, 2018) and was later extended to multi-layer neural networks (Araújo et al., 2019; Nguyen and Pham, 2020; Pham and Nguyen, 2020; Lu et al., 2020; Fang et al., 2021; Sirignano and Spiliopoulos, 2022; Ding et al., 2022). This theory allows the neural network to move beyond the lazy-training regime, so it can better capture the networks trained in practice. However, most mean-field analyses need either exponential in dimension or exponential in time number of neurons to attain small training error. Furthermore, most results do not offer a precise characterization of the training dynamics, therefore cannot be used to explain many phenomena that depend on the training dynamics.

Although over-parameterization is helpful for optimization, it creates many global minima of the training objective that cannot generalize to unseen data. Surprisingly, gradient descent usually converges to those global minima that also generalize. It was conjectured that gradient descent is implicitly optimizing certain norm/margin that's related to the generalization performance. Many such results have then been proved for linear models (Soudry et al., 2018; Nacson et al., 2019a; Ji and Telgarsky, 2019b, 2021), matrix sensing (Gunasekar et al., 2017), deep linear networks (Gunasekar et al., 2018b; Ji and Telgarsky, 2019a), homogeneous networks (Nacson et al., 2019b;

Lyu and Li, 2019), and very-wide neural networks (Jacot et al., 2018; Chizat and Bach, 2020). More recently, a series of works showed that gradient descent is implicitly minimizing the rank in some settings, which cannot be captured by any norm minimization (Arora et al., 2019c; Gidel et al., 2019; Gissin et al., 2019; Razin and Cohen, 2020; Li et al., 2020b; Razin et al., 2021).

1.4 Notations

We define some common notations as follows, and will define other notations in later sections.

We use $[n]$ as a shorthand for $\{1, 2, \cdots, n\}$. We use $O(\cdot), \Theta(\cdot), \Omega(\cdot)$ to hide the dependency on constant factors and use $\widetilde{O}(\cdot), \widetilde{\Theta}(\cdot), \widetilde{\Omega}(\cdot)$ to hide the dependency on poly-logarithmic factors. We use $\mathrm{poly}(\cdot)$ to represent a polynomial on the relevant parameters with constant degree.

We use I_d to denote $d \times d$ identity matrix, and omit the subscript d when the dimension is clear. We use $\delta\mathrm{Unif}(\mathbb{S}^{d-1})$ to denote the uniform distribution over $(d-1)$-dimensional sphere with radius δ. We use $\mathbb{N}(\mu, \Sigma)$ to denote the Gaussian distribution with mean μ and covariance Σ.

Tensor notations: We use $\otimes$ as the tensor product (outer product). An l-th order d-dimensional tensor is defined as an element in space $\mathbb{R}^d \otimes \cdots \otimes \mathbb{R}^d$, succinctly denoted as $(\mathbb{R}^d)^{\otimes l}$. For any $i_1, \cdots, i_l \in [d]$, we use $T_{i_1,\cdots,i_l}$ to refer to the $(i_1, \cdots, i_l)$-th entry of $T \in (\mathbb{R}^d)^{\otimes l}$ with respect to the canonical basis. For a vector $v \in \mathbb{R}^d$, we define $v^{\otimes l}$ as a tensor in $(\mathbb{R}^d)^{\otimes l}$ such that $\left(v^{\otimes l}\right)_{i_1,\cdots,i_l} = v_{i_1} v_{i_2} \cdots v_{i_l}$. A tensor is symmetric if the entry values remain unchanged for any permutation of its indices.

A tensor $T \in (\mathbb{R}^d)^{\otimes l}$ is rank-1 if it can be written as $T = w \cdot v_1 \otimes v_2 \otimes \cdots \otimes v_l$ for some $w \in \mathbb{R}$ and $v_1, \cdots, v_l \in \mathbb{R}^d$, and the rank of a tensor is defined as the minimum integer k such that this tensor equals the sum of k rank-1 tensors.

We use $\|\cdot\|$ to denote the ℓ_2 norm of a vector or the spectral norm of a matrix. For l-th order tensors $T, T' \in (\mathbb{R}^d)^{\otimes l}$ (vectors and matrices can be viewed as tensors with order 1 and 2, respectively), we define the inner product as $\langle T, T' \rangle := \sum_{i_1,\cdots,i_l \in [d]} T_{i_1,\cdots,i_l} T'_{i_1,\cdots,i_l}$ and the Frobenius norm as $\|T\|_F = \sqrt{\sum_{i_1,\cdots,i_l \in [d]} T^2_{i_1,\cdots,i_l}}$.

1.5 Bibliographic Notes

The work presented in this thesis is contained in previous published papers or preprints with co-authors. Chapter 2 is based on a joint work (Wang et al., 2020) with Chenwei Wu, Jason Lee, Tengyu Ma and Rong Ge. Chapter 3 is based on a joint work (Ge et al., 2021) with Rong Ge, Yunwei Ren and Mo Zhou. Chapter 4 is based on a joint work (Wang et al., 2021b) with Xinlei Chen, Simon S. Du and Yuandong Tian. Chapter 5 is based on a joint work (Wang et al., 2022) with Annie N. Wang, Mo Zhou and Rong Ge.

1.5.1 Omitted Works

During my Ph.D. study, I have also co-authored on the following publications and preprints that are not included in this thesis:

- Work on proving convergence rate for stochastic variance reduced gradient descent (Ge et al., 2019b).
- Work on applying spectral methods to optimize two-layer neural networks under symmetric inputs (Ge et al., 2019a).
- Work on explaining mode connectivity in the optimization landscape of neural networks (Kuditipudi et al., 2019).
- Work on proving guarantees for the learning-to-learn approach (Wang et al., 2021a).

- Work on understanding Mixup augmentation (Chidambaram et al., 2021, 2022).
- Work on analyzing the edge-of-stability training dynamics (Zhu et al., 2022).

2

Low-rank Tensor Decomposition

In this chapter, we study the gradient descent dynamics on an over-parameterized tensor decomposition problem. Given an l-th order tensor in $(\mathbb{R}^d)^{\otimes l}$ of rank r (where $r \ll d$), can variants of gradient descent find a rank m decomposition where $m > r$? We show that in a lazy training regime (similar to the NTK regime for neural networks) one needs at least $m = \Omega(d^{l-1})$, while a variant of gradient descent can find an approximate tensor when $m = O^*(r^{2.5l} \log d)$. Our results show that gradient descent on over-parametrized objective could go beyond the lazy training regime and utilize certain low-rank structure in the data.

2.1 Introduction

Given an order-l symmetric tensor T^* in $(\mathbb{R}^d)^{\otimes l}$ with rank r, we aim to decompose it into a sum of rank-1 tensors with as few components as possible. Finding the low-rank decomposition with the smallest possible rank r is known to be NP-hard (Hillar and Lim, 2013). The problem becomes easier if we relax the goal to finding a decomposition with m components where m is allowed to be larger than r. The

natural approach is to optimize the following objective using gradient descent

$$\min_{u_i \in \mathbb{R}^d, c_i \in \mathbb{R}} \left\| \sum_{i=1}^{m} c_i u_i^{\otimes l} - \sum_{i=1}^{r} c_i^* [u_i^*]^{\otimes l} \right\|_F^2 . \tag{2.1}$$

When $m = r$, gradient descent on the objective above will empirically get stuck at a bad local minimum even for orthogonal tensors (Ge et al., 2015a). On the other hand, when $m = \Omega(d^{l-1})$, gradient descent provably converges to a global minimum near the initialization. This result follows straightforwardly from the Neural Tangent Kernel (NTK) technique (Jacot et al., 2018), which was originally developed to analyze neural network training, and is referred to as the "lazy training" regime because essentially the algorithm is optimizing a convex function near the initialization (Chizat and Bach, 2018a).

The main goal of this chapter is to understand whether we can go beyond the lazy training regime for the tensor decomposition problem via better algorithm design and analysis. In other words, we aim to use a much milder over-parametrization than $m = \Omega(d^{l-1})$ and still converge to the global minimum of objective (2.1). We view the problem as an important first step towards analyzing neural network training beyond the lazy training regime.

We build upon the technical framework of mean-field analysis (Mei et al., 2018), which was developed to analyze overparameterized neural networks. It allows the parameters to move far away from the initialization and therefore has the potential to capture the realistic training regime of neural networks. However, to date, all the provable optimization results with mean-field analysis essentially operate in the infinite or exponential overparameterization regime (Chizat and Bach, 2018b; Wei et al., 2019), and applying these techniques to our problem naively would require m to be exponentially large in d, which is even worse than the NTK result. The exponential dependency is *not surprising* because the mean-field analyses in (Chizat

and Bach, 2018b; Wei et al., 2019) do not leverage or assume any particular structures of the data so they fail to produce polynomial-time guarantees on the worst-case data. Motivated by identifying problem structure that allows for polynomial-time guarantees, we study the mean-field analysis applied to tensor decomposition.

The main contribution of this chapter is to attain nearly dimension-independent over-parametrization for the mean-field analysis in Wei et al. (2019) by leveraging the particular structure of the tensor decomposition problem, and to show that with $m = O^*(r^{2.5l} \log d)$, a modified version of gradient descent on a variant of objective (2.1) converges to the global minimum and recovers the ground-truth tensor. This is a significant improvement over the NTK requirement of $m = \Omega(d^{l-1})$ and an exponential improvement upon the existing mean-field analysis that requires $m = \exp(d)$. Our analysis shows that unlike the lazy training regime, gradient descent with small initialization and appropriate regularizer can identify the subspace that the ground-truth components lie in, and automatically exploit such structure to reduce the number of necessary components. As shown in Ge et al. (2018b), the population-level objective of two-layer networks is a mixture of tensor decomposition objectives with different orders, so our analysis may be extendable to improve the over-parametrization necessary in analysis of two-layer networks.

2.1.1 Related work

Neural Tangent Kernel There has been a recent surge of research on connecting neural networks trained via gradient descent with the neural tangent kernel (NTK) (Jacot et al., 2018; Du et al., 2018b,a; Chizat and Bach, 2018a; Allen-Zhu et al., 2018b; Arora et al., 2019d,b; Zou and Gu, 2019; Oymak and Soltanolkotabi, 2020). This line of analysis proceeds by coupling the training dynamics of the nonlinear network with the training dynamics of its linearization in a local neighborhood of the initialization, and then analyzing the optimization dynamics of the linearization which is

convex.

Though powerful and applicable to any function class including tensor decomposition, NTK is not yet a completely satisfying theory for explaining the success of over-parametrization in deep learning. Neural tangent kernel analysis is essentially dataset independent and requires at least number of neurons $m \geqslant \frac{n}{d} = d^{l-1}$ to find a global optimum (Zou and Gu, 2019; Daniely, 2019)[1].

Beyond NTK approach The gap between linearized models and the full neural network has been established in theory by (Wei et al., 2019; Allen-Zhu and Li, 2019; Yehudai and Shamir, 2019; Ghorbani et al., 2019; Dyer and Gur-Ari, 2019; Woodworth et al., 2020) and observed in practice (Chizat and Bach, 2018a; Arora et al., 2019d; Lee et al., 2019). Higher-order approximations of the gradient dynamics such as Taylorized Training (Bai and Lee, 2019; Bai et al., 2020) and the Neural Tangent Hierarchy (Huang and Yau, 2019) have been recently proposed towards closing this gap. Unlike this work, existing results mostly try to improve the sample complexity instead of the level of over-parametrization for the NTK approach.

Mean field approach For two-layer networks, a series of works used the mean field approach to establish the evolution of the network parameters (Mei et al., 2018; Chizat and Bach, 2018b; Wei et al., 2018; Rotskoff and Vanden-Eijnden, 2018a; Sirignano and Spiliopoulos, 2018). In the mean field regime, the parameters move significantly from their initialization, unlike NTK regime, so it is *a priori* possible for the mean field approach to exploit data-dependent structure to utilize fewer neurons. However the current analysis techniques for mean field approach need either exponential in dimension or exponential in time number of neurons to attain small training error and do not exploit any data structure. One of the main contributions

[1] Here n is the number of samples which is effectively $\Theta(d^l)$ in our setting.

of our work is to show that gradient descent can benefit from the low-rank structure in T^*.

Tensor decomposition Tensor decomposition is in general an NP-hard problem (Hillar and Lim, 2013). There are many algorithms that find the exact decomposition (when $m = r$) under various assumptions. In particular Jenrich's algorithm (Harshman, 1970) works when $r \leqslant d$ and the components are linearly independent. In our setting, the components may not be linearly independent, this is similar to the over-complete tensor decomposition problem. Although there are some algorithms for overcomplete tensor decomposition (e.g., Cardoso (1991); Ma et al. (2016)), they require nondegeneracy conditions which we are not assuming. When the number of components m is allowed to be larger than r, one can use spectral algorithms to find a decomposition where $m = \Theta(r^{l-1})$. In this chapter our focus is to achieve similar guarantees using a direct optimization approach.

Neural network with polynomial activations Another model that sits between tensor decomposition and standard ReLU neural network is neural network with polynomial activations. Livni et al. (2013) gave an algorithm for training network with quadratic activations with specific algorithm. Andoni et al. (2014) gave a way to learn degree l polynomials over d variables using $\Omega(d^l)$ neurons, which is similar to the guarantee of (much later) NTK approach.

2.1.2 Notations

Most notations have been defined in Section 1.4. Below we define some specific notations for this chapter.

We use $O^*(\cdot)$ to hide constant factors and also the dependency on accuracy ϵ. We define $\mathrm{vec}(\cdot)$ to be the vectorize operator for tensors, mapping a tensor in $(\mathbb{R}^d)^{\otimes l}$ to a vector in $\mathbb{R}^{d^l}$: $\mathrm{vec}(T)_{(i_1-1)d^{l-1}+(i_2-1)d^{l-2}+\cdots+(i_{l-1}-1)d+i_l} := T_{i_1,i_2,\cdots,i_l}$.

2.2 Problem setup and challenges

In this section we discuss the objective for over-parameterized tensor decomposition and explain the challenges in optimizing this objective.

We consider tensor decomposition problems with general order $l \geqslant 3$. Throughout the chapter we consider l as a constant. Specifically, we assume that the ground-truth tensor is T^* of rank at most r:

$$T^* := \sum_{i=1}^{r} c_i^* [u_i^*]^{\otimes l},$$

where $\forall i \in [r], c_i^* \in \mathbb{R}$ and $u_i^* \in \mathbb{R}^d$. Without loss of generality, we assume that $\|T^*\|_F = 1$. We focus on the low rank setting where r is much smaller than d. Note we don't assume that u_i^*'s are linearly independent.

The vanilla over-parameterized model we use consists of m components (where $m \geqslant r$):

$$T_v := \sum_{i=1}^{m} c_i u_i^{\otimes l},$$

where $\forall i \in [m], c_i \in \mathbb{R}$ and $u_i \in \mathbb{R}^d$. We use $U \in \mathbb{R}^{d \times m}$ to denote the matrix whose i-th column is u_i, and denote $C \in \mathbb{R}^{m \times m}$ as the diagonal matrix with $C_{i,i} = c_i$. The vanilla loss function we are considering is the square loss:

$$f_v(U, C) = \frac{1}{2} \|T_v - T^*\|_F^2 = \frac{1}{2} \left\| \sum_{i=1}^{m} c_i u_i^{\otimes l} - \sum_{i=1}^{r} c_i^* [u_i^*]^{\otimes l} \right\|_F^2. \tag{2.2}$$

In other words, we are looking for a rank m approximation to a rank r tensor. When $m = r$, the problem of finding a decomposition is known to be NP-hard. Therefore, our goal is to get a small objective value with small m (which corresponds to the rank of T_v). In the following sub-sections, we will see that there are many challenges to directly optimize the vanilla over-parameterized model over the vanilla

objective, so we will need to modify the parametrization of the tensor T_v and the optimization algorithm to overcome them.

2.2.1 Challenge 0: lazy training requires immense over-parameterization

We show lazy training requires $\Omega(d^{l-1})$ components to fit a rank-one tensor in the following theorem.

Theorem 2.1. *Suppose the ground truth tensor $T^* = [u^*]^{\otimes l}$, where u^* is uniformly sampled from the unit sphere $\mathbb{S}^{d-1}$. Lazy training (defined as below) requires $\Omega(d^{l-1})$ components to achieve $o(1)$ error in expectation.*

In the lazy training regime, all the u_i's stay very close to the initialization. Assuming the final u_i' is equal to $u_i + \delta_i$, all the higher-order terms in δ_i can be ignored. Therefore, the model can only capture tensors in the linear subspace $S_U = \text{span}\{P_{sym}\text{vec}(u_i^{\otimes l-1} \otimes \delta_i)\}_{i=1}^m$ (here P_{sym} is the projection to the space of vectorized symmetric tensors, u_i's are the initialization and δ_i's are arbitrary vectors in $\mathbb{R}^d$). The dimension of this subspace is upperbounded by dm. Let W_l be the space of all vectorized symmetric tensors in $(\mathbb{R}^d)^{\otimes l}$ (with dimension $\Omega(d^l)$), and $S_U^\perp$ be the subspace of W_l orthogonal to S_U. We show that for a random rank-1 tensor T^*, it will often have a large projection in $S_U^\perp$ unless $m = \Omega(d^{l-1})$. Basically, the subspace S_U has to cover the whole space W_l to approximate a random rank-1 tensor. The proof of Theorem 2.1 is in Appendix A.1.

2.2.2 Challenge 1: zero is a high-order saddle point for vanilla objective

As Chizat and Bach (2018a) pointed out, lazy training regime corresponds to the case where the initialization has large norm. A natural way to get around lazy training is to use a much smaller initialization. However, for the vanilla objective, 0 will be a saddle point of order l on the loss landscape. This makes gradient descent really slow

at the beginning. In Section 2.3, we fix this issue by re-parameterizing the model into a 2-homogeneous model.

2.2.3 Challenge 2: existence of bad local minima far away from 0

It was shown that no bad local minima exist in matrix decomposition problems (Ge et al., 2016). Therefore, (stochastic) gradient descent is guaranteed to find a global minimum. In this section, we show that in contrary tensor decomposition problems with order at least 3 have bad local minima.

Theorem 2.2. *Let $f_v(U, C)$ be as defined in Equation 2.2. Assume $l \geqslant 3, d > r \geqslant 1$ and $m \geqslant r(l+1)+1$. There exists a symmetric ground truth tensor T^* with rank at most $r(l+1)+1$ such that a local minimum with function value $l(l-1)r/4$ exists while the global minimum has function value zero.*

In the construction, we set all the u_i's to be $e_1/m^{1/l}$ so that $T = e_1^{\otimes l}$. We define the ground truth tensor by setting the residual $T - T^*$ to be $\sum_{j=2}^{r+1} e_j^{\otimes 2} \otimes e_1^{\otimes l-2}$ plus its $\binom{l}{2}$ permutations. At this point, the gradient equals zero, so there is no first order change to the function value. Furthermore, we show if any component moves in one of the missing direction e_j for $2 \leqslant j \leqslant r+1$, it will incur a second order function value increase. So the tensor can only moves along e_1 direction, which cannot further decrease the function value because $e_1^{\otimes l}$ is orthogonal with the residual. Note this is a bad local min but not a strict bad local min because we can shrink one component to zero and meanwhile increase another component so that the tensor does not change. When we have a zero component (it's a saddle point), we can add a missing direction to decrease the function value.

In Appendix A.2, we prove Theorem 2.2 and also construct a bad local minimum for 2-homogeneous model defined in Section 2.3. To escape these spurious local minima, our algorithm re-initializes one component after a fixed number of iterations.

2.3 Algorithms and main results

In this section, we introduce our main algorithm, a modified version of gradient descent on a non-convex objective, and state our main results.

To address the high-order saddle point issue in Section 2.2.2, we introduce a new variant of the parameterized models.

$$T := \sum_{i=1}^{m} a_i c_i^{l-2} u_i^{\otimes l},$$

where $\forall i \in [m], a_i \in \{-1, 1\}, c_i = \frac{1}{\|u_i\|}$ and $u_i \in \mathbb{R}^d$.

Note that since $u_i^{\otimes l}$ is homogeneous, there is a redundancy in the vanilla parametrization between the coefficient and the norm of $\|u\|$. Here we do the rescaling to make sure that the model T is a 2-homogeneous function of u_i's. Using the new formulation of T, 0 will no longer be a high order saddle point.

Recall that we use $U \in \mathbb{R}^{d\times m}$ to denote the matrix whose i-th column is u_i. We use $C, A \in \mathbb{R}^{m\times m}$ to denote the diagonal matrices with $C_{ii} = c_i, A_{ii} = a_i$. The loss function we are considering is the square loss plus a regularization term:

$$f(U, C, \hat{C}, A) \triangleq \frac{1}{2} \left\| \sum_{i=1}^{m} a_i c_i^{l-2} u_i^{\otimes l} - T^* \right\|_F^2 + \lambda \sum_{i=1}^{m} \hat{c}_i^{l-2} \|u_i\|^l,$$

where $\forall i \in [m], \hat{c}_i \in \mathbb{R}^+$ and we use $\hat{C} \in \mathbb{R}^{m\times m}$ to denote the diagonal matrix with $\hat{C}_{ii} = \hat{c}_i$. For simplicity, we use $\bar{C}$ to denote the tuple $(C, \hat{C}, A)$. Therefore, we can write $f(U, C, \hat{C}, A)$ as $f(U, \bar{C})$.

The algorithm contains K epochs, where each epoch includes H iterations. At the initialization, we independently sample each u_i from $\delta\text{Unif}(\mathbb{S}^{d-1})$, where the radius δ will be set to be $\text{poly}(\epsilon, 1/d)$.

Denote the subspace of $\text{span}\{u_i^*\}$ as S and its orthogonal subspace in $\mathbb{R}^d$ as B. Let P_S, P_B be the projection matrices onto subspace S and B, respectively. Since

the components of the ground-truth tensor lies in the subspace S, ideally we want to make sure that the components of tensor T lies in the same subspace S. We also want to make sure c_i roughly equals $1/\|P_S u_i\|$ to ensure the improvement in S subspace is large enough. However, the algorithm does not know the subspace S. We address this problem using the observation that $\|P_S u_i\| \approx \frac{\sqrt{r}}{\sqrt{d}}\|u_i\|$ at initialization; and $\|P_S u_i\| \approx \|u_i\|$ if norm of u_i is large, but its projection in B has not grown larger. In our algorithm we introduce a "scalar mode switch" step between these two regimes by the separation between C and $\hat{C}$: For the i-th component, the coefficients c_i and $\hat{c}_i$ are initialized as $\sqrt{d(m+K)}/\|u_i\|$ and $1/\|u_i\|$, respectively, and we reduce c_i by a factor of $\sqrt{d(m+K)}$ (c_i will be equal to $\hat{c}_i$ afterwards) when $\|u_i\|$ exceeds $2\sqrt{m+K}\delta$ for the first time. For each $i \in [m], a_i$ is i.i.d. sampled from Unif$\{1, -1\}$.

We also re-initialize one component at the beginning of each epoch. At each iteration, we first update U by gradient descent: $U' \leftarrow U - \eta \nabla_U f(U, \bar{C})$. Note that when taking the gradient over U, we treat c_i's and $\hat{c}_i$'s as constants. Then we update each c_i and $\hat{c}_i$ using the updated value of u_i to preserve 2-homogeneity, i.e., $c_i' = \frac{\|u_i\|}{\|u_i'\|} c_i$ and $\hat{c}_i' = \frac{\|u_i\|}{\|u_i'\|}\hat{c}_i$. We fix a_i's during the algorithm except for the initialization and re-initialization steps.

The pseudocode is given in Algorithm 1. Using this variant of gradient descent, we can recover the ground truth tensor T^* with high probability using only $O\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right)$ number of components. The formal theorem is stated below.

Theorem 2.3. *Given any target accuracy $\epsilon > 0$, there exists $m = O\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right)$, $\lambda = O\left(\frac{\epsilon}{r^{0.5l}}\right), \delta = poly(\epsilon, 1/d), \eta = poly(\epsilon, 1/d), H = poly(1/\epsilon, d)$ such that with probability at least 0.99, our algorithm finds a tensor T satisfying*

$$\|T - T^*\|_F \leqslant \epsilon,$$

Algorithm 1 Variant of Gradient Descent for Tensor Decomposition

Input: number of epochs K, number of iterations in one epoch H, initialization size δ, step size η.

For each $i \in [m]$, initialize u_i i.i.d. from $\delta\mathrm{Unif}(\mathbb{S}^{d-1})$; initialize a_i i.i.d. from $\mathrm{Unif}\{1,-1\}$; initialize c_i as $\frac{\sqrt{d(m+K)}}{\|u_i\|}$ and $\hat{c}_i$ as $\frac{1}{\|u_i\|}$.

for epoch $k := 1$ to K **do**

Let u_j be any vector with the smallest ℓ_2 norm among all columns of U.

Re-initialize u_j from $\delta\mathrm{Unif}(\mathbb{S}^{d-1})$, re-initialize a_j from $\mathrm{Unif}\{1,-1\}$ and set $c_j = \frac{\sqrt{d(m+K)}}{\|u_j\|}, \hat{c}_j = \frac{1}{\|u_j\|}$.

for iteration $t := 1$ to H **do**

$U' \leftarrow U - \eta\nabla_U f(U, \bar{C})$.

for $i := 1$ to m **do**

$c_i' \leftarrow \frac{\|u_i\|}{\|u_i'\|}c_i$; $\hat{c}_i' \leftarrow \frac{\|u_i\|}{\|u_i'\|}\hat{c}_i$.

if $\|u_i\| \leqslant 2\sqrt{m+K}\delta < \|u_i'\|$ holds for the first time since it was (re)-initialized **then**

$c_i' \leftarrow \frac{c_i'}{\sqrt{d(m+K)}}$. ▷ Scalar Mode Switch

$U \leftarrow U', C \leftarrow C', \hat{C} \leftarrow \hat{C}$.

Output: $T := \sum_{i=1}^m a_i c_i^{l-2} u_i^{\otimes l}$.

within $K = O\left(\frac{r^{2l}}{\epsilon^4}\log(d/\epsilon)\right)$ *epochs.*

2.4 Summary of our techniques

In this section, we discuss the high-level ideas that we need to prove Theorem 2.3. The full proof is deferred into Appendix A.3.

Generally, doing gradient descent never increases the objective value (though this is not obvious for our algorithm as it is slightly different in handling the normalization $c_i, \hat{c}_i$'s). Our main concern is to address Challenge 2, namely, the algorithm might get stuck at a bad local minimum. We will show that this cannot happen with the re-initialization procedure.

More precisely, we rely on the following main lemma to show that as long as the objective is large, there is at least a constant probability to improve the objective within one epoch.

Lemma 2.1. *In the setting of Theorem 2.3, let* $(U_0', \bar{C}_0')$ *and* $(U_H, \bar{C}_H)$ *be the parameters at the beginning of an epoch and the parameters at the end of the same epoch. Assume* $\|T_0' - T^*\|_F \geqslant \epsilon$*, where* T_0' *is tensor with parameters* $(U_0', \bar{C}_0')$*. Then with probability at least* $1/6$*, we have*

$$f(U_H, \bar{C}_H) - f(U_0', \bar{C}_0') = -\Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right).$$

We complement this lemma by showing that even if an epoch does not decrease the objective, it will not overly increase the objective.

Lemma 2.2. *In the setting of Theorem 2.3, let* $(U_0', \bar{C}_0')$ *and* $(U_H, \bar{C}_H)$ *be the parameters at the beginning of an epoch and the parameters at the end of the same epoch. Assume* $f(U_0', \bar{C}_0') \geqslant \epsilon^2$*, where* ϵ *is the target accuracy in Theorem 2.3. Then, we have* $f(U_H, \bar{C}_H) - f(U_0', \bar{C}_0') = O(\frac{1}{\lambda m})$.

From these two lemmas, we know that in each epoch, the loss function can decrease by $\Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right)$ with probability at least $\frac{1}{6}$, and even if we fail to decrease the function value, the increase of function value is at most $O\left(\frac{1}{\lambda m}\right)$. By our choice of parameters in Theorem 2.3, $m = \Theta\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right), \lambda = \Theta\left(\frac{\epsilon}{r^{0.5l}}\right)$ and then $O(\frac{1}{\lambda m}) = O(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)})$. Choosing a large constant factor in m, we can ensure that the function value decrease will dominate the increase. This allows us to prove Theorem 2.3.

In the next two subsections, we will discuss how to prove Lemma 2.1 and Lemma 2.2, respectively.

2.4.1 Proof sketch for Lemma 2.2 - upper bound on function increase

To prove the increase of f is bounded in one epoch, we identify all the possible ways that the loss can increase and upper bound each of them. We first show that

a normal step (without scalar mode switch) of the algorithm will not increase the objective function

Lemma 2.3. *In the setting of Theorem 2.3, let* $(U, \bar{C})$ *be the parameters at the beginning of one iteration and let* $U', \bar{C}'$ *be the updated parameters (before potential scalar mode switch). Assuming* $f(U, \bar{C}) \leqslant 10$*, we have* $f(U', \bar{C}') - f(U, \bar{C}) \leqslant -\frac{\eta}{l} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2$.

Note that we treat C and $\hat{C}$ as constants when taking gradient with respect to U and then update C and $\hat{C}$ according to the updated value of U, so this lemma does not directly follow from standard optimization theory. The gradient descent on U decreases the function value when the step size is small enough while updating $C, \hat{C}$ can potentially increase the function value. In order to show that overall the function value decreases, we need to bound the function value increase due to updating $C, \hat{C}$. We are able to do this because of the special regularizer we choose. In particular, our regularizer guarantees that the change introduced by updating C and $\hat{C}$ is proportional to the change of the gradient step, and is smaller in scale. Therefore we maintain the decrease in the gradient step.

Since we already know that the function value cannot increase in a normal iteration (before potential scalar mode switch), the only causes of the function value increase are the re-initialization or scalar mode switches. According to the algorithm, we only switch the scalar mode when the norm of a component reaches $2\sqrt{m+K}\delta$ for the first time, so the number of scalar mode switches in each epoch is at most m. Choosing δ to be small enough, the effects of scalar mode switches should be negligible. In the re-intialization, we remove the component with smallest ℓ_2 norm, which can increase the function value by at most $O(\frac{1}{\lambda m})$. This is proved in Lemma 2.4.

Lemma 2.4. *In the setting of Theorem 2.3, let* $(U_0', \bar{C}_0')$ *and* $(U_0, \bar{C}_0)$ *be the parameters before and after the reinitialization step, respectively. Assume* $f(U_0', \bar{C}_0') \geqslant \epsilon^2$,

where ϵ comes from Theorem 2.3. Then, we have $f(U_0, \bar{C}_0) - f(U_0', \bar{C}_0') = O(\frac{1}{\lambda m})$.

In the proof, we can show the function value is at most a constant and then $\sum_{i=1}^{m} \|u_i\|^2 = O(1/\lambda)$ due to the regularizer. Since we choose the reinitialized component u as one of the component with smallest ℓ_2 norm, we know $\|u\|^2 = O(\frac{1}{\lambda m})$. This then allows us to bound the function value change from reinitialization by $O(\frac{1}{\lambda m})$. Lemma 2.2 follows from Lemma 2.3 and Lemma 2.4.

2.4.2 Proof sketch for Lemma 2.1 - escaping local minima

In this section, we will show how we can escape local minima by re-initialization. Intuitively, we will show that when a component is randomly re-initialized, it has a positive probability of having a good correlation with the current residual $T - T^*$. However, there is a major obstacle here: because the component is re-initialized in the full d-dimensional space, the correlation of this new component with $T - T^*$ is going to be of the order $d^{-l/2}$. If every epoch can only improve the objective function by $d^{-l/2}$ we would need a much larger number of epochs and components.

We solve this problem by observing that both T and T^* are almost entirely in the subspace S. If we only care about the projection in S, the random component will have a correlation of $r^{-l/2}$ with the residual $T - T^*$. We will show that such a correlation will keep increasing until the norm of the new component is large enough, therefore decreasing the objective function.

First of all, we need to show that the influence coming from the subspace B (the orthogonal subspace of the span of $\{u_i^*\}$) is small enough so that it can be ignored.

Lemma 2.5. *In the setting of Theorem 2.3, we have $\|P_B U\|_F^2 \leqslant (m+K)\delta^2$ throughout the algorithm.*

We prove Lemma 2.5 by showing the norm of $P_B U$ only increases at the (re-)initializations, so it will stay small throughout this algorithm. This lemma is also

the motivation of our algorithm, i.e., we treat C and $\hat{C}$ as constants when taking the gradient so that the gradient of U will never have negative correlation with $P_B U$.

Now let us focus on the subspace S. We denote the re-initialized vector at t-th step as u_t, and its sign as $a \in \{\pm 1\}$, and we will take a look at the change of $P_S u_t$. Our analysis focuses on the correlation between $P_S u_t$ and the residual tensor: $\langle (P_{S^{\otimes l}} T_t - T^*), a(\overline{P_S u_t}^{\otimes l}) \rangle$. Here $\overline{P_S u_t}$ is the normalized version $P_S u_t$. We will show that the norm of u_t will blow up exponentially if this correlation is significantly negative at every iteration.

Towards this goal, first we will show that the initial point $P_S u_0$ has a large negative correlation with the residual. We lower bound this correlation by anti-concentration of Gaussian polynomials:

Lemma 2.6. *Suppose the residual at the beginning of one epoch is $T_0' - T^*$. Suppose $a c_0^{l-2} u_0^{\otimes l}$ is the reinitialized component. With probability at least $1/5$,*

$$\left\langle P_{S^{\otimes l}} T_0' - T^*, a\overline{P_S u_0}^{\otimes l} \right\rangle \leqslant -\Omega\left(\frac{1}{r^{0.5l}}\right) \left\| P_{S^{\otimes l}} T_0' - P_{S^{\otimes l}} T^* \right\|_F,$$

where $\overline{P_S u_0} = P_S u_0 / \left\| P_S u_0 \right\|$.

Our next step argues that if this negative correlation is large in every step, then the norm of u_t blows up exponentially:

Lemma 2.7. *In the setting of Theorem 2.3, within one epoch, let T_0 be the tensor after the reinitilization and let T_τ be the tensor at the end of the τ-th iteration. Assume $\|P_S u_0\| \geqslant \Omega(\delta/\sqrt{d})$. For any $t \geqslant 1$, as long as $\left\langle P_{S^{\otimes l}} T_\tau - T^*, a\overline{P_S u_\tau}^{\otimes l} \right\rangle \leqslant -\Omega\left(\frac{\epsilon}{r^{0.5l}}\right)$ for all $t - 1 \geqslant \tau \geqslant 0$, we have*

$$\|P_S u_t\|^2 \geqslant \left(1 + \Omega\left(\frac{\eta\epsilon}{r^{0.5l}}\right)\right)^t \|P_S u_0\|^2.$$

Therefore the final step is to show that $P_S u_t$ always have a large negative correlation with $T_t - T^*$, unless the function value has already decreased. The difficulty here is that both the current reinitialized component u_t and other components are moving, therefore T_t is also changing.

We can bound the change of the correlation by separating it into two terms, which are the change of the re-initialized component and the change of the residual:

$$\left\langle P_{S^{\otimes l}} T_t - T^*, a\overline{P_S u_t}^{\otimes l} \right\rangle - \left\langle P_{S^{\otimes l}} T_0 - T^*, a\overline{P_S u_0}^{\otimes l} \right\rangle$$

$$\leqslant \sum_{\tau=1}^{t} \left(\left\langle P_{S^{\otimes l}} T_{\tau-1} - T^*, \overline{P_S u_\tau}^{\otimes l} \right\rangle - \left\langle P_{S^{\otimes l}} T_{\tau-1} - T^*, \overline{P_S u_{\tau-1}}^{\otimes l} \right\rangle \right) + \sum_{\tau=1}^{t} \left\| T_\tau - T_{\tau-1} \right\|_F .$$

The change of the re-initialized component has a small effect on the correlation because the change in S subspace can only improve the correlation, and the influence of the B subspace can be bounded. This is formally proved in the following lemma.

Lemma 2.8. *In the setting of Theorem 2.3, suppose at the beginning of one iteration, the tensor T has parameters $(U, \bar{C})$. Suppose u is one column vector in U with $\|P_S u\| = \Omega(\frac{\delta}{\sqrt{d}})$ and $u' = u - \eta \nabla_u f(U, \bar{C})$. We have*

$$\left\langle P_{S^{\otimes l}} T - T^*, a\overline{P_S u'}^{\otimes l} \right\rangle \leqslant \left\langle P_{S^{\otimes l}} T - P_{S^{\otimes l}} T^*, a\overline{P_S u}^{\otimes l} \right\rangle + \eta\delta \, poly(d),$$

where poly(d) does not hide any dependency on η, δ.

Therefore, the only way to change the residual term by a lot must be changing the tensor T, and the accumulated change of T is strongly correlated with the decrease of f. This is similar to the technique of bounding the function value decrease in Wei et al. (2019). The connection between them are formalized in the following lemma:

Lemma 2.9. *In the same setting of Lemma 2.7, within one epoch, let $(U_0, \bar{C}_0)$ be the parameters after the reinitialization step and let $(U_H, \bar{C}_H)$ be the parameters at*

the end of this epoch. We have

$$\sum_{\tau=1}^{H} \|T_\tau - T_{\tau-1}\|_F \leqslant O\left(\sqrt{\frac{\eta H}{\lambda}}\right)\sqrt{f(U_0, \bar{C}_0) - f(U_H, \bar{C}_H) + \delta^2 poly(d)} + \delta^2 poly(d),$$

where poly(d) does not hide dependency on δ.

Intuitively, Lemma 2.9 is true because a large accumulated change of T indicates large gradients along the trajectory, which suggests a large decrease in the function value. In fact, we choose the parameters such that $\lambda = \Theta(\frac{\epsilon}{r^{0.5l}}), \eta H = \Theta(\frac{r^{0.5l}}{\epsilon}\log(d/\epsilon))$. If the accumulated change of T is larger than $\Omega(\frac{\epsilon}{r^{0.5l}})$, the function value decreases by at least $\Omega(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)})$, as stated in Lemma 2.1.

Combining all the steps, we show that either the function value has already decreased (by Lemma 2.9), or the correlation remains negative and the norm $\|P_S u_t\|$ blows up exponentially (by Lemma 2.7). The norm cannot grow exponentially because of the regularizer, so the function value must eventually decrease. This finishes the proof of Lemma 2.1.

2.5 Conclusion

In this chapter we show that for an over-parameterized tensor decomposition problem, a variant of gradient descent can learn a rank r tensor using $O^*(r^{2.5l}\log(d/\epsilon))$ components. The result shows that gradient-based methods are capable of leveraging low-rank structure in the input data to achieve lower level of over-parametrization. There are still many open problems, in particular extending our result to a mixture of tensors of different orders which would have implications for two-layer neural network with ReLU activations. We hope this serves as a first step towards understanding what structures can help gradient descent to learn efficient representations.

3

Orthogonal Tensor Decomposition

In the last chapter, we show that gradient descent on low-rank tensor decomposition problems (the ground truth components may be non-orthogonal) stay in the low-rank subspace spanned by the ground truth components. In this chapter, we restrict our attention to the orthogonally decomposable tensor and give a more precise characterization of the gradient descent (flow) training dynamics. In particular, we prove that a slightly modified version of gradient flow would follow a tensor deflation process and recover all the tensor components one by one, from the larger components to the smaller components. Our proof suggests that for orthogonal tensors, gradient flow dynamics works similarly as greedy low-rank learning in the matrix setting, which is a first step towards understanding the implicit regularization effect of over-parametrized models for low-rank tensors.

3.1 Introduction

We are given a tensor of the form

$$T^* = \sum_{i=1}^{r} a_i (U[:, i])^{\otimes 4},$$

where $a_i \geqslant 0$ and $U[:, i]$ is the i-th column of $U \in \mathbb{R}^{d \times r}$. The goal is to fit T^* using a tensor T of a similar form:

$$T = \sum_{i=1}^{m} \frac{(W[:, i])^{\otimes 4}}{\|W[:, i]\|^2}.$$

Here W is a $d \times m$ matrix whose columns are components for tensor T. The model is over-parametrized when the number of components m is larger than r. The choice of normalization factor of $1/\|W[:, i]\|^2$ is made to accelerate gradient flow (similar to Li et al. (2020a); Wang et al. (2020)).

Suppose we run gradient flow on the standard objective $\frac{1}{2}\|T - T^*\|_F^2$, that is, we evolve W according to the differential equation:

$$\frac{\mathrm{d}W}{\mathrm{d}t} = -\nabla \left(\frac{1}{2}\|T - T^*\|_F^2 \right),$$

can we expect T to fit T^* with good accuracy? Empirical results (see Figure 3.1) show that this is true for orthogonal tensor T^*[1] as long as m is large enough. Further, the training dynamics exhibits a behavior that is similar to a *tensor deflation process*: it finds the ground truth components one-by-one from larger component to smaller component (if multiple ground truth components have similar norm they might be found simultaneously).

In this chapter we show that with a slight modification, gradient flow on over-parametrized tensor decomposition is guaranteed to follow this tensor deflation process, and can fit any orthogonal tensor to desired accuracy[2](see Section 3.4 for the algorithm and Theorem 3.1 for the main theorem). This shows that for orthogonal tensors, the trajectory of modified gradient-flow is similar to a greedy low-rank process that was used to analyze the implicit bias of low-rank matrix factorization (Li et al., 2020b). We emphasize that our goal is not to propose another tensor decomposition algorithm. Instead, we hope our results can serve as a first step in

[1] We say T^* is an orthogonal tensor if the ground truth components $U[:, i]$'s are orthonormal.

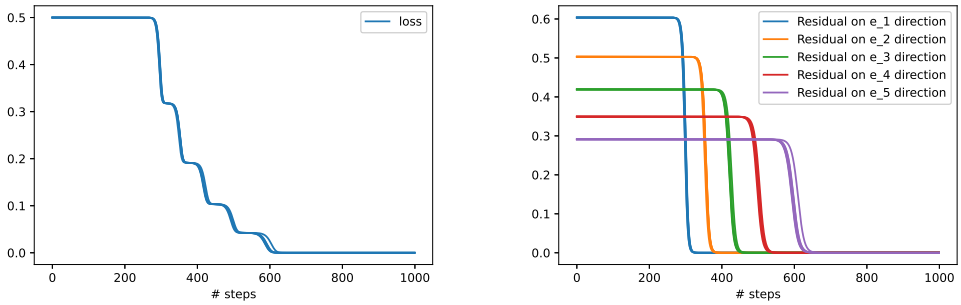

FIGURE 3.1: The training trajectory of gradient flow on orthogonal tensor decompositions. We chose $T^* = \sum_{i\in[5]} a_i e_i^{\otimes 4}$ with $e_i \in \mathbb{R}^{10}$ and $a_i/a_{i+1} = 1.2$. Our model T has 50 components and each component is randomly initialized with small norm 10^{-15}. We ran the experiments from 5 different initialization and plotted the results separately. The left figure shows the loss $\frac{1}{2}\|T - T^*\|_F^2$ and the right figure shows the residual on each e_i direction that is defined as $(T^* - T)(e_i^{\otimes 4})$.

understanding the implicit bias of over-parameterized gradient descent for low-rank tensor problems.

3.1.1 Our approach and technique

To understand the tensor deflation process shown in Figure 3.1, intuitively we can think about the discovery and fitting of a ground truth component in two phases. Consider the beginning of the gradient flow as an example. Initially all the components in T are small, which makes T negligible compared to T^*. In this case each component w in W will evolve according to a simpler dynamics that is similar to tensor power method, where one updates w to $T^*(w^{\otimes 3}, I)/\|T^*(w^{\otimes 3}, I)\|$ (see Section 3.3 for details).

For orthogonal tensors, it's known that tensor power method with random initializations would be able to discover the largest ground truth components (see Anandkumar et al. (2014)). Once the largest ground truth component has been discovered,

[2] Due to some technical challenges, we actually require the target accuracy to be at least $\exp(-o(d/\log d))$. This is only a very mild restriction since the dependence is exponential in d, and in practice, d is usually large and this lower bound can easily drop below the numerical precision.

the corresponding component (or multiple components) w will quickly grow in norm, which eventually fits the ground truth component. The flat regions in the trajectory in Figure 3.1 correspond to the period of time where the components w's are small and $T - T^*$ remains stable, while the decreasing regions correspond to the period of time where a ground truth component is being fitted.

However, there are many challenges in analyzing this process. The main problem is that the gradient flow would introduce a lot of dependencies throughout the trajectory, making it harder to analyze the fitting of later ground truth components, especially ones that are much smaller. We modify the algorithm to include a reinitialization step per epoch, which alleviates the dependency issue. Even after the modification we still need a few more techniques:

Local stability One major problem in analyzing the dynamics in a later stage is that the components used to fit the previous ground truth components are still moving according to their gradients, therefore it might be possible for these components to move away. To address this problem, we add a small regularizer to the objective, and give a new local stability analysis that bounds the distance to the fitted ground truth component both individually and on average. The idea of bounding the distance on average is important as just assuming each component w is close enough to the fitted ground truth component is not sufficient to prove that w cannot move far. While similar ideas were considered in Chizat (2021), the setting of tensor decomposition is different.

Norm/Correlation relation A key step in our analysis establishes a relationship between norm and correlation: we show if a component w crosses a certain norm threshold, then it must have a very large correlation with one of the ground truth components. This offers an initial condition for local stability and makes sure the

residual $T^* - T$ is almost close to an orthogonal tensor. Establishing this relation is difficult as unlike the high level intuition, we cannot guarantee $T^* - T$ remains unchanged even within a single epoch: it is possible that one ground truth component is already fitted while no large component is near another ground truth component of same size. In previous work, Li et al. (2020a) deals with a similar problem for neural networks using gradient truncation that prevents components from growing in the first phase (and as a result has super-exponential dependency on the ratio between largest and smallest a_i). We give a new technique to control the influence of ground truth components that are fitted within this epoch, so we do not need the gradient truncation and can characterize the deflation process.

3.1.2 Related works

Neural Tangent Kernel There is a recent line of work showing the connection between Neural Tangent Kernel (NTK) and sufficiently wide neural networks trained by gradient descent (Jacot et al., 2018; Allen-Zhu et al., 2018a; Du et al., 2018a, 2019; Li and Liang, 2018; Arora et al., 2019d,b; Zou et al., 2020; Oymak and Soltanolkotabi, 2020; Ghorbani et al., 2021). These papers show when the width of a neural network is large enough, it will stay around the initialization and its training dynamic is close to the dynamic of the kernel regression with NTK. In this work we go beyond the NTK setting and analyze the trajectory from a very small initialization.

Mean-field analysis There is another line of works that use mean-field approach to study the optimization for infinite-wide neural networks (Mei et al., 2018; Chizat and Bach, 2018b; Nguyen and Pham, 2020; Nitanda and Suzuki, 2017; Wei et al., 2019; Rotskoff and Vanden-Eijnden, 2018b; Sirignano and Spiliopoulos, 2020). Chizat et al. (2019) showed that, unlike NTK regime, the parameters can move away from its initialization in mean-field regime. However, most of the existing works need

width to be exponential in dimension and do not provide a polynomial convergence rate.

Beyond NTK There are many works showing the gap between neural networks and NTK (Allen-Zhu and Li, 2019; Allen-Zhu et al., 2018b; Yehudai and Shamir, 2019; Ghorbani et al., 2019, 2020; Dyer and Gur-Ari, 2019; Woodworth et al., 2020; Bai and Lee, 2019; Bai et al., 2020; Huang and Yau, 2020; Chen et al., 2020a). In particular, Li et al. (2020a) and Wang et al. (2020) are closely related with our setting. While Li et al. (2020a) focused on learning two-layer ReLU neural networks with orthogonal weights, they relied on the connection between tensor decomposition and neural networks (Ge et al., 2018b) and essentially worked with tensor decomposition problems. In their result, all the a_i's are within a constant factor and all components are learned simultaneously. We allow ground truth components with very different scale and show a deflation phenomenon. Wang et al. (2020) studied learning a low-rank non-orthogonal tensor, but they only showed the learned tensor T will eventually be close to the ground truth tensor T^* and does not guarantee the components of T will align with the components of T^*. On the other hand, we fully characterize the training trajectory and the components of the learned tensor.

Implicit regularization Many works recently showed that different optimization methods tend to converge to different optima and have different optimization trajectories in several settings (Saxe et al., 2014; Soudry et al., 2018; Nacson et al., 2019a; Ji and Telgarsky, 2018a,b, 2019c, 2020; Gunasekar et al., 2018a,b; Moroshko et al., 2020; Arora et al., 2019c; Lyu and Li, 2019; Chizat and Bach, 2020). In particular, Saxe et al. (2014) related the dynamics of gradient descent to the magnitude of the singular values of the target weight matrices for linear networks with orthogonal inputs. The phenomenon there is qualitatively similar to our results, but the settings and

the proof techniques are very different. The more related and recent works are Li et al. (2020b) and Razin et al. (2021). Li et al. (2020b) studied matrix factorization problem and showed gradient descent with infinitesimal initialization is similar to greedy low-rank learning, which is a multi-epoch algorithm that finds the best approximation within the rank constraint and relax the constraint after every epoch. Razin et al. (2021) studied the tensor factorization problem and showed that it biases towards low rank tensor. Both of these works considered partially observable matrix or tensor and are only able to fully analyze the first epoch (i.e., recover the largest direction). We focus on a simpler setting with fully-observable ground truth tensor and give a complete analysis of learning all the ground truth components.

3.1.3 Outline

In Section 3.2 we introduce the basic notations and problem setup. In Section 3.3 we review tensor deflation process and tensor power method. We then give our algorithm in Section 3.4. Section 3.5 gives the formal main theorem and discusses high-level proof ideas. We conclude in Section 3.6 and discuss some limitations of the work. The detailed proofs and additional experiments are left in the appendix.

3.2 Preliminaries

Notations Most notations have been defined in Section 1.4. Below we define some specific notations for this chapter.

We use v_k to denote the k-th entry of vector v, and use v_{-k} to denote vector v with its k-th entry removed. We use $\bar{v}$ to denote the normalized vector $\bar{v} = v/\|v\|$, and use $\bar{v}_k$ to denote the k-th entry of $\bar{v}$. For a matrix A, we use $A[:, i]$ to denote its i-th column and $\mathrm{col}(A)$ to denote the set of all column vectors of A.

For simplicity we restrict our attention to symmetric 4-th order tensors. Suppose $T = \sum_w w^{\otimes 4}$, we define $T(v^{\otimes 4})$ as $\sum_w \langle w, v\rangle^4$, $T(v^{\otimes 3}, I)$ as $\sum_w \langle w, v\rangle^3 w$, and

$T(v^{\otimes 2}, u, I) = \sum_w \langle w, v\rangle^2 \langle w, u\rangle w$.

For clarity, we always call a component in T^* as ground truth component and call a component in our model T simply as component.

Problem setup We consider the problem of fitting a 4-th order tensor. The components of the ground truth tensor is arranged as columns of a matrix $U \in \mathbb{R}^{d\times r}$, and the tensor T^* is defined as

$$T^* = \sum_{i=1}^{r} a_i (U[:, i]^{\otimes 4}),$$

where $a_1 \geqslant a_2 \geqslant \cdots \geqslant a_r \geqslant 0$ and $\sum_{i=1}^r a_i = 1$. For convenience in the analysis, we assume $a_i \geqslant \epsilon/\sqrt{d}$ for all $i \in [r]$. This is without loss of generality because the target accuracy is ϵ and we can safely ignore very small ground truth components with $a_i < \epsilon/\sqrt{d}$. In this chapter, we focus on the case where the components are orthogonal—that is, the columns $U[:, i]$'s are orthonormal. For simplicity we assume without loss of generality that $U[:, i] = e_i$ where e_i is the i-th standard basis vector[3]. To reduce the number of parameters we also assume $r = d$, again this is without loss of generality because we can simply set $a_i = 0$ for $i > r$.

There can be many different ways to parametrize the tensor that we use to fit T^*. Following previous works (Wang et al., 2020; Li et al., 2020a), we use an over-parameterized and two-homogeneous tensor

$$T = \sum_{i=1}^{m} \frac{W[:, i]^{\otimes 4}}{\|W[:, i]\|^2}.$$

Here $W \in \mathbb{R}^{d\times m}$ is a matrix with m columns that corresponds to the components in T. It is overparametrized when $m > r$.

[3] This is without loss of generality because gradient flow (and our modifications) is invariant under rotation of the ground truth parameters.

Since the tensor T only depends on the set of columns $W[:, i]$ instead of the orderings of the columns, for the most part of the chapter we will instead write the tensor T as

$$T = \sum_{w \in \mathrm{col}(W)} \frac{w^{\otimes 4}}{\|w\|^2},$$

where $\mathrm{col}(W)$ is the set of all the column vectors in W. This allows us to discuss the dynamics of coordinates for a component w without using the index for the component. In particular, w_i always represents the i-th coordinate of the vector w. This representation is similar to the mean-field setup (Chizat and Bach, 2018b; Mei et al., 2018) where one considers a distribution on w, however since we do not rely on analysis related to infinite-width limit we use the sum formulation instead. For the ease of presentation, we choose to restrict our setting to fourth-order tensor decomposition, but our results can be easily generalized to tensor with order at least three.

3.3 Tensor deflation process and tensor power method

In this section we will first discuss the basic tensor deflation process for orthogonal tensor decomposition. Then we show the connection between the tensor power method and gradient flow.

Tensor deflation For orthogonal tensor decomposition, a popular approach is to first fit the largest ground truth component in the tensor, then subtract it out and recurse on the residual. The general process is given in Algorithm 2. In this process, there are multiple ways to find the best rank-1 approximation. For example, Anandkumar et al. (2014) uses tensor power method, which picks many random vectors w, and update them as $w = T^*(w^{\otimes 3}, I)/\left\|T^*(w^{\otimes 3}, I)\right\|$.

Algorithm 2 Tensor Deflation Process

Input: Tensor T^*
Output: Components W such that $T^* \approx \sum_{w \in \mathrm{col}(W)} w^{\otimes 4}/\|w\|^2$
Initially let the residual R be T^*.
while $\|R\|_F$ is large **do**
    Find the best rank 1 approximation $w^{\otimes 4}/\|w\|^2$ for R.
    Add w as a new column in W, and let $R = R - w^{\otimes 4}/\|w\|^2$.

Tensor power method and gradient flow If we run tensor power method using a tensor T^* that is equal to $\sum_{i=1}^d a_i e_i^{\otimes 4}$, then a component w will converge to the direction of e_i where i is equal to $\arg\max_i a_i \bar{w}_i^2$. If there is a tie (which happens with probability 0 for random w), then the point will be stuck at a saddle point.

Let's consider running gradient flow on W with objective function $\frac{1}{2}\|T - T^*\|_F^2$ as $T := \sum_{w \in \mathrm{col}(W)} w^{\otimes 4}/\|w\|^2$. If T does not change much, the residual $R := T^* - T$ is close to a constant. In this case the trajectory of one component w is determined by the following differential equation:

$$\frac{\mathrm{d}w}{\mathrm{d}t} = 4R(\bar{w}^{\otimes 2}, w, I) - 2R(\bar{w}^{\otimes 4})w. \tag{3.1}$$

To understand how this process works, we can take a look at $\frac{\mathrm{d}w_i^2/\mathrm{d}t}{w_i^2}$ (intuitively this corresponds to the growth rate for w_i^2). If $R \approx T^*$ then we have:

$$\frac{\mathrm{d}w_i^2/\mathrm{d}t}{w_i^2} \approx 8a_i\bar{w}_i^2 - 4\sum_{j \in [d]} a_j \bar{w}_j^4.$$

From this formula it is clear that the coordinate with larger $a_i\bar{w}_i^2$ has a faster growth rate, so eventually the process will converge to e_i where i is equal to $\arg\max_i a_i\bar{w}_i^2$, same as the tensor power method. Because of their similarity later we refer to dynamics in Eqn. (3.1) as tensor power dynamics.

3.4 Our algorithm

Our algorithm is a modified version of gradient flow as described in Algorithm 3. First, we change the loss function to

$$L(W) = \frac{1}{2} \|T - T^*\|_F^2 + \frac{\lambda}{2} \|W\|_F^2 .$$

The additional small regularization $\frac{\lambda}{2} \|W\|_F^2$ allows us to prove a *local stability* result that shows if there are components w that are close to the ground truth components in direction, then they will not move too much (see Section 3.5.1).

Our algorithm runs in multiple epochs with increasing length. We use $W^{(s,t)}$ to denote the weight matrix in epoch s at time t. We use similar notation for tensor $T^{(s,t)}$. In each epoch we try to fit ground truth components with $a_i \geqslant \beta^{(s)}$. In general, the time it takes to fit one ground truth direction is inversely proportional to its magnitude a_i. The earlier epochs have shorter length so only large directions can be fitted, and later epochs are longer to fit small directions.

At the middle of each epoch, we reinitialize all components that do not have a large norm. This serves several purposes: first we will show that all components that exceed the norm threshold will have good correlation with one of the ground truth components, therefore giving an initial condition to the local stability result; second, the reinitialization will reduce the dependencies between different epochs and allow us to analyze each epoch almost independently. These modifications do not change the dynamics significantly, however they allow us to do a rigorous analysis.

3.5 Main theorem and proof sketch

In this section we discuss the ideas to prove the following main theorem[4]

[4] In the theorem statement, we have a parameter α that is not used in our algorithm but is very useful in the analysis (see for example Definition 3.1). Basically, α measures the closeness between a component and its corresponding ground truth direction (see more in Section 3.5.1).

Algorithm 3 Modified Gradient Flow

Input: Number of components m, initialization scale δ_0, re-initialization threshold δ_1, increasing rate of epoch length γ, target accuracy ϵ, regularization coefficient λ

Output: Tensor T satisfying $\|T - T^*\|_F \leqslant \epsilon$.

Initialize $W^{(0,0)}$ as a $d \times m$ matrix with each column $w^{(0,0)}$ i.i.d. sampled from $\delta_0 \mathrm{Unif}(\mathbb{S}^{d-1})$.

$\beta^{(0)} \leftarrow \|T^{(0,0)} - T^*\|_F$; $s \leftarrow 0$

while $\|T^{(s,0)} - T^*\|_F > \epsilon$ **do**

Phase 1: Starting from $W^{(s,0)}$, run gradient flow for time $t_1^{(s)} = O(\frac{d}{\beta^{(s)} \log(d)})$.

Reinitialize all components that have ℓ_2 norm less than δ_1 by sampling i.i.d. from $\delta_0 \mathrm{Unif}(\mathbb{S}^{d-1})$.

Phase 2: Starting from $W^{(s,t_1^{(s)})}$, run gradient flow for $t_2^{(s)} - t_1^{(s)} = O(\frac{\log(1/\delta_1)+\log(1/\lambda)}{\beta^{(s)}})$ time

$W^{(s+1,0)} \leftarrow W^{(s,t_2^{(s)})}$; $\beta^{(s+1)} \leftarrow \beta^{(s)}(1-\gamma)$; $s \leftarrow s+1$

Theorem 3.1. *For any $\epsilon \geqslant \exp(-o(d/\log d))$, there exists $\gamma = \Theta(1)$, $m = \mathrm{poly}(d)$, $\lambda = \min\{O(\log d/d), O(\epsilon/d^{1/2})\}$, $\alpha = \min\{O(\lambda/d^{3/2}), O(\lambda^2), O(\epsilon^2/d^4)\}$, $\delta_1 = O(\alpha^{3/2}/m^{1/2})$, $\delta_0 = \Theta(\delta_1 \alpha/\log^{1/2}(d))$ such that with probability $1 - 1/\mathrm{poly}(d)$ in the (re)-initializations, Algorithm 3 terminates in $O(\log(d/\epsilon))$ epochs and returns a tensor T such that*

$$\|T - T^*\|_F \leqslant \epsilon.$$

Intuitively, epoch s of Algorithm 3 will try to discover all ground truth components with a_i that is at least as large as $\beta^{(s)}$. The algorithm does this in two phases. In Phase 1, the small components w will evolve according to tensor power dynamics. For each ground truth component with large enough a_i that has not been fitted yet, we hope there will be at least one component in W that becomes large and correlated with e_i. We call such ground truth components "discovered". Phase 1 ends with a check that reinitilizes all components with small norm. Phase 2 is relatively short, and in Phase 2 we guarantee that every ground truth component that has been discovered become "fitted", which means the residual $T - T^*$ becomes small in

this direction.

However, there are still many difficulties in analyzing each of the steps. In particular, why would ground truth components that are fitted in previous epochs remain fitted? How to guarantee only components that are correlated with a ground truth component grow to a large norm? Why wouldn't the gradient flow in Phase 2 mess up with the initialization we require in Phase 1? We discuss the high level ideas to solve these issues. In particular, in Section 3.5.1 we first give an induction hypothesis that is preserved throughout the algorithm, which guarantees that every ground truth component that is fitted remains fitted. In Section 3.5.2 we discuss the properties in Phase 1, and in Section 3.5.3 we discuss the properties in Phase 2.

3.5.1 Induction hypothesis and local stability

In order to formally define what it means for a ground truth component to be "discovered" or "fitted", we need some more definitions and notations.

Definition 3.1. *Define* $S_i^{(s,t)} \subseteq [m]$ *as the subset of components that satisfy the following conditions: the* k*-th component is in* $S_i^{(s,t)}$ *if and only if there exists some time* (s', t') *that is no later than* (s, t) *and no earlier than the latest re-initialization of* $W[:, k]$ *such that*

$$\left\|W^{(s',t')}[:,k]\right\| = \delta_1 \text{ and } [\overline{W^{(s',t')}[:,k]}_i]^2 \geqslant 1 - \alpha^2.$$

We say that ground truth component i *is* discovered *in epoch* s *at time* t*, if* $S_i^{(s,t)}$ *is not empty.*

Intuitively, $S_i^{(s,t)}$ is a subset of components in W such that they have large enough norm and good correlation with the i-th ground truth component. Although such components may not have a large enough norm to fit a_i yet, their norm will eventually grow. Therefore we say ground truth component i is discovered when such components exist.

For convenience, we shorthand $w^{(s,t)} \in \{W^{(s,t)}[:,j] | j \in S_i^{(s,t)}\}$ by $w^{(s,t)} \in S_i^{(s,t)}$. Now we will discuss when a ground truth component is fitted, for that, let

$$\hat{a}_i^{(s,t)} = \sum_{w^{(s,t)} \in S_i^{(s,t)}} \left\| w^{(s,t)} \right\|^2.$$

Here $\hat{a}_i^{(s,t)}$ is the total squared norm for all the components in $S_i^{(s,t)}$. We say a ground truth component is *fitted* if $a_i - \hat{a}_i^{(s,t)} \leqslant 2\lambda$.

Note that one can partition the columns in W using sets $S_i^{(s,t)}$, giving d groups and one extra group that contains everything else. We define the extra group as $S_\varnothing^{(s,t)} := [m] \backslash \bigcup_{k \in [d]} S_k^{(s,t)}$.

For each of the non-empty $S_i^{(s,t)}$, we can take the average of its component (weighted by $\left\| w^{(s,t)} \right\|^2$):

$$\mathbb{E}_{i,w}^{(s,t)} f(w^{(s,t)}) := \frac{1}{\hat{a}_i^{(s,t)}} \sum_{w^{(s,t)} \in S_i^{(s,t)}} \left\| w^{(s,t)} \right\|^2 f(w^{(s,t)}).$$

If $S_i^{(s,t)} = \varnothing$, we define $\mathbb{E}_{i,w}^{(s,t)} f(w^{(s,t)})$ as zero. Now we are ready to state the induction hypothesis:

Proposition 3.1 (Induction hypothesis)**.** *In the setting of Theorem 3.1, for any epoch s and time t and every $k \in [d]$, the following hold.*

(a) For any $w^{(s,t)} \in S_k^{(s,t)}$, we have $\left[\bar{w}_k^{(s,t)} \right]^2 \geqslant 1 - \alpha$.

(b) If $S_k^{(s,t)}$ is nonempty, $\mathbb{E}_{k,w}^{(s,t)} \left[\bar{w}_k^{(s,t)} \right]^2 \geqslant 1 - \alpha^2 - 4sm\delta_1^2$.

(c) We always have $a_k - \hat{a}_k^{(s,t)} \geqslant \lambda/6 - sm\delta_1^2$; if $a_k \geqslant \frac{\beta^{(s)}}{1-\gamma}$, we further know $a_k - \hat{a}_k^{(s,t)} \leqslant \lambda + sm\delta_1^2$.

(d) If $w^{(s,t)} \in S_\varnothing^{(s,t)}$, then $\|w^{(s,t)}\| \leqslant \delta_1$.

We choose δ_1^2 small enough so that $sm\delta_1^2$ is negligible compared with α^2 and λ. Note that if Proposition 3.1 is maintained throughout the algorithm, all the large components will be fitted, which directly implies Theorem 3.1. Detailed proof is deferred to Appendix B.4.

Condition (c) shows that for a ground truth component k with large enough a_k, it will always be fitted after the corresponding epoch (recall from Theorem 3.1 that $\lambda = O(\varepsilon/\sqrt{d})$). Condition (d) shows that components that did not discover any ground truth components will always have small norm (hence negligible in most parts of the analysis). Conditions (a)(b) show that as long as a ground truth component k has been discovered, all components that are in $S_k^{(s,t)}$ will have good correlation, while the *average* of all such components will have even better correlation. The separation between individual correlation and average correlation is important in the proof. With only individual bound, we cannot maintain the correlation no matter how small α is. Here is an example below:

Claim 3.1. *Suppose* $T^* = e_k^{\otimes 4}$ *and* $T = v^{\otimes 4}/\|v\|^2 + w^{\otimes 4}/\|w\|^2$ *with* $\|w\|^2 + \|v\|^2 \in [2/3, 1]$. *Suppose* $\bar{v}_k^2 = 1 - \alpha$ *and* $\bar{v}_k = \bar{w}_k, \bar{v}_{-k} = -\bar{w}_{-k}$. *Assuming* $\|v\|^2 \leqslant c_1$ *and* $\alpha \leqslant c_2$ *for small enough constants* c_1, c_2, *we have* $\frac{\mathrm{d}}{\mathrm{d}t}\bar{v}_k^2 < 0$.

In the above example, both $\bar{v}$ and $\bar{w}$ are close to e_k but they are opposite in other directions ($\bar{v}_{-k} = \bar{w}_{-k}$). The norm of v is very small compared with that of w. Intuitively, we can increase v_{-k} so that the average of v and w is more aligned with e_k. See the rigorous analysis in Appendix B.1.6.

The induction hypothesis will be carefully maintained throughout the analysis. The following lemma guarantees that in the gradient flow steps the individual and average correlation will be maintained.

Lemma 3.1. *In the setting of Theorem 3.1, suppose Proposition 3.1 holds in epoch*

s at time t, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}[\bar{w}^{(s,t)}]^2 \geqslant 8\left(a_k - \hat{a}_k^{(s,t)}\right)\left(1 - [\bar{w}_k^{(s,t)}]^2\right) - O\left(\alpha^{1.5}\right),$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,w}^{(s,t)}[\bar{w}_k^{(s,t)}]^2 \geqslant 8\left(a_k - \hat{a}_k^{(s,t)}\right)\left(1 - \mathbb{E}_{k,w}^{(s,t)}[\bar{w}_k^{(s,t)}]^2\right) - O(\alpha^3).$$

In particular, when $a_k - \hat{a}_k^{(s,t)} \geqslant \Omega(\lambda) = \Omega(\sqrt{\alpha})$, *we have* $\frac{\mathrm{d}}{\mathrm{d}t}[\bar{w}_k^{(s,t)}]^2 > 0$ *when* $[\bar{w}_k^{(s,t)}]^2 = 1 - \alpha$ *and* $\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,w}^{(s,t)}[\bar{w}_k^{(s,t)}]^2 > 0$ *when* $\mathbb{E}_{k,w}^{(s,t)}[\bar{w}_k^{(s,t)}]^2 = 1 - \alpha^2$.

The detailed proof for the local stability result can be found in Appendix B.1. Of course, to fully prove the induction hypothesis one needs to talk about what happens when a component enters $S_i^{(s,t)}$, and what happens at the reinitialization steps. We discuss these details in later subsections.

3.5.2 Analysis of Phase 1

In Phase 1 our main goal is to discover all the components that are large enough. We also need to maintain Proposition 3.1. Formally we prove the following:

Lemma 3.2 (Main Lemma for Phase 1)**.** *In the setting of Theorem 3.1, suppose Proposition 3.1 holds at* $(s, 0)$. *For* $t_1^{(s)} := t_1^{(s)\prime} + t_1^{(s)\prime\prime} + t_1^{(s)\prime\prime\prime}$ *with* $t_1^{(s)\prime} = \Theta(d/(\beta^{(s)} \log d))$, $t_1^{(s)\prime\prime} = \Theta(d/(\beta^{(s)} \log^3 d))$, $t_1^{(s)\prime\prime\prime} = \Theta(\log(d/\alpha)/\beta^{(s)})$, *with probability* $1 - 1/\mathit{poly}(d)$ *we have*

1. *Proposition 3.1 holds at* (s, t) *for any* $0 \leqslant t < t_1^{(s)}$, *and also for* $t = t_1^{(s)}$ *after reinitialization.*

2. *If* $a_k \geqslant \beta^{(s)}$ *and* $S_k^{(s,0)} = \varnothing$, *we have* $S_k^{(s,t_1^{(s)})} \neq \varnothing$ *and* $\hat{a}_k^{(s,t_1^{(s)})} \geqslant \delta_1^2$.

3. *If* $S_k^{(s,0)} = \varnothing$ *and* $S_k^{(s,t_1^{(s)})} \neq \varnothing$, *we have* $a_k \geqslant C\beta^{(s)}$ *for universal constant* $0 < C < 1$.

Property 2 shows that large enough ground truth components are always discovered, while Property 3 guarantees that no small ground truth components can be discovered. Our proof relies on initial components being "lucky" and having higher than usual correlation with one of the large ground truth components. To make this clear we separate components into different sets (here we use v to denote a component in W):

Definition 3.2 (Partition of (re-)initialized components)**.** *For each direction $i \in [d]$, define the set of good components $S^{(s)}_{i,good}$ and the set of potential components $S^{(s)}_{i,pot}$ as follow, where $\Gamma^{(s)}_i := 1/(8a_i t_1^{(s)\prime})$ if $S^{(s,0)}_i = \varnothing$, and $\Gamma^{(s)}_i := 1/(8\lambda t_1^{(s)\prime})$ otherwise. Here $\rho^{(s)}_i := c_\rho \Gamma^{(s)}_i$ and c_ρ is a small enough absolute constant.*

$$S^{(s)}_{i,good} :=$$

$$\{k \mid [\bar{v}_i^{(s,0)}]^2 \geqslant \Gamma^{(s)}_i + \rho^{(s)}_i,\ [\bar{v}_j^{(s,0)}]^2 \leqslant \Gamma^{(s)}_j - \rho^{(s)}_j, \forall j \neq i \text{ and } v^{(s,0)} = W^{(s,0)}[:,k]\},$$

$$S^{(s)}_{i,pot} := \{k \mid [\bar{v}_i^{(s,0)}]^2 \geqslant \Gamma^{(s)}_i - \rho^{(s)}_i \text{ and } v^{(s,0)} = W^{(s,0)}[:,k]\}.$$

Let $S^{(s)}_{good} := \cup_i S^{(s)}_{i,good}$ and $S^{(s)}_{pot} := \cup_i S^{(s)}_{i,pot}$. We also define the set of bad components $S^{(s)}_{bad}$ as

$$\{k \mid \exists i \neq j \text{ s.t. } [\bar{v}_i^{(s,0)}]^2 \geqslant \Gamma^{(s)}_i - \rho^{(s)}_i,\ [\bar{v}_j^{(s,0)}]^2 \geqslant \Gamma^{(s)}_j - \rho^{(s)}_j \text{ and } v^{(s,0)} = W^{(s,0)}[:,k]\}.$$

For convenience, we shorthand $v^{(s,t)} \in \{W^{(s,t)}[:,j] | j \in S_{i,good}\}$ by $v^{(s,t)} \in S_{i,good}$ (same for $S_{i,pot}$ and S_{bad}). Intuitively, the good components will grow very quickly and eventually pass the norm threshold. Since both good and potential components only have one large coordinate, they will become correlated with that ground truth component when their norm is large. The bad components are correlated with two ground truth components so they can potentially have a large norm while not having a very good correlation with either one of them. In the proof we will guarantee with probability at least $1 - 1/\text{poly}(d)$ that good components exists for all large enough

ground truth components and there are no bad components. The following lemma characterizes the trajectories of different type of components:

Lemma 3.3. *In the setting of Lemma 3.2, for every $i \in [d]$*

1. *(Only good/potential components can become large) If $v^{(s,t)} \notin S_{pot}^{(s)}$, $\left\|v^{(s,t)}\right\| = O(\delta_0)$ and $[\bar{v}_i^{(s,t)}]^2 = O(\log(d)/d)$ for all $i \in [d]$ and $t \leqslant t_1^{(s)}$.*

2. *(Good components discover ground truth components) If $S_{i,good}^{(s)} \neq \varnothing$, there exists $v^{(s,t_1^{(s)})}$ such that $\left\|v^{(s,t_1^{(s)})}\right\| \geqslant \delta_1$ and $S_i^{(s,t_1^{(s)})} \neq \varnothing$.*

3. *(Large components are correlated with ground truth components) If $\left\|v^{(s,t)}\right\| \geqslant \delta_1$ for some $t \leqslant t_1^{(s)}$, there exists $i \in [d]$ such that $v^{(s,t)} \in S_i^{(s,t)}$.*

The proof of Lemma 3.3 is difficult as one cannot guarantee that all the ground truth components that we are hoping to fit in the epoch will be fitted simultaneously. However we are able to show that $T - T^*$ remains near-orthogonal and control the effect of changing $T - T^*$ within this epoch. The details are in Appendix B.2.

3.5.3 Analysis of Phase 2

In Phase 2 we will show that every ground truth component that's discovered in Phase 1 will become fitted, and the reinitialized components will preserve the desired initialization conditions.

Lemma 3.4 (Main Lemma for Phase 2)**.** *In the setting of Theorem 3.1, suppose Proposition 3.1 holds at $(s, t_1^{(s)})$, we have for $t_2^{(s)} - t_1^{(s)} := O(\frac{\log(1/\delta_1)+\log(1/\lambda)}{\beta^{(s)}})$*

1. *Proposition 3.1 holds at (s,t) for any $t_1^{(s)} \leqslant t \leqslant t_2^{(s)}$.*

2. *If $S_k^{(s,t_1^{(s)})} \neq \varnothing$, we have $a_k - \hat{a}_k^{(s,t_2^{(s)})} \leqslant 2\lambda$.*

3. For any component v that was reinitialized at $t_1^{(s)}$, we have $\left\|v^{(s,t_2^{(s)})}\right\|^2 = \Theta(\delta_0^2)$ and $\left[\bar{v}_i^{(s,t_2^{(s)})}\right]^2 = \left[\bar{v}_i^{(s,t_1^{(s)})}\right]^2 \pm o\left(\frac{\log d}{d}\right)$ for every $i \in [d]$.

The main idea is that as long as a direction has been discovered, the norm of the corresponding components will increase very fast. The rate of that is characterized by the following lemma.

Lemma 3.5 (informal)**.** *In the setting of Theorem 3.4, for any $t_1^{(s)} \leqslant t \leqslant t_2^{(s)}$,*

$$\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(s,t)} \geqslant \left(2(a_k - \hat{a}_k^{(s,t)}) - \lambda - O\left(\alpha^2\right)\right)\hat{a}_k^{(s,t)}.$$

In particular, after $O(\frac{\log(1/\delta_1)+\log(1/\lambda)}{a_k})$ time, we have $a_k - \hat{a}_k^{(s,t)} \leqslant \lambda$.

By the choice of δ_1 and λ, the length of Phase 2 is much smaller than the amount of time needed for the reinitialized components to move far, allowing us to prove the third property in Lemma 3.4. Detailed analysis is deferred to Appendix B.3.

3.6 Conclusion

In this chapter we analyzed the dynamics of gradient flow for over-parametrized orthogonal tensor decomposition. With very mild modification to the algorithm (a small regularizer and some re-initializations), we showed that the trajectory is similar to a tensor deflation process and the greedy low-rank procedure in Li et al. (2020b). These modifications allowed us to prove strong guarantees for orthogonal tensors of any rank, while not changing the empirical behavior of the algorithm. We believe such techniques would be useful in later analysis for the implicit bias of tensor problems.

A major limitation of our work is that it only applies to orthogonal tensors. Going beyond this would require significantly new ideas—we observed that for general tensors, overparametrized gradient flow may have a very different behavior compared to

the greedy low-rank procedure, as it is possible for two large component in the same direction to split into two different directions (see more details in Appendix B.5). We leave that as an interesting open problem.

4

Non-contrastive Self-supervised Learning

In this chapter, we explain the representation learning process in non-contrastive self-supervised learning by analyzing the gradient descent dynamics in a simple model. We prove in a linear network, non-contrastive methods learn a desirable projection matrix and also reduce the sample complexity on downstream tasks. Our analysis suggests that weight decay acts as an implicit threshold that discards the features with high variance under data augmentations, and keeps the features with low variance. Inspired by our theory, we design a simpler and more computationally efficient algorithm DirectCopy by removing the eigen-decomposition step in the original DirectPred algorithm in Tian et al. (2021). Our experiments show that DirectCopy rivals or even outperforms DirectPred on STL-10, CIFAR-10, CIFAR-100 and ImageNet.

4.1 Introduction

Recently, *non-contrastive* self-supervised learning (abbreviated as ***nc-SSL***) was proposed to learn representations by minimizing the distances between representations of two augmented views of the same data point (positive pairs). Presumably, nc-SSL

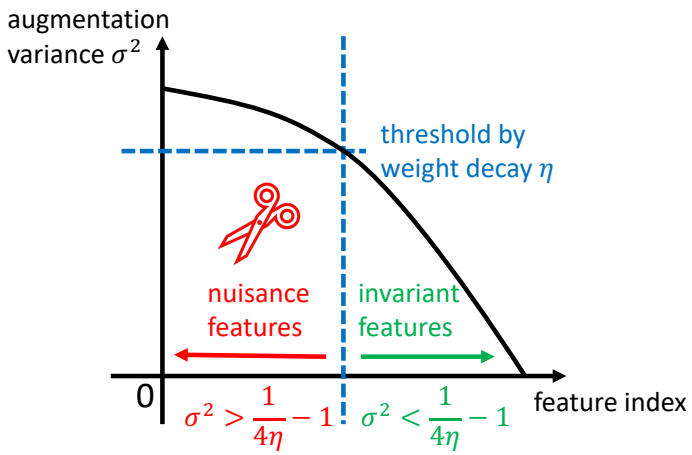

FIGURE 4.1: In non-contrastive self-supervised learning (abbr. ***nc-SSL***), the weight decay acts as an implicit threshold on the variance of features under data augmentations. With weight decay, the training process discards nuisance features that have high variance under data augmentations and keeps invariant features that have low variance.

might converge to the trivial constant representation that is a global minimizer of the loss function. However, in practice, nc-SSL is able to learn nontrivial representation and shows remarkable performance on downstream tasks (e.g., image classification (Grill et al., 2020; Chen and He, 2020)). This brings about two fundamental questions: (1) without negative pairs, why the learned representation does not collapse to trivial (i.e., constant) solutions, and (2) what representation nc-SSL learns from the training and how the learned representation reduces the sample complexity in downstream tasks.

While many theoretical results on contrastive SSL (Arora et al., 2019a; Lee et al., 2020; Tosh et al., 2020; Wen and Li, 2021) exist, similar study on nc-SSL has been very rare. As one of the first work towards this direction, Tian et al. (2021) showed that while the global optimum of the non-contrastive loss is indeed a trivial one, following gradient direction in nc-SSL, one can find a *local* optimum that admits a nontrivial representation. Based on their theoretical findings on gradient-based methods, they proposed a new approach, DirectPred, that directly sets the predictor

using the eigen-decomposition of the correlation matrix of inputs before the predictor, rather than updating it with gradient methods. As a method for nc-SSL, DirectPred shows comparable or better performance in multiple datasets, including CIFAR-10 (Krizhevsky and Hinton, 2009), STL-10 (Coates et al., 2011) and ImageNet (Deng et al., 2009), compared to BYOL (Grill et al., 2020) and SimSiam (Chen and He, 2020) that optimize the predictor using gradient descent.

While Tian et al. (2021) addressed the first question, i.e., why the learned representation does not collapse to zero, they did not address the second question, i.e., how the training dynamics in nc-SSL leads to a meaningful representation that depends on the data augmentations and reduces the sample complexity on down-stream tasks.

Main Contributions. In this chapter, we make a first attempt towards the second question, by studying a family of algorithms named **DirectSet**(α), in which the DirectPred algorithm proposed by Tian et al. (2021) is a special case with $\alpha = 1/2$. Our contribution is two-fold:

First, we perform a theoretical analysis on DirectSet(α) with linear networks. We prove that DirectSet(α) learns a desirable projection matrix onto the invariant features given polynomial number of unlabeled samples. Our analysis shows that there exists an *implicit threshold*, determined by weight decay parameter η, that governs which features are learned and which are discarded. As illustrated in Figure 4.1, the threshold is applied to the variance of the feature across different data augmentations (or "views") of the same instance: *nuisance features* (features with high variances under augmentation) are discarded, while *invariant features* (i.e., with low variances) are kept. We further prove the learned representation can reduce the sample complexity on downstream tasks. To the best of knowledge, this is the first result proving nc-SSL learns meaningful representations that reduce the sample complexity

on downstream tasks.

Second, we show that **DirectCopy**, a special case of DirectSet(α) when $\alpha = 1$, performs comparably with (or even outperforms) DirectPred on downstream tasks in CIFAR-10, CIFAR-100, STL-10 and ImageNet. In DirectCopy, the predictor can be set *without* the expensive eigen-decomposition operation, which makes DirectCopy much simpler and more efficient than DirectPred.

Organization. In Section 4.1.1, we discuss the related works. We introduce DirectSet(α) and DirectCopy in Section 4.2 and analyze them in a linear network setting in Section 4.3. Section 4.4 demonstrates the empirical performance of DirectCopy across various datasets and Section 4.5 shows ablation experiments. Finally, we conclude the chapter in Section 4.6.

4.1.1 Related Works

Contrastive methods: Contrastive learning (Oord et al., 2018; Tian et al., 2019; Bachman et al., 2019; He et al., 2020; Chen et al., 2020b) learns representations by minimizing the distances of positive pairs and maximizing distances of negative pairs. There are many theoretical works trying to explain contrastive learning (Arora et al., 2019a; Wang and Isola, 2020; Tian et al., 2020b; Tsai et al., 2020; Tosh et al., 2021). HaoChen et al. (2021) proposed a contrastive loss that implicitly performs spectral decomposition on the augmentation graph. Tian et al. (2020a) showed that gradient updates tend to amplify the features invariant to augmentations. Wen and Li (2021) proved that data augmentations decouple the correlations between spurious dense features and force the network to learn desired sparse features.

Non-contrastive methods: Without negative samples, non-contrastive methods use other techniques to avoid representational collapse. BYOL (Grill et al., 2020) and SimSiam (Chen and He, 2020) use an extra predictor and a stop-gradient operation.

SwAV (Caron et al., 2020) clusters the data while ensuring different views of the same data falls in the same cluster. Zbontar et al. (2021); Bardes et al. (2021); Hua et al. (2021) de-correlate the variables in the features. Ermolov et al. (2021) proposed a new loss function based on the whitening of the latent space features. DINO (Caron et al., 2021) applies centering and sharpening on the target network outputs. In this work, we study BYOL and SimSiam as representative nc-SSL methods.

Comparison with Tian et al. (2021) Tian et al. (2021) only explained why the representation in nc-SSL does not collapse to zero, but did not study what representation is learned and how the representation is related to the data distribution and augmentation process. In particular, they assumed the augmentation is isotropic in all dimensions and did not define the invariant features and nuisance features. In our model, we relax the isotropic assumption and allow the augmentation to act only in the nuisance subspace. Our analysis for the first time explains the representation learning mechanism in nc-SSL: weight decay discards the nuisance features and keeps the invariant features. Motivated by the analysis, we also design a simpler and more efficient algorithm (DirectCopy), which achieves comparable or even better performances than the original DirectPred proposed by Tian et al. (2021).

4.1.2 Notations

Most notations have been defined in Section 1.4. Below we define some specific notations for this chapter.

For any linear subspace S in $\mathbb{R}^d$, we use $P_S \in \mathbb{R}^{d\times d}$ to denote the projection matrix on S. More precisely, the projection matrix P_S equals $UU^\top$, where the columns of U constitute a set of orthonormal bases for subspace S.

For a real symmetric matrix $A \in \mathbb{R}^{d\times d}$ whose eigen-decomposition is $\sum_{i=1}^d \lambda_i u_i u_i^\top$, we use $|A|$ to denote $\sum_{i=1}^d |\lambda_i|\, u_i u_i^\top$. If A is also positive semi-definite, we use A^α to

denote $\sum_{i=1}^{d} \lambda_i^{\alpha} u_i u_i^{\top}$ for any positive $\alpha \in \mathbb{R}$.

4.2 Preliminaries

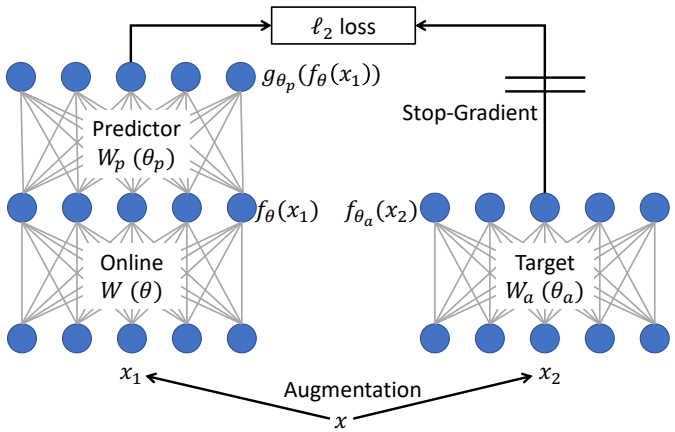

FIGURE 4.2: Problem Setup of a linear network. Both BYOL and DirectSet(α) update online W by gradient methods and set target W_a as EMA of W. BYOL also updates predictor W_p by gradient methods, while DirectSet(α) sets W_p based on the correlation matrix of predictor inputs ($\mathbb{E}_{x_1} f_\theta(x_1) f_\theta(x_1)^{\top}$).

In nc-SSL, recent methods as BYOL (Grill et al., 2020) and SimSiam (Chen and He, 2020) employ a dual pair of Siamese networks (Bromley et al., 1994): one side is a composition of an online network (including a projector) and a predictor network, the other side is a target network (see Figure 4.2 for a simple example). The target network has the same architecture as the online network, but has potentially different weights. Given an input x, two augmented views x_1, x_2 are generated, and the network is trained to match the representation of x_1 (through the online network and the predictor network) and the representation of x_2 (through the target network). More precisely, suppose the online network and the target network are two mappings $f_\theta, f_{\theta_a} : \mathbb{R}^d \mapsto \mathbb{R}^h$ and the predictor network is a mapping $g_{\theta_p} : \mathbb{R}^h \mapsto \mathbb{R}^h$, the network is trained to minimize the following loss $L(\theta, \theta_p, \theta_a)$:

$$\frac{1}{2}\mathbb{E}_{x_1,x_2}\left\| \frac{g_{\theta_p}\left(f_\theta(x_1)\right)}{\left\|g_{\theta_p}\left(f_\theta(x_1)\right)\right\|} - \text{StopGrad}\left(\frac{f_{\theta_a}(x_2)}{\|f_{\theta_a}(x_2)\|}\right)\right\|^2.$$

In BYOL and SimSiam, the online network and the target network are trained by running gradient methods on L. The target network is not trained by gradient methods; instead, it is directly set with the weights in the online network (SimSiam) or an exponential moving average (EMA) of the online network (BYOL).

Tian et al. (2021) proposed DirectPred that directly sets the predictor based on the correlation matrix of the predictor inputs. DirectPred achieves comparable performance as BYOL and admits much cleaner theoretical analysis. Therefore, in this chapter, we focus our theoretical analysis on DirectSet(α) (a family of algorithms that include DirectPred as a special case), although we expect some of the insights also apply to BYOL/SimSiam.

Given a positive scalar α, DirectSet(α) sets the predictor based on the correlation matrix F of the predictor inputs:

$$W_p = \frac{F^\alpha}{\|F^\alpha\|} + \epsilon I,$$

where $F = \mathbb{E}_{x_1} f_\theta(x_1) f_\theta(x_1)^\top$. In practice, F is estimated by a moving average over batches. That is,

$$\hat{F} = \mu \hat{F} + (1-\mu)\mathbb{E}_B[f_\theta(x_1) f_\theta(x_1)^\top],$$

where $\mathbb{E}_B$ is the expectation over one batch. The predictor regularization ϵI, when properly chosen, can improve the quality of the learned representations (see the experiments and analysis in Section 4.5).

In the original DirectPred algorithm, α is fixed at $1/2$. To compute $\hat{F}^{1/2}$, one needs to first compute the eigen-decomposition of $\hat{F}$, and then taking the square root of each eigenvalue. This step of eigen-decomposition can be expensive especially when the representation dimension h is high. To avoid the eigen-decomposition step, we propose DirectCopy ($\alpha = 1$), in which the predictor W_p is a direct copy of the $\hat{F}$ (with normalization and regularization)[1]. As we shall see, DirectCopy enjoys both

[1] Computing the spectral norm of $\hat{F}$ is much faster than computing the eigen-decomposition of

theoretical guarantees and strong empirical performance.

4.3 Theoretical Analysis of DirectSet(α)

We prove DirectSet(α) learns meaningful representations and reduces sample complexity of down-stream tasks when the online/target network is a linear network. For simplicity, we focus on the setting where the online network is a single-layer network in this section, although our analysis also extends to deep linear networks (see Appendix C.3). Deep linear networks have been widely used as a tractable theoretical model for studying nonconvex loss landscapes (Kawaguchi, 2016; Du and Hu, 2019; Laurent and Brecht, 2018) and nonlinear learning dynamics (Saxe et al., 2013, 2019; Lampinen and Ganguli, 2018; Arora et al., 2018) in supervised learning setting. Tian et al. (2021) also analyzed nc-SSL on a linear network, but did not analyze their proposed approach DirectPred.

4.3.1 Setup

In this subsection, we define the network model, data distribution and simplify DirectSet(α) algorithm for our theoretical analysis. We consider the following network model (see Figure 4.2),

Assumption 4.1 (Linear network model)**.** *The online, predictor and target network are all single-layer linear network without bias, with weight matrices denoted as* $W, W_p, W_a \in \mathbb{R}^{d\times d}$ *respectively.*

For the data distribution, we assume the input space is a direct sum of an invariant feature subspace and a nuisance feature subspace (see Figure 4.3). Specifically, we assume

$\hat{F}$, because the former only needs the top eigen-vector of $\hat{F}$. Table 4.4 shows that the spectral normalization can also be removed or be replaced by Frobenius normalization without hurting the performance.

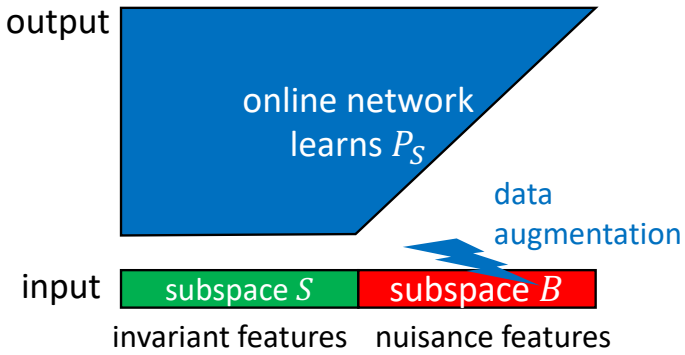

FIGURE 4.3: The input space is a direct sum of subspace S of invariant features and subspace B of nuisance features. With the data augmentations only applying on subspace B, the online network converges to the projection matrix onto S subspace after training.

Assumption 4.2 (Data distribution)**.** *The input x is sampled from $\mathbb{N}(0, I_d)$, and its augmented view x_1, x_2 are independently sampled from $\mathbb{N}(x, \sigma^2 P_B)$, where B is a $(d-r)$-dimensional subspace. We denote S as the orthogonal subspace of B in $\mathbb{R}^d$.*

In this simple data distribution, subspace S corresponds to the features that are invariant to augmentations and its orthogonal subspace B is the nuisance subspace which the augmentation changes. We will prove that DirectSet(α) can learn the projection matrix onto S subspace. Note in the previous work (Tian et al., 2021), they assumed the covariance of the augmentation distribution to be $\sigma^2 I$ and did not study what representation is learned.

Algorithm simplification: For the convenience of analysis, we consider a simplified version of DirectSet(α). We compute the loss function without normalizing the two representations, so the population loss $L(W, W_a, W_p)$ is

$$\frac{1}{2}\mathbb{E}_{x_1,x_2}\left\|W_p W x_1 - \text{StopGrad}\left(W_a x_2\right)\right\|^2, \tag{4.1}$$

and the empirical loss $\hat{L}(W, W_p, W_a)$ is

$$\frac{1}{2n}\sum_{i=1}^{n}\left\|W_p W x_1^{(i)} - \text{StopGrad}(W_a x_2^{(i)})\right\|^2, \tag{4.2}$$

where $x^{(i)}$'s are independently sampled from $\mathbb{N}(0, I)$, and augmented views $x_1^{(i)}$ and $x_2^{(i)}$ are independently sampled from $\mathbb{N}(x^{(i)}, \sigma^2 P_B)$. To train our model, we initialize the online network as a scaled identity matrix, which greatly facilitates our analysis.

Assumption 4.3 (Identity initialization)**.** *The online network weight W is initialized as δI with δ a positive real number.*

We run gradient flow or gradient descent on online network W with weight decay η, and set the the target network $W_a = W$. For clarity of presentation, when training on the population loss, we set W_p as $(W\mathbb{E}_x xx^\top W^\top)^\alpha = (WW^\top)^\alpha$ instead of $(W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha$ as in practice; when training on the empirical loss, we set W_p as $(W\frac{1}{n}\sum_{i=1}^{n} x^{(i)}[x^{(i)}]^\top W^\top)^\alpha$. Here, we set the predictor regularization $\epsilon = 0$ and its influence will be studied in Section 4.5.

4.3.2 Gradient Flow on Population Loss

In this subsection, we show that DirectSet(α) running on the population loss with infinitesimal learning rate can learn the projection matrix onto the invariant feature subspace S.

Theorem 4.1. *Suppose network architecture and data distribution follow Assumption 4.1 and Assumption 4.2, respectively. Suppose we initialize online network W as δI, and run DirectSet(α) on population loss (see Eqn. 4.1) with infinitesimal step size and η weight decay. If $\eta \in \left(\frac{1}{4(1+\sigma^2)}, \frac{1}{4}\right)$ and $\delta > \left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$, then W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$ when time goes to infinity.*

Theorem 4.1 shows that when the weight decay is in certain range, and when the initialization is large enough, the online network can converge to the desired projection matrix P_S [2]. In sequel, we explain how the dynamics of W leads to a projection matrix and how the weight decay and initialization scale come into play. We leave the full proof in Appendix C.2.1. We also consider the setting when W_p is set as $(W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha$ in Appendix C.2.4 and extend the result to deep linear networks in Appendix C.3.

Due to the identity initialization, we can ensure that W is always a real symmetric matrix and is simultaneously diagonalizable with P_B. We can then analyze the evolution of each eigenvalue in W separately. Under our assumptions, it turns out that all the eigenvalues whose eigenvectors lie in the B subspace share the same value λ_B, and all the eigenvalues in the S subspace share the value λ_S as shown in the following time dynamics:

$$\dot{\lambda}_B = \lambda_B \left[-(1+\sigma^2) |\lambda_B|^{4\alpha} + |\lambda_B|^{2\alpha} - \eta \right],$$
$$\dot{\lambda}_S = \lambda_S \left[- |\lambda_S|^{4\alpha} + |\lambda_S|^{2\alpha} - \eta \right].$$

Next, we show λ_B converges to zero and λ_S converges to a positive number, which immediately implies that W converges to some scaling of P_S.

Similar as the analysis in Tian et al. (2021), when $\eta > \frac{1}{4(1+\sigma^2)}$, we know $\dot{\lambda}_B < 0$ for any $\lambda_B > 0$ and $\lambda_B = 0$ is a stable stationary point, as illustrated in Figure 4.4 (**Left**). Therefore, as long as $\eta > \frac{1}{4(1+\sigma^2)}$, λ_B must converge to zero. On the other hand, there are three non-negative solutions to $\dot{\lambda}_S = 0$, which are $0, \lambda_S^- = \left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$ and $\lambda_S^+ = \left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$ when $0 < \eta < \frac{1}{4}$. As illustrated in Figure 4.4 (**Right**), if

[2] Note that Theorem 4.1 also holds with negative initialization $\delta < -\left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$, in which case W converges to $-\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$. Our other results can be extended to negative δ in a similar way.

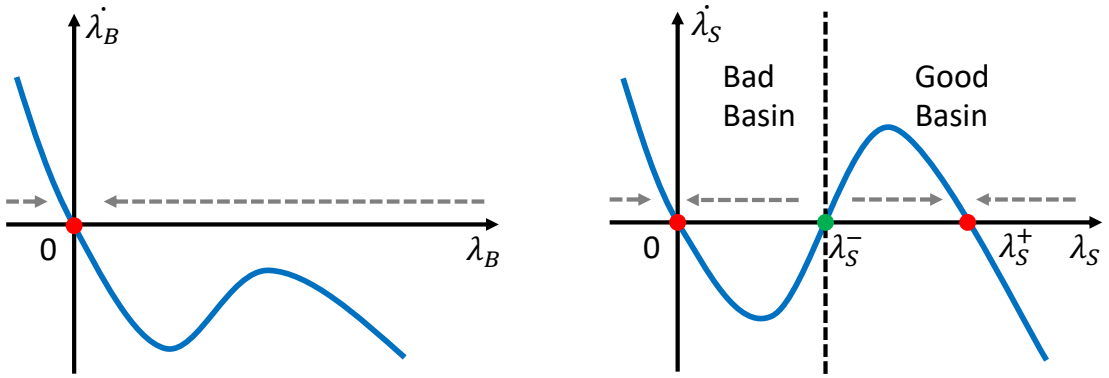

FIGURE 4.4: (**Left**) with appropriate weight decay, λ_B always converges to zero ; (**Right**) λ_S converges to zero when it's initialized in the bad basin and converges to positive λ_S^+ when it's initialized in the good basin.

initialization $\delta > \lambda_S^-$ (good basin), λ_S converges to a positive value λ_S^+; if $0 < \delta < \lambda_S^-$ (bad basin), λ_S converges to zero.

Thresholding role of weight decay in feature learning: While Tian et al. (2021) showed why nc-SSL does not collapse, one key question is how nc-SSL learns useful features and how the method determines which feature is learned. Now it is clear: the weight decay factor η makes a call on what features should be learned. As illustrated in Figure 4.1, *Nuisance features* subject to significant changes under data augmentations have larger variance ($\sigma^2 > \frac{1}{4\eta} - 1$), the eigenspace corresponding to these features goes to zero; on the other hand, *invariant features* that are robust to data augmentations have much smaller variance ($\sigma^2 < \frac{1}{4\eta} - 1$) and these features are kept. In our above analysis, B subspace corresponds to the nuisance features and collapses to zero; S subspace corresponds to the invariant features (whose variance was assumed as zero for simplicity) and is kept after training.

Figure 4.5 shows the spectrum of F (which is the correlation matrix of the predictor inputs) when the network is trained by DirectCopy under different weight decay η on CIFAR10. The larger the weight decay is, the fewer significant eigenvalues F

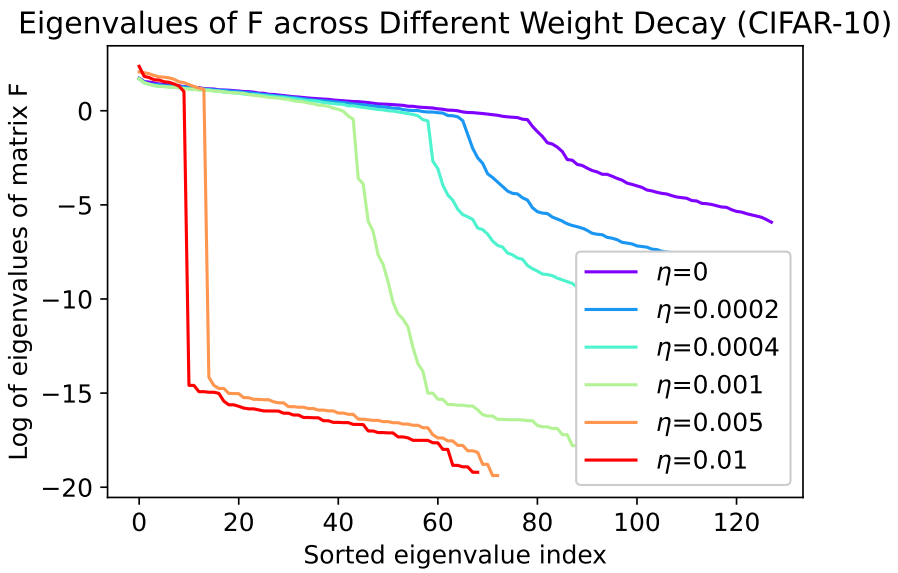

FIGURE 4.5: Eigenvalues of correlation matrix F at 100-th epoch when it's trained by DirectCopy under different weight decays.

has [3]. This suggests that the features are better suppressed when larger weight decay is adopted.

Therefore, it is crucially important to choose weight decay appropriately: a too small η may not be sufficient to suppress the nuisance features; a too large η can also collapse the invariant features. As shown in Section 4.5, both cases lead to worse downstream performance.

4.3.3 Sample Complexity of nc-SSL

In this subsection, we prove that DirectCopy (one special case of DirectSet(α) with $\alpha = 1$) learns the projection matrix given polynomial number of unlabeled samples.

Theorem 4.2. *Suppose network architecture and data distribution are as defined in Assumption 4.1 and Assumption 4.2, respectively. Suppose we initialize online network as δI, and run DirectCopy on empirical loss (see Eqn. 4.2) with γ step size*

[3] Notice that there is a natural drop in the eigenvalues of F even without weight decay ($\eta = 0$) since features along different eigen-directions of F can have very different magnitudes.

and η *weight decay. Assume* $\sigma^2 = \Theta(1), \eta \in \left(\frac{1+\sigma^2/4}{4(1+\sigma^2)}, \frac{1+3\sigma^2/4}{4(1+\sigma^2)}\right), \delta \in (1/2, O(1))$ *and* $\gamma = \Theta(1)$. *For any accuracy* $\hat{\epsilon} > 0$, *given* $n \geqslant poly(d, 1/\hat{\epsilon})$ *number of samples, with probability at least* 0.99 *there exists* $t = O(\log(1/\hat{\epsilon}))$ *such that*

$$\left\| \widetilde{W}_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| \leqslant \hat{\epsilon},$$

where $\widetilde{W}_t$ *is the online network weights at the* t*-th step.*

The proof proceeds by first proving that gradient descent on the population loss converges in linear rate and then couples the gradient descent dynamics on empirical loss and that on population loss. See the detailed proof in Appendix C.2.2.

4.3.4 Sample Complexity on Downstream Tasks

In this subsection, we show that the learned representations can indeed reduce the sample complexity on the downstream tasks. We consider the following data distribution for the down-stream task:

Assumption 4.4 (Downstream data distribution)**.** *Each input* $z^{(i)}$ *is sampled from* $\mathcal{N}(0, I_d)$ *and its label* $y^{(i)} = \left\langle z^{(i)}, w^* \right\rangle + \xi^{(i)}$, *where* w^* *is the ground truth vector with unit* ℓ_2 *norm and* $\xi^{(i)}$ *is independently sampled from* $\mathcal{N}(0, \beta^2)$. *We assume the ground truth* w^* *lies on an* r*-dimensional subspace* S *and we denote the projection matrix on subspace* S *simply as* P.

In practice, usually the semantically relevant features (S subspace here) are invariant to augmentations and the nuisance features (orthogonal subspace of S) have high variance under augmentations. Therefore, by previous analysis, we expect DirectSet(α) to learn the projection matrix P.

Suppose $\{(z^{(i)}, y^{(i)})\}_{i=1}^n$ are n training samples. Each input $z^{(i)}$ is transformed by a matrix $\hat{P} \in \mathbb{R}^{d \times d}$ (for example the learned online network W) to get its representation

$\hat{P}z^{(i)}$. The regularized loss is then defined as $\hat{L}(w) := \frac{1}{2n}\sum_{i=1}^{n}\left\|\left\langle \hat{P}z^{(i)}, w\right\rangle - y^{(i)}\right\|^2 + \frac{\rho}{2}\|w\|^2$, where the regularization coefficient ρ will be chosen carefully to prevent w from overfitting the noise in labels. Note here the regularization has nothing to do with the predictor regularization ϵI in DirectSet(α) algorithm.

In the below theorem, we show that when $\left\|P - \hat{P}\right\|_F$ is small, the above ridge regression can recover the ground truth w^* given only $O(r)$ number of samples, where r is the dimension of the subspace on which w^* lies.

Theorem 4.3. *Suppose the downstream data distribution is as defined in Assumption 4.4. Suppose* $\left\|\hat{P} - P\right\|_F \leqslant \hat{\epsilon}$ *with* $\hat{\epsilon} < 1$. *Choose the regularizer coefficient* $\rho = \hat{\epsilon}^{1/3}$. *For any* $\zeta < 1/2$, *given* $n \geqslant O(r + \log(1/\zeta))$ *number of samples, with probability at least* $1 - \zeta$, *the training loss minimizer* $\hat{w}$ *satisfies*

$$\left\|\hat{P}\hat{w} - w^*\right\| \leqslant O\left(\hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right).$$

In the above theorem, when n is at least $O\left(\frac{\beta^2(r+\log(1/\zeta))}{\hat{\epsilon}^{2/3}}\right)$, we have $\left\|\hat{P}\hat{w} - w^*\right\| \leqslant O(\hat{\epsilon}^{1/3})$. Note that if we directly estimate $\hat{w}$ without transforming the inputs by $\hat{P}$, we need $\Omega(d)$ number of samples to ensure that $\|\hat{w} - w^*\| \leqslant o(1)$ (Wainwright, 2019). The proof of Theorem 4.3 follows from bounding the difference between $\hat{P}\hat{w}$ and w^* by matrix concentration inequalities and matrix perturbation bounds. The full proof is in Appendix C.2.3.

4.4 Empirical Performance of DirectCopy

In the previous analysis, we show DirectSet(α), and in particular DirectCopy (DirectSet(α) with $\alpha = 1$), could recover the input feature structure with polynomial samples and make the downstream task more sample efficient in a simple linear

setting. Compared with the original DirectPred (DirectSet(α) with $\alpha = 1/2$), DirectCopy is a simpler and computationally more efficient algorithm since it directly set the predictor as the correlation matrix F, without the eigen-decomposition step. By our analysis in Theorem 4.1, DirectCopy also learns the projection matrix P_S with larger scale [4] compared with DirectPred, which suggests that the invariant features learned by DirectCopy are stronger and more distinguishable. Next, we show that DirectCopy is on par with (or even outperforms) the original DirectPred in various datasets, when coupling with deep nonlinear models on real datasets.

4.4.1 Results on STL-10, CIFAR-10 and CIFAR-100

We use ResNet-18 (He et al., 2016) as the backbone network, a two-layer nonlinear MLP as the projector, and a linear predictor. Unless specified otherwise, SGD is used as the optimizer with weight decay $\eta = 0.0004$. To evaluate the quality of the pre-trained representations, we follow the linear evaluation protocol. Each setting is repeated 5 times to compute the mean and standard deviation. The accuracy is reported as "mean±std". Unless explicitly specified, we use learning rate $\gamma = 0.01$, regularization $\epsilon = 0.2$ on STL-10; $\gamma = 0.02, \epsilon = 0.3$ on CIFAR-10 and $\gamma = 0.03, \epsilon = 0.3$ on CIFAR-100. See more detailed experiment settings in Appendix C.1.

STL-10: We evaluate the quality of the learned representation after each epoch, and report the best accuracy in the first 100/300/500 epochs in Table 4.1. DirectCopy achieves substantially better performance than DirectPred and SGD baseline, especially when trained with longer epochs. DirectPred (freq=5) means the predictor is set by DirectPred every 5 batchs, and is trained with gradient updates in other batchs, which outperforms DirectPred in later epochs, but is still much worse than DirectCopy. The SGD baseline is obtained by training the linear predictor using

[4] Recall that in Theorem 4.1 under DirectSet(α), online matrix W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$. So with a larger α, the scalar in front of P_S becomes larger.

Table 4.1: STL-10/CIFAR-10/CIFAR-100 Top-1 accuracy of DirectCopy and other algorithms. The numbers for DirectPred, DirectPred (freq=5) and SGD baseline on STL-10/CIFAR-10 are obtained from Tian et al. (2021).

	Num of epochs		
	100	300	500
STL-10			
DirectCopy	77.83±0.56	**82.01±0.28**	**82.95±0.29**
DirectPred	**77.86±0.16**	78.77±0.97	78.86±1.15
DirectPred (freq=5)	77.54±0.11	79.90±0.66	80.28±0.62
SGD baseline	75.06±0.52	75.25±0.74	75.25±0.74
CIFAR-10			
DirectCopy	84.02±0.37	**89.17±0.12**	**89.62±0.10**
DirectPred	**85.21±0.23**	88.88±0.15	89.52±0.04
DirectPred (freq=5)	84.93±0.29	88.83±0.10	89.56±0.13
SGD baseline	84.49±0.20	88.57±0.15	89.33±0.27
CIFAR-100			
DirectCopy	55.40±0.19	61.06±0.14	62.23±0.06
DirectPred	**56.60±0.27**	61.65±0.18	62.68±0.35
DirectPred (freq=5)	56.43±0.21	**62.01±0.22**	**63.15±0.27**
SGD baseline	54.94±0.50	60.88±0.59	61.42±0.89

SGD.

CIFAR-10/100: For CIFAR-10, DirectCopy is slighly worse than DirectPred at epoch 100, but catches up and gets even better performance in epoch 300 and 500 (Table 4.1). For CIFAR-100, at earlier epochs, the performance of DirectCopy is not as good as DirectPred, but the gap gradually diminishes in later epochs. Both DirectCopy and DirectPred outperfom the SGD baseline. DirectPred (freq=5) achieves even better performance, but at the cost of a more complicated algorithm.

4.4.2 Results on ImageNet

Table 4.2: ImageNet Top-1 accuracy of DirectCopy, DirectPred and BYOL baseline with one/two-layer predictor after 100 epochs.

	DirectCopy	DirectPred	1-layer BYOL	2-layer BYOL
ImageNet	**68.8**	68.5	68.6	66.5

Following BYOL (Grill et al., 2020), we use ResNet-50 as the backbone and a

two-layer MLP as the projector. We use LARS (You et al., 2017) optimizer and train the model for 100 epochs. See more detailed experiment settings in Appendix C.1.

For fairness, we compare DirectCopy to the gradient-based baseline which uses the same-sized linear predictor as ours. As shown in Table 4.2, at 100-epoch, this baseline achieves 68.6 top-1 accuracy, which is already significantly higher than BYOL with two-layer predictor reported in the literature (e.g., Chen and He (2020) reported 66.5 top-1 under 100-epoch training). DirectCopy using normalized F with regularization parameter $\epsilon = 0.01$ achieves 68.8 under the same setting, better than this strong baseline. In contrast, DirectPred achieves 68.5, slightly lower than the BYOL baseline with linear predictor.

4.5 Ablation Study

In this section, we study the influence of predictor regularization ϵ, normalization method, weight decay and degree α on the performance of DirectCopy.

Table 4.3: STL-10/CIFAR-10 Top-1 accuracy of DirectCopy with varying regularization ϵ.

	Number of epochs	
	100	300
STL-10		
$\epsilon = 0$	76.57±0.66	81.19±0.39
$\epsilon = 0.1$	**78.05±0.14**	81.60±0.15
$\epsilon = 0.2$	77.83±0.56	**82.01±0.28**
$\epsilon = 1$	31.10±0.80	31.10±0.80
CIFAR-10		
$\epsilon = 0$	80.53±1.14	86.07±0.71
$\epsilon = 0.1$	83.97±0.25	88.58±0.11
$\epsilon = 0.3$	**84.02±0.37**	**89.17±0.12**
$\epsilon = 1$	57.38±11.62	83.15±4.24

Predictor regularization: Table 4.3 shows that when the predictor regularization ϵ increases, the performance of DirectCopy on STL-10 and CIFAR-10 improves at

first and then deteriorates. On STL-10, DirectCopy with $\epsilon = 1$ completely fails. On CIFAR-10, although DirectCopy with $\epsilon = 1$ achieved reasonable performance at epoch 300, it's still much worse than $\epsilon = 0.3$.

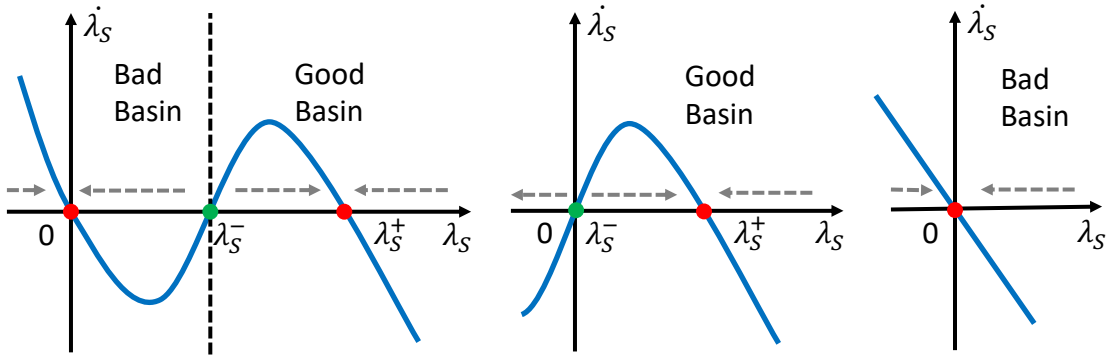

FIGURE 4.6: Increasing ϵ shifts the two positive stationary points λ_S^- and λ_S^+ towards zero. (**Left**) when $0 \leqslant \epsilon < \frac{1-\sqrt{1-4\eta}}{2}$, increasing ϵ expands the good basin ($\lambda_S > \lambda_S^-$) by reducing λ_S^-. (**Middle**) when $\frac{1-\sqrt{1-4\eta}}{2} \leqslant \epsilon < \frac{1+\sqrt{1-4\eta}}{2}$, λ_S^- becomes zero and λ_S converges to positive λ_S^+ from any positive value; further increasing ϵ decreases λ_S^+. (**Right**) when $\frac{1+\sqrt{1-4\eta}}{2} \leqslant \epsilon$, λ_S^+ becomes zero and λ_S always converges to zero.

To better understand the role of ϵ, we analyze the simple linear setting as in Section 4.3.1 while setting $W_p = WW^\top + \epsilon I$. Recall that λ_B is the eigenvalue of W in B subspace and λ_S is that in S subspace. When the weight decay is appropriate, λ_B still converges to zero. On the other hand, the dynamics for λ_S is as follows: $\dot{\lambda}_S = -\lambda_S \left(\lambda_S^2 + \epsilon - \frac{1-\sqrt{1-4\eta}}{2}\right)\left(\lambda_S^2 + \epsilon - \frac{1+\sqrt{1-4\eta}}{2}\right)$. Increasing ϵ shifts the two positive stationary points λ_S^-, λ_S^+ towards zero. As illustrated in Figure 4.6, as ϵ increases, when λ_S^+ is still positive, the good attraction basin expands, which means λ_S can converge to a positive value from a smaller initialization; when λ_S^+ shifts to zero, λ_S converges to zero regardless the initialization size. See the full analysis in Appendix C.4.

Intuitively, a reasonable ϵ can alleviate representation collapse, but a too large ϵ also encourages representation collapse. As shown in Figure 4.7, when ϵ increases from zero, more eigenvalues of F becomes large; but when ϵ exceeds 0.3, eigenvalues

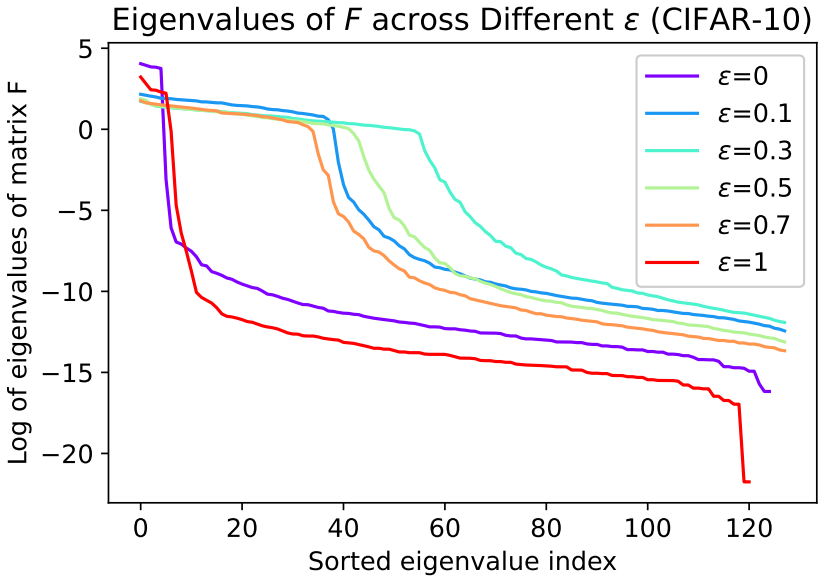

FIGURE 4.7: Eigenvalues of F when trained by DirectCopy under different predictor regularization ϵ on CIFAR-10 for 100 epochs.

of F begin to collapse.

Table 4.4: STL-10/CIFAR-10 Top-1 accuracy of DirectCopy with F matrix normalized by spectral norm/Frobenius norm or no normalization.

	Number of epochs	
	100	300
STL-10		
Spectral	**77.83±0.56**	82.01±0.28
Frobenius	77.71±0.18	**82.06±0.28**
None	77.81±0.20	82.00±1.24
CIFAR-10		
Spectral	84.02±0.37	89.17±0.12
Frobenius	**84.33±0.25**	**89.62±0.14**
None	81.76±0.34	89.21±0.17

Normalization on F: In our experiments, we have been normalizing F by its spectral norm before adding the regularization: $W_p = F/\|F\| + \epsilon I$. It turns out that we can also normalize F by its Frobenius norm or simply skip the normalization step. In Table 4.4, we see comparable performance from DirectCopy with Frobenius

normalization or no normalization, especially when trained longer.

Table 4.5: STL-10/CIFAR-10 Top-1 accuracy of DirectCopy with varying weight decay.

	Number of epochs	
	100	300
STL-10		
$\eta = 0$	71.94±0.93	78.53±0.40
$\eta = 0.0004$	**77.83±0.56**	**82.01±0.28**
$\eta = 0.001$	77.65±0.16	80.28±0.16
$\eta = 0.01$	58.12±0.94	58.53±0.76
CIFAR-10		
$\eta = 0$	79.15±0.08	85.35±0.31
$\eta = 0.0004$	**84.02±0.37**	**89.17±0.12**
$\eta = 0.001$	83.91±0.33	87.75±0.16
$\eta = 0.01$	65.31±1.19	65.63±1.30

Weight decay: Table 4.5 shows that when weight decay η increases, the performance of DirectCopy improves at first and then deteriorates. This fits our analysis on simple linear networks. Basically, when the weight decay η increases, it can suppress the nuisance features more effectively, but a too large weight decay also collapses the useful features.

Predictor degree: We compare DirectCopy against DirectSet(α) with $\alpha = 2, 1/2, 1/4$. Table 4.6 shows that DirectCopy outperforms other algorithms on STL-10. On CIFAR-10, DirectCopy is slightly worse at epoch 100, but catches up in later epochs. According to our analysis, $\alpha = 2$ is supposed to learn stronger invariant features than $\alpha = 1$, but it does not lead to better performance in experiments. This suggests that the benefits from more distinguishable features diminish beyond $\alpha = 1$.

4.6 Conclusion

In this chapter, we have proved DirectSet(α) can learn the desirable projection matrix in a linear network setting and can reduce the sample complexity on down-stream

Table 4.6: STL-10/CIFAR-10 Top-1 accuracy of DirectSet(α) with varying degree α.

	Number of epochs	
	100	300
STL-10		
$\alpha = 2$	76.80±0.22	80.90±0.18
$\alpha = 1$	**77.83±0.56**	**82.01±0.28**
$\alpha = 1/2$	77.82±0.37	77.83±0.37
$\alpha = 1/4$	76.82±0.36	76.82±0.36
CIFAR-10		
$\alpha = 2$	82.96±0.56	88.60±0.11
$\alpha = 1$	84.02±0.37	**89.17±0.12**
$\alpha = 1/2$	**84.88±0.21**	88.32±0.57
$\alpha = 1/4$	84.78±0.21	87.82±0.32

tasks. Our analysis sheds light on the crucial role of weight decay in nc-SSL, which discards the features that have high variance under augmentations and keeps the invariant features. Inspired by the analysis, we designed a simpler and more efficient algorithm DirectCopy, which achieved comparable or even better performance than the original DirectPred (Tian et al., 2021) on various datasets.

We view our work as an initial step towards demystifying the representation learning in nc-SSL. Many mysteries still lie beyond the explanation of the current theory and we leave them for future work.

5

Plateau in Monotonic Linear Interpolation

Monotonic linear interpolation (MLI) — on the line connecting a random initialization with the minimizer it converges to, the loss and accuracy are monotonic — is a phenomenon that is commonly observed in the training of neural networks. Such a phenomenon may seem to suggest that optimization of neural networks is easy. In this chapter, we show that the MLI property is not necessarily related to the hardness of optimization problems, and empirical observations on MLI for deep neural networks depend heavily on the biases. In particular, we show that interpolating both weights and biases linearly leads to very different influences on the final output, and when different classes have different last-layer biases on a deep network, there will be a long plateau in both the loss and accuracy interpolation (which existing theory of MLI cannot explain). We also show how the last-layer biases for different classes can be different even on a perfectly balanced dataset by analyzing the gradient descent dynamics on a simple model. Empirically we demonstrate that similar intuitions hold on practical networks and realistic datasets.

5.1 Introduction

Deep neural networks can often be optimized using simple gradient-based methods, despite the objectives being highly nonconvex. Intuitively, this suggests that the loss landscape must have nice properties that allow efficient optimization. To understand the properties of loss landscape, Goodfellow et al. (2014) studied the linear interpolation between a random initialization and the local minimum found after training. They observed that the loss interpolation curve is monotonic and approximately convex (see the MNIST curve in Figure 5.1) and concluded that these tasks are easy to optimize. However, other recent empirical observations, such as Frankle (2020) observed that for deep neural networks on more complicated datasets, both the loss and the error curves have a long plateau along the interpolation path, i.e., the loss and error remain high until close to the optimum (see the CIFAR-10 curve in Figure 5.1). Does the long plateau along the linear interpolation suggest these tasks are harder to optimize? Not necessarily, since the hardness of optimization problems does not need to be related to the shape of interpolation curves (see examples in Appendix D.1).

In this chapter we give the first theory that explains the plateau in both loss and error interpolations. We attribute the plateau to simple reasons as the bias terms, the network initialization scale and the network depth, which may not necessarily be related to the difficulty of optimization.

There are many different theories for the optimization of overparametrized neural networks, in particular the neural tangent kernel (NTK) analysis (Jacot et al., 2018; Du et al., 2018a; Allen-Zhu et al., 2019; Arora et al., 2019b) and mean-field analysis (Chizat and Bach, 2018b; Mei et al., 2018). However they don't explain the plateau in both loss and error interpolations. For NTK regime, the network output is nearly linear in the parameters and the loss interpolation curve is monotonically

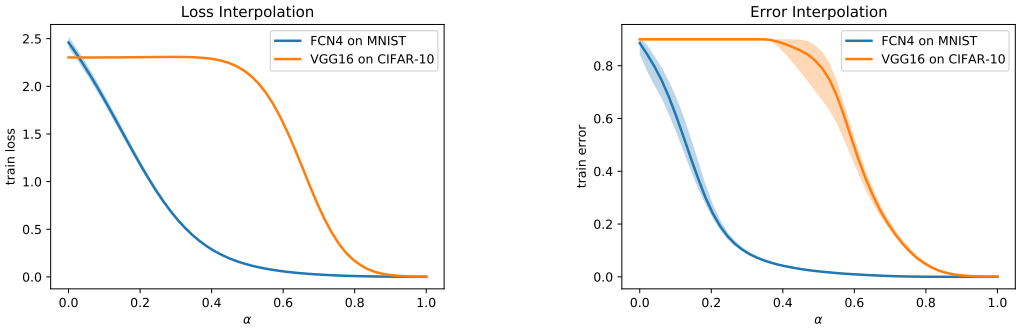

FIGURE 5.1: Loss interpolation curve and error interpolation curve for a four-layer fully-connected network (FCN4) on MNIST and for VGG16 on CIFAR-10.

decreasing and convex — no plateau in the loss interpolation. Mean-field regime often uses a smaller initialization on a homogeneous neural network (as considered in Chizat and Bach (2018b); Mei et al. (2018)). In this case, the interpolated network output is basically a scaled version of the network output at the minimum and has same label predictions — no plateau in the error interpolation curve.

5.1.1 Our results

Our theoretical results consist of two parts. In the first part (see Section 5.3), we give a plausible explanation for the plateau in both the loss and error curves.

Claim 5.1 (informal)**.** *If a deep network has a relatively small initialization, and its last-layer biases are significantly different for different classes, then both the loss and error curves will have a plateau. The length of the plateau is longer for a deeper network.*

We formalize this claim in Theorem 5.1. For intuition, consider an r-layer neural network that only has bias on the last layer, and consider Xavier initialization (Glorot and Bengio, 2010) which typically gives small output and zero bias. If we consider the α-interpolation point (with coefficient α for the minimum and $(1-\alpha)$ for the initialization), then the weight "signal" from the minimum scales as α^r (as it is the product of r layers) while the bias scales as α. As illustrated in Figure 5.2 (right),

when r is large and there is a difference in biases, the bias will dominate, which creates a plateau in error. For the loss, one can also show that the weight signal is near 0 for small α, so the network output is dominated by the biases and the loss cannot beat the random guessing at initialization. Note that this explanation for the plateau does not have any implication on the hardness of optimization.

However, why would the last-layer biases be different for different classes, especially in cases when the biases are initialized as zeros and all classes are balanced? In the second part (see Section 5.4), we focus on a simple model that we call r-homogeneous-weight network. This is a two-layer network whose i-th output is $\langle W_{i,:}, x\rangle^r + b_i$, where $x \in \mathbb{R}^d$ is the network input, $W_{i,:} \in \mathbb{R}^d$ is the weight vector and $b_i \in \mathbb{R}$ is the bias (see Figure 5.2 (left)). This model captures the intuition that the weights are r-homogeneous while the (last-layer) bias is 1-homogeneous. Under this model we can show

Claim 5.2 (informal)**.** *For the r-homogeneous-weight network on a simple balanced dataset, the class that is learned last has the largest bias.*

Here, a class is learned when all the samples in this class get classified correctly with good confidence. We basically show that once a class gets learned, the bias associated with this class starts decreasing and eventually the class that is learned last has the largest bias. We formalize this claim in Theorem 5.2.

In Section 5.5, we verify these ideas empirically on fully-connected networks for MNIST (Deng, 2012), Fashion-MNIST (Xiao et al., 2017) and on VGG-16 (Simonyan and Zisserman, 2014) for CIFAR-10, CIFAR-100 (Krizhevsky and Hinton, 2009). We first show that if we train a neural network *without* using any bias, then the error curve has much shorter plateau or no plateau at all. Even for networks that are trained normally with biases, we design a homogeneous interpolation scheme for biases to make sure that both biases and weights are r-homogeneous. Such an

interpolation indeed significantly shortens the plateau for the error. We also show that decreasing the initialization scale or increasing the network depth can produce a longer plateau in both the error and loss curves. Finally, we show that the bias is correlated with the ordering in which the classes are being learned for small datasets, which suggests that even though the model we consider in the convergence analysis is simple, it captures some of the behavior in practice.

5.1.2 Related works

There are two major lines of work studying interpolation between different points for neural networks, one on monotonic linear interpolation that interpolates the initial network and the learned network, and the other on mode connectivity that connects two learned networks.

Monotonic linear interpolation. Goodfellow et al. (2014) first studied the linear interpolation between the network at initialization and the network after training on MNIST. Frankle (2020) extended the experiments to modern networks on CIFAR-10 and ImageNet and found that though the loss/error is still monotonically non-increasing along the path, it remains high until close to the optimum. Lucas et al. (2021) showed that MLI holds when the network output curve along the interpolation path is close to linear (measured by Gaussian length). However, the Gaussian length can only be formally controlled in the NTK regime.

Mode connectivity. Mode connectivity considers the interpolation between two learned networks (modes) found by SGD. In general, a linear interpolation between two different local minima crosses regions of high loss (Goodfellow et al., 2014). Surprisingly, Draxler et al. (2018) and Garipov et al. (2018) observed that local minima found by SGD from different initializations can be connected via a piecewise linear path of low loss. Frankle et al. (2020) and Fort et al. (2020) observed that local minima trained from the same initialization can also be connected using a

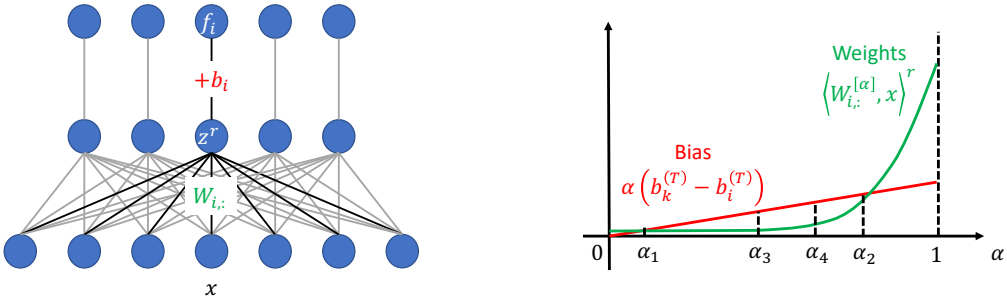

FIGURE 5.2: **(Left)** Our r-homogeneous-weight model with $f_i(x) = \langle W_{i,:}, x \rangle^r + b_i$. **(Right)** The comparison between interpolated bias $\alpha(b_k^{(T)} - b_i^{(T)})$ and interpolated weights signal $\left\langle W_{i,:}^{[\alpha]}, x \right\rangle^r$.

linear path. Freeman and Bruna (2016); Venturi et al. (2018b); Nguyen (2019, 2021); Kuditipudi et al. (2019); Shevchenko and Mondelli (2020); Nguyen et al. (2021) gave several theoretical explanations for this phenomenon.

5.2 Preliminaries

We first formally define the linear interpolation between the network at initialization and the network after training. Then we describe the notations that we will use in the chapter.

Linear interpolation: Consider a network with parameters $\theta \in \mathbb{R}^p$. Suppose the network is initialized with parameters $\theta^{(0)}$ and it converges to $\theta^{(T)}$. A *linear interpolation* is constructed by setting the parameters $\theta^{[\alpha]} = (1-\alpha)\theta^{(0)} + \alpha\theta^{(T)}$ for $\alpha \in [0,1]$. The *loss interpolation curve* is defined as $\gamma_{\text{loss}}(\alpha) : [0,1] \to \mathbb{R}$ such that $\gamma_{\text{loss}}(\alpha)$ is the training loss of the network at $\theta^{[\alpha]}$. Similarly, the *error interpolation curve* is defined as $\gamma_{\text{error}}(\alpha) : [0,1] \to [0,1]$ with $\gamma_{\text{error}}(\alpha)$ as the training error of the network at $\theta^{[\alpha]}$. Here, the training error is simply the ratio of training samples that get classified incorrectly by the network.

Notations: Most notations have been defined in Section 1.4. Below we define some specific notations for this chapter.

For any non-zero vector v, we use $\bar{v}$ to denote $v/\|v\|$. For any time t, we use $\theta^{(t)}, f^{(t)}$ to denote the parameters and the network at time t. For any $\alpha \in [0,1]$, we use $\theta^{[\alpha]}, f^{[\alpha]}$ to denote the α interpolation point, which means $\theta^{[\alpha]} := (1-\alpha)\theta^{[0]} + \alpha\theta^{[T]}$ and $f^{[\alpha]}$ is the network with parameters $\theta^{[\alpha]}$.

5.3 Plateau for loss and error interpolations

We prove that the long plateau exists in the loss and error curves when the initialization is small and the network is deep on fully-connected networks. The detailed proof can be found in Appendix D.2.3.

We consider an r-layer fully-connected neural network with r at least three. Given input $x \in \mathbb{R}^{n_0}$, the network output is

$$g(x) := V_r \sigma\left(V_{r-1} \cdots \sigma(V_1 x) \cdots\right) + b, \tag{5.1}$$

where $V_i \in \mathbb{R}^{n_r \times n_{r-1}}$ for each layer $i \in [r]$ and $b \in \mathbb{R}^{n_r}$. Here the activation function $\sigma(\cdot)$ can be either identity function or ReLU function. The output layer width equals to the number of classes, i.e., $n_r = k$. We use $L(\{V_i\}, b)$ to denote the sum of cross entropy loss over all samples.

For the biases, we initialize them as zeros and assume after training there exists a gap between the largest bias and the second largest, which also holds empirically (see Figure 5.8).

Assumption 5.1 (Bias Gap)**.** *Choosing $i^* \in \arg\max_{i\in[k]} b_i^{(T)}$, we have*

$$b_{i^*}^{(T)} - \max_{i \neq i^*} b_i^{(T)} > 0.$$

Without loss of generality, we assume that $b_k^{(T)} > \max_{i\in[k-1]} b_i^{(T)}$. We denote $\Delta_{\min} := b_k^{(T)} - \max_{i\in[k-1]} b_i^{(T)}$ and $\Delta_{\max} := b_k^{(T)} - \min_{i\in[k-1]} b_i^{(T)}$.

Then, we show both the loss and error interpolation curves have a long plateau in Theorem 5.1.

Theorem 5.1. *Suppose the network is defined as in Equation* (5.1) *and suppose the weights satisfy* $\left\|V_i^{(0)}\right\| \leqslant \delta, \left\|V_i^{(T)}\right\| \leqslant V_{\max}$ *for all layers* $i \in [r]$. *On a* k*-class balanced dataset whose inputs have* ℓ_2 *norm at most* 1, *if Assumption 5.1 holds, for any* $\epsilon > 0$, *as long as* $\delta < \min\left(\frac{\epsilon^{1/r}}{r}, \frac{1}{r^2}, \left(\frac{1}{2e}\right)^{\frac{2}{r-2}}\right)$, *there exist* $\alpha_1 = \frac{\delta}{\Delta_{\min}}, \alpha_2 = \left(\frac{1}{1+\sqrt{\delta}}\right)^{\frac{r}{r-1}} \left(\frac{\Delta_{\min}}{2V_{\max}^r}\right)^{\frac{1}{r-1}}$ *and* $\alpha_3 = \frac{\epsilon^{1/r}}{V_{\max}}$ *such that*

1. *for all* $\alpha \in [\alpha_1, \alpha_2]$, *the error is* $1 - 1/k$;

2. *for all* $\alpha \in [0, \alpha_3]$, *we have*

$$\log k - 2e\epsilon \leqslant \frac{1}{N} L\left(\left\{V_i^{[\alpha]}\right\}, b^{[\alpha]}\right) \leqslant \log k + \alpha\Delta_{\max} + 2e\epsilon,$$

where N *is the number of training examples.*

The above theorem shows that for all $\alpha \in [\alpha_1, \alpha_2]$, the error remains at $1 - 1/k$ that is the same as random guessing. We skip the very short initial region $[0, \frac{\delta}{\Delta_{\min}}]$ since the bias is very small and the error can be unpredictable due to the randomness in initial weights. When initialization scale δ is small, this error plateau region is roughly $[0, (\frac{\Delta_{\min}}{2V_{\max}^r})^{\frac{1}{r-1}}]$. Empirically, $\frac{\Delta_{\min}}{2V_{\max}^r}$ is smaller than 1 and does not change much when depth increases. So the plateau becomes longer in a deeper network.

Intuitively, the plateau in error curve is there because for a small initialization, the output is close to
$\alpha^r V_r^{(T)} \sigma\left(V_{r-1}^{(T)} \cdots \sigma(V_1^{(T)} x) \cdots\right) + \alpha b^{(T)}$. When α is not large enough α^r is much smaller than α, so for every class $i \neq k$, the first term (signal part) cannot overcome the bias gap $\alpha(b_k^{(T)} - b_i^{(T)})$. This implies that all samples are predicted as class k and the error is $1 - 1/k$.

We also show that the average loss cannot be lower than $\log k - 2e\epsilon$ when $\alpha \leqslant \frac{\epsilon^{1/r}}{V_{\max}}$. Note a small random initialization can achieve loss approximately $\log k$. Usually the

bias gap $\Delta_{\max}$ in practice is not very large, so the loss curve remains nearly flat during this interpolation region. Again, the loss plateau is becoming longer when depth r increases. This is because the weights signal remains near 0 for a larger range of α.

5.4 Training dynamics for creating a bias gap

In this section, we explain how the gradient flow dynamics generates a bias gap on a balanced dataset by analyzing a simple model. Below, we first define the network model, training dataset and optimization procedure for our analysis.

r-homogeneous-weight network: We consider a two-layer and k-output neural network with activation function $\sigma(z) := z^r$, where r is a positive constant that is at least three. As illustrated in Figure 5.2 (left), under input $x \in \mathbb{R}^d$, the i-th output $f_i(x)$ is $\langle W_{i,:}, x\rangle^r + b_i$, where weight vector $W_{i,:} \in \mathbb{R}^d$ is the i-th row of weight matrix $W \in \mathbb{R}^{k\times d}$ and $b_i \in \mathbb{R}$ is the i-th entry of vector $b \in \mathbb{R}^k$. In output $f_i(x)$, we call $\langle W_{i,:}, x\rangle^r$ the signal since it is input-dependent and call b_i the bias.

Dataset: We consider a k-class balanced dataset, with k as a constant. We denote the whole dataset as $\mathcal{S}$ and denote the subset for each class $i \in [k]$ as $\mathcal{S}_i$. Each subset $\mathcal{S}_i$ has exactly N/k samples and each sample $x \in \mathbb{R}^d$ is independently sampled as $v_i + \xi$, where the noise $\xi \sim \mathcal{N}(0, \frac{\sigma^2}{d} I)$. To differentiate the noise terms among different samples, we denote the noise associated with sample x as ξ_x. We assume all v_i's are orthonormal; without loss of generality, we assume $v_i = e_i$ for each class i. Here, we assume the orthogonal features to facilitate the convergence analysis beyond the NTK regime, following previous works (Allen-Zhu and Li, 2020b; Ge et al., 2021).

Optimization: We initialize each entry in weight matrix W by independently sampling from Gaussian distribution $\mathcal{N}(0, \delta^2)$ and then taking the absolute value [1].

[1] In the Xavier initialization, each entry in weight matrix W is sampled from $\mathcal{N}(0, 1/d)$, so we can

Our analysis can be trivially generalized to standard Gaussian initialization (without taking absolute value) when r is an even integer. We initialize all bias terms as zeros. We use cross-entropy loss

$$L(W, b) = \sum_{i\in[k]} \sum_{x\in\mathcal{S}_i} -\log\left(\frac{\exp\left(f_i(x)\right)}{\sum_{j\in[k]}\exp\left(f_j(x)\right)}\right),$$

and run gradient flow on $\frac{k}{N}L(W, b)$ for time T. Our analysis can also be extended to gradient descent with a small step size.

Next we show that running gradient flow from a small initialization can converge to a model with zero error and constant bias gap.

Theorem 5.2. *Suppose the neural network, dataset and optimization procedure are as defined in Section 5.2. Suppose initialization scale $\delta \leqslant \Theta(1)$, noise level $\sigma \leqslant \widetilde{\Theta}(1)$, dimension $d \geqslant \widetilde{\Theta}(1/\delta^{2r-2})$ and number of samples $N \geqslant \widetilde{\Theta}(1/\delta^{r-1})$, with probability at least 0.99 in the initialization, there exists time $T = \Theta(\log(1/\delta)/\delta^{r-2})$ such that we have*

1. *zero error: for all different $i, j \in [k]$ and for all $x \in \mathcal{S}_i$, $f_i^{(T)}(x) \geqslant f_j^{(T)}(x) + \Omega(1)$;*
2. *bias gap: $b_{i^*}^{(T)} - \max_{i\neq i^*} b_i^{(T)} \geqslant \Omega(1)$ with $i^* = \arg\max_{i\in[k]} b_i^{(T)}$.*

Due to space limit, we only give a proof sketch here and leave the detailed proof in Section D.3. Since our dataset is perfectly balanced, it might seem surprising that gradient flow learns diverse biases. We can compute the time derivative on the bias, $\dot{b}_i = 1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_i(x)$, where $u_i(x)$ is the softmax output for class i, that is $\frac{\exp(f_i(x))}{\sum_{i'\in[k]}\exp(f_{i'}(x))}$. At the beginning, all logits are small, we have $u_i(x) \approx 1/k$ and $\dot{b}_i \approx 0$. If all the samples are learned at the same time, we have $u_i(x) \approx 1, u_i(x') \approx 0$ for $x \in \mathcal{S}_i, x' \in \mathcal{S}\backslash\mathcal{S}_i$, which again leads to $\dot{b}_i \approx 0$.

think of $\delta^2 = 1/d$ that is small when input dimension d is large.

On the other hand, we can consider what happens if all samples in one class (e.g., class i) are learned before any sample in any other class (e.g., class j) is learned [2]. In this case we have

$$\begin{aligned}
\dot{b}_i =& 1 - \frac{k}{N}\sum_{x\in\mathcal{S}_i} u_i(x) - \frac{k}{N}\sum_{x\in\mathcal{S}\backslash\mathcal{S}_i} u_i(x) \\
\approx& 1 - \frac{k}{N}\cdot\frac{N}{k}\cdot 1 - \frac{k}{N}\cdot\frac{N(k-1)}{k}\cdot\frac{1}{k} = -\frac{k-1}{k}, \\
\dot{b}_j =& 1 - \frac{k}{N}\sum_{x\in\mathcal{S}_i} u_j(x) - \frac{k}{N}\sum_{x\in\mathcal{S}\backslash\mathcal{S}_i} u_j(x) \\
\approx& 1 - \frac{k}{N}\cdot\frac{N}{k}\cdot 0 - \frac{k}{N}\cdot\frac{N(k-1)}{k}\cdot\frac{1}{k} = \frac{1}{k},
\end{aligned}$$

where for any learned sample $x \in \mathcal{S}_i$, we have $u_i(x) \approx 1, u_j(x) \approx 0$; for any not yet learned sample $x \in \mathcal{S}\backslash\mathcal{S}_i$, we have $u_i(x), u_j(x) \approx 1/k$. The above calculation shows that b_i starts to decrease and all the other bias terms increase. Generalizing this intuition, we show that $b_{i'}$ starts to decrease whenever class i' is learned, and the class that is learned last will have the largest bias.

As the weights are initialized randomly, by standard anti-concentration, one can argue that there is a gap between $W_{i,i}^{(0)}$'s. Without loss of generality, we assume $W_{1,1}^{(0)} > W_{2,2}^{(0)} > \cdots > W_{k,k}^{(0)}$. The initial difference in the weights will lead to different classes being learned at different time. We show that by doing induction on the following hypothesis through training:

Proposition 5.1 (Induction Hypothesis)**.** *In the same setting of Theorem 5.2, with probability at least 0.99 in initialization, there exist time points* $0 =: s_1 < t_1 < s_2 < t_2 < \cdots < s_{k-1} < t_{k-1} < s_k := T$ *with* $t_i - s_i = \Theta(\log(1/\delta)/\delta^{r-2})$ *and* $s_{i+1} - t_i = \Theta(1)$ *for* $i \in [k-1]$ *such that for any* $t \in [s_i, s_{i+1}]$,

[2] This is indeed possible since all samples of one class only differ in the noise terms in our setting. In the analysis, we can show that the noise term has negligible contribution to the network output and all samples in one class are learned almost at the same time.

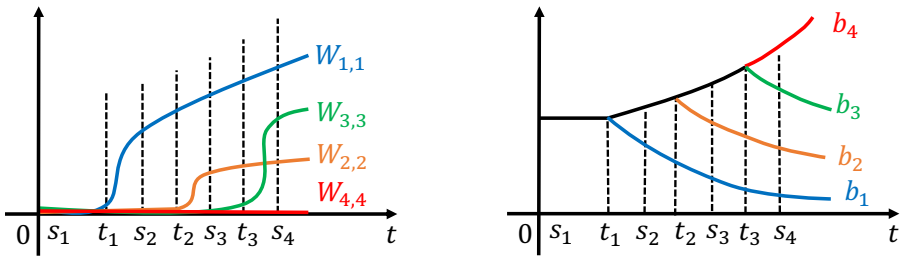

FIGURE 5.3: The training dynamics of W and b in a four-class example.

1. ***(classes not yet learned)*** *for any class* $j, j' \geqslant i+1$, *we have (1)* $b_j^{(t)} \geqslant \max_{i' \in [k]} b_{i'}^{(t)} - O(\delta^r)$, *(2)* $\left| b_j^{(t)} - b_{j'}^{(t)} \right| \leqslant O(\delta^r)$ *and (3)* $W_{j,j}^{(t)} \leqslant O(\delta)$;

2. ***(classes already learned)*** *for any class* $j \leqslant i-1$, *we have (1)* $b_j^{(t)} \leqslant \max_{i' \in [k]} b_{i'}^{(t)} - \Omega(1)$, *(2)* $f_j^{(t)}(x) \geqslant f_{i'}^{(t)}(x) + \Omega(1)$ *for* $i' \neq j, x \in \mathcal{S}_j$ *and (3)* $W_{j,j}^{(t)} \geqslant \Omega(1)$;

3. ***(parameters movement)*** *(1) for any* $j \in [k], \Theta(\delta) = W_{j,j}^{(0)} < W_{j,j}^{(t)}$, *(2) for any distinct* $j, j' \in [k], 0 < W_{j,j'}^{(t)} \leqslant O(\delta)$ *and (3) for any* $j, j' \in [k]$ *and any* $x \in \mathcal{S}_{j'}, \left| \left\langle W_{j,:}^{(t)}, \xi_x \right\rangle \right| \leqslant \min\left(O(\delta), W_{j,j'}^{(t)} \right)$.

This proposition shows that gradient flow learns k classes one by one, from class 1 to class k. More precisely, each class i is learned during time $[s_i, s_{i+1}]$. All the not yet learned classes $j \geqslant i+1$ have close to maximum biases and their weights $W_{j,j}^{(t)}$'s are small. All the already learned classes $j \leqslant i-1$ have small biases and large weights $W_{j,j}^{(t)}$'s. For the parameters movement, we know that all the diagonal entries $W_{j,j}^{(t)}$'s are larger than the initialization and all the off-diagonal entries $W_{j,j'}^{(t)}$'s are only $O(\delta)$. The correlation between the weights and noise terms also remains small.

When learning class i during time $[s_i, s_{i+1}]$, the weight $W_{i,i}^{(t)}$ slowly grows to a small constant in $[s_i, t_i]$ and then quickly grows large in $[t_i, s_{i+1}.]$ As a result, all $x \in \mathcal{S}_i$ become classified correctly. During the same time, $b_i^{(t)}$ decreases and becomes

smaller than the largest bias by at least a constant. At the end time $T = s_k$, although $W_{k,k}^{(T)}$ remains small, all $x \in \mathcal{S}_k$ are also classified correctly because $b_k^{(T)}$ is the largest bias. See an illustration of this learning process in Figure 5.3.

Although we consider a simple neural network and data distribution, the analysis for the training dynamics is still non-trivial. There are three major challenges in our proof: (1) How to ensure that class $i+1$ is learned much later than class i? (2) For any class j that has not been learned, how to maintain that its bias is close to the maximum? (3) For any learned class j, how to maintain the large bias gap from the top bias? Next, we give the proof ideas for these questions. Since all the off-diagonal entries and correlations with noise terms in $W^{(t)}$ are negligible, in our proof we can essentially focus on the movement of $W_{i,i}^{(t)}$'s and $b_i^{(t)}$'s.

Lower bounding $s_{i+1} - s_i$. During time $[s_i, t_i]$, the dynamics of $W_{i,i}^{(t)}$ is similar as in the tensor power method (Allen-Zhu and Li, 2020b; Ge et al., 2021). The initial gap between $W_{i,i}^{(0)}$ and $W_{j,j}^{(0)}$ ensures that when $W_{i,i}^{(t_i)}$ rises to a small constant, $W_{j,j}^{(t_i)}$ is still $O(\delta)$ for all $j \geqslant i+1$. Then after constant time $s_{i+1} - t_i$, $W_{j,j}^{s_{i+1}}$ is still $O(\delta)$ since the increasing rate of $W_{j,j}^{(t)}$ is merely $O(\delta^{r-1})$.

Bias for classes that are not learned. For $j \geqslant i+1$, we maintain that $b_j^{(t)} \geqslant \max_{i' \in [k]} b_{i'}^{(t)} - O(\delta^r)$. First, we use the below lemma to show biases for any two classes $j, j' \geqslant i+1$ remain close.

Lemma 5.1 (Coupling Biases). *Assuming $W_{j',j'}, W_{j,j} \leqslant O(\delta)$ and $b_{j'}, b_j \geqslant \max_{i' \in [k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j > 0$ if $b_{j'} - b_j \leqslant -\mu\delta^r$, and $\dot{b}_{j'} - \dot{b}_j < 0$ if $b_{j'} - b_j \geqslant +\mu\delta^r$ for some positive constant μ.*

Second we show that any already learned or being learned class $j' \leqslant i$ cannot have bias much larger than any class $j \geqslant i+1$ not yet learned.

Lemma 5.2 (Bias Gap Control I). *For any different $j', j \in [k]$, if $W_{j',j'} \geqslant W_{j,j}, W_{j,j} \leqslant O(\delta)$ and $b_{j'} - b_j \geqslant O(\delta^r), b_j \geqslant \max_{i' \in [k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j < 0$.*

Bias for learned classes. At time s_{j+1}, we can prove that $1 - u_j^{(s_{j+1})}(x) \leqslant C_1$ for all $x \in \mathcal{S}_j$ and $b_j^{(s_{j+1})} - b_k^{(s_{j+1})} \leqslant -C_2$. According to the below lemma, we can ensure that $b_j^{(t)} - b_k^{(t)} \leqslant -C_2$ for any $t \geqslant s_{j+1}$.

Lemma 5.3 (Bias Gap Control II)**.** *There exist small positive constants C_1, C_2 such that for any $j \in [k-1]$ and any $x \in \mathcal{S}_j$, if $1 - u_j(x) \leqslant C_1, W_{k,k} \leqslant O(\delta)$ and $b_j - b_k \geqslant -C_2$, we have $\dot{b}_j - \dot{b}_k < -\Omega(1)$.*

5.4.1 Plateau and monotonicity for r-homogeneous-weight network

Now assuming the network at initialization and after training satisfies the properties described in Theorem 5.2 and Proposition 5.1, we can prove a tighter bound on the plateau region and also show the monotonicity in error and loss curve. See the complete proofs in Appendix D.2.1 and Appendix D.2.2.

Same as in Assumption 5.1, we use Δ_i to denote the bias gap $b_k^{(T)} - b_i^{(T)}$ for $i \in [k-1]$ and denote $\Delta_{\min} := \min_{i \in [k-1]} \Delta_i$ and $\Delta_{\max} = \max_{i \in [k-1]} \Delta_i$. For the weights, we denote $W_{\min} = \min_{i \in [k-1]} W_{i,i}^{(T)}$ and $W_{\max} = \max_{i \in [k]} W_{i,i}^{(T)}$. We denote $R_{\min} = \min_{i \in [k-1]} \Delta_i / [W_{i,i}^{(T)}]^r$, $R_{\max} = \max_{i \in [k-1]} \Delta_i / [W_{i,i}^{(T)}]^r$. Below, we show the plateau and monotonicity of loss and error interpolations in Theorem 5.3.

Theorem 5.3. *Suppose the neural network, dataset and optimization procedure are as defined in Section 5.2. Suppose the network at initialization and after training satisfies the properties described in Theorem 5.2 and Proposition 5.1. For any $\epsilon \in (0,1)$, suppose $\delta \leqslant \min(\Theta(\epsilon^{1/r}), \Theta(R_{\min}^{\frac{1}{r-1}} \Delta_{\min}^{1/r}), \Theta((\frac{W_{\min}}{W_{\max}})^{\frac{2r}{r-2}}))$. There exist $\alpha_1 = \frac{\delta}{\Delta_{\min}}, \alpha_2 = (\frac{1}{1+O(\sqrt{\delta})})^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}}, \alpha_3 = \frac{\epsilon^{1/r}}{W_{\max}}$ and $\alpha_4 = (1+O(\delta))^{\frac{1}{r-1}} \left(\frac{R_{\max}}{r}\right)^{\frac{1}{r-1}}$ such that*

1. *for all $\alpha \in [\alpha_1, \alpha_2]$, the error is $1 - 1/k$; for all $\alpha \in [\alpha_1, 1]$, the error is non-increasing;*

2. *for all* $\alpha \in [0, \alpha_3]$*, we have* $\log k - e\epsilon \leqslant \frac{1}{N} L(W^{[\alpha]}, b^{[\alpha]}) \leqslant \log k + \alpha\Delta_{\max} + e\epsilon$*; for all* $\alpha \in [\alpha_4, 1]$*, the loss is monotonically decreasing.*

For the error plateau, we prove a tighter bound on the right boundary α_2 than in Theorem 5.1. We also show the error is non-increasing for $\alpha \in [\delta/\Delta_{\min}, 1]$ by arguing that once a sample is correctly classified at interpolated point $\alpha' \geqslant \delta/\Delta_{\min}$, it will remain so for any $\alpha \geqslant \alpha'$. Similar as in Theorem 5.1, we can show that the loss is no smaller than $\log k - e\epsilon$ when $\alpha \leqslant \frac{\epsilon^{1/r}}{W_{\max}}$. To show the monotonicity of loss after α_4, we show that $f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)$ is increasing in α for $i \neq j$ and $x \in \mathcal{S}_i$.

In summary, for $\alpha \in [\alpha_1, \alpha_2]$, the signal is smaller than the bias gap and the error remains at $1 - 1/k$. Before α_3, the signal is very small and the loss remains large; after α_4, the signal starts to overcome the bias gap and the loss is decreasing. See an illustration in Figure 5.2 (right).

5.5 Experiments

In this section we empirically show that intuitions from our simple theoretical model can also be applied to more realistic datasets and architectures. First, we show that bias plays an important role in creating the plateau in the error interpolation, as predicted by Theorem 5.1. We then demonstrate the influence of initialization size and network depth (also see Theorem 5.1). Finally, we show that similar to Proposition 5.1 the class that is learned last often has larger bias. Due to space constraint, we only show the results on MNIST and CIFAR-100 in this section, while similar results also hold on Fashion-MNIST and CIFAR-10 (see Appendix D.4).

Unless specified otherwise, we use a depth-10 and width-1024 fully-connected ReLU neural network (FCN10) for MNIST and use VGG-16 (without batch normalization) for CIFAR-100. We use Kaiming initialization (He et al., 2015) for the weights and set all bias terms as zeros. For FCN10 on MNIST, we use a small ini-

tialization by scaling the weights of each layer by $(0.001)^{1/10}$ so the output is scaled by 0.001. We train each network using SGD for 100 epochs. See more experiment settings in Appendix D.4.

We linearly interpolate using 50 evenly spaced points between the network at initialization and the network at the end of training. We evaluate error and loss on the train set. For each setting, we repeat the experiments three times from different random seeds and plot the mean and deviation.

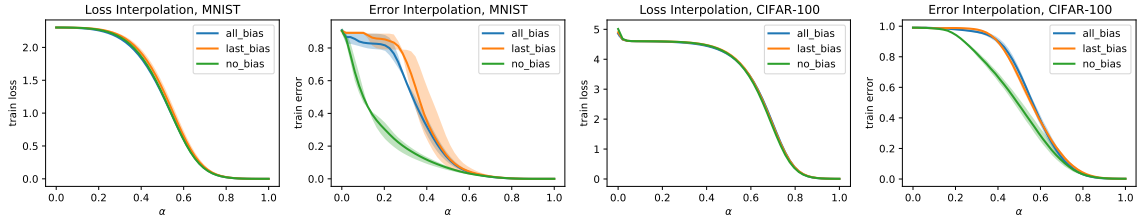

FIGURE 5.4: Loss and error curves across networks with all bias, last bias and no bias.

Role of bias in creating plateau. We demonstrate the importance of bias using two experiments. In the first experiment, we compare the loss/error interpolation curves between networks equipped with bias for all the layers (*all bias*), with bias only for the output layer (*last bias*), and with no bias at all (*no bias*). Figure 5.4 shows that networks with all bias and last bias have a much longer error plateau than networks without bias. Three bias settings have similar loss interpolation curves.

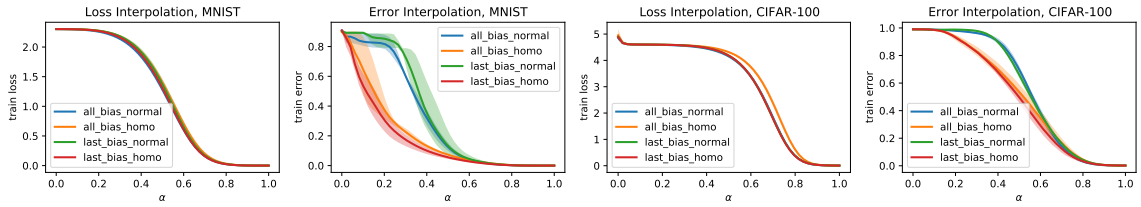

FIGURE 5.5: Loss and error curves across networks with normal and homogeneous interpolation on bias.

By our theory, the bias dominates the signal at the beginning of the interpolation because the bias term scales as α while the signal scales as α^r. In the second experiment, to correct this discrepancy, we interpolate the bias at the h-th layer (input is at the 0-th layer) as $b_h^{[\alpha]} = (1-\alpha)^h b_h^{(0)} + \alpha^h b_h^{(T)} = \alpha^h b_h^{(T)}$. We call this the *homogeneous interpolation* as now terms involving bias and weights all have α^r coefficients. We compare this with the *normal interpolation* that linearly interpolates the bias terms. Figure 5.5 shows that for networks with all bias or last bias, using homogeneous interpolation can significantly reduce the plateau in the error interpolation, but does not affect the loss interpolation.

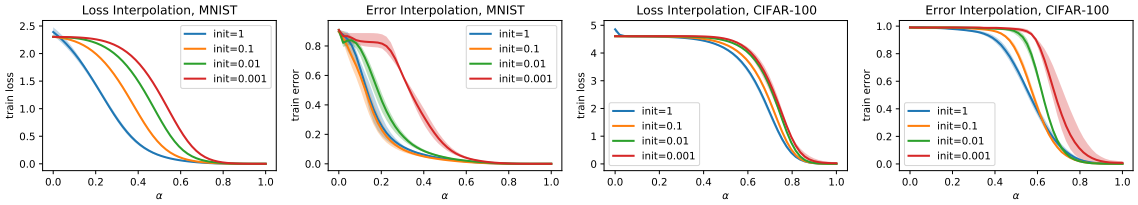

FIGURE 5.6: Loss and error curves across networks with different initialization scales.

Role of initialization scale and network depth. Our theory suggests that with a smaller initialization, the signal magnitude at the initial interpolation is smaller, which can create longer plateau in both loss interpolation and error interpolation. We compare networks under initialization scales $1, 0.1, 0.01$ and 0.001, where scale 1 corresponds to the standard Kaiming initialization. For other initialization β, we rescale each layer by the same factor so the output is rescaled by β. According to Figure 5.6, smaller initialization does create longer plateau in loss and error interpolation.

With a deeper network, the signal grows slower at the initial interpolation phase, which can potentially create a longer plateau in both loss interpolation and error interpolation. We compare FCN4, FCN6, FCN8, FCN10 on MNIST and compare VGG11, VGG13, VGG16, VGG19 on CIFAR-100. According to Figure 5.7, deeper networks do have longer plateau in loss and error interpolation.

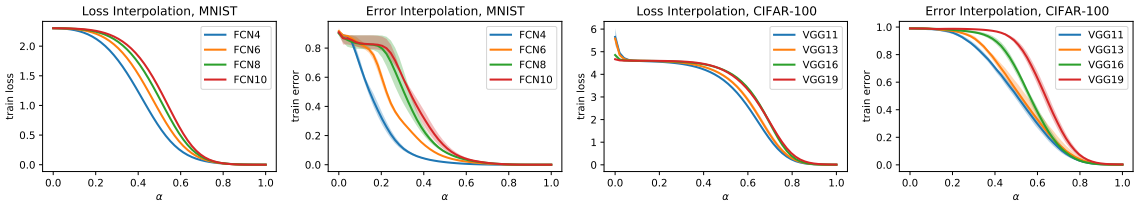

FIGURE 5.7: Loss and error curves across networks with different depths.

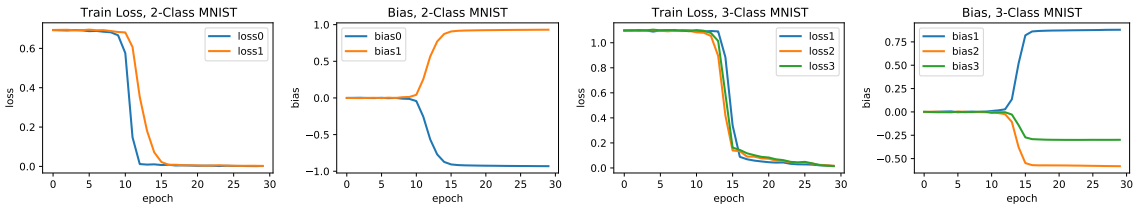

FIGURE 5.8: Train loss for each class and bias term dynamics on 2-class MNIST and 3-class MNIST.

Bias learning dynamics. Our dynamics analysis in Section 5.4 shows that gradient descent can learn diverse biases on a balanced dataset by learning different classes at different time points. In particular, the last learned class should have the highest bias term. We verify this theory by studying FCN10 with only output bias on balanced 2-class or 3-class MNIST. To separate the learning of different classes, we compute the per-class loss by only considering the examples in that particular class. According to Figure 5.8, in the 2-class MNIST, number 1 is learned last and its bias is larger, which fits our theory. Also in the 3-class MNIST, class 2 is learned first, class 3 the second and class 1 the last; for the learned bias, class 2 bias is smallest, class 3 bias in the middle and class 1 bias the highest.

5.6 Conclusion

Our theory suggests that the plateau in loss/error interpolation curves may be attributed to simple reasons, and it's unclear if these reasons are related to the difficulty/easiness of optimization. In our experiments although the training succeeds in all the settings, the loss and error interpolation curves can be easily manipulated

by changing the initialization size, network depth and bias terms. Therefore, we believe one needs to look at structures more complicated than linear interpolation to understand why optimization succeeds for deep neural networks.

Though our theory requires a small initialization, we also observe plateau in CIFAR-100 with standard initialization, which suggests that the useful signal is still a high order term in α. We also observe that sometimes the ordering of the biases does not exactly follow the ordering of the learning. We believe this is partially due to the correlation between different-class features and offer a preliminary explanation in Appendix D.4.5. We leave the thorough study of these problems in the future work.

6

Conclusion

Analyzing the training dynamics of gradient descent is necessary for explaining many phenomena in deep learning. In this thesis, we developed the analysis techniques of training dynamics in over-parameterized tensor decompositions. We also applied training dynamics analysis to explain the phenomena in non-contrastive self-supervised learning and monotonic linear interpolation.

Though significant progress has been achieved, we still can only characterize the training dynamics in very restricted settings, such as deep linear networks and shallow neural networks. Many insights can be obtained by analyzing these simple networks, yet some phenomena happen only on complicated non-linear networks. To build a solid theoretical foundation for deep learning, we will need to develop techniques to analyze the training of deep non-linear networks.

Appendix A

Supplementary materials for Chapter 2

Notations

Besides the notations defined in Section 2.1.2, we also use the following notations in the proofs.

We use $\odot$ for the Khatri-Rao product. We denote e_i as the i-th basis vector in $\mathbb{R}^d$.

We define $\mathrm{mat}(\cdot)$ to be the matrixize operator for tensors, mapping a tensor in $(\mathbb{R}^d)^{\otimes l}$ to a matrix in $\mathbb{R} \times \mathbb{R}^{d^{l-1}}$: $\mathrm{mat}(T)_{i_1,(i_2-1)d^{l-2}+\cdots+(i_{l-1}-1)d+i_l} := T_{i_1,i_2,\cdots,i_l}$ for any $i_1, i_2, \cdots, i_l \in [d]$.

We view a tensor $T \in (\mathbb{R}^d)^{\otimes l}$ as a multilinear form. For matrices $M_1 \in \mathbb{R}^{d\times k_1}, \cdots, M_l \in \mathbb{R}^{d\times k_l}$, the tensor $T(M_1, M_2, \cdots, M_l) \in \mathbb{R}^{k_1\times k_2\times\cdots\times k_l}$ is defined such that $T(M_1, M_2, \cdots, M_l)_{j_1,\cdots,j_l} := \sum_{i_1,\cdots,i_l\in[d]} T_{i_1,\cdots,i_l}(M_1)_{i_1,j_1}\cdots(M_l)_{i_l,j_l}$, for any $j_1 \in [k_1], \cdots, j_l \in [k_l]$. For notation simplicity, we use $T(M^{\otimes k}, M_{k+1}, \cdots, M_l)$ to denote $T(M, M, \cdots, M, M_{k+1}, \cdots, M_l)$. In particular, for any $v \in \mathbb{R}^d$, $T(v^{\otimes l})$ is a scalar equals to $\left\langle T, v^{\otimes l}\right\rangle = \sum_{i_1,\cdots,i_l\in[d]} T_{i_1,\cdots,i_l} v_{i_1} v_{i_2}\cdots v_{i_l}$.

A.1 Lower Bound for the Number of Components Needed for Kernels

In this section, we will prove that a lazy training model requires $\Omega(d^{l-1})$ components to fit a random rank-one tensor with $o(1)$ loss. Recall Theorem 2.1 as follows.

Theorem 2.1. *Suppose the ground truth tensor $T^* = [u^*]^{\otimes l}$, where u^* is uniformly sampled from the unit sphere $\mathbb{S}^{d-1}$. Lazy training (defined as below) requires $\Omega(d^{l-1})$ components to achieve $o(1)$ error in expectation.*

Recall that in our definition, a lazy training model can only capture tensors in the linear subspace $S_U = \text{span}\{P_{sym}\text{vec}(u_i^{\otimes l-1} \otimes \delta_i)\}_{i=1}^m$ (here P_{sym} is the projection to the space of vectorized symmetric tensors, δ_i's are arbitrary vectors in $\mathbb{R}^d$). The dimension of this subspace is upperbounded by dm. Let W_l be the space of all vectorized symmetric tensors in $(\mathbb{R}^d)^{\otimes l}$, and $S_U^\perp$ be the subspace of W_l orthogonal to S_U. We only need to show that for a random rank-one tensor, in expectation its projection on the orthogonal subspace $S_U^\perp$ is at least a constant unless $m = \Omega(d^{l-1})$. In the following lemma, we first lower bound the projection of the ground truth tensor on a fixed direction. The proof of Lemma A.1 is deferred into Section A.1.2.

Lemma A.1. *Let $u \in \mathbb{R}^d$ be a vector sampled uniformly on the unit sphere $\mathbb{S}^{d-1}$. For any vectorized symmetric l-th order tensor $b \in \mathbb{R}^{d^l}$ with unit ℓ_2 norm, we have*

$$b^\top \mathbb{E}[vec(u^{\otimes l})vec(u^{\otimes l})^\top]b \geqslant \frac{\Gamma\left(\frac{d}{2}\right)}{2^l\Gamma\left(l+\frac{d}{2}\right)}l!,$$

where $\Gamma(\cdot)$ is the Gamma function.

Next, we lower bound the projection of $\text{vec}(T^*)$ on subspace $S_U^\perp$ by summation up the projections on the subspace bases, each of which can be bounded by Lemma A.1. We give the proof of Theorem 2.1 as follows.

Proof of Theorem 2.1. Recall that W_l is the space of all vectorized symmetric tensors in $(\mathbb{R}^d)^{\otimes l}$. Due to the symmetry, the dimension of W_l is $\binom{d+l-1}{l}$. Since

the dimension of S_U is at most dm, we know that the dimension of $S_U^\perp$ is at least $\binom{d+l-1}{l} - dm$. Assuming $S_U^\perp$ is an $\bar{m}$-dimensional space, we have $\bar{m} \geqslant \binom{d+l-1}{l} - dm \geqslant \frac{d^l}{l!} - dm$. Let $\{e_1, \cdots, e_{\bar{m}}\}$ be a set of orthonormal bases of $S_U^\perp$, and $\Pi_U^\perp$ be the projection matrix from $\mathbb{R}^{d^l}$ onto $S_U^\perp$, then we know that the smallest possible error that we can get given U is

$$\frac{1}{2}\mathbb{E}_{u^*}\left[\left\|\Pi_U^\perp \mathrm{vec}(T^*)\right\|_F^2\right] = \frac{1}{2}\mathbb{E}_{u^*}\left[\sum_{i=1}^{\bar{m}} \langle \mathrm{vec}(T^*), e_i\rangle^2\right] = \frac{1}{2}\sum_{i=1}^{\bar{m}} \mathbb{E}_{u^*}\left[\langle \mathrm{vec}(T^*), e_i\rangle^2\right],$$

where the expectation is taken over $u^* \sim \mathrm{Unif}(\mathbb{S}^{d-1})$.

By Lemma A.1, we know that for any $i \in [m]$,

$$\begin{aligned}\mathbb{E}_{u^*}\left[\langle \mathrm{vec}(T^*), e_i\rangle^2\right] =& e_i^\top \mathbb{E}_{u^*}[(\mathrm{vec}([u^*]^{\otimes l})\mathrm{vec}([u^*]^{\otimes l})^\top] e_i \\ \geqslant& \frac{\Gamma\left(\frac{d}{2}\right)}{2^l \Gamma\left(l+\frac{d}{2}\right)} l! \geqslant \mu \frac{l!}{d^l},\end{aligned}$$

where μ is a positive constant only related to l.

Therefore,

$$\frac{1}{2}\mathbb{E}_{u^*}\left[\left\|\Pi_U^\perp T^*\right\|_F^2\right] = \frac{1}{2}\sum_{i=1}^{\bar{m}} \mathbb{E}_{u^*}\left[\langle \mathrm{vec}(T^*), e_i\rangle^2\right] \geqslant \left(\frac{d^l}{l!} - dm\right)\frac{\mu l!}{2d^l} = \frac{\mu}{2} - \frac{\mu l!}{2}\cdot\frac{m}{d^{l-1}}.$$

Note that we assume l is a constant. If $m = o(d^{l-1})$, i.e., $\frac{m}{d^{l-1}} = o(1)$, then the expectation of the smallest possible error is at least some constant. Thus, if we want the error to be $o(1)$, we must have $m = \Omega(d^{l-1})$. This finishes the proof of Theorem 2.1. □

A.1.1 Numerical verification of the lower bound

In this section, we plot the projection of the ground truth tensor on the orthogonal subspace $\mathbb{E}_{u^*}\left\|\Pi_U^\perp T^*\right\|_F^2$ under different dimension d and number of components m. For convenience, we only plot the lower bound for the projection that is $\left(\binom{d+l-1}{l} - dm\right)\frac{\Gamma\left(\frac{d}{2}\right)}{2^l\Gamma\left(l+\frac{d}{2}\right)} l!$ as we derived previously.

Figure A.1 shows that under different dimensions, $\mathbb{E}_{u*}\left\|\Pi_U^{\perp}T^*\right\|_F^2$ is at least a constant until $\log_d m$ gets close to $l-1=3$. As dimension d increases, the threshold when the orthogonal projection significantly drops becomes closer to 3.

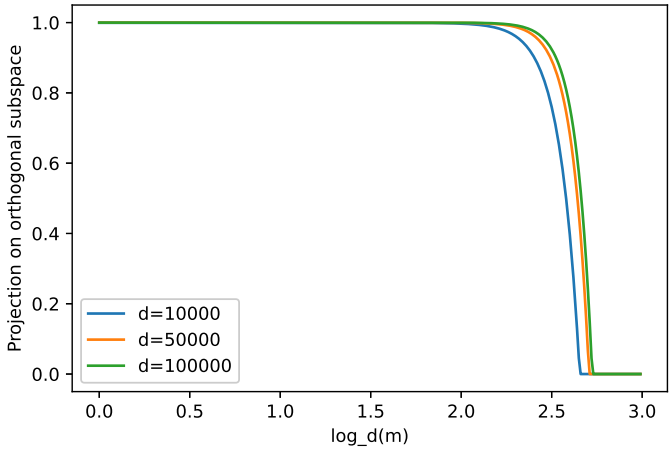

FIGURE A.1: The projection of the ground truth tensor on the orthogonal subspace when $l=4$.

A.1.2 Proofs of technical lemmas

To prove Lemma A.1, we need another technical lemma, which is stated and proved below.

Lemma A.2. *Let $u\in\mathbb{R}^d$ be a standard normal vector. For any vectorized symmetric l-th order tensor $b\in\mathbb{R}^{d^l}$ with unit norm, we have*

$$b^\top\mathbb{E}[vec(u^{\otimes l})vec(u^{\otimes l})^\top]b\geqslant l!.$$

Proof of Lemma A.2. First we define the notion of symmetry: For a vector $x\in\mathbb{R}^{d^B}$, where $B\in\mathbb{N}^*$, if for all permutation σ of $[d]$, and for all $i_1,\cdots,i_B\in[d]$, $x_{i_1i_2\cdots i_B}=x_{\sigma(i_1)\sigma(i_2)\cdots\sigma(i_B)}$, then we say vector x is symmetric.

Besides, for a vector $v\in\mathbb{R}^{d^l}$, we use $v_{i_1,i_2,\cdots,i_l}$ to refer to $\text{Tensor}(v)_{i_1,i_2,\cdots,i_l}$ where Tensor is the inverse translation of vec, i.e., Tensor translates a $\mathbb{R}^{d^l}$ vector back to a

$\mathbb{R}^{d^{\otimes l}}$ tensor. In other words, we use $v_{i_1,i_2,\cdots,i_l}$ to refer to the entry $v_{(i_1-1)d^{l-1}+(i_2-1)d^{l-2}+\cdots+(i_{l-1}-1)d+i_l}$. Similarly, for a $d^l \times d^l$ matrix M, we use $M_{i_1,i_2,\cdots,i_l,j_1,j_2,\cdots,j_l}$ to denote $\text{Tensor}(M)_{i_1,i_2,\cdots,i_l,j_1,j_2,\cdots,j_l}$, or in other words, $M_{(i_1-1)d^{l-1}+(i_2-1)d^{l-2}+\cdots+(i_{l-1}-1)d+i_l,(j_1-1)d^{l-1}+(j_2-1)d^{l-2}+\cdots+(j_{l-1}-1)d+j_l}$.

Assume that $v = u^{\otimes l}$, then $v \in \mathbb{R}^{d^{\otimes l}}$, and v is a symmetric tensor. Note that

$$\forall i_1, \cdots, i_l \in [d], v_{i_1,\cdots,i_l} = u_{i_1} \cdots u_{i_l}.$$

Define $M \triangleq \text{vec}(v)\text{vec}(v)^\top = \text{vec}(u^{\otimes l})\text{vec}(u^{\otimes l})^\top$, then

$$M_{i_1,\cdots,i_l,j_1,\cdots,j_l} = u_{i_1} \cdots u_{i_l} \cdot u_{j_1} \cdots u_{j_l}.$$

By Wick's theorem, we know that

$$\mathbb{E}[M_{i_1,\cdots,i_l,j_1,\cdots,j_l}] = \sum_{\sigma \in P} \prod_{t \in [l]} \mathbb{E}[u_{\sigma(2t-1)} u_{\sigma(2t)}],$$

where P is the set that containing all distinct partitions of $S = \{i_1, \cdots, i_l, j_1, \cdots, j_l\}$ into l pairs. Each two variables in S are considered different even if the values of them are the same, e.g., $\{(i_1, i_2), (j_1, j_2)\}$ and $\{(i_1, j_2), (j_1, i_2)\}$ are different partitions even if $i_2 = j_2$. In other words, the partition is independent of the value of those variables. Thus, we can decompose matrix M into the sum of $(2l-1)!!$ matrices, i.e.,

$$M = \sum_{\sigma \in P} M_\sigma.$$

Assume σ_1 is the partition $\{(i_1, j_1), \cdots, (i_l, j_l)\}$, so

$$\mathbb{E}[M_{\sigma_1}] = \prod_{t \in [l]} \mathbb{E}[u_{i_t} u_{j_t}].$$

Since $\mathbb{E}[u_{i_t} u_{j_t}] = \mathbb{I}\{i_t = j_t\}$($\mathbb{I}$ is the indicator function), we know that all elements on the diagonal of $\mathbb{E}[M_{\sigma_1}]$ are 1, and all other elements are 0, which means that $\mathbb{E}[M_{\sigma_1}]$ is the identity matrix. Hence,

$$b^\top \mathbb{E}[M_{\sigma_1}] b = 1.$$

Note that b is a symmetric vector, meaning $b^\top \mathbb{E}[M_{\sigma_1}]b$ doesn't change if we permute $\{i_1, \cdots, i_l\}$. Thus, as long as each i is paired with a j in σ, we will have $b^\top \mathbb{E}[M_{\sigma_1}]b = b^\top \mathbb{E}[M_\sigma]b$. There are $n!$ such partitions, so summing them up gives us $l!$..

For any other partition σ, we can always permute $\{i_1, \cdots, i_l\}$ and $\{j_1, \cdots, j_l\}$ such that the partition becomes

$$\{(i_1, i_2), \cdots, (i_{2t-1}, i_{2t}), (j_1, j_2), \cdots, (j_{2t-1}, j_{2t}), (i_{2t+1}, j_{2t+1}), \cdots, (i_l, j_l)\}$$

. Then

$$\mathbb{E}[M_\sigma] = I_{d^{l-2t}} \otimes (ww^\top),$$

where $w \in \mathbb{R}^{d^{2t}}$ and $w_{i_1,\cdots,i_{2t}} = \mathbb{I}\{i_1 = i_2\} \cdots \mathbb{I}\{i_{2t-1} = i_{2t}\}$. Therefore, $\mathbb{E}[M_\sigma]$ is a positive semi-definite matrix, i.e., $b^\top \mathbb{E}[M_\sigma]b \geqslant 0$.

In a word, we can divide $\mathbb{E}[M]$ into the sum of two sets of matrices. In the symmetric sense, the first set of matrices are equivalent to identity matrices while the second set of matrices are equivalent to some semi-definite matrices. Therefore,

$$b^\top \mathbb{E}[\mathrm{vec}(u^{\otimes l})\mathrm{vec}(u^{\otimes l})^\top]b \geqslant l!.$$

□

Proof of Lemma A.1. Let $u \in \mathbb{R}^d$ be a standard normal vector, i.e., $u \sim \mathbb{N}(0, I_d)$, then from Theorem 2 in Vignat and Bhatnagar (2008) we know that

$$b^\top \mathbb{E}\left[\mathrm{vec}\left((u/\|u\|)^{\otimes l}\right)\mathrm{vec}\left((u/\|u\|)^{\otimes l}\right)^\top\right]b = \frac{\Gamma\left(\frac{d}{2}\right)}{2^l\Gamma\left(l+\frac{d}{2}\right)} b^\top \mathbb{E}[\mathrm{vec}(u^{\otimes l})\mathrm{vec}(u^{\otimes l})^\top]b.$$

Furthermore, from Lemma A.2 we know that

$$^\top \mathbb{E}[\mathrm{vec}(u^{\otimes l})\mathrm{vec}(u^{\otimes l})^\top]b \geqslant l!.$$

Note that $u/\|u\|$ is distributed as a uniform vector from the unit sphere $\mathbb{S}^{d-1}$. Combining the above equality and inequality, we finish the proof of this lemma. □

A.2 Construction of Bad Local Minimum

In this section, we construct a bad local min for the vanilla loss function with vanilla parameterization of T. That is, $T := \sum_{i=1}^{m} c_i u_i^{\otimes l}$ and $f_v(U, C) = 1/2 \left\| T - T^* \right\|_F^2$. Recall Theorem 2.2 as follows.

Theorem 2.2. *Let $f_v(U, C)$ be as defined in Equation 2.2. Assume $l \geqslant 3, d > r \geqslant 1$ and $m \geqslant r(l+1)+1$. There exists a symmetric ground truth tensor T^* with rank at most $r(l+1)+1$ such that a local minimum with function value $l(l-1)r/4$ exists while the global minimum has function value zero.*

In our example, the model fits one direction in T^* but misses all the other directions. Moving any component towards one of the missing directions would actually make the approximation worse because of the cross terms. The proof of Theorem 2.2 is in Section A.2.1.

We also extend the local min to the vanilla loss function with 2-homogeneous parameterization of T. That is, $T := \sum_{i=1}^{m} a_i c_i^{l-2} u_i^{\otimes l}$ and $f(U, C) = 1/2 \left\| T - T^* \right\|_F^2$. For simplicity, we assume half of the a_i's are 1's.

Theorem A.1. *Let $f(U, C) := 1/2 \left\| \sum_{i=1}^{m} a_i c_i^{l-2} u_i^{\otimes l} - T^* \right\|_F^2$. Assume $\lfloor m/2 \rfloor$ of a_i's are 1's and the remaining are -1's. Assume $l \geqslant 3, d-2 \geqslant r \geqslant 1$ and $m \geqslant 4r(l+1)+2$. There exists a symmetric ground truth tensor T^* with rank at most $2r(l+1)+2$ such that a local minimum with function value $l(l-1)r/2$ exists while the global minimum has function value zero.*

In the above Theorem, we treat c_i's as separate variables from u_i's. That is, at a local min (U, C), we allow arbitrary perturbations to c_i's regardless of the perturbations to u_i's and show none of these perturbations can decrease the function value. Note our result trivially extends to the case when $c_i = 1/\left\| u_i \right\|$ since the coupling be-

tween c_i and u_i only restricts the set of possible perturbations to (U, C). The detailed proof of Theorem A.1 is in Section A.2.1.

A.2.1 Detailed Proofs

Proof of Theorem 2.2. In this proof, we first construct a ground truth tensor and a local min with non-zero loss. To further prove this local min is indeed spurious, we show there exists a global min with zero loss under the same ground truth tensor.

We first define the local min. For every $i \in [m]$, let c_i be 1 and u_i be $e_1/m^{\frac{1}{l}}$. Then, we know at this point $T = e_1^{\otimes l}$.

We define the ground truth tensor T^* by defining the residual $R := T - T^*$. The residual R is defined as the summation of $\hat{R}$ and all its permutation. We define $\hat{R}$ as follows,

$$\hat{R} := \sum_{j=2}^{r+1} e_j^{\otimes 2} \otimes e_1^{\otimes l-2}.$$

Then, R is defined as the summation of all $\binom{l}{2}$ permutations of $\hat{R}$. It's clear that R is symmetric and therefore T^* is symmetric.

Let U be a $d \times m$ matrix whose i-th row is u_i, and C be an $m \times m$ diagonal matrix with $C_{ii} = c_i, \forall i \in [m]$. Suppose we perform a local change to U and C such that $U' = U + \Delta U, C' = C + \Delta C$ and $\|\Delta U\|_F, \|\Delta C\|_F \to 0$. We prove that for any $\Delta U, \Delta C$, we have $f(U', C') \geqslant f(U, C)$. Let's first show that the gradient at U is zero, which means there is no locally first-order change on the function value.

First-order Change Let's first show the gradient of f_v w.r.t. U and C at (U, C) is zero. Here we first compute the gradient in terms of one column u_i,

$$\forall i \in [m], \nabla_{u_i} f_v(U, C) = lR(u_i^{\otimes l-1}, I)c_i = \frac{l}{m^{\frac{l-1}{l}}} R(e_1^{\otimes l-1}, I).$$

$$\forall i \in [m], \nabla_{c_i} f_v(U, C) = R(u_i^{\otimes l}) = \frac{1}{m} R(e_1^{\otimes l}).$$

In order to compute $R(e_1^{\otimes l-1}, I)$, we first consider $\hat{R}(e_1^{\otimes l-1}, I)$. We have

$$\hat{R}(e_1^{\otimes l-1}, I) = \sum_{j=2}^{r+1} e_j \langle e_j, e_1 \rangle \langle e_1, e_1 \rangle^{l-2} = 0.$$

Similarly,

$$\hat{R}(e_1^{\otimes l}) = \sum_{j=2}^{r+1} \langle e_j, e_1 \rangle^2 \langle e_1, e_1 \rangle^{l-2} = 0.$$

The computation for other permutations of $\hat{R}$ is the same. Overall, we have $\nabla f_v(U, C) = 0$.

Second-order change The second order change of $f_v(U, C)$ is as follows,

$$\frac{1}{2} \left\| \sum_{i=1}^{m} \left(c_i(\Delta u_i) \otimes u_i^{\otimes l-1} + \cdots + c_i u_i^{\otimes l-1} \otimes (\Delta u_i) \right) + (\Delta c_i) u_i^{\otimes l} \right\|_F^2$$

$$+ \sum_{i=1}^{m} \left(l(l-1) R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i) c_i + l R(u_i^{\otimes l-1}, \Delta u_i) \Delta c_i \right).$$

The first term is always non-negative, and the second term can be further computed as follows:

$$\begin{aligned} l(l-1) \sum_{i=1}^{m} R(u_i^{l-2}, \Delta u_i, \Delta u_i) =& l(l-1) \frac{1}{m^{(l-2)/l}} \sum_{i=1}^{m} R(e_1^{\otimes l-2}, \Delta u_i, \Delta u_i) \\ =& l(l-1) \frac{1}{m^{(l-2)/l}} \sum_{i=1}^{m} \sum_{j=2}^{r+1} [\Delta u_i]_j^2. \end{aligned}$$

Similar to the computations in the first-order change part, we know $R(u_i^{\otimes l-1}, \Delta u_i) = 0$. Therefore, the second-order change of $f(U, C)$ can be lower bounded by $l(l-1) \frac{1}{m^{(l-2)/l}} \sum_{i=1}^{m} \sum_{j=2}^{r+1} [\Delta u_i]_j^2$.

For any $\Delta U, \Delta C$, if there exists $i \in [m], 2 \leqslant j \leqslant r+1$ such that $[\Delta u_i]_j \neq 0$, we know the second order change is positive. Combining with the fact that the gradient is zero at (U, C), this implies the function value increases.

Otherwise, if $[\Delta u_i]_j = 0$ for all $i \in [m]$ and all $2 \leqslant j \leqslant r+1$, we know $\Delta u_i \in B$ for all $i \in [m]$ where B is the span of $\{e_1, e_k | r+2 \leqslant k \leqslant d\}$. Let the perturbed tensor be T', we know $T' - T$ lies in the $B^{\otimes l}$ subspace. Note perturbing c_i introduces changes in $e_1^{\otimes l}$ direction that is also in the $B^{\otimes l}$ subspace. This type of perturbation cannot decrease the function value because the residual R is orthogonal with $B^{\otimes l}$ subspace.

Overall, we have proved that (U, C) is a local minimizer. Notice that residual R contains $r\binom{l}{2}$ orthogonal components with unit norm. Therefore, the function value at (U, C) is $f(U, C) = \frac{1}{2}\|R\|_F^2 = \frac{1}{2} \times r \times \binom{l}{2} = \frac{l(l-1)r}{4}$.

Construction of global minimizer: Next we will show that when $m \geqslant r(l+1)+1$, there exists U and C such that $f(U, C) = 0$. Therefore, the local minimizer we found above must be a spurious local minimizer. We only need to show that T^* can be expressed as the summation of $r(l+1)+1$ rank-one symmetric tensors.

Define $\hat{R}_j := e_j^{\otimes 2} \otimes e_1^{\otimes l-2}$, and define R_j to be the sum of all $\binom{l}{2}$ permutations of $\hat{R}_j$. Then we can write T^* as

$$T^* = e_1^{\otimes l} + \sum_{j=2}^{r+1} R_j.$$

Note that R_j is a symmetric tensor with entries equal to 1 if the index of the entry has 2 j's and $(l-2)$ 1's, and entries equal to 0 otherwise. Define $v_{i,j} := e_1 + b_{i,j} e_j$ where $b_{i,j} \in \mathbb{R}$, and consider the tensor $\bar{T}_j := \sum_{i=1}^{l+1} \bar{b}_{i,j} v_{i,j}^{\otimes l}$. Then we know that $\bar{T}_j$ is also a symmetric tensor with entries equal to $\sum_{i=1}^{l+1} \bar{b}_{i,j} b_{i,j}^k$ if the index of the entry has k j's and $(l-k)$ 1's, and entries equal to 0 otherwise. Therefore, if $\forall k \in \{0, 1, \cdots, l\} \backslash \{2\}$,

$\sum_{i=1}^{l+1} \bar{b}_{i,j} b_{i,j}^k = 0$ and $\sum_{i=1}^{l+1} \bar{b}_{i,j} b_{i,j}^2 = 1$, then $\bar{T}_j = R_j$. In other words, we want

$$\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ b_{1,j} & b_{2,j} & \cdots & b_{l+1,j} \\ b_{1,j}^2 & b_{2,j}^2 & \cdots & b_{l+1,j}^2 \\ \vdots & \vdots & \vdots & \vdots \\ b_{1,j}^l & b_{2,j}^l & \cdots & b_{l+1,j}^l \end{pmatrix} \begin{pmatrix} \bar{b}_{1,j} \\ \bar{b}_{2,j} \\ \vdots \\ \bar{b}_{l+1,j} \end{pmatrix}.$$

Denote the matrix in the middle by M_j, then

$$|M_j| = \prod_{1 \leqslant s < t \leqslant l+1} (b_{s,j} - b_{t,j}).$$

Thus, as long as the $b_{i,j}$'s are mutually different, the matrix M_j will be full rank, and there must exist a set of $\bar{b}_{i,j}$'s such that the equation above holds. In other words, we have shown that there exists such $b_{i,j}$'s and $\bar{b}_{i,j}$'s that $\forall 2 \leqslant j \leqslant r+1$, $\bar{T}_j = R_j$. Therefore, we know each R_j can be expressed as the summation of $(l+1)$ rank-one symmetric tensors.

To summarize, when $m \geqslant r(l+1)+1$, we can construct T^* such that there exists a local minimum with function value $\frac{l(l-1)r}{4}$ while the global minimum has function value zero.

□

Proof of Theorem A.1. The proof is very similar as the proof of Theorem 2.2. The only difference is that in the 2-homogeneous, all the c_i's are positive and we need to rely on positive and negative a_i's to fit the ground truth tensors. We need to define a slightly different ground truth tensor and bad local min.

Same as in the proof of Theorem 2.2, we first construct a ground truth tensor and a local min with non-zero loss. To further prove this local min is indeed spurious, we show there exists a global min with zero loss under the same ground truth tensor.

We first define the local min. Let $m' = \lfloor m/2 \rfloor$. Without loss of generality, assume $a_i = 1$ for all $i \in [m']$ and $a_i = -1$ for all $i \in [m'+1, m]$. For any $i \in [m']$, let

$u_i = \sqrt{1/m'}e_1$ and $c_i = 1/\|u_i\|$. For any $i \in [m'+1, m]$, let $u_i = \sqrt{1/(m-m')}e_d$ and $c_i = 1/\|u_i\|$. With this choice of parameters, it's not hard to verify that $T = e_1^{\otimes l} - e_d^{\otimes l}$.

We define the ground truth tensor T^* by defining the residual $R := T - T^*$. The residual R is defined as the summation of $\hat{R}$ and all its permutation, where $\hat{R}$ is defined as:

$$\hat{R} := \sum_{j=2}^{r+1} \left(e_j^{\otimes 2} \otimes e_1^{\otimes l-2} - e_j^{\otimes 2} \otimes e_d^{\otimes l-2} \right).$$

Since we assume $r \leqslant d-2$, we know $r+1 \leqslant d-1$ and e_j is orthogonal with e_1, e_d for all $2 \leqslant j \leqslant r+1$. Then, R is defined as the summation of all $\binom{l}{2}$ permutations of $\hat{R}$. It's clear that R is symmetric and T^* is also symmetric.

Suppose we perform a local change to U and C such that $U' = U + \Delta U, C' = C + \Delta C$ and $\|\Delta U\|_F, \|\Delta C\|_F \to 0$, where ΔC is a diagonal matrix. We prove that for all possible $\Delta U, \Delta C$, $f(U', C') \geqslant f(U, C)$.

First-order Change Let's first show the derivative of f in terms of all u_i's and c_i's at (U, C) is zero. This means there is no first order decrease direction at (U, C).

For any $i \in [m']$, we can compute the derivative in terms of u_i and c_i:

$$\nabla_{u_i} f(U, C) = lR(u_i^{\otimes l-1}, I)c_i^{l-2} = \frac{l}{\sqrt{m'}} R(e_1^{\otimes l-1}, I),$$

$$\nabla_{c_i} f(U, C) = (l-2)R(u_i^{\otimes l})c_i^{l-3} = \frac{l-2}{(m')^{3/2}} R(e_1^{\otimes l}).$$

It's not hard to verify that $R(e_1^{\otimes l-1}, I) = 0$ and $R(e_1^{\otimes l}) = 0$ using the orthogonality between e_1 and e_j for all $2 \leqslant j \leqslant r+1$. For the same reason, we also have $\nabla_{u_i} f(U, C) = 0, \nabla_{c_i} f(U, C) = 0$ for all $i \in [m'+1, 2m]$.

Second-order change The second order change of $f(U', C')$ compared with $f(U, C)$ is as follows,

$$\frac{1}{2}\left\|\sum_{i=1}^{m} a_i \left(c_i^{l-2}\left((\Delta u_i)\otimes u_i^{\otimes l-1} + \cdots + u_i^{\otimes l-1}\otimes(\Delta u_i)\right) + (l-2)c_i^{l-3}(\Delta c_i)u_i^{\otimes l}\right)\right\|_F^2$$

$$+ \sum_{i=1}^{m} a_i \left(l(l-1)R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i)c_i^{l-2} + l(l-2)R(u_i^{\otimes l-1}, \Delta u_i)c_i^{l-3}\Delta c_i\right)$$

$$+ \sum_{i=1}^{m} a_i \left((l-2)(l-3)R(u_i^{\otimes l})c_i^{l-4}(\Delta c_i)^2\right).$$

(When $l = 3$, we do not have the c_i^{l-4} term)

The first term is always non-negative. By the previous argument, we also know $R(u_i^{\otimes l-1}, \Delta u_i) = 0$ and $R(u_i^{\otimes l}) = 0$. Therefore, the second order change is lower bounded by $\sum_{i=1}^{m} a_i l(l-1)R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i)c_i^{l-2}$. Let's first consider the components from 1 to m' for which $a_i = 1$,

$$\begin{aligned} l(l-1)a_i \sum_{i=1}^{m'} R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i)c_i^{l-2} =& l(l-1)\sum_{i=1}^{m'} R(e_1^{\otimes l-2}, \Delta u_i, \Delta u_i) \\ =& l(l-1)\sum_{i=1}^{m'}\sum_{j=2}^{r+1}[\Delta u_i]_j^2. \end{aligned}$$

For the components from $m' + 1$ to m, we have

$$\begin{aligned} l(l-1)a_i \sum_{i=m'+1}^{m} R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i)c_i^{l-2} =& l(l-1)\sum_{i=m'+1}^{m} -R(e_d^{\otimes l-2}, \Delta u_i, \Delta u_i) \\ =& l(l-1)\sum_{i=m'+1}^{m}\sum_{j=2}^{r+1}[\Delta u_i]_j^2. \end{aligned}$$

Overall, we have

$$\sum_{i=1}^{m} a_i l(l-1)R(u_i^{\otimes l-2}, \Delta u_i, \Delta u_i)c_i^{l-2} \geqslant l(l-1)\sum_{i=1}^{m}\sum_{j=2}^{r+1}[\Delta u_i]_j^2.$$

For any $\Delta U, \Delta C$, if there exists $i \in [m], 2 \leqslant j \leqslant r+1$ such that $[\Delta u_i]_j \neq 0$, we know the second order change is positive. Combining with the fact that the gradient is zero at (U, C), this implies the function value increases.

Otherwise, if $[\Delta u_i]_j = 0$ for all $i \in [m]$ and all $2 \leqslant j \leqslant r+1$, we know $\Delta u_i \in B$ for all $i \in [m]$ where B is the span of $\{e_1, e_k | r+2 \leqslant k \leqslant d\}$. Let the perturbed tensor be T', we know $T' - T$ lies in the $B^{\otimes l}$ subspace. Note perturbing c_i introduces changes in $e_1^{\otimes l}$ direction or $e_d^{\otimes l}$ direction that are also in the $B^{\otimes l}$ subspace. This type of perturbation cannot decrease the function value because the residual R is orthogonal with $B^{\otimes l}$ subspace.

Overall, we have proved that (U, C) is a local minimizer. Notice that residual R contains $2r\binom{l}{2}$ orthogonal components with unit norm. Therefore, the function value at (U, C) is $f(U, C) = \frac{1}{2}\|R\|_F^2 = \frac{1}{2} \times 2r \times \binom{l}{2} = \frac{l(l-1)r}{2}$.

Construction of global minimizer: Next, we show as long as $m \geqslant 4r(l+1)+2$, there exists parameters (U, C) such that $f(U, C) = 0$. To prove this, we first write T^* as summation of rank-one symmetric tensors.

For any $2 \leqslant j \leqslant r+1$, define $\hat{R}_{j,1} := e_j^{\otimes 2} \otimes e_1^{\otimes l-2}$ and $\hat{R}_{j,d} = e_j^{\otimes 2} \otimes e_d^{\otimes l-2}$, and define $R_{j,1}, R_{j,d}$ to be the sum of all $\binom{l}{2}$ permutations of $\hat{R}_{j,1}$ and $\hat{R}_{j,d}$ respectively. Then we can write T^* as

$$T^* = e_1^{\otimes l} - e_d^{\otimes l} - \sum_{j=2}^{r+1} R_{j,1} + \sum_{j=2}^{r+1} R_{j,d}.$$

Same as in the proof of Theorem 2.2, we can show each $R_{j,1}$ or $R_{j,d}$ can be written as the sum of $(l+1)$ rank-one symmetric tensors.

Therefore, we know the ground truth T^* can be expressed as the summation of $2 + 2r(l+1)$ rank-one symmetric tensors. For each component, we can re-scale it to make it fit the form of $a_i c_i^{l-2} u_i^{\otimes l}$, with $a_i = \pm 1$ and $c_i = 1/\|u_i\|$. In these rank-one

tensors, at most $2r(l+1)+1$ has positive (negative) a_i. So, as long as $m' \geqslant 2r(l+1)+1$ or $m \geqslant 4r(l+1)+2$ our model is able to fit this ground truth tensor and achieve zero loss.

To summarize, when $m \geqslant 4r(l+1)+2$, we can construct T^* with rank at most $2r(l+1)+2$ such that there exists a local minimum with function value $\frac{l(l-1)r}{2}$ while the global minimum has function value zero. □

A.3 Detailed Proofs of Theorem 2.3

In this section, we give the proof of Theorem 2.3. We first state a formal version of Theorem 2.3.

Theorem A.2. *Given any target accuracy $\epsilon > 0$, there exists $m = O\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right)$, $\lambda = O\left(\frac{\epsilon}{r^{0.5l}}\right)$, $\delta = O\left(\frac{\epsilon^{5l-1.5}}{d^{l-1.5}(\log(d/\epsilon))^{l+0.5}r^{2.5l^2-0.75l}}\right)$, $\eta = O\left(\frac{\epsilon^{15l-4.5}}{d^{3l-4.5}(\log(d/\epsilon))^{3l+1.5}r^{7.5l^2-2.25l}}\right)$ and $H = O\left(\frac{d^{3l-4.5}(\log(d/\epsilon))^{3l+2.5}r^{7.5l^2-1.75l}}{\epsilon^{15l-3.5}}\right)$ such that with probability at least 0.99, our algorithm finds a tensor T satisfying*

$$\|T - T^*\|_F \leqslant \epsilon,$$

within $K = O\left(\frac{r^{2l}}{\epsilon^4}\log(d/\epsilon)\right)$ epochs.

We follow the proof strategy outlined in Section 2.4.

As we discussed in Challenge 2, bad local minima exist for our loss function. Therefore, gradient descent might get stuck at a bad local minima. This issue is fixed in our algorithm by re-initializing one component at the beginning of each epoch. In Lemma 2.1, we show as long as the objective is large, there is at least a constant probability to improve the objective within one epoch. We state the formal version of Lemma 2.1 as follows. The proof of Lemma A.3 is in Section A.3.2.

Lemma A.3. *Let $(U_0', \bar{C}_0')$ and $(U_H, \bar{C}_H)$ be the parameters at the beginning of an epoch and the parameters at the end of the same epoch. For the target accuracy $\epsilon > 0$ in Theorem A.2, assume $K \leqslant \frac{\lambda m}{14}$ and $\|T_0' - T^*\|_F \geqslant \epsilon$ where T_0' is the tensor with parameters $(U_0', \bar{C}_0')$. There exists $m = O\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right)$, $\lambda = O\left(\frac{\epsilon}{r^{0.5l}}\right)$, $\delta = O\left(\frac{\epsilon^{5l-1.5}}{d^{l-1.5}(\log(d/\epsilon))^{l+0.5}r^{2.5l^2-0.75l}}\right)$, $\eta = O\left(\frac{\epsilon^{15l-4.5}}{d^{3l-4.5}(\log(d/\epsilon))^{3l+1.5}r^{7.5l^2-2.25l}}\right)$, $H = O\left(\frac{d^{3l-4.5}(\log(d/\epsilon))^{3l+2.5}r^{7.5l^2-1.75l}}{\epsilon^{15l-3.5}}\right)$, such that with probability at least $\frac{1}{6}$, we have*

$$f(U_H, \bar{C}_H) - f(U_0', \bar{C}_0') \leqslant -\Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right).$$

We compliment this lemma by showing that even if an epoch does not improve the objective, it will not increase the function value by too much. The formal version of Lemma 2.2 is as follows. We prove Lemma A.4 in Section A.3.1.

Lemma A.4. *Assume $K \leqslant \frac{\lambda m}{14}$, $\delta \leqslant \frac{\mu_1 \epsilon}{m^{\frac{3}{4}} d^{\frac{l-2}{2}} (m+K)^{\frac{l-1}{2}} \lambda^{\frac{1}{2}}}$, and $\eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} d^{\frac{l-1}{2}} (m+K)^{\frac{l-2}{2}}}$ for some constants μ_1, μ_2, and $\frac{10}{m} \leqslant \lambda \leqslant 1$. Let $(U_0', \bar{C}_0')$ and $(U_H, \bar{C}_H)$ be the parameters at the beginning of an epoch and the parameters at the end of the same epoch. Assume $f(U_0', \bar{C}_0') \geqslant \epsilon^2$, where ϵ is the target accuracy in Theorem A.2. Then we have*

$$f(U_H, \bar{C}_H) - f(U_0', \bar{C}_0') \leqslant O\left(\frac{1}{\lambda m}\right).$$

From these two lemmas, we know that in each epoch, the loss function can decrease by $\Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right)$ with probability at least $\frac{1}{6}$, and even if we fail to decrease the function value, the increase of function value is at most $O\left(\frac{1}{\lambda m}\right)$. Therefore, choosing a large enough m, the function value decrease will dominate the increase. This allows us to prove Theorem A.2.

Proof of Theorem A.2. We use a contradiction proof to show that with high probability our algorithm finds a tensor T satisfying $\|T - T^*\|_F \leqslant \epsilon$ within K epochs.

For the sake of contradiction, we assume $\|T - T^*\|_F > \epsilon$ through the first K epochs. Under this assumption, we show with high probability the function value will decrease below zero.

Note that under the choice of parameters of this theorem, all the conditions of Lemma A.3 and Lemma A.4 are satisfied. By Lemma A.3, we know that with probability at least 1/6, the function value decreases by at least $\Lambda := \Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right)$ in each epoch. By Lemma A.4, we show that the function value at most increases by $\Lambda' := O(\frac{1}{\lambda m})$ in each epoch. Using our choice of the parameters in Theorem A.2, we know that $O(\frac{1}{\lambda m}) = O\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right)$. Choosing a large enough constant factor for m ensures that $\Lambda' \leqslant \frac{\Lambda}{10}$.

For each $1 \leqslant k \leqslant K$, let $\mathcal{E}_k$ be the event that in the beginning of the k-th epoch, the reinitialized component $ac_0^{l-2}u_0^l$ has good correlation with the residual (see Lemma A.9) and $\|P_S u_0\| \geqslant \frac{\mu\delta}{\sqrt{d}}$, where μ is some constant. We know $\mathcal{E}_k$'s are independent with each other and $\Pr[\mathcal{E}_k] \geqslant 1/6$. By Hoeffding's inequality, we know as long as $K \geqslant \mu'$ for certain constant μ', we have $\sum_{k=1}^K \mathbb{1}_{\mathcal{E}_k} \geqslant K/7$ with probability at least 0.99, where $\mathbb{1}_{\mathcal{E}_k}$ is the indicator function of event $\mathcal{E}_k$.

By the proof of Lemma A.3, we know conditioning on $\mathcal{E}_k$, the function value decreases by at least Λ in the k-th epoch. Since $\sum_{k=1}^K \mathbb{1}_{\mathcal{E}_k} \geqslant K/7$, we know the total function value decrease is at least $K\Lambda/7 - K\Lambda/10 = K\Omega\left(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)}\right)$. Therefore, there exists $K = O\left(\frac{r^{2l}\log(d/\epsilon)}{\epsilon^4}\right)$ such that $K\Lambda/7 - K\Lambda/10 \geqslant 4$.

By the analysis in Lemma A.7, the function value is upper bounded by 3 at initialization. However, with probability at least 0.99, the decrease of the function value is at least 4, meaning that the function value must be negative, which is a contradiction. Therefore, we know that with probability at least 0.99, our algorithm finds a tensor T satisfying $\|T - T^*\|_F \leqslant \epsilon$ within $K = O\left(\frac{r^{2l}\log(d/\epsilon)}{\epsilon^4}\right)$ epochs. □

A.3.1 Upper bound on function increase

In this section, we prove Lemma A.4.

To prove the increase of f is bounded in one epoch, we identify all the possible ways that the loss can increase and upper bound each of them. We first show that a normal step (without re-initialization or scalar mode switch) of the algorithm will not increase the objective function. Note that many parts of our proofs rely on an upperbound on function value. To get such a bound the proof includes an induction component: when we prove Lemma A.5 and Lemma A.6, we assume that the function value is upper bounded by a constant, and we will inductively prove that these conditions are satisfied in Lemma A.7. This induction ensures that the conclusions of all the lemmas in this section hold throughout the entire algorithm.

The following lemma is a formal version of Lemma 2.3 in the main text.

Lemma A.5. *Let $(U, \bar{C})$ be the parameters at the beginning of one iteration and let $U', \bar{C}'$ be the updated parameters (before potential scalar mode switch). Assuming $f(U, \bar{C}) \leqslant 10$, $\lambda \leqslant 1$, there exists constants μ_1, μ_2 such that*

$$f(U', \bar{C}') - f(U, \bar{C}) \leqslant -\frac{\eta}{l} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2$$

as long as $\delta \leqslant \frac{\mu_1}{m^{\frac{1}{4}} \sqrt{\lambda} d^{\frac{l-2}{2}} (m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} d^{\frac{l-1}{2}} (m+K)^{\frac{l-2}{2}}}$.

Recall that in an iteration, we first update U by gradient descent, then update C and $\hat{C}$ by the updated value of U. The gradient descent step on U cannot increase the function value as long as the step size is small enough. The update on C and $\hat{C}$ can potentially increase the function value. In the proof of Lemma A.5, we show the increase due to updating C and $\hat{C}$ is proportional to the decrease by updating U and smaller in scale.

Proof of Lemma A.5.

According to the algorithm, each iteration contains two steps: update U as $U' \leftarrow U - \eta \nabla_U f(U, \bar{C})$; update c_i and $\hat{c}_i$ as $c_i' = c_i \frac{\|u_i\|}{\|u_i'\|}$ and $\hat{c}_i' = \hat{c}_i \frac{\|u_i\|}{\|u_i'\|}$. We can divide the function value change into these two steps: $f(U', \bar{C}') - f(U, \bar{C}) = (f(U', \bar{C}) - f(U, \bar{C})) + (f(U', \bar{C}') - f(U', \bar{C}))$. We will show that the function value decrease in the first step and does not increase by too much in the second step. At the end, we will combine them to show that overall the function value decreases.

Since we assume $f(U, \bar{C}) \leqslant 10$. According to the definition of the loss function, we know $\|T - T^*\|_F \leqslant \sqrt{20}, \sum_{i=1}^m \|u_i\|^2 \leqslant \frac{10}{\lambda}$. We also know that $\sum_{i=1}^m \|u_i\|^4 \leqslant \left(\sum_{i=1}^m \|u_i\|^2\right)^2 \leqslant \frac{100}{\lambda^2}$. For convenience, denote $\Gamma = 10, M_4^2 := \frac{100}{\lambda^2}$ and $M_2^2 := \frac{10}{\lambda}$.

$f(U', \bar{C}) - f(U, \bar{C})$ is negative: In the first step, we update U by gradient descent, which should decrease the function value as long as we choose the step size to be small enough. To prove that an inverse polynomially step size suffices, we need to bound the second derivative of f in terms of U at $(U'', \bar{C})$ for any $U'' \in \{(1-\theta)U + \theta U' | 0 \leqslant \theta \leqslant 1\}$. Let $\mathcal{H}''$ be the Hessian of f in terms of U at $(U'', \bar{C})$. We will bound the Frobenius norm of $\mathcal{H}''$.

Let's first show that $\|u_i''\| \leqslant (1 + 1/(4l)) \|u_i\|$ when η is small enough. Recall the derivative in u_i is,

$$\nabla_{u_i} f(U, \bar{C}) = l(T - T^*)(u_i^{\otimes(l-1)}, I) c_i^{l-2} a_i + \lambda l u_i.$$

Therefore, we can bound the derivative as

$$\left\|\nabla_{u_i} f(U, \bar{C})\right\| \leqslant \left(l\sqrt{2\Gamma}(\sqrt{d(m+K)})^{l-2} + \lambda l\right) \|u_i\|.$$

Thus, as long as $\eta \leqslant \frac{1}{4l^2(\sqrt{2\Gamma}(\sqrt{d(m+K)})^{l-2}+\lambda)}$, we have

$$\eta \left\|\nabla_{u_i} f(U, \bar{C})\right\| \leqslant \eta \left(l\sqrt{2\Gamma}(\sqrt{d(m+K)})^{l-2} + \lambda l\right) \|u_i\| \leqslant \frac{1}{4l} \|u_i\|.$$

Since $u_i'' = u_i - \theta\eta\nabla_{u_i} f(U, \bar{C})$ for $0 \leqslant \theta \leqslant 1$, we know that

$$\begin{aligned}\|u_i''\| \leqslant & \|u_i\| + \eta \left\|\nabla_{u_i} f(U, \bar{C})\right\| \\ \leqslant & (1 + \frac{1}{4l}) \|u_i\|,\end{aligned}$$

Let T'' be the tensor parameterized by $(U'', \bar{C})$. We can bound $\|T'' - T^*\|_F$ as follows,

$$\begin{aligned}\|T'' - T^*\|_F \leqslant & \|T - T^*\|_F + \sum_{i=1}^{m}\sum_{k=1}^{l} \binom{l}{k} \|u_i\|^{l-k} \left\|\eta\nabla_{u_i} f(U, \bar{C})\right\|^k c_i^{l-2} \\ \leqslant & \|T - T^*\|_F + \sum_{i=1}^{m}\sum_{k=1}^{l} \binom{l}{k} \frac{1}{l^k} \|u_i\|^l c_i^{l-2} \\ \leqslant & \|T - T^*\|_F + l\sum_{i=1}^{m} \left(4(\sqrt{d(m+K)})^{l-2}(m+K)\delta^2 + \|u_i\|^2\right) \\ \leqslant & \|T - T^*\|_F + 4lm(\sqrt{d(m+K)})^{l-2}(m+K)\delta^2 + lM_2^2 \\ \leqslant & \sqrt{2\Gamma} + 2lM_2^2,\end{aligned}$$

where the last inequality assumes $\delta \leqslant \frac{M_2}{\sqrt{4m(\sqrt{d(m+K)})^{l-2}(m+K)}}$. For convenience, denote $\beta := \sqrt{2\Gamma} + 2lM_2^2$.

With the bound on $\|T'' - T^*\|$ and $\|u_i''\|$, we are ready to bound the Frobenius norm of $\mathcal{H}''$. For each $i \in [m]$, we have

$$\frac{\partial}{\partial u_i} f(U'', \bar{C}) = l(T'' - T^*)((u_i'')^{l-1}, I)c_i^{l-2}a_i + \lambda l \left(\frac{\|u_i''\|}{\|u_i\|}\right)^{l-2} u_i''. \tag{A.1}$$

We know $\mathcal{H}''$ is a $dm \times dm$ matrix that contains $m \times m$ block matrices with dimension $d \times d$. Each block corresponds to the second-order derivative of $f(U'', \bar{C})$ in terms of u_i, u_j. We will bound the Frobenius norm of $\mathcal{H}''$ by bounding the Frobenius norm of each block.

For each i, we can compute $\frac{\partial^2}{\partial u_i \partial u_i} f(U'', \bar{C})$ as follows,

$$\frac{\partial^2}{\partial u_i \partial u_i} f(U'', \bar{C}) = l(l-1)(T'' - T^*)((u_i'')^{l-2}, I, I)c_i^{l-2}a_i + l^2 c_i^{2l-4} \|u_i''\|^{2l-4} u_i'' \otimes u_i''$$
$$+ \lambda l \left(\frac{\|u_i''\|}{\|u_i\|}\right)^{l-2} I + \lambda l(l-2)\frac{\|u_i''\|^{l-4} u_i'' \otimes u_i''}{\|u_i\|^{l-2}}.$$

For the first term, we have $l(l-1)\left\|(T'' - T^*)((u_i'')^{l-2}, I, I)c_i^{l-2}\right\|_F$
$\leqslant l^2\sqrt{e}\beta(\sqrt{d(m+K)})^{l-2}$ since $\|u_i''\| / \|u_i\| \leqslant (1 + 1/(4l))$.

For the second term, we have

$$l^2 \left\| c_i^{2l-4} \|u_i''\|^{2l-4} u_i'' \otimes u_i'' \right\|_F \leqslant l^2\sqrt{e} \max\{4(d(m+K))^{l-2}(m+K)\delta^2, \|u_i\|^2\}.$$

For the third term, we have

$$\left\| \lambda l \left(\frac{\|u_i''\|}{\|u_i\|}\right)^{l-2} \mathrm{vec}(I) \right\|_F \leqslant \lambda l\sqrt{e}\sqrt{d}.$$

For the fourth term, we have

$$\left\| \lambda l(l-2)\frac{\|u_i''\|^{l-4} u_i'' \otimes u_i''}{\|u_i\|^{l-2}} \right\|_F \leqslant \lambda l^2\sqrt{e}.$$

Combing the bounds on these terms and assuming $\lambda \leqslant 1$, we have

$$\left\| \frac{\partial^2}{\partial u_i \partial u_i} f(U'', \bar{C}) \right\|_F$$
$$\leqslant l^2\sqrt{e}\beta(\sqrt{d(m+K)})^{l-2} + l^2\sqrt{e}\max\{4d^{l-2}(m+K)^{l-1}\delta^2, \|u_i\|^2\} + 2l^2\sqrt{e}\sqrt{d}.$$

Thus,

$$\left\| \frac{\partial^2}{\partial u_i \partial u_i} f(U'', \bar{C}) \right\|_F^2$$
$$\leqslant 3el^4\beta^2 d^{l-2}(m+K)^{l-2} + 3el^4 \max\{16d^{2l-4}(m+K)^{2l-2}\delta^4, \|u_i\|^4\} + 12el^4 d$$
$$\leqslant 15el^4\beta^2 d^{l-2}(m+K)^{l-2} + 3el^4 \max\{16d^{2l-4}(m+K)^{2l-2}\delta^4, \|u_i\|^4\}.$$

For each pair of $i \neq j$, we can compute $\frac{\partial^2}{\partial u_i \partial u_j} f(U'', \bar{C})$ as follows

$$\frac{\partial^2}{\partial u_i \partial u_j} f(U'', C) = l^2 a_i a_j c_i^{l-2} c_j^{l-2} \left\langle u_i'', u_j'' \right\rangle^{l-2} u_i'' \otimes u_j''.$$

The Frobenius norm square can be bounded as

$$\begin{aligned}
&\left\| \frac{\partial^2}{\partial u_i \partial u_j} f(U'', \bar{C}) \right\|_F^2 \\
\leqslant & el^4 \max\{4d^{l-2}(m+K)^{l-1}\delta^2, \|u_i\|^2\} \max\{4d^{l-2}(m+K)^{l-1}\delta^2, \|u_j\|^2\} \\
\leqslant & el^4 \max\{\max\{\|u_i\|^2, \|u_j\|^2\}^2, 16d^{2l-4}(m+K)^{2l-2}\delta^4\} \\
\leqslant & el^4 \left(\|u_i\|^4 + \|u_j\|^4 + 16d^{2l-4}(m+K)^{2l-2}\delta^4 \right)
\end{aligned}$$

Summing over the bounds on blocks, we can bound the Frobeneius norm of $\mathcal{H}''$,

$$\begin{aligned}
\|\mathcal{H}''\|_F^2 = & \sum_{i,j} \left\| \frac{\partial^2}{\partial u_i \partial u_j} f(U'', \bar{C}) \right\|_F^2 \\
\leqslant & 15eml^4\beta^2 d^{l-2}(m+K)^{l-2} + 48eml^4 d^{2l-4}(m+K)^{2l-2}\delta^4 + 3el^4 \sum_{i=1}^m \|u_i\|^4 \\
& + (m-1)el^4 \sum_{i=1}^m \|u_i\|^4 + 16el^4 m(m-1)d^{2l-4}(m+K)^{2l-2}\delta^4 \\
= & 15eml^4\beta^2 d^{l-2}(m+K)^{l-2} + 16el^4 m(m+2)d^{2l-4}(m+K)^{2l-2}\delta^4 \\
& + (m+2)el^4 \sum_{i=1}^m \|u_i\|^4 \\
\leqslant & 15eml^4\beta^2 d^{l-2}(m+K)^{l-2} + 2(m+2)el^4 M_4^2
\end{aligned}$$

where the last inequality assumes $\delta \leqslant \left(\frac{M_4^2}{16md^{2l-4}(m+K)^{2l-2}} \right)^{1/4}$.

Denoting $L_1 := \sqrt{15eml^4\beta^2 d^{l-2}(m+K)^{l-2} + 2(m+2)el^4 M_4^2}$, we have

$$f(U', \bar{C}) - f(U, \bar{C}) \leqslant -\eta \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 + \frac{\eta^2 L_1}{2} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2$$

$f(U', \bar{C}') - f(U', \bar{C})$ is bounded: Next, we show that setting c_i' as $c_i \frac{\|u_i\|}{\|u_i'\|}$ and $\hat{c}_i'$ as $\hat{c}_i \frac{\|u_i\|}{\|u_i'\|}$ does not increase the function value by too much. We use $\nabla_{\hat{u}_i} f$ to denote the gradient of u_i through c_i and $\hat{c}_i$, which means

$$\nabla_{\hat{u}_i} f = \frac{\partial f}{\partial c_i}\frac{\partial c_i}{\partial u_i} + \frac{\partial f}{\partial \hat{c}_i}\frac{\partial \hat{c}_i}{\partial u_i}.$$

In the following we first bound the Frobenius norm of the Hessian of f in terms of $\hat{U}$ evaluated at $(U', \bar{C}'')$ for any $C'' \in \{\mathrm{diag}(c_1'', \cdots, c_m'') | c_i'' = c_i \frac{\|u_i\|}{\|(1-\theta)u_i + \theta u_i'\|}, 0 \leqslant \theta \leqslant 1\}$ and $\hat{C}'' \in \{\mathrm{diag}(\hat{c}_1'', \cdots, \hat{c}_m'') | \hat{c}_i'' = \hat{c}_i \frac{\|u_i\|}{\|(1-\theta)u_i + \theta u_i'\|}, 0 \leqslant \theta \leqslant 1\}$. We denote the Hessian at $(U', \bar{C}'')$ as $\hat{\mathcal{H}}''$, which is a $md \times md$ matrix. Hessian $\hat{\mathcal{H}}''$ contains $m \times m$ blocks with dimension $d \times d$, each of which corresponds to $\frac{\partial^2}{\partial \hat{u}_i \partial \hat{u}_j} f(U', \bar{C}'')$ for some $(i, j) \in [m] \times [m]$.

Note that $\frac{\|u_i'\|}{\|u_i''\|} \leqslant 1 + 1/l$ since $\|u_i' - u_i\| \leqslant 1/(4l)\|u_i\|$. Let T''' be the tensor corresponds to $(U', \bar{C}'')$, we can bound $\|T''' - T^*\|_F$ as follows,

$$\begin{aligned}
\|T''' - T^*\|_F \leqslant & \left\| \sum_{i=1}^m a_i (c_i'')^{l-2} (u_i')^{\otimes l} \right\|_F + \|T^*\|_F \\
\leqslant & e \left(\sum_{i=1}^m \|u_i\|^2 + 4(\sqrt{d(m+K)})^{l-2} m(m+K)\delta^2 \right) + 1 \\
\leqslant & 2eM_2^2 + 1,
\end{aligned}$$

where the last step assumes $\delta \leqslant \frac{M_2}{\sqrt{4(\sqrt{d(m+K)})^{l-2} m(m+K)}}$. For convenience, denote $\alpha := 2eM_2^2 + 1$.

Let's first compute the derivative of f in terms of $\hat{u}_i$,

$$\frac{\partial}{\partial \hat{u}_i} f(U', \bar{C}'') = \tag{A.2}$$

$$- (l-2) a_i (T''' - T^*)((u_i')^{\otimes l}) \frac{u_i''}{\|u_i''\|^l} \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_i'' = \sqrt{d(m+K)}/\|u_i''\|} + \mathbb{1}_{c_i''=1/\|u_i''\|} \right)$$

$$- \lambda (l-2) \frac{u_i''}{\|u_i''\|^l} \|u_i'\|^l . \tag{A.3}$$

For each i, we have

$$\begin{aligned}
&\frac{\partial^2}{\partial \hat{u}_i \partial \hat{u}_i} f(U', \bar{C}'') \frac{1}{l-2} \\
=& - a_i (T''' - T^*)((u_i')^{\otimes l}) \frac{I}{\|u_i''\|^l} \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_i'' = \sqrt{d(m+K)}/\|u_i''\|} + \mathbb{1}_{c_i''=1/\|u_i''\|} \right) \\
&+ l a_i (T''' - T^*)((u_i')^{\otimes l}) \frac{u_i''(u_i'')^\top}{\|u_i''\|^{l+2}} \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_i'' = \sqrt{d(m+K)}/\|u_i''\|} + \mathbb{1}_{c_i''=1/\|u_i''\|} \right) \\
&+ (l-2) \|u_i'\|^{2l} \frac{u_i''(u_i'')^\top}{\|u_i''\|^{2l}} \left(d^{l-2}(m+K)^{l-2} \mathbb{1}_{c_i'' = \sqrt{d(m+K)}/\|u_i''\|} + \mathbb{1}_{c_i''=1/\|u_i''\|} \right) \\
&- \lambda \frac{I}{\|u_i''\|^l} \|u_i'\|^l \\
&+ \lambda l \frac{u_i''(u_i'')^\top}{\|u_i''\|^{l+2}} \|u_i'\|^l .
\end{aligned}$$

We bound its Frobenius norm square by

$$\begin{aligned}
&\left\| \frac{\partial^2}{\partial \hat{u}_i \partial \hat{u}_i} f(U', \bar{C}'') \right\|_F^2 \\
\leqslant& 5 \left(l^2 \alpha^2 e^2 d^{l-1} (m+K)^{l-2} + l^4 \alpha^2 e^2 d^{l-2} (m+K)^{l-2} \right) \\
&+ 5 \left(l^4 e^4 \max\{\|u_i\|^4, 16(m+K)^{2l-2} \delta^4 d^{2l-4}\} + \lambda^2 l^2 d e^2 + \lambda^2 l^4 e^2 \right) \\
\leqslant& 20 e^2 \alpha^2 l^4 d^{l-1} (m+K)^{l-2} + 5 e^4 l^4 \left(\|u_i\|^4 + 16(m+K)^{2l-2} \delta^4 d^{2l-4} \right),
\end{aligned}$$

where we assume that $\lambda \leqslant 1$.

For $i \neq j$, we have

$$\begin{aligned}
&\frac{\partial^2}{\partial \hat{u}_i \partial \hat{u}_j} f(U', \bar{C}'') \\
=&(l-2)^2 (\left\langle u_i', u_j' \right\rangle^l) \frac{u_i''(u_j'')^\top}{\|u_i''\|^l \|u_j''\|^l} \\
&\cdot \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_i''=\sqrt{d(m+K)}/\|u_i''\|} + \mathbb{1}_{c_i''=1/\|u_i''\|} \right) \\
&\cdot \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_j''=\sqrt{d(m+K)}/\|u_j''\|} + \mathbb{1}_{c_j''=1/\|u_j''\|} \right)
\end{aligned}$$

We can bound its Frobenius norm by

$$\begin{aligned}
&\left\| \frac{\partial^2}{\partial \hat{u}_i \partial \hat{u}_j} f(U', \bar{C}'') \right\|_F^2 \\
\leqslant& l^4 e^4 \max\{4d^{l-2}(m+K)^{l-1}\delta^2, \|u_i\|^2\} \max\{4d^{l-2}(m+K)^{l-1}\delta^2, \|u_j\|^2\} \\
\leqslant& l^4 e^4 \left(\|u_i\|^4 + \|u_j\|^4 + 16(m+K)^{2l-2}\delta^4 d^{2l-4} \right)
\end{aligned}$$

Combing the bounds on all blocks, we have

$$\begin{aligned}
\left\| \hat{\mathcal{H}}'' \right\|_F^2 \leqslant& \sum_{i=1}^m \left(20e^2\alpha^2 l^4 d^{l-1}(m+K)^{l-2} + 5e^4 l^4 \left(\|u_i\|^4 + 16(m+K)^{2l-2}\delta^4 d^{2l-4} \right) \right) \\
&+ \sum_{\substack{i,j \in [m] \\ i \neq j}} l^4 e^4 \left(\|u_i\|^4 + \|u_j\|^4 + 16(m+K)^{2l-2}\delta^4 d^{2l-4} \right) \\
\leqslant& 20me^2\alpha^2 l^4 d^{l-1}(m+K)^{l-2} + 80ml^4 e^4 (m+K)^{2l-2}\delta^4 d^{2l-4} + 5l^4 e^4 \sum_{i=1}^m \|u_i\|^4 \\
&+ 16m^2 l^4 e^4 (m+K)^{2l-2}\delta^4 d^{2l-4} + 2l^4 e^4 m \sum_{i=1}^m \|u_i\|^4 \\
\leqslant& 20me^2\alpha^2 l^4 d^{l-1}(m+K)^{l-2} + 7l^4 e^4 m M_4^2 + 96m^2 l^4 e^4 (m+K)^{2l-2}\delta^4 d^{2l-4} \\
\leqslant& 20me^2\alpha^2 l^4 d^{l-1}(m+K)^{l-2} + 8l^4 e^4 m M_4^2,
\end{aligned}$$

where the last inequality assumes $\delta \leqslant \left(\frac{M_4^2}{96m(m+K)^{2l-2}d^{2l-4}}\right)^{1/4}$.

Denote $L_2 := \sqrt{20me^2\alpha^2 l^4 d^{l-1}(m+K)^{l-2} + 8l^4e^4mM_4^2}$. Then, we have

$$f(U', \bar{C}') - f(U', \bar{C}) \leqslant \left\langle \nabla_{\hat{U}} f(U', \bar{C}), -\eta \nabla_U f(U, \bar{C}) \right\rangle + \frac{\eta^2 L_2}{2} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2.$$

From equation A.1 and A.3 we know that $\forall i \in [m]$, $\nabla_{\hat{u}_i} f(U, \bar{C}) = -\frac{l-2}{l} \cdot \frac{u_i u_i^\top (\nabla_{u_i} f(U, \bar{C}))}{\|u_i\|^2}$. Thus,

$$\left\| \nabla_{\hat{U}} f(U, \bar{C}) \right\|_F \leqslant \frac{l-2}{l} \left\| \nabla_U f(U, \bar{C}) \right\|_F.$$

In order to bound $\left\| \nabla_{\hat{U}} f(U', \bar{C}) \right\|_F$, we still need to show that $\nabla_{\hat{U}} f(U', \bar{C})$ is close to $\nabla_{\hat{U}} f(U, \bar{C})$.

Bounding $\left\| \nabla_{\hat{U}} f(U', \bar{C}) - \nabla_{\hat{U}} f(U, \bar{C}) \right\|_F$: Define U'' as $(1-\theta)U + \theta U'$ for all $0 \leqslant \theta \leqslant 1$. We will show that the derivative of $\nabla_{\hat{U}} f$ in U evaluated $(U'', \bar{C})$ is bounded. We denote this derivative as $\tilde{\mathcal{H}}''$ that is a $dm \times dm$ matrix. Matrix $\tilde{\mathcal{H}}''$ contains $m \times m$ blocks each of which has dimension $d \times d$ and corresponds to $\frac{\partial^2}{\partial \hat{u}_i \partial u_j} f(U'', \bar{C})$. Denote T'' as the tensor parameterized by $(U'', \bar{C})$. Recall that,

$$\begin{aligned}
&\frac{\partial}{\partial \hat{u}_i} f(U'', \bar{C}) \\
=& -(l-2)a_i(T'' - T^*)((u_i'')^{\otimes l}) \frac{u_i}{\|u_i\|^l} \left((\sqrt{d(m+K)})^{l-2} \mathbb{1}_{c_i = \sqrt{d(m+K)}/\|u_i\|} + \mathbb{1}_{c_i = 1/\|u_i\|} \right) \\
&- \lambda(l-2) \frac{u_i}{\|u_i\|^l} \|u_i''\|^l.
\end{aligned}$$

For any $i \in [m]$, we have

$$
\begin{aligned}
&-\frac{\partial^2}{\partial u_i \partial \hat{u}_i} f(U'', \bar{C}) = \\
&l(l-2)a_i(T''-T^*)((u_i'')^{\otimes l-1}, I)\frac{u_i^\top}{\|u_i\|^l}\left((\sqrt{d(m+K)})^{l-2}\mathbb{1}_{c_i=\sqrt{d(m+K)}/\|u_i\|}+\mathbb{1}_{c_i=1/\|u_i\|}\right)\\
&+l(l-2)c_i^{l-2}\|u_i''\|^{2l-2}\frac{u_i(u_i'')^\top}{\|u_i\|^l}\left((\sqrt{d(m+K)})^{l-2}\mathbb{1}_{c_i=\sqrt{d(m+K)}/\|u_i\|}+\mathbb{1}_{c_i=1/\|u_i\|}\right)\\
&+\lambda l(l-2)\frac{u_i(u_i'')^\top}{\|u_i\|^l}\|u_i''\|^{l-2}.
\end{aligned}
$$

The Frobenius norm of $\frac{\partial^2}{\partial u_i \partial \hat{u}_i} f(U'', \bar{C})$ can be bounded as follows,

$$
\begin{aligned}
&\left\|\frac{\partial^2}{\partial u_i \partial \hat{u}_i} f(U'', \bar{C})\right\|_F\\
\leqslant&\sqrt{e}l^2\beta(\sqrt{d(m+K)})^{l-2}+\sqrt{e}l^2\max\left(\|u_i\|^2, 4d^{l-2}(m+K)^{l-1}\delta^2\right)+\sqrt{e}\lambda l^2\\
\leqslant&2\sqrt{e}l^2\beta(\sqrt{d(m+K)})^{l-2}+\sqrt{e}l^2\max\left(\|u_i\|^2, 4d^{l-2}(m+K)^{l-1}\delta^2\right),
\end{aligned}
$$

where the last inequality assumes $\lambda \leqslant 1$. Therefore,

$$
\begin{aligned}
&\left\|\frac{\partial^2}{\partial u_i \partial \hat{u}_i} f(U'', \bar{C})\right\|_F^2\\
\leqslant&8el^4\beta^2 d^{l-2}(m+K)^{l-2}+2el^4\max\left(\|u_i\|^4, 16d^{2l-4}(m+K)^{2l-2}\delta^4\right)
\end{aligned}
$$

For $i \neq j$, we have

$$
\begin{aligned}
&\frac{\partial^2}{\partial u_j \partial \hat{u}_i} f(U'', \bar{C})\frac{1}{l(l-2)}\\
=&-a_i a_j c_j^{l-2}\left\langle u_i'', u_j''\right\rangle^{l-1}\frac{u_i(u_i'')^\top}{\|u_i\|^l}\left((\sqrt{d(m+K)})^{l-2}\mathbb{1}_{c_i=\sqrt{d(m+K)}/\|u_i\|}+\mathbb{1}_{c_i=1/\|u_i\|}\right)
\end{aligned}
$$

The Frobenius norm square can be bounded as

$$\begin{aligned}
&\left\|\frac{\partial^2}{\partial\hat{u}_i\partial u_j}f(U'',C)\right\|_F^2\\
\leqslant& el^4\max\{4d^{l-2}(m+K)^{l-1}\delta^2,\|u_i\|^2\}\max\{4d^{l-2}(m+K)^{l-1}\delta^2,\|u_j\|^2\}\\
\leqslant& el^4\left(\|u_i\|^4+\|u_j\|^4+16(m+K)^{2l-2}\delta^4d^{2l-4}\right).
\end{aligned}$$

Summing over the bounds on blocks, we can bound the Frobeneius norm of $\tilde{\mathcal{H}}''$,

$$\begin{aligned}
&\left\|\tilde{\mathcal{H}}''\right\|_F^2\\
=&\sum_{i,j}\left\|\frac{\partial^2}{\partial u_j\partial\hat{u}_i}f(U'',\bar{C})\right\|_F^2\\
\leqslant&8mel^4\beta^2d^{l-2}(m+K)^{l-2}+2el^4\sum_{i=1}^m\|u_i\|^4+32mel^4d^{2l-4}(m+K)^{2l-2}\delta^4\\
&+2mel^4\sum_{i=1}^m\|u_i\|^4+16m^2el^4(m+K)^{2l-2}\delta^4d^{2l-4}\\
\leqslant&8mel^4\beta^2d^{l-2}(m+K)^{l-2}+48m^2el^4d^{2l-4}(m+K)^{2l-2}\delta^4+3el^4mM_4^2\\
\leqslant&8mel^4\beta^2d^{l-2}(m+K)^{l-2}+4el^4mM_4^2,
\end{aligned}$$

where the last inequality assumes $\delta\leqslant\left(\frac{M_4^2}{48md^{2l-4}(m+K)^{2l-2}}\right)^{1/4}$.

Denoting $L_3:=\sqrt{8mel^4\beta^2d^{l-2}(m+K)^{l-2}+4el^4mM_4^2}$, we have

$$\left\|\nabla_{\hat{U}}f(U',\bar{C})-\nabla_{\hat{U}}f(U,\bar{C})\right\|_F\leqslant L_3\left\|\eta\nabla_Uf(U,\bar{C})\right\|_F\leqslant\frac{1}{3l}\left\|\nabla_Uf(U,\bar{C})\right\|_F,$$

where the second inequality assumes $\eta\leqslant\frac{1}{3L_3l}$. Therefore, we have

$$\begin{aligned}
\left\|\nabla_{\hat{U}}f(U',\bar{C})\right\|_F\leqslant\left\|\nabla_{\hat{U}}f(U,\bar{C})\right\|_F+\frac{1}{3l}\left\|\nabla_Uf(U,\bar{C})\right\|_F&\leqslant\left(\frac{l-2}{l}+\frac{1}{3l}\right)\left\|\nabla_Uf(U,\bar{C})\right\|_F\\
&\leqslant\left(1-\frac{5}{3l}\right)\left\|\nabla_Uf(U,\bar{C})\right\|_F.
\end{aligned}$$

Overall, we have proved that as long as η is small enough,

$$\begin{aligned}
f(U', \bar{C}') - f(U, \bar{C}) =& f(U', \bar{C}) - f(U, \bar{C}) + f(U', \bar{C}') - f(U', \bar{C}) \\
\leqslant& -\eta \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 + \frac{\eta^2 L_1}{2} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 \\
& + \eta \left\| \nabla_{\hat{U}} f(U', \bar{C}) \right\|_F^2 + \frac{\eta^2 L_2}{2} \left\| \nabla_{\hat{U}} f(U', \bar{C}) \right\|_F^2 \\
\leqslant& -\eta \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 + \frac{\eta^2 L_1}{2} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 \\
& + \eta \left(1 - \frac{5}{3l} \right) \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 + \frac{\eta^2 L_2}{2} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2 \\
\leqslant& -\frac{\eta}{l} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2,
\end{aligned}$$

where the last inequality assumes $\eta \leqslant \frac{4}{3l(L_1+L_2)}$. Combining all the bounds on δ, η, we know there exists constant μ_1, μ_2 such that as long as $\delta \leqslant \frac{\mu_1}{m^{\frac{1}{4}} \sqrt{\lambda} d^{\frac{l-2}{2}} (m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} d^{\frac{l-1}{2}} (m+K)^{\frac{l-2}{2}}}$, we have

$$f(U', \bar{C}') - f(U, \bar{C}) \leqslant -\frac{\eta}{l} \left\| \nabla_U f(U, \bar{C}) \right\|_F^2.$$

□

Then, we know in an epoch, the function value can only increase because of the initialization and the scalar mode switches. In Lemma A.6, we show these operations cannot increase the function by too much. Note that Lemma A.6 is the formal version of Lemma 2.4 together with the bound for scalar mode switches in the main text (these two arguments in the main text correspond to the two claims in Lemma A.6).

Lemma A.6. *Assume $f(U_0', \bar{C}_0') \leqslant \tilde{\Gamma} \leqslant 10$ at the beginning of an epoch, $\delta \leqslant \frac{\mu_1 \sqrt{\tilde{\Gamma}}}{m^{\frac{3}{4}} d^{\frac{l-2}{2}} (m+K)^{\frac{l-1}{2}} \lambda^{\frac{1}{2}}}$, and $\eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} d^{\frac{l-1}{2}} (m+K)^{\frac{l-2}{2}}}$ for some constants μ_1, μ_2. Also assume that $\lambda m \geqslant 10$. Denote the parameters at the end of this epoch as $(U_H, \bar{C}_H)$,*

then

$$f(U_H, \bar{C}_H) \leqslant \exp\left(\frac{14}{\lambda m}\right)\tilde{\Gamma}.$$

Proof of Lemma A.6. By Lemma A.5, we know the function value does not increase in any iteration (before potential scalar mode switch) as long as the initial function value is at most 10 and $\delta \leqslant \frac{\mu_1}{m^{\frac{1}{4}}\sqrt{\lambda} d^{\frac{l-2}{2}}(m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} d^{\frac{l-1}{2}}(m+K)^{\frac{l-2}{2}}}$ for some constants μ_1, μ_2. Thus, the function value can only increase when we reinitialize a component or when we switch the scaling from $\sqrt{d(m+K)}/\left\|u_i\right\|$ to $1/\left\|u_i\right\|$. In the following, we first show that reinitializing a component can only increase the function value by a small factor.

Claim A.1. *Suppose $f(U, \bar{C}) \leqslant \hat{\Gamma} \leqslant 10$. Reinitialize any vector with the smallest ℓ_2 norm among all columns of U, and let the updated parameters be $(U', \bar{C}')$, then*

$$f(U', \bar{C}') \leqslant \left(1 + \frac{13}{\lambda m}\right)\hat{\Gamma}.$$

According to the definition of the function value, we know $\left\|T - T^*\right\|_F \leqslant \sqrt{2\hat{\Gamma}} \leqslant \sqrt{20}$, $\sum_{j=1}^m \left\|u_j\right\|^2 \leqslant \frac{\hat{\Gamma}}{\lambda}$, and $\sum_{j=1}^m \left\|u_j\right\|^4 \leqslant \left(\sum_{j=1}^m \left\|u_j\right\|^2\right)^2 \leqslant \frac{\hat{\Gamma}^2}{\lambda^2}$. Suppose u_i is one of the vectors in U with the smallest ℓ_2 norm, then $\left\|u_i\right\|^2 \leqslant \frac{\hat{\Gamma}}{\lambda m}$. Suppose $u_i', c_i', \hat{c}_i', a_i'$ are the corresponding reinitialized vector and coefficients, and we have

$$\begin{aligned}
&\left\|a_i c_i^{l-2} u_i^{\otimes l} - a_i'(c_i')^{l-2}(u_i')^{\otimes l}\right\|_F \\
\leqslant &\left\|a_i c_i^{l-2} u_i^{\otimes l}\right\|_F + \left\|a_i'(c_i')^{l-2}(u_i')^{\otimes l}\right\|_F \\
\leqslant &\max\left(\left\|u_i\right\|^2, (\sqrt{d(m+K)})^{l-2} 4(m+K)\delta^2\right) + (\sqrt{d(m+K)})^{l-2}\delta^2 \\
\leqslant &\frac{\hat{\Gamma}}{\lambda m} + (\sqrt{d(m+K)})^{l-2} 4(m+K)\delta^2 + (\sqrt{d(m+K)})^{l-2}\delta^2 \leqslant \frac{2\hat{\Gamma}}{\lambda m},
\end{aligned}$$

where the last inequality assumes $\delta^2 \leqslant \frac{\hat{\Gamma}}{5\lambda m(m+K)^{\frac{l}{2}} d^{\frac{l-2}{2}}}$. Therefore, we can bound

$f(U', \bar{C}')$ as

$$
\begin{aligned}
& f(U', \bar{C}') \\
=& \frac{1}{2} \left\| T - T^* - a_i c_i^{l-2} u_i^{\otimes l} + a_i'(c_i')^{l-2}(u_i')^{\otimes l} \right\|_F^2 + \lambda \sum_{j=1}^{m} \hat{c}_j^{l-2} \left\| u_j \right\|^l - \lambda \hat{c}_i^{l-2} \left\| u_i \right\|^{2l} \\
& + \lambda(\hat{c}_i')^{l-2} \left\| u_i' \right\|^l \\
\leqslant & f(U, C) + \left\| T - T^* \right\|_F \left\| a_i c_i^{l-2} u_i^{\otimes l} - a_i'(c_i')^{l-2}(u_i')^{\otimes l} \right\|_F + \frac{1}{2} \left\| a_i c_i^{l-2} u_i^l - a_i'(c_i')^{l-2}(u_i')^l \right\|_F^2 \\
& + \lambda(\hat{c}_i')^{l-2} \left\| u_i' \right\|^l \\
\leqslant & f(U, C) + \sqrt{20} \cdot \frac{2\hat{\Gamma}}{\lambda m} + \frac{1}{2} \left(2 \frac{\hat{\Gamma}}{\lambda m} \right)^2 + \lambda \delta^2 \\
\leqslant & \left(1 + \frac{12}{\lambda m} \right) \hat{\Gamma} + \lambda \delta^2 \\
\leqslant & \left(1 + \frac{13}{\lambda m} \right) \hat{\Gamma},
\end{aligned}
$$

where the second last inequality assumes $\lambda m \geqslant \hat{\Gamma}$ and the last inequality assumes $\delta^2 \leqslant \frac{\hat{\Gamma}}{\lambda^2 m}$.

Switching the scaling from $\sqrt{d(m+K)}/\left\|u_i\right\|$ to $1/\left\|u_i\right\|$ can also potentially increase the function value. In the following, we show that the function value increase is small because we only switch the scaling mode when $\left\|u_i\right\| \leqslant 2\sqrt{m+K}\delta$.

Claim A.2. *Assume $f(U', \bar{C}') \leqslant \bar{\Gamma}$. Suppose at this iteration we switch the scaling of c_i', i.e., we set c_i' as $c_i'/\sqrt{d(m+K)}$. Let the updated parameters be $(U', C'', \hat{C}', A')$, we have*

$$
f(U', C'', \hat{C}', A') \leqslant \left(1 + \frac{1}{\lambda m^2} \right) \bar{\Gamma}.
$$

Suppose u_i is the parameter which is one step of gradient descent before u_i'. According to the algorithm, we know $\left\|u_i\right\| \leqslant 2\sqrt{m+K}\delta$. According to the proof in

Lemma A.5 where we bound the derivative of f with respect to u_i, we know that as long as $\eta \leqslant \frac{1}{l(\sqrt{2\bar{\Gamma}}(\sqrt{d(m+K)})^{l-2}+\lambda)}$, we have $\|u_i'\| \leqslant 2\|u_i\| \leqslant 4\sqrt{m+K}\delta$. Therefore,

$$\begin{aligned}\left\|(c_i')^{l-2}(u_i')^{\otimes l} - (c_i'')^{l-2}(u_i')^{\otimes l}\right\|_F &= \left\|(c_i')^{l-2}(u_i')^{\otimes l} - \frac{1}{(\sqrt{d(m+K)})^{l-2}}(c_i')^{l-2}(u_i')^{\otimes l}\right\|_F \\ &\leqslant \left\|(c_i')^{l-2}(u_i')^{\otimes l}\right\|_F \leqslant 16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}.\end{aligned}$$

Suppose the tensor at $(U', \bar{C}')$ is T', then $\|T' - T^*\|_F \leqslant \sqrt{2\bar{\Gamma}}$.

Thus, we can bound $f(U', C'', \hat{C}', A')$ as follows:

$$\begin{aligned}&f(U', C'', \hat{C}', A') \\ \leqslant& f(U', \bar{C}'') + \|T' - T^*\|_F \left\|(c_i')^{l-2}(u_i')^{\otimes l} - (c_i'')^{l-2}(u_i')^{\otimes l}\right\|_F \\ &+ \frac{1}{2}\left\|(c_i')^{l-2}(u_i')^{\otimes l} - (c_i'')^{l-2}(u_i')^{\otimes l}\right\|_F^2 \\ \leqslant& \bar{\Gamma} + \sqrt{2\bar{\Gamma}}\left(16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}\right) + \frac{1}{2}\left(16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}\right)^2 \\ \leqslant& \left(1 + \frac{1}{\lambda m^2}\right)\bar{\Gamma},\end{aligned}$$

where the last inequality assumes $\delta^2 \leqslant \frac{\sqrt{\bar{\Gamma}}}{32\sqrt{2}\lambda m^2(m+K)^{\frac{l}{2}}d^{\frac{l-2}{2}}}$ and $\lambda m^2 \geqslant 1$.

We are now ready to bound the increase of the function value during this epoch. According to the algorithm, each epoch contains at most m scaling mode switches. Therefore, following Claim A.2, all the scaling switches in one epoch can increase the upper bound of the function value by at most a factor of $\left(1 + \frac{1}{\lambda m^2}\right)^m \leqslant \exp\left(\frac{1}{\lambda m}\right)$. Combining with Claim A.1 which considers the re-initialization, we know that in each epoch, the upper bound of the function value increase by at most a factor of $\exp\left(\frac{1}{\lambda m}\right)\left(1 + \frac{13}{\lambda m}\right) \leqslant \exp\left(\frac{14}{\lambda m}\right)$. □

Following Lemma A.5 and Lemma A.6, we are ready to show that the function value is upper bounded by a constant by an induction proof. At the beginning of our algorithm, the function value is bounded by a constant as long as the initialization

radius δ is small enough. According to Lemma A.6, the increase in each epoch is bounded by a factor of $O(\frac{1}{\lambda m})$. Therefore, as long as the total number of epochs do not exceed $O(\lambda m)$, f will always be bounded by a constant. As a consequence, the Frobenius norm square of U must be bounded by $O(\frac{1}{\lambda})$ due to the design of the regularizer. These results are summarized into Lemma A.7.

Lemma A.7. *Assume* $\delta \leqslant \frac{\mu_1}{m^{\frac{3}{4}} d^{\frac{l-2}{2}} (m+K)^{\frac{l-1}{2}} \lambda^{\frac{1}{2}}}$, *and* $\eta \leqslant \frac{\mu_2 \lambda}{m^{\frac{1}{2}} l^4 d^{\frac{l-1}{2}} (m+K)^{\frac{l-2}{2}}}$ *for some constants* μ_1, μ_2. *Also assume* $K \leqslant \frac{\lambda m}{14}$ *and* $\frac{10}{m} \leqslant \lambda \leqslant 1$. *We know throughout the algorithm*

$$f(U, \bar{C}) \leqslant 10 \text{ and } \sum_{i=1}^{m} \|u_i\|^2 \leqslant \frac{10}{\lambda}.$$

Proof of Lemma A.7. Let's first show that the function value is bounded at the initialization if we choose δ to be small enough. At initialization, we have

$$\begin{aligned}
f(U, \bar{C}) =& \frac{1}{2} \|T - T^*\|_F^2 + \lambda \sum_{i=1}^{m} \hat{c}_i^{l-2} \|u_i\|^l \\
\leqslant& \frac{1}{2} \left(\sum_{i=1}^{m} \left\| c_i^{l-2} u_i^{\otimes l} \right\|_F + \|T^*\|_F \right)^2 + \lambda \sum_{i=1}^{m} \|u_i\|^2 \\
\leqslant& \frac{1}{2} \left(m (\sqrt{d(m+K)})^{l-2} \delta^2 + 1 \right)^2 + \lambda m \delta^2 \\
\leqslant& m^2 d^{l-2} (m+K)^{l-2} \delta^4 + 1 + \lambda m \delta^2 \leqslant 3,
\end{aligned}$$

where the last inequality assumes $\delta^4 \leqslant \frac{1}{m^2 d^{l-2} (m+K)^{l-2}}$ and $\lambda \leqslant 1$.

We use an inductive proof to prove that the function value at the end of the k-th ($k \leqslant K$) iteration is at most $3 \exp(\frac{14k}{\lambda m})$: At the initialization, the function value is at most 3. For every epoch, assume that our induction hypothesis is true, then at each step (a step can be a re-initialization, a gradient descent update, or a scalar mode switch), from Lemma A.6 we know that the function value is upper bounded

by $3\exp(\frac{14k}{\lambda m}) \leqslant 10$, so at this step Lemma A.5 is correct, meaning that Lemma A.6 is still correct at the next step.

Therefore, throughout the algorithm, we have

$$f(U, C) \leqslant 3\exp\left(\frac{14K}{\lambda m}\right) \leqslant 10,$$

where we assume $K \leqslant \frac{\lambda m}{14}$. This immediately implies that $\sum_i \|u_i\|^2 \leqslant \frac{10}{\lambda}$ by the design of our regularizer. □

Now we are ready to prove Lemma A.4.

Proof of Lemma A.4. From Lemma A.7, we know that the function value is upper bounded by 10 throughout our algorithm. Besides, from Lemma A.6 we know that the function value increase is at most $(\exp(\frac{14}{\lambda m}) - 1)$ times the function value at the beginning of this epoch. Choosing $\tilde{\Gamma} = f(U_0', \bar{C}_0')$, we know the function value increase at each epoch cannot exceed $10(\exp(\frac{14}{\lambda m}) - 1) = O(\frac{1}{\lambda m})$, which finishes the proof of Lemma A.4. □

A.3.2 Escaping local minima

In this section, we will give a formal proof of Lemma A.3. We again follow the proof ideas outlined in Section 2.4.2. Recall that the proof goes in the following steps:

1. We first show that the projection of U in the B subspace must be very small, therefore the influence from incorrect subspace B is small (Lemma A.8).

2. We then focus on the correlation in the correct subspace S. First we show that the correlation can be significantly negative at re-initialization with constant probability (Lemma A.9).

3. If the correlation is always significantly negative, then the re-initialized component will grow exponentially and eventually decrease the function value (Lemma A.10).

4. If the correlation changes significantly, the function value must also decrease (Lemma A.11 and Lemma A.12).

First of all, we need to show that the influence coming from B is small enough so that it can be ignored. The following lemma is the formal version of Lemma 2.5. Note that the assumption $\|U\|_F \leqslant \sqrt{\frac{10}{\lambda}}$ has been verified in Lemma A.7 so Lemma A.8 holds for the entire algorithm.

Lemma A.8. *Assume $\|U\|_F \leqslant \sqrt{\frac{10}{\lambda}}$ throughout the algorithm. Assume $\lambda \leqslant \sqrt{10}, \delta \leqslant \frac{\sqrt{10}}{2\sqrt{\lambda m d^{l-2}(m+K)^{l-1}}}$ and $\eta \leqslant \frac{\lambda}{20l}$. Then, we know $\|P_B U\|_F^2 \leqslant (m+K)\delta^2$ throughout the algorithm.*

Proof of Lemma A.8. At the initialization,

$$\|P_B U\|_F^2 \leqslant \|U\|_F^2 = m\delta^2.$$

At the beginning of each epoch, we re-initialize one column of U, which at most increases $\|P_B U\|_F^2$ by δ^2. Thus, the total increase due to the re-initialization process is at most $K\delta^2$.

Then, we only need to show that running gradient descent does not increase the norm of $P_B U$. Suppose at the beginning of one iteration, the tensor T is parameterized by $(U, \bar{C})$. Let U' be the updated parameter, which means $U' = U - \eta \nabla_U f(U, \bar{C})$. We have,

$$\begin{aligned}
&\|P_B U'\|_F^2 - \|P_B U\|_F^2 \\
=& \left\|P_B(U - \eta \nabla_U f(U, \bar{C}))\right\|_F^2 - \|P_B U\|_F^2 \\
=& -2\eta \left\langle P_B U, P_B \nabla_U f(U, \bar{C}) \right\rangle + \eta^2 \left\|P_B \nabla_U f(U, \bar{C})\right\|_F^2. \qquad \text{(A.4)}
\end{aligned}$$

We will show that the first term is negative and dominates the second term when η is small enough, which implies that gradient descent never increases the norm of

$P_B U$. We first compute the gradient as follows,

$$\nabla_U f(U, \bar{C}) = l\text{mat}(T - T^*)U^{\odot l-1}C^{l-2}A + l\lambda U.$$

where $\text{mat}(T - T^*) = UC^{l-2}A(U^{\odot l-1})^\top - U^*C^*[(U^*)^{\odot l-1}]^\top$ is a $d \times d^{l-1}$ matrix. Therefore, the projection of the gradient on B subspace is

$$P_B \nabla_U f(U, \bar{C}) = lP_B\text{mat}(T)U^{\odot l-1}C^{l-2}A + l\lambda P_B U.$$

Now, we show that the first term in (A.4) is negative.

$$\begin{aligned}
&-2\eta \left\langle P_B U, P_B \nabla_U f(U, \bar{C}) \right\rangle \\
=&-2l\eta \left\langle P_B U, P_B \text{mat}(T)U^{\odot l-1}C^{l-2}A + \lambda P_B U \right\rangle \\
=&-2l\eta \left\langle P_B UC^{l-2}A[U^{\odot l-1}]^\top, P_B \text{mat}(T) \right\rangle - 2l\eta \left\langle P_B U, \lambda P_B U \right\rangle \\
=&-2l\eta \left\| P_B \text{mat}(T) \right\|_F^2 - 2l\lambda\eta \left\| P_B U \right\|_F^2 .
\end{aligned}$$

Next, we show the second term in (A.4) is bounded. We have,

$$\begin{aligned}
\eta^2 \left\| P_B \nabla_U f(U, \bar{C}) \right\|_F^2 =& \eta^2 \left\| lP_B \text{mat}(T)U^{\odot l-1}C^{l-2}A + l\lambda P_B U \right\|_F^2 \\
\leqslant& 2\eta^2 l^2 \left(\left\| P_B \text{mat}(T)U^{\odot l-1}C^{l-2}A \right\|_F^2 + \lambda^2 \left\| P_B U \right\|_F^2 \right)
\end{aligned}$$

Recall that $M_2 = \sqrt{\frac{10}{\lambda}}$. Note that

$$\begin{aligned}
&\left\| P_B \text{mat}(T)U^{\odot l-1}C^{l-2}A \right\|_F^2 \\
\leqslant& \left\| P_B \text{mat}(T) \right\|_F^2 \left\| U^{\odot l-1}C^{l-2} \right\|_F^2 \\
=& \left\| P_B \text{mat}(T) \right\|_F^2 \sum_{i=1}^m c_i^{2l-4} \left\| u_i \right\|^{2l-2} \\
\leqslant& \left\| P_B \text{mat}(T) \right\|_F^2 \sum_{i=1}^m \max\{4(d(m+K))^{l-2}(m+K)\delta^2, \left\| u_i \right\|^2\} \\
\leqslant& M_2^2 \left\| P_B \text{mat}(T) \right\|_F^2
\end{aligned}$$

where the second inequality holds since $c_i = \frac{\sqrt{d(m+K)}}{\|u_i\|}$ only when $\|u_i\| \leqslant 2\sqrt{m+K}\delta$, and otherwise $c_i = \frac{1}{\|u_i\|}$. The last inequality assumes $\delta \leqslant \frac{M_2}{2\sqrt{md^{l-2}(m+K)^{l-1}}}$.

Overall, we have

$$\begin{aligned}
&\|P_B U'\|_F^2 - \|P_B U\|_F^2 \\
=& -2\eta \left\langle P_B U, P_B \nabla_U f(U, \bar{C}) \right\rangle + \eta^2 \left\| P_B \nabla_U f(U, \bar{C}) \right\|_F^2 \\
\leqslant& -2l\eta \left(\|P_B \mathrm{mat}(T)\|_F^2 + \lambda \|P_B U\|_F^2 \right) + 2\eta^2 l^2 \left(M_2^2 \|P_B \mathrm{mat}(T)\|_F^2 + \lambda^2 \|P_B U\|_F^2 \right) \\
\leqslant& -l\eta \left(\|P_B \mathrm{mat}(T)\|_F^2 + \lambda \|P_B U\|_F^2 \right),
\end{aligned}$$

where the last inequality assumes $\lambda \leqslant M_2^2$ and $\eta \leqslant \frac{1}{2lM_2^2}$. □

Lemma A.8 shows that the norm of $P_B U$ only increases at the (re-)initializations, so it will stay small throughout this algorithm. This allows us to bound the influence to our algorithm from the orthogonal subspace B and only focus on subspace S. We denote the re-initialized vector at t-th step as u_t, and its sign as $a \in \{\pm 1\}$. Our analysis focuses on the correlation between $P_S u_t$ and the residual tensor

$$\langle P_{S^{\otimes l}} T_t - T^*, a\overline{P_S u_t}^{\otimes l} \rangle.$$

Here $\overline{P_S u_t}$ is the normalized version $P_S u_t$. We will show that if this correlation is significantly negative at every iteration the norm of u_t will blow up exponentially.

Towards this goal, first we will show that the initial point $P_S u_0$ has a large negative correlation with the residual. We will lower bound this correlation by anti-concentration of Gaussian polynomials, and the following lemma is the formal version of Lemma 2.6. Note in our notation, we have $\langle P_{S^{\otimes l}} T_t - T^*, a\overline{P_S u_t}^{\otimes l} \rangle = a \left(P_{S^{\otimes l}} T_t - T^* \right) \left(\overline{P_S u_t}^{\otimes l} \right)$.

Lemma A.9. *Suppose the residual at the beginning of one epoch is $T_0' - T^*$. Suppose $a c_0^{l-2} u_0^{\otimes l}$ is the reinitialized component. There exists absolute constant μ such that*

with probability at least $1/5$*,*

$$\left\langle P_{S^{\otimes l}}T_0' - T^*, a\overline{P_S u_0}^{\otimes l} \right\rangle \leqslant -\frac{1}{(\mu r l)^{l/2}} \left\| P_{S^{\otimes l}}T_0' - P_{S^{\otimes l}}T^* \right\|_F,$$

where $\overline{P_S u_0} = P_S u_0 / \left\| P_S u_0 \right\|$.

Proof of Lemma A.9. Let's restrict into the r^l-dimensional space $S^{\otimes l}$, and let $P_{S^{\otimes l}}$ be the projection operator that projects a d^l-dimensional tensor to the r^l-dimensional space $S^{\otimes l}$. Then, we can think of $(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)$ as an r^l dimensional vector, and $\overline{P_S u}$ comes from uniform distribution on $\mathbb{S}^{r-1}$. Let v be an r-dimensional standard normal vector, then $a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(\overline{P_S u}^{\otimes l})$ has the same distribution as $a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\frac{1}{\|v\|^l}$.

Let's first show that the variance of $a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)$ is large:

$$\begin{aligned}\operatorname{Var}\left[a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\right] =& \mathbb{E}\left|a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\right|^2 \\ \geqslant& l! \left\| P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^* \right\|_F^2,\end{aligned}$$

where the equality holds because $\mathbb{E}\left[a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\right] = 0$ and the inequality follows from Lemma A.2. It's not hard to see that $a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)$ is an l-th order polynomial of standard Gaussian vectors. By anti-concentration inequality of Gaussian polynomials (Lemma A.16), we know there exists constant κ such that

$$\Pr\left[\left|a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\right| \leqslant \epsilon\sqrt{l!}\left\| P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^* \right\|_F\right] \leqslant \kappa l \epsilon^{1/l}.$$

Choosing $\epsilon = \frac{1}{2^l \kappa^l l^l}$, we know with probability at least half,

$$\begin{aligned}\left|a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\right| \geqslant& \frac{1}{2^l\kappa^l l^l}\sqrt{l!}\left\| P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^* \right\|_F \\ \geqslant& \frac{1}{2^l\kappa^l l^l}\left(\frac{l}{e}\right)^{l/2}\left\| P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^* \right\|_F \\ =& \frac{1}{2^l e^{l/2}\kappa^l l^{l/2}}\left\| P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^* \right\|_F.\end{aligned}$$

Since the distribution of $a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)$ is symmetric, we know with probability at least 1/4,

$$a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l) \leqslant -\frac{1}{2^l e^{l/2}\kappa^l l^{l/2}} \left\|P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*\right\|_F.$$

According to Lemma A.15, we know that with probability at least 19/20,

$$\|v\| \leqslant \kappa'\sqrt{r},$$

where κ' is some constant. This further implies that with probability at least 1/5,

$$a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(v^l)\frac{1}{\|v\|^l} \leqslant -\frac{1}{2^l e^{l/2}(\kappa\kappa')^l (rl)^{l/2}} \left\|P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*\right\|_F.$$

Choosing $\mu = 4e\kappa^2(\kappa')^2$ finishes the proof. □

Our next step argues that if this negative correlation is large in every step, then the norm of u_t blows up exponentially. Intuitively, this is due to the fact that the correlation is basically the dominating term in the gradient, so when it is significantly negative the vector u_t behaves similar to a vector doing matrix power method (here it is important that our model is 2-homogeneous so the behavior of power method is similar to the matrix setting). Below is the formal version of Lemma 2.7.

Lemma A.10. *In the setting of Theorem A.2, within one epoch, let T_0 be the tensor after the reinitilization and let T_τ be the tensor at the end of the τ-th iteration. Assume $\|P_S u_0\| \geqslant \frac{\mu_2\delta}{\sqrt{d}}$ for some constant $\mu_2 \in (0,1)$. For any $H \geqslant t \geqslant 1$, as long as $\left\langle P_{S^{\otimes l}}T_\tau - T^*, a\overline{P_S u_\tau}^{\otimes l}\right\rangle \leqslant \frac{-\epsilon}{5(\mu_1 rl)^{l/2}}$ for some constant μ_1 for all $t-1 \geqslant \tau \geqslant 0$, we have*

$$\|P_S u_t\|^2 \geqslant \left(1 + \eta\left(\frac{\mu_2}{2}\right)^{l-2}\frac{\epsilon}{10(\mu_1 rl)^{\frac{l}{2}}}\right)^t \|P_S u_0\|^2.$$

Proof of Lemma A.10. We will use inductive proof for this lemma. At the first step, we have

$$\begin{aligned}\|P_S u_1\|^2 &= \left\|P_S u_0 - \eta P_S \nabla_u f(U_0, \bar{C}_0)\right\|^2 \\ &= \|P_S u_0\|^2 - \eta \left\langle P_S u_0, P_S \nabla_u f(U_0, \bar{C}_0)\right\rangle + \eta^2 \left\|P_S \nabla_u f(U_0, \bar{C}_0)\right\|^2 \\ &\geqslant \|P_S u_0\|^2 - \eta \left\langle P_S u_0, P_S \nabla_u f(U_0, \bar{C}_0)\right\rangle .\end{aligned}$$

We can write down the $P_S \nabla_u f(U_0, \bar{C}_0)$ as follows,

$$P_S \nabla_u f(U_0, \bar{C}_0) = al(T_0 - T^*)(u_0^{\otimes(l-1)}, P_S) c_0^{l-2} + \lambda l P_S u_0.$$

Let's first consider $al(T_0 - T^*)((u_0)^{\otimes(l-1)}, P_S) c_0^{l-2}$. We can decompose u_0 into $P_S u_0$ and $P_B u_0$, so we can divide $al(T_0 - T^*)(u_0^{\otimes(l-1)}, P_S) c_0^{l-2}$ into 2^{l-1} terms, each of which corresponds to the projection of $u_0^{\otimes(l-1)}$ on a subspace in $\{S, B\}^{\otimes l-1}$. For subspace $S^{\otimes(l-1)}$, the projection is $al(P_{S^{\otimes l}} T_0 - P_{S^{\otimes l}} T^*)((P_S u_0)^{\otimes l-1}, P_S) c_0^{l-2}$. Its inner product with $-P_S u_0$ is

$$\begin{aligned}&\left\langle -P_S u_0, al(P_{S^{\otimes l}} T_0 - P_{S^{\otimes l}} T^*)((P_S u_0)^{\otimes l-1}, P_S) c_0^{l-2}\right\rangle \\ =& - al(P_{S^{\otimes l}} T_0 - P_{S^{\otimes l}} T^*)((P_S u_0)^{\otimes l}) c_0^{l-2} \\ =& - al(P_{S^{\otimes l}} T_0 - P_{S^{\otimes l}} T^*)((\overline{P_S u_0})^{\otimes l}) \left(\|P_S u_0\| c_0\right)^{l-2} \|P_S u_0\|^2 .\end{aligned}$$

Now, we only need to show that $\|P_S u_0\| c_0$ is lower bounded. We have

$$\|P_S u_0\| c_0 = \frac{\sqrt{d(m+K)}\, \|P_S u_0\|}{\|u_0\|} \geqslant \frac{\sqrt{d(m+K)}\mu_2 \delta/\sqrt{d}}{\delta} \geqslant \frac{\mu_2}{2},$$

where the first inequality uses $\|P_S u_0\| \geqslant \mu_2 \delta/\sqrt{d}$. Therefore,

$$\left\langle -P_S u_0, al(P_{S^{\otimes l}} T_0 - P_{S^{\otimes l}} T^*)((P_S u_0)^{\otimes l-1}, P_S) c_0^{l-2}\right\rangle \geqslant \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{5(\mu_1 r l)^{l/2}} \|P_S u_0\|^2 .$$

We then bound the remaining terms in $al(T_0 - T^*)((u_0)^{\otimes l-1}, P_S) c_0^{l-2}$: For any $l - 1 \geqslant k \geqslant 1$, we consider the subspace $B^{\otimes k} \otimes S^{\otimes l-1-k}$ and all of its permutations,

we bound the norm of $al(T_0 - T^*)((P_B u_0)^{\otimes k}, (P_S u_0)^{\otimes l-1-k}, P_S)c_0^{l-2}$ as follows.

$$\begin{aligned}
&\left\| al(T_0 - T^*)\left((P_B u_0)^{\otimes k}, (P_S u_0)^{\otimes l-1-k}, P_S\right) c_0^{l-2} \right\| \\
=& l \left\| T_0((P_B u_0)^{\otimes k}, (P_S u_0)^{\otimes l-1-k}, P_S) c_0^{l-2} \right\| \\
\leqslant& l \sum_{i=1}^{m} c_{0,i}^{l-2} \left\| P_B u_{0,i} \right\|^k \left\| P_S u_{0,i} \right\|^{l-k} \left\| P_B u_0 \right\|^k \left\| P_S u_0 \right\|^{l-1-k} c_0^{l-2} \\
\leqslant& l \sum_{i=1}^{m} c_{0,i}^{l-2} \left\| P_B u_{0,i} \right\| \left\| u_{0,i} \right\|^{l-1} \left\| P_B u_0 \right\| \left\| u_0 \right\|^{l-2} c_0^{l-2} \\
\leqslant& l \sum_{i=1}^{m} d^{l-2} (m+K)^{l-1} \delta^2 \left\| u_{0,i} \right\| \\
\leqslant& l\sqrt{m} d^{l-2} (m+K)^{l-1} \delta^2 M_2,
\end{aligned}$$

where $M_2 = \sqrt{\frac{10}{\lambda}}$ is the upper bound of $\|U\|_F$. Denote R_0 as the summation of terms in all subspaces except for $S^{\otimes l-1}$. We have $\|R_0\| \leqslant (2^{l-1}-1) l\sqrt{m} d^{l-2}(m+K)^{l-1}\delta^2 M_2$. Therefore, we have

$$\begin{aligned}
|\langle P_S u_0, R_0 \rangle| \leqslant \left\| P_S u_0 \right\| \left\| R_0 \right\| \leqslant& 2^l l\sqrt{m} d^{l-2} (m+K)^{l-1} \delta^2 M_2 \left\| P_S u_0 \right\| \\
\leqslant& \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{20(\mu_1 r l)^{l/2}} \left\| P_S u_0 \right\|^2
\end{aligned}$$

where the last inequality uses $\|P_S u_0\| \geqslant \frac{\mu_2 \delta}{\sqrt{d}}$ and assumes $\delta \leqslant \frac{1}{2^l l\sqrt{m} d^{l-2}(m+K)^{l-1} M_2} \cdot \frac{\mu_2}{\sqrt{d}} \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{20(\mu_1 r l)^{l/2}}$.

Next, let's analyze the regularizer $\lambda l P_S u_0$. Its norm can be bounded as follows,

$$\left\| \lambda l P_S u_0 \right\| \leqslant \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{20(\mu_1 r l)^{l/2}} \left\| P_S u_0 \right\|,$$

where we assume $\lambda \leqslant \frac{1}{l} \cdot \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{20(\mu_1 r l)^{l/2}}$.

Overall, we have

$$\begin{aligned}\|P_S u_1\|^2 &\geqslant \|P_S u_0\|^2 - \eta \left\langle P_S u_0, \nabla_u f(U_0, \bar{C}_0)\right\rangle \\ &\geqslant \|P_S u_0\|^2 + \eta \left(\frac{\mu_2}{2}\right)^{l-2} \left(\frac{\epsilon}{5(\mu_1 r l)^{l/2}} - \frac{\epsilon}{20(\mu_1 r l)^{l/2}} - \frac{\epsilon}{20(\mu_1 r l)^{l/2}}\right) \|P_S u_0\|^2 \\ &= \left(1 + \eta \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right) \|P_S u_0\|^2 .\end{aligned}$$

Induction Step: Suppose $\|P_S u_t\|^2 \geqslant \left(1 + \eta \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right)^t \|P_S u_0\|^2$, we will prove that $\|P_S u_{t+1}\|^2 \geqslant \left(1 + \eta \left(\frac{\mu_2}{2}\right)^{l-2} \frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right)^{t+1} \|P_S u_0\|^2$. Actually, we have assumed that $a(P_{S^{\otimes l}} T_t - P_{S^{\otimes l}} T^*)(\overline{P_S u_t}^{\otimes l}) \leqslant -\frac{\epsilon}{5(\mu_1 r l)^{l/2}}$, so we only need to show that $c_t \|P_S u_t\| \geqslant \frac{\mu_2}{2}$. Based on these two properties, the remaining proofs are exactly the same as that for $t = 0$.

The latter property is not hard to verify:

If $\|u_t\| > 2\sqrt{m + K}\delta$, we know $c_t = 1/\|u_t\|$. Then, we have $\|P_S u_t\| c_t = \frac{\|P_S u_t\|}{\|u_t\|} \geqslant \frac{\|u_t\| - \|P_B u_t\|}{\|u_t\|} \geqslant \frac{1}{2}$, where we use $\|P_B u_t\| \leqslant \sqrt{m + K}\delta$.

If $\|u_t\| \leqslant 2\sqrt{m + K}\delta$, we do not necessarily have $c_t = \frac{\sqrt{d(m+K)}}{\|u_t\|}$ because the norm of $\|u_t\|$ might first exceed the threshold and then drop below the threshold later. Note, by the induction proof, we only know the norm of $P_S u_t$ monotonically increase, which does not imply that $\|u_t\|$ monotonically increases. So, we have to consider both cases here. If $c_t = \frac{\sqrt{d(m+K)}}{\|u_t\|}$, we have $\|P_S u_t\| c_t = \frac{\sqrt{d(m+K)}\|P_S u_t\|}{\|u_t\|} \geqslant \frac{\mu_2}{2}$, which is because $\|P_S u_t\| \geqslant \|P_S u_0\| \geqslant \mu_2\delta/\sqrt{d}$. If $c_t = 1/\|u_t\|$, we know there exists $\tau \leqslant t$ such that $\|u_\tau\| > 2\sqrt{m + K}\delta$. Since $\|P_B u_\tau\| \leqslant \sqrt{m + K}\delta$, we know $\|P_S u_\tau\| \geqslant \sqrt{m + K}\delta$. By the induction proof, we know $\|P_S u_t\| \geqslant \|P_S u_\tau\| \geqslant \sqrt{m + K}\delta$. Then, we have $\|P_S u_t\| c_t = \frac{\|P_S u_t\|}{\|u_t\|} \geqslant \frac{1}{2}$.

This finishes the proof of Lemma A.10. □

Therefore the final step is to show that $a\overline{P_S u_t}^{\otimes l}$ always has a large negative correlation with $P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*$, unless the function value has already decreased. The difficulty here is that both the current reinitialized component u_t and other components are moving, therefore T_t is also changing.

We can bound the change of $T - T^*$ by separating it into two terms, which are the change of the re-initialized component and the change of the residual:

$$\begin{aligned}
&\left|a(P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*)(\overline{P_S u_t}^{\otimes l}) - a(P_{S^{\otimes l}}T_0 - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l})\right| \\
\leqslant &\left|\sum_{\tau=1}^{t}\left((P_{S^{\otimes l}}T_{\tau-1} - P_{S^{\otimes l}}T^*)(\overline{P_S u_\tau}^{\otimes l}) - (P_{S^{\otimes l}}T_{\tau-1} - P_{S^{\otimes l}}T^*)(\overline{P_S u_{\tau-1}}^{\otimes l})\right)\right| \\
&+ \sum_{\tau=1}^{t}\|T_\tau - T_{\tau-1}\|_F .
\end{aligned}$$

The change of the re-initialized component has a small effect on the correlation because the change in S subspace can only improve the correlation, and the influence of the B subspace can be bounded. This is formally proved in the following lemma, which is the formal version of Lemma 2.8.

Lemma A.11. *Assume $\delta \leqslant \frac{\mu_1}{m^{\frac{3}{4}}\sqrt{\lambda}d^{\frac{l-2}{2}}(m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2}{\lambda^{\frac{3}{2}}m^{\frac{9}{4}}d^{\frac{3l-6}{2}}(m+K)^{\frac{3l-3}{2}}}$ for some constants μ_1, μ_2. Assume $K \leqslant \frac{\lambda m}{14}$ and $\frac{10}{m} \leqslant \lambda \leqslant 1$. Suppose at the beginning of one iteration, the tensor T is parameterized by $(U, \bar{C})$. Suppose u is one column vector in U with $\|P_S u\| \geqslant \frac{\mu_3\delta}{\sqrt{d}}$ where μ_3 is a constant. Suppose u' is u after one step of gradient descent: $u' = u - \eta\nabla_u f(U, \bar{C})$. We have*

$$\begin{aligned}
&[a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(\overline{P_S u'}^{\otimes l}) \\
\leqslant &a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(\overline{P_S u}^{\otimes l}) + \mu l^4 2^l d^{l-1.5}m^{1/2}(m+K)^{l-1}\eta\delta\lambda,
\end{aligned}$$

where μ is some constant.

Proof of Lemma A.11. Define $g(u) := a(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)(\overline{P_S u}^{\otimes l})$. Note that in function $g(u)$, we view T as fixed. We will show that the change of g is bounded when the input changes from u to u'.

Bounding first order change: Let's first compute the gradient of g at u.

$$\begin{aligned}
\nabla g(u) =& al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, P_S)\frac{1}{\|P_S u\|^l} \\
&- al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l})\frac{P_S u}{\|P_S u\|^{l+2}} \\
=& al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, P_S)\frac{1}{\|P_S u\|^l} \\
&- al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, \overline{P_S u})\frac{\overline{P_S u}}{\|P_S u\|^l} \\
=& al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, P_S - \overline{P_S u}\cdot\overline{P_S u}^\top)\frac{1}{\|P_S u\|^l} \\
=& al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, P_D)\frac{1}{\|P_S u\|^l},
\end{aligned}$$

where P_D is the projection matrix on the span of $S\backslash\{u\}$. We can also compute the projection of $\nabla_u f(U, \bar{C})$ on D as follows,

$$P_D \nabla_u f(U, \bar{C}) = al(T - T^*)(u^{\otimes l-1}, P_D)c^{l-2}.$$

We can divide $l(T - T^*)(u^{\otimes l-1}, P_D)c^{l-2}$ into 2^{l-1} terms, each of which corresponds to the projection of u^{l-1} on a subspace. For subspace $S^{\otimes l-1}$, we have

$$al(T - T^*)((P_S u)^{\otimes l-1}, P_D)c^{l-2} = al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u)^{\otimes l-1}, P_D)c^{l-2},$$

which has non-negative inner product with $\nabla g(u)$. We can bound the norm of all the other terms. For any $l - 1 \geqslant k \geqslant 1$, consider subspace $B^{\otimes k} \otimes S^{\otimes(l-1-k)}$, we can

bound the norm of $al(T-T^*)((P_Bu)^{\otimes k},(P_Su)^{\otimes(l-1-k)},P_D)c^{l-2}$ as follows:

$$\begin{aligned}
&\left\|al(T-T^*)((P_Bu)^{\otimes k},(P_Su)^{\otimes l-1-k},P_D)c^{l-2}\right\| \\
=&l\left\|T((P_Bu)^{\otimes k},(P_Su)^{\otimes l-1-k},P_D)c^{l-2}\right\| \\
\leqslant&l\sum_{i=1}^{m}c_i^{l-2}\left\|P_Bu_i\right\|^k\left\|P_Su_i\right\|^{l-k}\left\|P_Bu\right\|^k\left\|P_Su\right\|^{l-1-k}c^{l-2} \\
\leqslant&l\sum_{i=1}^{m}c_i^{l-2}\left\|P_Bu_i\right\|\left\|u_i\right\|^{l-1}\left\|P_Bu\right\|\left\|u\right\|^{l-2}c^{l-2} \\
\leqslant&l\sum_{i=1}^{m}d^{l-2}(m+K)^{l-1}\delta^2\left\|u_i\right\| \\
\leqslant&l\sqrt{m}d^{l-2}(m+K)^{l-1}\delta^2M_2,
\end{aligned}$$

where the second last inequality comes from Lemma A.8.

Denote R as the summation of terms in all subspaces except for $S^{\otimes l-1}$, then

$$\|R\|\leqslant(2^{l-1}-1)l\sqrt{m}d^{l-2}(m+K)^{l-1}\delta^2M_2.$$

Therefore, the first order change of g can be bounded as follows,

$$\begin{aligned}
&\left\langle\nabla g(u),-\eta\nabla_u f(U,\bar{C})\right\rangle \\
\leqslant&\eta\left\|l(P_{S^{\otimes l}}T-P_{S^{\otimes l}}T^*)((P_Su)^{\otimes l-1},P_D)\frac{1}{\left\|P_Su\right\|^l}\right\|\|R\| \\
\leqslant&\eta l\sqrt{20}\frac{\sqrt{d}}{\mu_3\delta}\cdot(2^{l-1}-1)l\sqrt{m}d^{l-2}(m+K)^{l-1}\delta^2M_2 \\
\leqslant&\frac{10l^22^l}{\mu_3}\eta d^{l-1.5}\sqrt{m}(m+K)^{l-1}\delta M_2,
\end{aligned}$$

where the second last inequality assumes $||P_Su||\geqslant\frac{\mu_3\delta}{\sqrt{d}}$.

Bounding higher order change: For all $u'' = (1-\theta)u + \theta u'$ with $0 \leqslant \theta \leqslant 1$, we prove a uniform upper bound for $\|\nabla^2 g(u'')\|_F$. Recall the gradient of g at u'',

$$\nabla g(u'') = al(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u'')^{\otimes l-1}, P_D'')\frac{1}{\|P_S u''\|^l},$$

where P_D'' is the projection matrix to $S\backslash\{u''\}$. We compute $\|\nabla^2 g(u'')\|$ as follows,

$$\begin{aligned}\nabla^2 g(u'') =& al(l-1)(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u'')^{\otimes l-2}, P_S, P_D'')\frac{1}{\|P_S u''\|^l} \\ &- al^2(P_{S^{\otimes l}}T - P_{S^{\otimes l}}T^*)((P_S u'')^{\otimes l-1}, P_D'') \otimes \frac{P_S u''}{\|P_S u''\|^{l+2}}.\end{aligned}$$

Therefore,

$$\left\|\nabla^2 g(u'')\right\|_F \leqslant 2l^2\sqrt{20}\frac{1}{\|P_S u''\|^2}.$$

Assume that $\eta \leqslant \frac{\mu_3\delta\sqrt{\lambda}}{2\sqrt{10d}\left(\sqrt{20}l(\sqrt{d(m+K)})^{l-2}+\lambda l\right)}$ and from the proof of Lemma A.5 where we bound the gradient, we know that

$$\left\|\eta\nabla_u f(U,\bar{C})\right\| \leqslant \eta\left(l\sqrt{20}(\sqrt{d(m+K)})^{l-2} + \lambda l\right)\|u\| \leqslant \frac{\sqrt{\frac{\lambda}{10}}\mu_3\delta}{2\sqrt{d}}\sqrt{\frac{10}{\lambda}} = \frac{\mu_3\delta}{2\sqrt{d}}.$$

Thus,

$$\begin{aligned}\|P_S u''\| &\geqslant \|P_S u\| - \|P_S u'' - P_S u\| \\ &\geqslant \|P_S u\| - \left\|\theta\eta\nabla_u f(U,\bar{C})\right\| \\ &\geqslant \frac{\mu_3\delta}{\sqrt{d}} - \frac{\mu_3\delta}{2\sqrt{d}} = \frac{\mu_3\delta}{2\sqrt{d}}.\end{aligned}$$

Therefore,

$$\left\|\nabla^2 g(u'')\right\|_F \leqslant 2l^2\sqrt{20}\frac{4d}{\mu_3^2\delta^2}.$$

Overall, we have

$$g(u') - g(u) \leqslant \left\langle\nabla g(u), -\eta\nabla_u f(U,\bar{C})\right\rangle + \frac{\eta^2}{2}2l^2\sqrt{20}\frac{4d}{\mu_3^2\delta^2}\left\|\nabla_u f(U,\bar{C})\right\|^2.$$

Recall that,

$$\nabla_u f(U, \bar{C}) = al(T - T^*)(u^{\otimes l-1}, I)c^{l-2} + \lambda l u,$$

we have

$$\begin{aligned}
\left\|\nabla_u f(U, \bar{C})\right\|_F \leqslant & l\sqrt{20}\max\left(\|u\|, 2\sqrt{m+K}\delta(\sqrt{d(m+K)})^{l-2}\right) + \lambda l \|u\| \\
\leqslant & l\sqrt{20}\max\left(M_2, 2\sqrt{m+K}\delta(\sqrt{d(m+K)})^{l-2}\right) + \lambda l M_2 \\
\leqslant & l\sqrt{20}M_2 + \lambda l M_2,
\end{aligned}$$

where the last inequality assumes $\delta \leqslant \frac{M_2}{2\sqrt{m+K}(\sqrt{d(m+K)})^{l-2}}$.

Finally, we have

$$\begin{aligned}
& g(u') - g(u) \\
\leqslant & \left\langle \nabla g(u), -\eta \nabla_u f(U, \bar{C}) \right\rangle + \frac{\eta^2}{2} 2l^2\sqrt{20}\frac{4d}{\mu_3^2\delta^2}\left\|\nabla_u f(U, \bar{C})\right\|^2 \\
\leqslant & \frac{10l^2 2^l}{\mu_3}\eta d^{l-1.5}\sqrt{m}(m+K)^{l-1}\delta M_2 + \frac{\eta^2}{2} 2l^2\sqrt{20}\frac{4d}{\mu_3^2\delta^2}\left(l\sqrt{20}M_2 + \lambda l M_2\right)^2 \\
\leqslant & \frac{10l^2 2^l}{\mu_3}\eta d^{l-1.5}\sqrt{m}(m+K)^{l-1}\sqrt{\frac{10}{\lambda}}\delta + \sqrt{20}\eta^2 l^2\frac{4d}{\mu_3^2\delta^2}\left(40l^2\frac{10}{\lambda} + 2\lambda^2 l^2\frac{10}{\lambda}\right) \\
\leqslant & \mu l^4 2^l d^{l-1.5} m^{1/2}(m+K)^{l-1}\eta\delta\lambda,
\end{aligned}$$

where the last inequality assumes $l \geqslant 3$, $\eta \leqslant \delta^3$ and μ is some constant. □

Therefore, the only way to change the residual term by a lot must be changing the tensor T, and the accumulated change of T is strongly correlated with the decrease of f. This is similar to the technique of bounding the function value decrease in Wei et al. (2019). The connection between them are formalized in the following lemma, which is the formal version of Lemma 2.9:

Lemma A.12. *Assume that* $\delta \leqslant \frac{\mu_1}{m^{\frac{3}{4}}\sqrt{\lambda}d^{\frac{l-2}{2}}(m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2\lambda}{m^{\frac{1}{2}}l^4 d^{\frac{l-1}{2}}(m+K)^{\frac{l-2}{2}}}$ *for some constants* μ_1, μ_2, *and* $\frac{10}{m} \leqslant \lambda \leqslant 1$. *Within one epoch, let* T_0 *be the tensor after*

reinitialization, and let T_t be the tensor at the end of the t-th iteration. Let $(U_0, \bar{C}_0)$ be the parameters after the reinitialization step and let $(U_H, \bar{C}_H)$ be the parameters at the end of this epoch. We have

$$\begin{aligned}
&\sum_{\tau=1}^{H} \|T_\tau - T_{\tau-1}\|_F \\
\leqslant &200l^{2.5}\sqrt{\frac{1}{\lambda}}\sqrt{\eta H}\sqrt{f(U_0,\bar{C}_0) - f(U_H,\bar{C}_H) + 160m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}} \\
&+ 16m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}.
\end{aligned}$$

Intuitively, if we are doing a standard gradient descent, at each step the change in function value is going to be proportional to the square of the change in the tensor T, and the guarantee similar to the Lemma above can be proved by applying Cauchy-Schwartz. However, in our setting the proof becomes more complicated because of the normalization steps and in particular the scalar mode switch.

Before proving Lemma A.12, we first prove the following lemma which guarantees the function value decrease in one step (without scalar mode switch):

Lemma A.13. *Assume $\delta \leqslant \frac{\mu_1}{m^{\frac{3}{4}}\sqrt{\lambda}d^{\frac{l-2}{2}}(m+K)^{\frac{l-1}{2}}}, \eta \leqslant \frac{\mu_2\lambda}{m^{\frac{1}{2}}l^4 d^{\frac{l-1}{2}}(m+K)^{\frac{l-2}{2}}}$ for some constants μ_1, μ_2, and $\eta \leqslant \delta^3$. Assume $K \leqslant \frac{\lambda m}{14}$. Starting from T parameterized by $(U, \bar{C})$, suppose after one iteration (before potential scalar mode switch) the tensor becomes T' parameterized by $(U', \bar{C}')$. We have*

$$\|T' - T\|_F \leqslant 200l^2\sqrt{\frac{1}{\lambda}}\left\|\eta\nabla_U f(U,\bar{C})\right\|_F.$$

Proof of Lemma A.13. According to the algorithm, we know each iteration is composed of two steps: update U by gradient descent ($U' = U - \eta\nabla_U f(U,\bar{C})$) and update C and $\hat{C}$ according to U'. Let $\hat{T}$ be the intermediate tensor parameterized by $(U', \bar{C})$. We will bound $\|T' - T\|_F$ by bounding $\left\|\hat{T} - T\right\|_F$ and $\left\|T' - \hat{T}\right\|_F$ separately.

According to Lemma A.7, we know $\sum_{i=1}^m \|u_i\|^2, \sum_{i=1}^m \|u_i'\|^2 \leqslant \frac{10}{\lambda}$. Denote $M_2^2 = \frac{10}{\lambda}$.

Bounding $\left\|\hat{T}-T\right\|_F$*:* From T to $\hat{T}$, U is updated to $U' = U - \eta\nabla_U f(U,\bar{C})$ while C and $\hat{C}$ remains the same. Therefore,

$$\begin{aligned}\left\|\hat{T}-T\right\|_F &= \left\|\sum_{i=1}^m a_i c_i^{l-2}\left(u_i - \eta\nabla_{u_i} f(U,\bar{C})\right)^{\otimes l} - \sum_{i=1}^m a_i c_i^{l-2} u_i^{\otimes l}\right\|_F \\ &\leqslant \sum_{i=1}^m l\left\|u_i\right\|^{l-1}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\| c_i^{l-2} + \sum_{i=1}^m\sum_{k=2}^{l}\binom{l}{k}\left\|u_i\right\|^{l-k}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\|^k c_i^{l-2}.\end{aligned}$$

We can further bound the linear term as follows:

$$\begin{aligned}&\sum_{i=1}^m l\left\|u_i\right\|^{l-1}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\| c_i^{l-2} \\ \leqslant& l\sum_{i=1}^m\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\|\max(\left\|u_i\right\|, 2\sqrt{m+K}\delta(\sqrt{d(m+K)})^{l-2}) \\ \leqslant& l\sqrt{\sum_{i=1}^m\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\|^2}\sqrt{\sum_{i=1}^m\max(\left\|u_i\right\|^2, 4(m+K)^{l-1}\delta^2 d^{l-2})} \\ \leqslant& \sqrt{2}lM_2\eta\left\|\nabla_U f(U,\bar{C})\right\|_F,\end{aligned}$$

where the last inequality assumes $\delta^2 \leqslant \frac{M_2^2}{4m(m+K)^{l-1}d^{l-2}}$.

According to the proof in Lemma A.5, we know $\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\| \leqslant \frac{1}{l}\left\|u_i\right\|$. Therefore, for the higher order terms, for each $k \geqslant 2$,

$$\begin{aligned}\sum_{i=1}^m\binom{l}{k}\left\|u_i\right\|^{l-k}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\|^k c_i^{l-2} &\leqslant \sum_{i=1}^m l^k\left\|u_i\right\|^{l-k}\frac{\left\|u_i\right\|^{k-1}}{l^{k-1}}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\| c_i^{l-2} \\ &\leqslant \sum_{i=1}^m l\left\|u_i\right\|^{l-1}\left\|\eta\nabla_{u_i} f(U,\bar{C})\right\| c_i^{l-2} \\ &\leqslant \sqrt{2}lM_2\eta\left\|\nabla_U f(U,\bar{C})\right\|_F.\end{aligned}$$

Overall, we have

$$\left\|\hat{T}-T\right\|_F \leqslant 2\sqrt{2}l^2 M_2\eta\left\|\nabla_U f(U,\bar{C})\right\|_F.$$

Bounding $\left\|T' - \hat{T}\right\|_F$*:* From $\hat{T}$ to T', we update C to C' and $\hat{C}$ to $\hat{C}'$ such that $\forall i \in [m], c_i' = c_i \frac{\|u_i\|}{\|u_i'\|}$ and $\hat{c}_i' = \hat{c}_i \frac{\|u_i\|}{\|u_i'\|}$. Thus,

$$\begin{aligned}\left\|T' - \hat{T}\right\|_F &= \left\|\sum_{i=1}^m a_i (c_i')^{l-2} (u_i')^{\otimes l} - \sum_{i=1}^m a_i c_i^{l-2} (u_i')^{\otimes l}\right\|_F \\ &\leqslant \sum_{i=1}^m \left|(c_i')^{l-2} - c_i^{l-2}\right| \|u_i'\|^l .\end{aligned}$$

Now, let's focus on the change in c_i^{l-2}. Define $g(u) = \frac{1}{\|u\|^{l-2}}$. We have,

$$\nabla g(u) = -(l-2)\frac{u}{\|u\|^l} \text{ and } \nabla^2 g(u) = -(l-2)\frac{I}{\|u\|^l} + l(l-2)\frac{uu^\top}{\|u\|^{l+2}}.$$

Therefore, the spectral norm of $\nabla^2 g(u)$ is bounded by $l^2/\|u\|^l$.

For any $i \in [m]$, let u_i'' be any point on the line segment between u_i and u_i', then $\|\nabla^2 g(u_i'')\|_2 \leqslant l^2/\|u_i''\|^l$. If $c_i = 1/\|u_i\|$, we have

$$\begin{aligned}\left|(c_i')^{l-2} - c_i^{l-2}\right| &\leqslant \|\nabla g(u_i)\| \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\| + \frac{1}{2} \max_{u_i''} \left\|\nabla^2 g(u_i'')\right\|_2 \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\|^2 \\ &\leqslant \frac{l-2}{\|u_i\|^{l-1}} \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\| + \frac{1}{2} \max_{u_i''} \frac{l\|u_i\|}{\|u_i''\|^l} \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\|.\end{aligned}$$

If $c_i = \sqrt{d(m+K)}/\|u_i\|$, we have

$$\begin{aligned}\left|(c_i')^{l-2} - c_i^{l-2}\right| \leqslant& \frac{l-2}{\|u_i\|^{l-1}} \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\| (\sqrt{d(m+K)})^{l-2} \\ &+ \frac{1}{2} \max_{u_i''} \frac{l\|u_i\|}{\|u_i''\|^l} \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\| (\sqrt{d(m+K)})^{l-2}.\end{aligned}$$

Therefore, we have

$$\begin{aligned}\left\|T' - \hat{T}\right\|_F \leqslant& 2el \sum_{i=1}^m \left\|\eta \nabla_{u_i} f(U, \bar{C})\right\| \max(\|u_i\|, 2\sqrt{m+K}\delta(\sqrt{d(m+K)})^{l-2}) \\ \leqslant& 2\sqrt{2} elM_2 \left\|\eta \nabla_U f(U, \bar{C})\right\|_F,\end{aligned}$$

where the first inequality holds because $\|u_i'\| \leqslant \left(1+\frac{1}{l}\right)\|u_i\|$ and the second inequality assumes $\delta^2 \leqslant \frac{M_2^2}{4m(m+K)^{l-1}d^{l-2}}$.

Overall, combing the bounds on $\left\|\hat{T}-T\right\|_F$ and $\left\|T'-\hat{T}\right\|_F$, we have

$$\|T'-T\|_F \leqslant 200l^2\sqrt{\frac{1}{\lambda}}\left\|\eta\nabla_U f(U,\bar{C})\right\|_F.$$

□

Now we are ready to prove Lemma A.12.

Proof of Lemma A.12. Let's first bound the tensor change and function value change due to scalar mode switches. Following the proof of Claim A.2 in Lemma A.6, setting $\tilde{\Gamma}=10$ and assuming $\eta \leqslant \frac{1}{l(\sqrt{2\Gamma}(\sqrt{d(m+K)})^{l-2}+\lambda)}$, we know each scalar mode switch can at most change the tensor Frobenius norm by $16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}$. Furthermore, using the same argument as Claim A.2, the function value can increase by at most

$$\begin{aligned}&\sqrt{20}\left(16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}\right)+\frac{1}{2}\left(16(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}\right)^2\\ \leqslant&160(m+K)\delta^2(\sqrt{d(m+K)})^{l-2},\end{aligned}$$

where we assume $\delta^2 \leqslant \frac{5}{8(m+K)(\sqrt{d(m+K)})^{l-2}}$.

According to the algorithm, we know each epoch contains at most m scalar mode switches. Suppose T_τ' be the tensor before potential scalar mode switch in the τ-th iteration. Then, we have

$$\begin{aligned}\sum_{\tau=1}^{t}\|T_\tau-T_{\tau-1}\|_F &\leqslant \sum_{\tau=1}^{t}\|T_\tau'-T_{\tau-1}\|_F+\sum_{\tau=1}^{t}\|T_\tau-T_\tau'\|_F\\ &\leqslant \sum_{\tau=1}^{t}\|T_\tau'-T_{\tau-1}\|_F+16m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}.\end{aligned}$$

According to Lemma A.13, we know

$$\|T'_\tau - T_{\tau-1}\|_F \leqslant 200l^2\sqrt{\frac{1}{\lambda}}\left\|\eta\nabla_U f(U_{\tau-1},\bar{C}_{\tau-1})\right\|_F.$$

Therefore, we have

$$\begin{aligned}\sum_{\tau=1}^{t}\|T'_\tau - T_{\tau-1}\|_F \leqslant& 200l^2\sqrt{\frac{1}{\lambda}}\sum_{\tau=1}^{t}\left\|\eta\nabla_U f(U_{\tau-1},\bar{C}_{\tau-1})\right\|_F\\ \leqslant& 200l^2\sqrt{\frac{1}{\lambda}}\sqrt{t}\sqrt{\sum_{\tau=1}^{t}\left\|\eta\nabla_U f(U_{\tau-1},\bar{C}_{\tau-1})\right\|_F^2}.\end{aligned}$$

According to Lemma A.5, we know

$$f(U'_\tau,\bar{C}'_\tau) - f(U_{\tau-1},\bar{C}_{\tau-1}) \leqslant -\frac{\eta}{l}\left\|\nabla_U f(U_{\tau-1},\bar{C}_{\tau-1})\right\|_F^2.$$

Therefore, we have

$$\sum_{\tau=1}^{t}\|T'_\tau - T_{\tau-1}\|_F \leqslant 200l^2\sqrt{\frac{1}{\lambda}}\sqrt{t}\sqrt{\sum_{\tau=1}^{t}\eta l\left(f(U_{\tau-1},\bar{C}_{\tau-1}) - f(U'_\tau,\bar{C}'_\tau)\right)}.$$

Since scalar mode switches in total change the function value by at most $160m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}$, we know

$$\begin{aligned}&\sum_{\tau=1}^{t}\left(f(U_{\tau-1},\bar{C}_{\tau-1}) - f(U'_\tau,\bar{C}'_\tau)\right)\\ \leqslant& f(U_0,\bar{C}_0) - f(U_t,\bar{C}_t) + 160m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}.\end{aligned}$$

Overall, we have

$$\begin{aligned}&\sum_{\tau=1}^{t}\|T_\tau - T_{\tau-1}\|_F\\ \leqslant& 200l^{2.5}\sqrt{\frac{1}{\lambda}}\sqrt{\eta H}\sqrt{f(U_0,\bar{C}_0) - f(U_t,\bar{C}_t) + 160m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}}\\ &+ 16m(m+K)\delta^2(\sqrt{d(m+K)})^{l-2}.\end{aligned}$$

□

Combining all the steps above, we are now ready to prove Lemma A.3.

Proof of Lemma A.3. Let u_0 be the reinitialized vector. According to Lemma A.9, we know with probability at least 1/5,

$$a(P_{S^{\otimes l}}T_0' - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l}) \leqslant \frac{-1}{(\mu_1 rl)^{l/2}} \left\|P_{S^{\otimes l}}T_0' - P_{S^{\otimes l}}T^*\right\|_F \leqslant -\frac{\epsilon}{(\mu_1 rl)^{l/2}},$$

where μ_1 is some constant. According to Lemma A.14, we know with probability at least $1 - 1/30$, $\|P_S u_0\| \geqslant \frac{\mu_2 \delta}{\sqrt{d}}$ for some constant $\mu_2 < 1$. Taking a union bound, we know both properties hold with probability at least 1/6.

Conditioning on both properties, we will prove that

$$f(U_0, \bar{C}_0) - f(U_H, \bar{C}_H) \geqslant \frac{\lambda}{32000000(\mu_1 rl)^l \eta H l^5}\epsilon^2.$$

For the sake of contradiction, assume that

$$f(U_0, \bar{C}_0) - f(U_H, \bar{C}_H) \leqslant \frac{\lambda}{32000000(\mu_1 rl)^l \eta H l^5}\epsilon^2$$

. According to Lemma A.12, we know

$$\sum_{\tau=1}^{H} \left\|T_\tau - T_{\tau-1}\right\|_F \leqslant \frac{\epsilon}{10(\mu_1 rl)^{l/2}}$$

as long as $\delta^2 \leqslant \frac{\epsilon}{320(\mu_1 rl)^{l/2} m(m+K)^{\frac{l}{2}} d^{\frac{l-2}{2}}}$ and $\delta^2 \leqslant \frac{\lambda\epsilon^2}{32000000(\mu_1 rl)^l \eta H l^5 \cdot 160 m(m+K)^{\frac{l}{2}} d^{\frac{l-2}{2}}}$.

We will prove that $a(P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*)(\overline{P_S u_t}^{\otimes l}) \leqslant -\frac{\epsilon}{5(C_1 rl)^{l/2}}$ for all $0 \leqslant t \leqslant H-1$, so from Lemma A.10 we know that the norm of $P_S u_t$ must increase exponentially.

Let's first prove the case at the beginning of an epoch: Let T_0 be the tensor after reinitialization. According to the proof of Claim A.1 in Lemma A.6, we know

$$\|T_0 - T_0'\|_F \leqslant 2\sqrt{\frac{10}{\lambda m}} \leqslant \frac{\epsilon}{2(\mu_1 rl)^{l/2}},$$

where the last inequality assumes $\lambda m \geqslant \frac{160(\mu_1 rl)^l}{\epsilon^2}$. This implies that

$$\begin{aligned}
&a(P_{S^{\otimes l}}T_0 - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l}) \\
\leqslant &a(P_{S^{\otimes l}}T_0' - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l}) + \|T_0 - T_0'\|_F \\
\leqslant &-\frac{\epsilon}{2(\mu_1 rl)^{l/2}}.
\end{aligned}$$

For later steps, we will show that $a(P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*)(\overline{P_S u_t}^{\otimes l})$ is close to $a(P_{S^{\otimes l}}T_0 - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l})$. Actually,

$$\begin{aligned}
&\left|a(P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*)(\overline{P_S u_t}^{\otimes l}) - a(P_{S^{\otimes l}}T_0 - P_{S^{\otimes l}}T^*)(\overline{P_S u_0}^{\otimes l})\right| \\
\leqslant &\left|\sum_{\tau=1}^{t}\left((P_{S^{\otimes l}}T_{\tau-1} - P_{S^{\otimes l}}T^*)(\overline{P_S u_\tau}^{\otimes l}) - (P_{S^{\otimes l}}T_{\tau-1} - P_{S^{\otimes l}}T^*)(\overline{P_S u_{\tau-1}}^{\otimes l})\right)\right| \\
&+ \left|\sum_{\tau=1}^{t}\left((P_{S^{\otimes l}}T_{\tau} - P_{S^{\otimes l}}T^*)(\overline{P_S u_\tau}^{\otimes l}) - (P_{S^{\otimes l}}T_{\tau-1} - P_{S^{\otimes l}}T^*)(\overline{P_S u_\tau}^{\otimes l})\right)\right| \\
\leqslant &H\mu l^4 2^l d^{l-1.5} m^{1/2}(m+K)^{l-1}\eta\delta\lambda + \sum_{\tau=1}^{t}\|T_\tau - T_{\tau-1}\| \\
\leqslant &H\mu l^4 2^l d^{l-1.5} m^{1/2}(m+K)^{l-1}\eta\delta\lambda + \frac{\epsilon}{10(\mu_1 rl)^{l/2}} \leqslant \frac{\epsilon}{5(\mu_1 rl)^{l/2}}.
\end{aligned}$$

The second inequality above comes from Lemma A.11, and the last inequality assumes $\delta \leqslant \frac{1}{\mu l^4 2^l d^{l-1.5} m^{1/2}(m+K)^{l-1}\eta\lambda H} \cdot \frac{\epsilon}{10(\mu_1 rl)^{l/2}}$.

This then implies that for all $0 \leqslant t \leqslant H-1$,

$$a(P_{S^{\otimes l}}T_t - P_{S^{\otimes l}}T^*)(\overline{P_S u_t}^{\otimes l}) \leqslant -\frac{\epsilon}{2(\mu_1 rl)^{l/2}} + \frac{\epsilon}{5(\mu_1 rl)^{l/2}} \leqslant -\frac{\epsilon}{5(\mu_1 rl)^{l/2}}.$$

Then according to Lemma A.10,

$$\begin{aligned}\|P_S u_H\|^2 &\geqslant \left(1+\eta\left(\frac{\mu_2}{2}\right)^{l-2}\frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right)^H \|P_S u_0\|^2 \\ &\geqslant \left(1+\eta\left(\frac{\mu_2}{2}\right)^{l-2}\frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right)^H \frac{\mu_2^2\delta^2}{d} \\ &\geqslant \exp\left(\frac{1}{2}\eta H\left(\frac{\mu_2}{2}\right)^{l-2}\frac{\epsilon}{10(\mu_1 r l)^{l/2}}\right)\frac{\mu_2^2\delta^2}{d},\end{aligned}$$

where the last inequality assumes $\eta \leqslant \left(\frac{2}{\mu_2}\right)^{l-2}\frac{10(\mu_1 r l)^{l/2}}{\epsilon}$. Therefore, $\|P_S u_H\|^2$ exceeds M_2 as long as $\eta H \geqslant 2\left(\frac{2}{\mu_2}\right)^{l-2}\frac{10(\mu_1 r l)^{l/2}}{\epsilon}\log\left(\frac{dM_2}{\mu_2^2\delta^2}\right)$. Since $M_2 = \sqrt{\frac{10}{\lambda}}$ is the upper bound of $\|U\|_F$, this finishes the contradiction proof.

We have shown that

$$f(U_0,\bar{C}_0) - f(U_H,\bar{C}_H) \geqslant \frac{\lambda}{32000000(\mu_1 r l)^l \eta H l^5}\epsilon^2.$$

In order to show $f(U_0',\bar{C}_0') - f(U_H,\bar{C}_H)$ is large, we still need to bound $|f(U_0',\bar{C}_0') - f(U_0,\bar{C}_0)|$ that comes from reinitialization. According to Lemma A.6, we know

$$|f(U_0',\bar{C}_0') - f(U_0,\bar{C}_0)| \leqslant \frac{200}{\lambda m} \leqslant \frac{\lambda}{64000000(\mu_1 r l)^l \eta H l^5}\epsilon^2,$$

where the second inequality assumes $\lambda^2 m \geqslant 1.28\times 10^{11}(\mu_1 r l)^l \eta H l^5$. Therefore,

$$\begin{aligned}&f(U_H,\bar{C}_H) - f(U_0',\bar{C}_0') \\ \leqslant& \big(f(U_0,\bar{C}_0) - f(U_H,\bar{C}_H)\big) + |f(U_0,\bar{C}_0) - f(U_0',\bar{C}_0')| \\ \leqslant& -\frac{\lambda}{3.2\times 10^7(\mu_1 r l)^l \eta H l^5}\epsilon^2 + \frac{\lambda}{6.4\times 10^8(\mu_1 r l)^l \eta H l^5}\epsilon^2 \\ \leqslant& -\frac{\lambda}{6.4\times 10^7(\mu_1 r l)^l \eta H l^5}\epsilon^2.\end{aligned}$$

We choose $m = O\left(\frac{r^{2.5l}}{\epsilon^5}\log(d/\epsilon)\right)$, $\lambda = O\left(\frac{\epsilon}{r^{0.5l}}\right)$,

$\delta = O\left(\frac{\epsilon^{5l-1.5}}{d^{l-1.5}(\log(d/\epsilon))^{l+0.5} r^{2.5l^2-0.75l}}\right)$, $\eta = O\left(\frac{\epsilon^{15l-4.5}}{d^{3l-4.5}(\log(d/\epsilon))^{3l+1.5} r^{7.5l^2-2.25l}}\right)$,

$H = O\left(\frac{d^{3l-4.5}(\log(d/\epsilon))^{3l+2.5}r^{7.5l^2-1.75l}}{\epsilon^{15l-3.5}}\right)$ and $K = O\left(\frac{r^{2l}}{\epsilon^4}\log(d/\epsilon)\right)$ such that all the conditions are satisfied and the function value decreases by $\Omega(\frac{\epsilon^4}{r^{2l}\log(d/\epsilon)})$ in each epoch. Note that there does exist some circular dependency between the parameters. This turns out to be not an issue in our proof because for example δ depends on $\frac{1}{\eta H}$ while ηH only depends logarithmically on $1/\delta$. Other circular dependencies can be resolved in the same manner. □

A.4 Tools

In this section, we give the technical lemmas we use in the proof.

A.4.1 Random projection on a subspace

We use the following lemma to show that with good probability, the projection of the reinitialized component on the good subspace is lower bounded.

Lemma A.14 (Lemma 2.2 in Dasgupta and Gupta (2003))**.** *Let Y be a d-dimensional vector uniformly sampled from sphere $\mathbb{S}^{d-1}$. Let $Z \in \mathbb{R}^k$ be the projection of Y onto its first k coordinates ($k < d$). For any $\beta < 1$, we have*

$$\Pr\left[\|Z\|^2 \leqslant \frac{\beta k}{d}\right] \leqslant \exp\left(\frac{k}{2}(1-\beta+\ln\beta)\right).$$

A.4.2 Norm of random Gaussian vectors

The following lemma gives the concentration of ℓ_2 norm of a random Gaussian vector.

Lemma A.15 (Theorem 3.1.1 in Vershynin (2018))**.** *Let $X = (X_1, X_2, \cdots, X_n) \in \mathbb{R}^n$ be a random vector with each entry independently sampled from $\mathcal{N}(0,1)$. Then*

$$\Pr\left[\left|\|x\| - \sqrt{n}\right| \geqslant t\right] \leqslant 2\exp\left(-t^2/C^2\right),$$

where C is an absolute constant.

A.4.3 Anti-concentration of Gaussian polynomials

We use anti-concentration of Gaussian polynomials to argue that a randomly initialized component has good correlation with the residual.

Lemma A.16 (Theorem 8 in Carbery and Wright (2001))**.** *Let $x \in \mathbb{R}^n$ be a Gaussian variable $x \in N(0, I)$, for any polynomial $p(x)$ of degree l, there exists a constant κ such that*

$$\Pr\left[|p(x)| \leqslant \epsilon\sqrt{Var[p(x)]}\right] \leqslant \kappa l\epsilon^{1/l}.$$

Appendix B

Supplementary materials for Chapter 3

Overview of Supplementary Materials

In the supplementary material we will give detailed proof for Theorem 3.1. We will first highlight a few technical ideas that goes into the proof, and then give details for each part of the proof.

Continuity Argument Continuity argument is the main tool we use to prove Proposition 3.1. Intuitively, the continuity argument says that if whenever a property is about to be violated, there exists a positive speed that pulls it back, then that property will never be violated. In some sense, this is the continuous version of the mathematical induction or, equivalently, the minimal counterexample method. See Section 1.3 of Tao (2006) for a short discussion on this method.

However, since our algorithm is not just gradient flow, and in particular involves reinitialization steps that are not continuous, we need to generalize continuity argument to handle impulses. We give detailed lemmas in Section B.1.1 as the continuity argument is mostly used to prove Proposition 3.1.

Approximating residual In many parts of the proof, we approximate the residual $T^* - T$ as:

$$T^* - T = \sum_{i=1}^{d} \tilde{a}_i e_i^{\otimes 4} + \Delta,$$

where $\tilde{a}_i = a_i - \hat{a}_i$. That is, we think of $T^* - T$ as an orthogonal tensor with some perturbations. The norm of the perturbation $\|\Delta\|_F$ is going to be bounded by $O(\alpha + m\delta_1^2)$, which is sufficient in several parts of the proof that only requires crude estimates. However, in several key steps of our proof (including conditions (a) and (b) of Proposition 3.1 and the analysis of the first phase), it is important to use extra properties of Δ. In particular we will expand Δ to show that for a basis vector e_i we always have $\Delta(e_i^{\otimes 4}) = o(\alpha)$, which gives us tighter bounds when we need them.

Radial and tangent movement Throughout the proof, we often need to track the movement of a particular component w (a column in W). It is beneficial to separate the movement of w into radial and tangent movement, where radial movement is defined as $\left\langle \frac{dw}{dt}, w \right\rangle$ and tangent movement is defined as $P_{w^\perp} \frac{dw}{dt}$ (where $P_{w^\perp}$ is the projection to the orthogonal subspace of w). Intuitively, the radial movement controls the norm of the component w, and the tangent movement controls the direction of w. When the component w has small norm, it will not significantly change the residual $T^* - T$, therefore we mostly focus on the tangent movement; on the other hand when norm of w becomes large in our proof we show that it must already be correlated with one of the ground truth components, which allow us to better control its norm growth.

Overall structure of the proof The entire proof is a large induction/continuity argument which maintains Proposition 3.1 as well as properties of the two phases (summarized later in Assumption B.1). In each part of the proof, we show that if we

assume these conditions hold for the previous time, then they will continue to hold during the phase/after reinitialization.

In Section B.1 we prove Proposition 3.1 assuming Assumption B.1 holds before. In Section B.2.2 we prove guarantees of Phase 1 and reinitialization assuming Proposition 3.1. In Seciton B.3 we prove guarantees for Phase 2 assuming Proposition 3.1. Finally in Section B.4 we give the proof of the main theorem.

Experiments Finally in Section B.5.1 we give details about experiments that illustrate the deflation process, and show why such a process may not happen for non-orthgonal tensors.

B.1 Proofs for Proposition 3.1

The goal of this section is to prove Proposition 3.1 under Assumption B.1. We also prove Claim 3.1 in Section B.1.6.

Notations Recall we defined

$$\mathbb{E}_{i,w}^{(s,t)} f(w^{(s,t)}) := \frac{1}{\hat{a}_i^{(s,t)}} \sum_{w^{(s,t)} \in S_i^{(s,t)}} \left\| w^{(s,t)} \right\|^2 f(w^{(s,t)}).$$

We will use this notation extensively in this section. For simplicity, we shall drop the superscript of epoch s. Further, we sometimes consider expectation with two variables v and w:

$$\mathbb{E}_{i,v,w}^{(s,t)} f(w^{(s,t)}) := \frac{1}{\left[\hat{a}_i^{(s,t)}\right]^2} \sum_{v^{(s,t)}, w^{(s,t)} \in S_i^{(s,t)}} \left\| v^{(s,t)} \right\|^2 \left\| w^{(s,t)} \right\|^2 f(w^{(s,t)}, v^{(s,t)}).$$

We will also use z_t to denote $z^{(t)} := \left\langle \bar{v}^{(t)}, \bar{w}^{(t)} \right\rangle$ and $\tilde{a}_k^{(t)} := a_k - \hat{a}_k^{(t)}$. Note that v and w in this section (and later in the proof) just serve as arbitrary components in columns of W.

Assumption B.1. *Throughout this section, we assume the following.*

(a) For any $k \in [d]$, in phase 1, when $\|v^{(t)}\|$ enters $S_k^{(t)}$, that is, $\|v^{(t)}\| = \delta_1$, we have $[\bar{v}_k^{(t)}]^2 \geqslant 1 - \alpha^2$ if $\hat{a}_k^{(t)} < \alpha$ and $[\bar{v}_k^{(t)}]^2 \geqslant 1 - \alpha$ if $\hat{a}_k^{(t)} \geqslant \alpha$.

(b) There exists a small constant $c > 0$ s.t. for any $k \in [d]$ with $a_k < c\beta^{(s)}$, in phase 1, no components will enter $S_k^{(t)}$.

(c) For any $k \in [d]$, in phase 2, no components will enter $S_k^{(t)}$.

(d) For the parameters, we assume $m\delta_1^2 \leqslant \alpha^3$ and $\Omega\left(\sqrt{\alpha}\right) \leqslant \lambda \leqslant O\left(\min_s \beta^{(s)}\right) = O(\varepsilon/\sqrt{d})$.

Remark. As we mentioned, the entire proof is an induction and we only need the assumption up to the point that we are analyzing. The assumption will be proved later in Appendix B.2 and B.3 to finish the induction/continuity argument. The reason we state this assumption here, and state it as an assumption, is to make the dependencies more transparent.

Remark on the choice of λ. The lower bound $\lambda = \Omega(\sqrt{\alpha})$ comes from Lemma B.1. For the upper bound, first note that when λ is larger than a_k, actually the norm of components in $S_k^{(t)}$ can decrease (cf. Lemma B.6). Hence, we require $\lambda < c \min_s \beta^{(s)}/10$ where c is the constant in (c). This makes sure in phase 2 the growth rate of $\hat{a}_k^{(t)}$ is not too small.

Proposition 3.1 (Induction hypothesis)**.** *In the setting of Theorem 3.1, for any epoch s and time t and every $k \in [d]$, the following hold.*

(a) For any $w^{(s,t)} \in S_k^{(s,t)}$, we have $\left[\bar{w}_k^{(s,t)}\right]^2 \geqslant 1 - \alpha$.

(b) If $S_k^{(s,t)}$ is nonempty, $\mathbb{E}_{k,w}^{(s,t)}\left[\bar{w}_k^{(s,t)}\right]^2 \geqslant 1 - \alpha^2 - 4sm\delta_1^2$.

(c) We always have $a_k - \hat{a}_k^{(s,t)} \geqslant \lambda/6 - sm\delta_1^2$*; if* $a_k \geqslant \frac{\beta^{(s)}}{1-\gamma}$*, we further know* $a_k - \hat{a}_k^{(s,t)} \leqslant \lambda + sm\delta_1^2$.

(d) If $w^{(s,t)} \in S_{\varnothing}^{(s,t)}$*, then* $\|w^{(s,t)}\| \leqslant \delta_1$.

Before we move on to the proof, we collect some further remarks on Proposition 3.1 and the proof overview here.

Remark on the epoch correction term. Note that conditions (b) and (c) have an additional term with form $O(sm\delta_1^2)$. This is because these average bounds may deteriorate a little when the content of $S_k^{(t)}$ changes, which will happen when new components enter $S_k^{(t)}$ or the reinitialization throw some components out of $S_k^{(t)}$. The norm of the components involved in these fluctuations is upper bounded by δ_1 and the number by m. Thus the $O(m\delta_1^2)$ factor. The factor s accounts for the accumulation across epochs. We need this to guarantee at the beginning of each epoch, the conditions hold with some slackness (cf. Lemma B.5). Though this issue can be fixed by a slightly sharper estimations for the ending state of each epoch, adding one epoch correction term is simpler and, since we only have $\log(d/\epsilon)$ epochs, it does not change the bounds too much and, in fact, we can always absorb them into the coefficients of λ and α^2, respectively.

Remark on condition (a). Note that Assumption B.1 makes sure that when a component enters $S_k^{(t)}$, we always have $[\bar{v}_k^{(t)}]^2 \geqslant 1 - \alpha$. Hence, essentially this condition says that it will remain basis-like. Following the spirit of the continuity argument, to maintain this condition, it suffices to prove Lemma B.1, the proof of which is deferred to Section B.1.3. Also note that by Assumption B.1 and the definition of $S_k^{(s,t)}$, neither the entrance of new components nor the reinitialization will break this condition.

Lemma B.1. *Suppose that at time t, Proposition 3.1 is true. Assuming $\delta_1^2 = O(\alpha^{1.5}/m)$, then for any $v^{(t)} \in S_k^{(t)}$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 \geqslant 8\tilde{a}^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - O\left(\alpha^{1.5}\right),$$

In particular, if $\lambda = \Omega\left(\sqrt{\alpha}\right)$, then $\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 > 0$ whenevner $[\bar{v}_k^{(t)}]^2 = 1-\alpha$.

Remark on condition (b). The proof idea of condition (b) is similar to condition (a) and we prove Lemma B.2 in Section B.1.4. In Section B.1.4, we also handle the impulses caused by the entrance of new components and the reinitialization.

Lemma B.2. *Suppose that at time t, Proposition 3.1 is true and $S_k^{(t)} \neq \varnothing$. Assuming $\delta_1^2 = O(\alpha^3/m)$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 \geqslant 8\tilde{a}_k^{(t)}(1-\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2) - O(\alpha^3).$$

In particular, if $\lambda = \Omega(\alpha)$, then $\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 > 0$ when $\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 < 1-\alpha^2/2$.

Remark on condition (c). This condition says that the residual along direction k is always $\Omega(\lambda)$. This guarantees the existence of a small attraction region around e_k, which will keep basis-like components basis-like. We rely on the regularizer to maintain this condition. The second part of condition (c) means fitted directions will remain fitted. We prove Lemma B.3 and handle the impulses in Section B.1.5.

Lemma B.3 (Lemma B.17 and Lemma B.18)**.** *Suppose that at time t, Proposition 3.1 is true. and no impulses happen at time t. Then at time t, we have*

$$\frac{1}{\hat{a}_k^{(t)}}\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} = 2\tilde{a}_k^{(t)} - \lambda \pm O\left(\alpha^2\right).$$

In particular, $\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)}$ is negative (resp. positive) when $\hat{a}_k^{(t)} > a_k - \lambda/6$ (resp. $\hat{a}_k^{(t)} < a_k - \lambda$).

B.1.1 Continuity argument

We mostly use the following version of continuity argument, which is adapted from Proposition 1.21 of Tao (2006).

Lemma B.4. *Let $\mathbf{I}^{(t)}$ be a statement about the structure of some object. $\mathbf{I}^{(t)}$ is true for all $t \geqslant 0$ as long as the following hold.*

(a) $\mathbf{I}^{(0)}$ is true.

(b) $\mathbf{I}$ is closed in the sense that for any sequence $t_n \to t$, if $\mathbf{I}^{(t_n)}$ is true for all n, then $\mathbf{I}^{(t)}$ is also true.

(c) If $\mathbf{I}^{(t)}$ is true, then there exists some $\delta > 0$ s.t. $\mathbf{I}^{(s)}$ is true for $s \in [t, t+\delta)$.

In particular, if $\mathbf{I}^{(t)}$ has form $\bigwedge_{i=1}^{N} \bigvee_{j=1}^{N} p_{i,j}^{(t)} \leqslant q_{i,j}$. Then, we can replace (b) and (c) by the following.

(b') $p_{i,j}^{(t)}$ is C^1 for all i, j.

(c') Suppose at time t, $\mathbf{I}^{(t)}$ is true but some clause $\bigvee_{j=1}^{N} p_{i,j}^{(t)} \leqslant q_{i,j}$ is tight, in the sense that $p_{i,j}^{(t)} \geqslant q_{i,j}$ for all j with at least one equality. Then there exists some k s.t. $p_{i,k}^{(t)} = q_{i,k}$ and $\dot{p}_{i,k}^{(t)} < 0$.

Proof. Define $t' := \sup\{t \geqslant 0 \ : \ \mathbf{I}^{(t)} \text{ is true}\}$. Since $\mathbf{I}^{(0)}$ is true, $t' \geqslant 0$. Assume, to obtain a contradiction, that $t' < \infty$. Since $\mathbf{I}$ is closed, $\mathbf{I}^{(t')}$ is true, whence there exists a small $\delta > 0$ s.t. $\mathbf{I}^{(t)}$ is true in $[t', t'+\delta)$. Contradiction.

For the second set of conditions, first note that the continuity of $p_{i,j}^{(t)}$ and the non-strict inequalities imply that $\mathbf{I}$ is closed. Now we show that (b') and (c') imply (c). If none of the clause is tight at time t, by the continuity of $p_{i,j}^{(t)}$, $\mathbf{I}$ holds in a small neighborhood of t. If some constraint is tight, by (c') and the C^1 condition, we have $p_{i,k}^{(t)} < q_{i,k}$ in a right small neighborhood of t. □

Remark. Despite the name "continuity argument", it is possible to generalize it to certain classes of discontinuous functions. In particular, we consider impulsive differential equations here, that is, for almost every t, $p^{(t)}$ behaves like a usual differential equation, but at some t_i, it will jump from $p^{(t_i-)}$ to $p^{(t_i)} = p^{(t_i-)} + \delta_i$. See, for example, Lakshmikantham et al. (1989) for a systematic treatment on this topic. Suppose that we still want to maintain the property $p^{(t)} \leqslant 0$. If the total amount of impulses is small and we have some cushion in the sense that $\dot{p}^{(t)} < 0$ whenever $p^{(t)} \in [-\varepsilon, 0]$, then we can still hope $p^{(t)} \leqslant 0$ to hold for all t, since, intuitively, only the jumps can lead $p^{(t)}$ into $[-\varepsilon, 0]$, and the normal $\dot{p}^{(t)}$ will try to take it back to $(-\infty, -\varepsilon)$. As long as the amount of impulses is smaller than the size ε of the cushion, then the impulses will never break things. We formalize this idea in the next lemma.

Lemma B.5 (Continuity argument with impulses)**.** *Let $0 < t_1 < \cdots < t_N < \infty$ be the moments at which the impulse happens and $\delta_1, \ldots, \delta_N \in \mathbb{R}$ the size of the impulses at each t_i. Let $p : [0, \infty) \to \mathbb{R}$ be a function that is C^1 on $[0, t_1)$, every (t_i, t_{i+1}) and (t_N, ∞), and $p^{(t_i)} = p^{(t_i-)} + \delta_i$. Write $\Delta = \sum_{i=1}^{N} \max\{0, \delta_i\}$. If (a) $p^{(0)} \leqslant -\Delta$ and (b) for every $t \notin \{t_i\}_{i=1}^{N}$ with $p^{(t)} \in [-\Delta, 0]$, we have $\dot{p}^{(t)} < 0$, then $p^{(t)} \leqslant 0$ always holds.*

Remark. Note that if there is no impulses, then $p^{(t)}$ is a usual C^1 function and we recover conditions (b') and (c') of Lemma B.4. Also, though the statement here only concerns one a_t, one can incorporate it into Lemma B.4 by replacing (b') and (c') with the hypotheses of this lemma and modify (a) to be $p_{i,j}^{(0)} \leqslant p_{i,j} - \Delta_{i,j}$.

Proof. We claim that $p^{(t)} \leqslant -\Delta + \sum_{i=1}^{N} \mathbb{1}_{t \leqslant t_k} \max\{0, \delta_i\} =: q^{(t)}$. Define $t' = \sup\{t \geqslant 0 \ : \ p^{(t)} \leqslant q^{(t)}\}$. Since $p^{(t)} \leqslant -\Delta$ and $t_1 > 0$, $t' \geqslant 0$. Assume, to obtain a contradiction, that $t' < \infty$ and consider $p^{(t')}$. If $t' = t_k$ for some k, then, by the definition of t', $p^{(t'-)} \leqslant -\Delta + \sum_{i=1}^{k-1} \max\{0, \delta_i\}$, whence, $p^{(t')} = p^{(t'-)} + \delta_k \leqslant -\Delta + \sum_{i=1}^{k} \max\{0, \delta_i\}$. Contradiction. If $t' \notin \{t_i\}_{i=1}^{N}$, then by the continuity of p, we have

$p^{(t')} = q^{(t')}$. Then, since $\dot{p}^{(t')} < 0$ and p is C^1, we have $p^{(t)} < p^{(t')} = q^{(t')} = q^{(t)}$ in $[t', t'+\tau]$ for some small $\tau > 0$, which contradicts the maximality of t'. Thus, $p^{(t)} \leqslant 0$ holds for all $t \geqslant 0$. □

B.1.2 Preliminaries

The next two lemmas give formulas for the norm growth rate and tangent speed of each component.

Lemma B.6 (Norm growth rate)**.** *For any $v^{(t)}$, we have*

$$\frac{1}{2\left\|v^{(t)}\right\|^2}\frac{\mathrm{d}}{\mathrm{d}t}\left\|v^{(t)}\right\|^2 = \sum_{i=1}^{d} a_i[\bar{v}_i^{(t)}]^4 - \sum_{i=1}^{d} \hat{a}_i^{(t)}\mathbb{E}_{i,w}^{(t)}\left\{[z^{(t)}]^4\right\} - T_{\varnothing}^{(t)}\left([\bar{v}^{(t)}]^{\otimes 4}\right) - \frac{\lambda}{2}.$$

Proof. Due to the 2-homogeneity, we have[1]

$$\frac{1}{2\left\|v^{(t)}\right\|^2}\frac{\mathrm{d}}{\mathrm{d}t}\left\|v^{(t)}\right\|^2 = \left(T^* - T^{(t)}\right)\left([\bar{v}^{(t)}]^{\otimes 4}\right) - \frac{\lambda}{2}.$$

The ground truth terms can be rewritten as

$$T^*\left([\bar{v}^{(t)}]^{\otimes 4}\right) = \sum_{i=1}^{d} a_i[\bar{v}_i^{(t)}]^4.$$

Decompose the $T^{(t)}$ term accordingly and we get

$$T^{(t)}\left([\bar{v}^{(t)}]^{\otimes 4}\right) = \sum_{i=1}^{d} \hat{a}^{(t)}\mathbb{E}_{i,w}^{(t)}\left\{[z^{(t)}]^4\right\} + T_{\varnothing}^{(t)}\left([\bar{v}^{(t)}]^{\otimes 4}\right).$$

□

Lemma B.7 (Tangent speed)**.** *Suppose that at time t, Proposition 3.1 is true. Then at time t, for any $v^{(t)} \in W^{(t)}$ and any $k \in [d]$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 = G_1 - G_2 - G_3 \pm O(m\delta_1^2),$$

[1] In the mean-field terminologies, the RHS is just the first variation (or functional derivative) of the loss at $\bar{v}^{(t)}$.

where

$$G_1 := 8a_k\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - 8\hat{a}_k^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)\mathbb{E}_{k,w}^{(t)}\left\{[z^{(t)}]^4\right\}$$
$$+ 8\hat{a}_k^{(t)}\mathbb{E}_{k,w}^{(t)}\left\{[z^{(t)}]^3\langle \bar{w}_{-k}, \bar{v}_{-k}\rangle\right\},$$
$$G_2 = 8\sum_{i\neq k}\hat{a}_i^{(t)}\mathbb{E}_{i,w}^{(t)}\left\{[z^{(t)}]^3 v_k^{(t)} w_k^{(t)}\right\},$$
$$G_3 = 8[\bar{v}_k^{(t)}]^2\sum_{i\neq k}\left(a_i[\bar{v}_i^{(t)}]^4 - \hat{a}_i^{(t)}\mathbb{E}_{i,w}^{(t)}\left\{[z^{(t)}]^4\right\}\right).$$

Remark. Intuitively, G_1 captures the local dynamics around e_k and G_2 characterize the cross interaction between different ground truth directions.

Proof. Let's compute the derivative of $[\bar{v}_k^{(t)}]^2$ in terms of time t:

$$\begin{aligned}
\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} &= 2\bar{v}_k^{(t)}\cdot\frac{d}{dt}\frac{v_k^{(t)}}{\|v^{(t)}\|}\\
&= 2\bar{v}_k^{(t)}\cdot\frac{1}{\|v^{(t)}\|}\frac{d}{dt}v_k^{(t)} + 2[\bar{v}_k^{(t)}]^2\cdot\frac{d}{dt}\frac{1}{\|v^{(t)}\|}\\
&= 2\bar{v}_k^{(t)}\cdot\frac{1}{\|v^{(t)}\|}[-\nabla L(v^{(t)})]_k - 2[\bar{v}_k^{(t)}]^2\cdot\frac{\left\langle\bar{v}^{(t)}, -\nabla L(v^{(t)})\right\rangle}{\|v^{(t)}\|}\\
&= 2\bar{v}_k^{(t)}\cdot\frac{1}{\|v^{(t)}\|}[-(I-\bar{v}^{(t)}[\bar{v}^{(t)}]^\top)\nabla L(v^{(t)})]_k.
\end{aligned}$$

Note that

$$\nabla f(v^{(t)}) = 4(T^{(t)}-T^*)([\bar{v}^{(t)}]^{\otimes 2}, \bar{v}^{(t)}, I) - 2(T^{(t)}-T^*)([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)} + \lambda\bar{v}^{(t)},$$

where the last two terms left multiplied by $(I-\bar{v}^{(t)}[\bar{v}^{(t)}]^\top)$ equals to zero. Therefore,

$$\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} = 8\bar{v}_k^{(t)}\left[(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 3}), I) - (T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k$$

We can write T^* as $\sum_{i\in[d]} a_i e_i^{\otimes 4}$ and write $T^{(t)}$ as $\sum_{i\in[d]} T_i^{(t)} + T_\varnothing^{(t)}$. Since Proposition 3.1 is true at time t, we know any $w^{(t)}$ in $W_\varnothing^{(t)}$ has norm upper bounded by δ_1,

which implies $\left\|T_{\varnothing}^{(t)}\right\|_F \leqslant m\delta_1^2$. Therefore, we have

$$\left|8\bar{v}_k^{(t)}\left[-T_{\varnothing}^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) + T_{\varnothing}^{(t)}([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k\right| \leqslant O(m\delta_1^2).$$

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

$$\begin{aligned}\left[T_i^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I)\right]_k =& \sum_{w^{(t)} \in S_i^{(t)}} \left\|w^{(t)}\right\|^2 \left\langle \bar{w}^{(t)}, \bar{v}^{(t)}\right\rangle^3 \bar{w}_k^{(t)} \\ =& \hat{a}_k^{(t)} \mathbb{E}_{k,w}^{(t)} \left\langle \bar{w}^{(t)}, \bar{v}^{(t)}\right\rangle^3 \bar{w}_k^{(t)},\end{aligned}$$

and

$$\begin{aligned}\left[T_i^{(t)}([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k =& \sum_{w^{(t)} \in S_i^{(t)}} \left\|w^{(t)}\right\|^2 \left\langle \bar{w}^{(t)}, \bar{v}^{(t)}\right\rangle^4 \bar{v}_k^{(t)} \\ =& \hat{a}_k^{(t)} \mathbb{E}_{k,w}^{(t)} \left\langle \bar{w}^{(t)}, \bar{v}^{(t)}\right\rangle^4 \bar{v}_k^{(t)}.\end{aligned}$$

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

$$\left[T^*([\bar{v}^{(t)}]^{\otimes 3}, I)\right]_k = [\bar{v}_k^{(t)}]^3 \mathbb{1}_{i=k}$$

and

$$\left[T^*([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k = [\bar{v}_i^{(t)}]^4 \bar{v}_k^{(t)}$$

Based on the above calculations, we can see that

$$\begin{aligned}G_1 &= 8\bar{v}_k^{(t)}\left[(T_k^* - T_k^{(t)})([\bar{v}^{(t)}]^{\otimes 3}, I) - (T_k^* - T_k^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k \\ G_2 &= 8\bar{v}_k^{(t)}\left[\sum_{i \neq k} T_i^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I)\right]_k \\ G_3 &= 8[\bar{v}_k^{(t)}]^2 \sum_{i \neq k} (T_i^* - T_i^{(t)})([\bar{v}^{(t)}]^{\otimes 4}),\end{aligned}$$

and the error term $O(m\delta_1^2)$ comes from $T_{\varnothing}^{(t)}$. To complete the proof, use the identity $\langle \bar{w}, \bar{v}\rangle = \bar{w}_k\bar{v}_k + \langle \bar{w}_{-k}, \bar{v}_{-k}\rangle$ to rewrite G_1. □

One may wish to skip all following estimations and come back to them when needed.

Lemma B.8. *For any $\bar{v}$ with $\bar{v}_k^2 \geqslant 1-\alpha$ and any $\bar{w} \in \mathbb{S}^{d-1}$, we have $|\langle \bar{v}, \bar{w}\rangle| = |\bar{w}_k| \pm \sqrt{\alpha}$.*

Proof. Assume w.o.l.g. that $k = 1$. Note that the set $\{\bar{v} \in \mathbb{S}^{d-1} \ : \ \bar{v}_k^2 \geqslant 1-\alpha\}$ is invariant under rotation of other coordinates, whence we may further assume w.o.l.g. that $\bar{w} = \bar{w}_1 e_1 + \sqrt{1-\bar{w}_1^2} e_2$. Then,

$$\begin{aligned}
|\langle \bar{w}, \bar{v}\rangle| &= \left|\bar{w}_1\bar{v}_1 + \sqrt{1-\bar{v}_1^2}\sqrt{1-\bar{w}_1^2}\right| \\
&\geqslant |\bar{w}_1|\sqrt{1-\alpha} - \sqrt{\alpha}\sqrt{1-\bar{w}_1^2} \\
&= \frac{\bar{w}_1^2(1-\alpha) - \alpha(1-\bar{w}_1^2)}{|\bar{w}_1|\sqrt{1-\alpha} + \sqrt{\alpha}\sqrt{1-\bar{w}_1^2}} \\
&= \frac{\bar{w}_1^2 - \alpha}{|\bar{w}_1|\sqrt{1-\alpha} + \sqrt{\alpha}\sqrt{1-\bar{w}_1^2}} \geqslant \frac{\bar{w}_1^2 - \alpha}{|\bar{w}_1| + \sqrt{\alpha}} = |\bar{w}_1| - \sqrt{\alpha}.
\end{aligned}$$

The other direction follows immediately from

$$|\langle \bar{w}, \bar{v}\rangle| \leqslant |\bar{w}_1||\bar{v}_1| + \left|\sqrt{1-\bar{v}_1^2}\sqrt{1-\bar{w}_1^2}\right| \leqslant |\bar{w}_1| + \sqrt{\alpha}.$$

□

The next two lemmas bound the cross interaction between different $S_k^{(t)}$.

Lemma B.9. *Suppose that at time t, Proposition 3.1 is true. Then for any $v^{(t)} \in S_k^{(t)}$ and $l \neq k$, the following hold.*

(a) $[\bar{v}_l^{(t)}]^4 \leqslant \alpha^2$.

(b) $\mathbb{E}_{l,w}^{(t)}\{[z_t]^4\} \leqslant O(\alpha^2)$.

(c) $\mathbb{E}_{l,w}^{(t)}\left\{[z_t]^3\bar{v}_l\bar{w}_l\right\} \leqslant O(\alpha^2)$.

Proof. (a) follows immediately from $[v_l^{(t)}]^4 \leqslant (1-[v_l^{(t)}]^2) \leqslant \alpha^2$. For (b), apply Lemma B.8 and we get

$$\begin{aligned}
&\mathbb{E}_{l,w}^{(t)}\left\{[z_t]^4\right\}\\
\leqslant&\mathbb{E}_{l,w}^{(t)}\left\{\left(|\bar{w}_k|+\sqrt{\alpha}\right)^4\right\}\\
\leqslant&\mathbb{E}_{l,w}^{(t)}\left\{[\bar{w}_k]^4+4|\bar{w}_k|^3\sqrt{\alpha}+6[\bar{w}_k]^2\alpha+4|\bar{w}_k|\alpha^{1.5}+\alpha^2\right\}.
\end{aligned}$$

For the first three terms, it suffices to note that $\mathbb{E}_{l,w}^{(t)}\left\{[\bar{w}_k]^2\right\} \leqslant \alpha^2$. For the fourth term, it suffices to additionally recall Jensen's inequality. Combine these together and we get $\mathbb{E}_{l,w}^{(t)}\left\{[z_t]^4\right\} = O(\alpha^2)$. The proof of (b), *mutatis mutandis*, yields (c). □

Lemma B.10. *Suppose that at time t, Proposition 3.1 is true. Then for any $k \neq l$, the following hold.*

(a) $\mathbb{E}_{k,v}^{(t)}[\bar{v}_l^{(t)}]^4 \leqslant O(\alpha^3)$.

(b) $\mathbb{E}_{k,v}^{(t)}\mathbb{E}_{l,w}^{(t)}[z^{(t)}]^4 \leqslant O(\alpha^3)$.

(c) $\mathbb{E}_{k,v}^{(t)}\mathbb{E}_{l,w}^{(t)}\left\{[z^{(t)}]^3\bar{v}_k\bar{w}_k\right\} \leqslant O(\alpha^3)$.

Proof. For (a), we compute

$$\mathbb{E}_{k,v}^{(t)}[\bar{v}_l^{(t)}]^4 \leqslant \mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)^2\right\} \leqslant \alpha\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\} \leqslant O(\alpha^3),$$

where the second inequality comes from the condition (a) of Proposition 3.1 and the third from condition (b) of Proposition 3.1. Now we prove (b). (c) can be proved in a similar fashion. For simplicity, write $x^{(t)} = \left\langle \bar{w}_{-l}^{(t)}, \bar{v}_{-l}^{(t)} \right\rangle$. Clear that $|x^{(t)}| \leqslant \sqrt{1-[\bar{w}_l^{(t)}]^2}$ and by Jensen's inequality and condition (b) of Proposition 3.1,

$\mathbb{E}^{(t)}_{l,w}\sqrt{1-[\bar{w}^{(t)}_l]^2} \leqslant O(\alpha)$. We compute

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}[z^{(t)}]^4 = \mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}\Big\{[\bar{w}^{(t)}_l]^4[\bar{v}^{(t)}_l]^4 + 4[\bar{w}^{(t)}_l]^3[\bar{v}^{(t)}_l]^3x^{(t)} + 6[\bar{w}^{(t)}_l]^2[\bar{v}^{(t)}_l]^2[x^{(t)}]^2 + 4\bar{w}^{(t)}_l\bar{v}^{(t)}_l[x^{(t)}]^3 + [x^{(t)}]^4\Big\}.$$

We bound each of these five terms as follows.

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}\left\{[\bar{w}^{(t)}_l]^4[\bar{v}^{(t)}_l]^4\right\} \leqslant \mathbb{E}^{(t)}_{k,v}[\bar{v}^{(t)}_l]^4 \leqslant O(\alpha^3),$$

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}\left\{[\bar{w}^{(t)}_l]^3[\bar{v}^{(t)}_l]^3x^{(t)}\right\} \leqslant \mathbb{E}^{(t)}_{k,v}[\bar{v}^{(t)}_l]^3\mathbb{E}^{(t)}_{l,w}\left\{\sqrt{1-[\bar{w}^{(t)}_l]^2}\right\} \leqslant O(\alpha^3),$$

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}\left\{[\bar{w}^{(t)}_l]^2[\bar{v}^{(t)}_l]^2[x^{(t)}]^2\right\} \leqslant \mathbb{E}^{(t)}_{k,v}[\bar{v}^{(t)}_l]^2\mathbb{E}^{(t)}_{l,w}\left\{1-[\bar{w}^{(t)}_l]^2\right\} \leqslant O(\alpha^3),$$

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}\left\{\bar{w}^{(t)}_l\bar{v}^{(t)}_l[x^{(t)}]^3\right\} \leqslant \mathbb{E}^{(t)}_{k,v}\bar{v}^{(t)}_l\mathbb{E}^{(t)}_{l,w}\left\{\left(1-[\bar{w}^{(t)}_l]^2\right)^{1.5}\right\} \leqslant O(\alpha^3),$$

$$\mathbb{E}^{(t)}_{k,v}\mathbb{E}^{(t)}_{l,w}[x^{(t)}]^4 \leqslant \mathbb{E}^{(t)}_{l,w}\left\{\left(1-[\bar{w}^{(t)}_l]^2\right)^2\right\} \leqslant O(\alpha^3).$$

Combine these together and we complete the proof. □

Lemma B.11. *Suppose that at time t, Proposition 3.1 is true. Then, for any $v^{(t)} \in S^{(t)}_k$, we have $\mathbb{E}^{(t)}_{k,w}\left\{[z^{(t)}]^4\right\} = [\bar{v}^{(t)}_k]^4 \pm O(\alpha^{1.5})$.*

Proof. For simplicity, put $x^{(t)} = \left\langle \bar{w}^{(t)}_{-k}, \bar{v}^{(t)}_{-k}\right\rangle$. Note that

$|x^{(t)}| \leqslant \sqrt{1-[\bar{v}^{(t)}_k]^2}\sqrt{1-[\bar{w}^{(t)}_k]^2} \leqslant \sqrt{\alpha}\sqrt{1-[\bar{w}^{(t)}_k]^2}$. Then

$$\mathbb{E}^{(t)}_{k,w}\left\{[z^{(t)}]^4\right\} = \mathbb{E}^{(t)}_{k,w}\left\{\left[\bar{w}^{(t)}_k\bar{v}^{(t)}_k + x^{(t)}\right]^4\right\} = [\bar{v}^{(t)}_k]^4\mathbb{E}^{(t)}_{k,w}\left\{[\bar{w}^{(t)}_k]^4\right\} \pm O(1)\mathbb{E}^{(t)}_{k,w}x^{(t)}.$$

For the first term, note that

$$\mathbb{E}^{(t)}_{k,w}\left\{[\bar{w}^{(t)}_k]^4\right\} = 1 - \mathbb{E}^{(t)}_{k,w}\left\{(1-[\bar{w}^{(t)}_k]^2)(1+[\bar{w}^{(t)}_k]^2)\right\} \geqslant 1-2\alpha^2.$$

For the second term, by Jensen's inequality, we have

$$\left|\mathbb{E}_{k,w}^{(t)} x^{(t)}\right| \leqslant \sqrt{\alpha \mathbb{E}_{k,w}^{(t)}[1-[\bar{w}_k^{(t)}]^2]} \leqslant \alpha^{1.5}.$$

Thus,

$$\mathbb{E}_{k,w}^{(t)}\left\{[z^{(t)}]^4\right\} = [\bar{v}_k^{(t)}]^4\left(1 \pm 2\alpha^2\right) \pm O(\alpha^{1.5}) = [\bar{v}_k^{(t)}]^4 \pm O(\alpha^{1.5}).$$

□

Lemma B.12. *Suppose that at time t, Proposition 3.1 is true. Then we have* $\mathbb{E}_{k,v,w}^{(t)}\left\{[z^{(t)}]^4\right\} \geqslant 1 - O(\alpha^2)$.

Proof. For simplicity, put $x^{(t)} = \left\langle \bar{w}_{-k}^{(t)}, \bar{v}_{-k}^{(t)} \right\rangle$. We have

$$\begin{aligned}\mathbb{E}_{k,v,w}^{(t)}\left\{[z^{(t)}]^4\right\} &= \mathbb{E}_{k,v,w}^{(t)}\left\{\left(\bar{w}_k^{(t)}\bar{v}_k^{(t)} + x^{(t)}\right)^4\right\} \\ &\geqslant \mathbb{E}_{k,v,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^4[\bar{v}_k^{(t)}]^4 + [\bar{w}_k^{(t)}]^3[\bar{v}_k^{(t)}]^3 x + \bar{w}_k^{(t)}\bar{v}_k^{(t)}x^3\right\}.\end{aligned}$$

Note that

$$\begin{aligned}\mathbb{E}_{k,v,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^3[\bar{v}_k^{(t)}]^3 x\right\} &= \sum_{i\neq k}\mathbb{E}_{k,v,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^3[\bar{v}_k^{(t)}]^3\bar{w}_i^{(t)}\bar{v}_i^{(t)}\right\} \\ &= \sum_{i\neq k}\left(\mathbb{E}_{k,v,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^3\bar{w}_i^{(t)}\right\}\right)^2 \geqslant 0.\end{aligned} \tag{B.1}$$

Similarly, $\mathbb{E}_{k,v,w}^{(t)}\left\{\bar{w}_k^{(t)}\bar{v}_k^{(t)}x^3\right\} \geqslant 0$ also holds. Finally, by Jensen's inequality, we have

$$\begin{aligned}\mathbb{E}_{k,v,w}^{(t)}\left\{[z^{(t)}]^4\right\} &\geqslant \mathbb{E}_{k,v,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^4[\bar{v}_k^{(t)}]^4\right\} \\ &= \left(\mathbb{E}_{k,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^4\right\}\right)^2 \geqslant \left(\mathbb{E}_{k,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^2\right\}\right)^4 \geqslant \left(1-\alpha^2\right)^4 = 1 - O(\alpha^2).\end{aligned}$$

□

B.1.3 Condition (a): the individual bound

In this section, we show Lemma B.1, which implies condition (a) of Proposition 3.1 always holds.

Lemma B.1. *Suppose that at time t, Proposition 3.1 is true. Assuming $\delta_1^2 = O(\alpha^{1.5}/m)$, then for any $v^{(t)} \in S_k^{(t)}$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 \geqslant 8\tilde{a}^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - O\left(\alpha^{1.5}\right),$$

In particular, if $\lambda = \Omega\left(\sqrt{\alpha}\right)$, then $\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 > 0$ whenevner $[\bar{v}_k^{(t)}]^2 = 1-\alpha$.

Proof. Recall the definition of G_1, G_2 and G_3 from Lemma B.7. Now we estimate each of these three terms. By Lemma B.11, the first two terms of G_1 can be lower bounded by $8\tilde{a}^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - O(\hat{a}_k^{(t)}\alpha^{1.5})$ and, for the third term, replace $|z^{(t)}|$ with 1, and then, by the Cauchy-Schwarz inequality and Jensen's inequality, it is bounded $O(\hat{a}_k^{(t)}\alpha^{1.5})$. By Lemma B.9, G_2 and G_3 can be bounded by $O(1)\sum_{i\neq k}\hat{a}_i^{(t)}\alpha^2$. Thus,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 &\geqslant 8\tilde{a}^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - O(1)\sum_{i=1}^{d}\hat{a}_k^{(t)}\alpha^{1.5} - O(m\delta_1^2)\\ &\geqslant 8\tilde{a}^{(t)}\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4 - O\left(\alpha^{1.5}\right).\end{aligned}$$

Now suppose that $[\bar{v}_k^{(t)}]^2 = 1-\alpha$. By Proposition 3.1, we have $\tilde{a}^{(t)} \geqslant \lambda/6$. Hence,

$$\frac{\mathrm{d}}{\mathrm{d}t}[\bar{v}^{(t)}]^2 \geqslant \lambda\alpha(1-\alpha)^2 - O\left(\alpha^{1.5}\right) \geqslant \lambda\alpha - O\left(\alpha^{1.5}\right).$$

□

B.1.4 Condition (b): the average bound

Bounding the total amount of impulses

Note that there are two sources of impulses. First, when $\hat{a}_k^{(t)}$ is larger, the correlation of the newly-entered components is $1-\alpha$ instead of $1-\alpha^2$ and, second, the

reinitialization may throw some components out of $S_k^{(t)}$.

First we consider the first type of impulses. Suppose that at time t, $\hat{a}_k^{(t)} \geqslant \alpha$, $\mathbb{E}_{k,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^2\right\} = B$, and one particle $v^{(t)}$ enters $S_k^{(t)}$. The deterioration of the average bound can be bounded as

$$\begin{aligned} B - \left(\frac{\hat{a}_k^{(t)}}{\hat{a}_k^{(t)} + \left\|v^{(t)}\right\|^2} B + \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)} + \left\|v^{(t)}\right\|^2}(1-\alpha)\right) &= \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)} + \left\|v^{(t)}\right\|^2}\left(B - (1-\alpha)\right) \\ &\leqslant \frac{\left\|v^{(t)}\right\|^2}{\alpha} 2\alpha \\ &= 2\left\|v^{(t)}\right\|^2. \end{aligned}$$

Hence, the total amount of impulses caused by the entrance of new components can be bounded by $2m\delta_1^2$.

Now we consider the reinitialization. Again, it suffices to consider the case where $\hat{a}_k^{(t)} \geqslant \alpha$. Suppose that at time t, $\hat{a}_k^{(t)} \geqslant \alpha$, $\mathbb{E}_{k,w}^{(t)}\left\{[\bar{w}_k^{(t)}]^2\right\} = B$ and one particle $v^{(t)} \in S_k^{(t)}$ is reinitialized. By the definition of the algorithm, its norm is at most δ_1. Hence, The deterioration of the average bound can be bounded as[2]

$$\begin{aligned} B - \frac{\hat{a}_k^{(t)}}{\hat{a}_k^{(t)} - \left\|v^{(t)}\right\|^2}\left(B - \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)}}[\bar{v}_k^{(t)}]^2\right) &= \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)} - \left\|v^{(t)}\right\|^2}\left([\bar{v}_k^{(t)}]^2 - B\right) \\ &\leqslant \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)}} 2\alpha \\ &\leqslant 2\left\|v^{(t)}\right\|^2. \end{aligned}$$

Since there are at most m components, the amount of impulses caused by reinitialization is bounded by $2m\delta_1^2$.

Combine these two estimations together and we know that the total amount of impulses is bounded by $4m\delta_1^2$. This gives the epoch correction term of condition (c).

[2] The second term is obtained by solving the equation $B = \frac{\hat{a}_k^{(t)} - \left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)}} B' + \frac{\left\|v^{(t)}\right\|^2}{\hat{a}_k^{(t)}}[\bar{v}_k^{(t)}]^2$ for B'.

The average bound

First we derive a formula for the evolution of $\mathbb{E}_{k,w}^{(t)}\left\{[\bar{v}_k^{(t)}]^2\right\}$.

Lemma B.13. *For any k with $S_k^{(t)} \neq \varnothing$, we have*

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 =&\mathbb{E}_{k,v}^{(t)}\left[\frac{d}{dt}[\bar{v}_k^{(t)}]^2\right]\\ &+4\mathbb{E}_{k,v}^{(t)}\left[\left((T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left([\bar{v}_k^{(t)}]^2\right)\right]\\ &-4\left(\mathbb{E}_{k,v}^{(t)}(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left(\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2\right).\end{aligned}$$

Remark. The first term corresponds to the tangent movement and the two terms in the second line correspond to the norm change of the components.

Proof. Recall that

$$\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 = \frac{1}{\hat{a}_k^{(t)}}\sum_{v^{(t)}\in S_k^{(t)}}\left\|v^{(t)}\right\|^2[\bar{v}_k^{(t)}]^2.$$

Taking the derivative, we have

$$\begin{aligned}\frac{d}{dt}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 =&\frac{1}{\hat{a}_k^{(t)}}\sum_{v^{(t)}\in S_k^{(t)}}\left\|v^{(t)}\right\|^2\left(\frac{d}{dt}[\bar{v}_k^{(t)}]^2\right)+\frac{1}{\hat{a}_k^{(t)}}\sum_{v^{(t)}\in S_k^{(t)}}\left(\frac{d}{dt}\left\|v^{(t)}\right\|^2\right)[\bar{v}_k^{(t)}]^2\\ &+\left(\frac{d}{dt}\frac{1}{\hat{a}_k^{(t)}}\right)\sum_{v^{(t)}\in S_k^{(t)}}\left\|v^{(t)}\right\|^2[\bar{v}_k^{(t)}]^2.\end{aligned}$$

The first term is just $\mathbb{E}_{k,v}^{(t)}\frac{d}{dt}[\bar{v}_k^{(t)}]^2$. Denote $R(\bar{v}^{(t)}) = 2(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})-\lambda$. We can write the second term as follows:

$$\begin{aligned}\frac{1}{\hat{a}_k^{(t)}}\sum_{v^{(t)}\in S_k^{(t)}}\left(\frac{d}{dt}\left\|v^{(t)}\right\|^2\right)[\bar{v}_k^{(t)}]^2 =&\frac{1}{\hat{a}_k^{(t)}}\sum_{v^{(t)}\in S_k^{(t)}}2R(\bar{v}^{(t)})\left\|v^{(t)}\right\|^2[\bar{v}_k^{(t)}]^2\\ =&2\mathbb{E}_{k,v}^{(t)}\left[R(\bar{v}^{(t)})[\bar{v}_k^{(t)}]^2\right]\end{aligned}$$

Finally, let's consider $\frac{d}{dt}\frac{1}{\hat{a}_k^{(t)}}$ in the third term,

$$\begin{aligned}
\frac{d}{dt}\frac{1}{\hat{a}_k^{(t)}} =& -\frac{1}{[\hat{a}_k^{(t)}]^2}\frac{d}{dt}\hat{a}_k^{(t)} \\
=& -\frac{1}{[\hat{a}_k^{(t)}]^2}\frac{d}{dt}\sum_{v^{(t)}\in S_k^{(t)}}\left\|v^{(t)}\right\|^2 \\
=& -\frac{2}{[\hat{a}_k^{(t)}]^2}\sum_{v^{(t)}\in S_k^{(t)}}R(\bar{v}^{(t)})\left\|v^{(t)}\right\|^2 \\
=& -\frac{2}{\hat{a}_k^{(t)}}\mathbb{E}_{k,v}^{(t)}R(\bar{v}^{(t)}).
\end{aligned}$$

Overall, we have

$$\begin{aligned}
\frac{d}{dt}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 =& \mathbb{E}_{k,v}^{(t)}\left[\frac{d}{dt}[\bar{v}_k^{(t)}]^2\right] \\
&+ 4\mathbb{E}_{k,v}^{(t)}\left[\left((T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left([\bar{v}_k^{(t)}]^2\right)\right] \\
&- 4\left(\mathbb{E}_{k,v}^{(t)}(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left(\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2\right)
\end{aligned}$$

□

Lemma B.14 (Bound for the average tangent speed)**.** *Suppose that $m\delta_1^2 = O(\alpha^3)$ and, at time t, Proposition 3.1 is true and $S_k^{(t)} \neq \varnothing$. Then we have*

$$\mathbb{E}_{k,v}^{(t)}\left[\frac{d}{dt}[\bar{v}_k^{(t)}]^2\right] \geqslant 8(a_k - \hat{a}_k^{(t)})(1 - \mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2) - O(\alpha^3).$$

Proof. Recall the definition of G_1, G_2 and G_3 from Lemma B.7.

- **Lower bound for $\mathbb{E}_{k,v}^{(t)}G_1$.** By (B.1), we have $\mathbb{E}_{k,v,w}^{(t)}\left\{[z^{(t)}]^3\langle\bar{w}_{-k},\bar{v}_{-k}\rangle\right\} \geqslant 0$, whence can be ignored. Meanwhile, note that $\mathbb{E}_{k,w}^{(t)}\left\{[z^{(t)}]^4\right\} \leqslant 1$. Therefore,

$$\mathbb{E}_{k,v}^{(t)}G_1 \geqslant 8a_k\mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4\right\} - 8\hat{a}_k^{(t)}\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\}.$$

For the first term, we compute

$$\begin{aligned}
&\mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)[\bar{v}_k^{(t)}]^4\right\}\\
=&\mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)\left(1-\left(1+[\bar{v}_k^{(t)}]^4\right)\right)\right\}\\
=&\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\}-\mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)^2\left(1+[\bar{v}_k^{(t)}]^2\right)\right\}\\
\geqslant&\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\}-2\mathbb{E}_{k,v}^{(t)}\left\{\left(1-[\bar{v}_k^{(t)}]^2\right)^2\right\}\\
\geqslant&\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\}-O(\alpha^3).
\end{aligned}$$

Thus,

$$\mathbb{E}_{k,v}^{(t)}G_1\geqslant 8\tilde{a}_k^{(t)}\mathbb{E}_{k,v}^{(t)}\left\{1-[\bar{v}_k^{(t)}]^2\right\}-O\left(\hat{a}_k^{(t)}\alpha^3\right).$$

- **Upper bound for $\mathbb{E}_{k,v}^{(t)}|G_2|$ and $\mathbb{E}_{k,v}^{(t)}|G_2|$.** It follows from Lemma B.10 that both terms are $O(1)\sum_{i\neq k}\hat{a}_i^{(t)}\alpha^3$.

Combine these two bounds together, absorb $m\delta_1^2$ into $O(\alpha^3)$, and we complete the proof. □

Lemma B.15 (Bound for the norm fluctuation)**.** *Suppose that at time t, Proposition 3.1 is true and $S_k^{(t)}\neq\varnothing$. Then at time t, we have*

$$4\mathbb{E}_{k,v}^{(t)}\left[\left((T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left([\bar{v}_k^{(t)}]^2\right)\right]-4\left(\mathbb{E}_{k,v}^{(t)}(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\right)\left(\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2\right)$$
$$\geqslant -O(\alpha^3)$$

Proof. We can express $(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})$ as follows:

$$\begin{aligned}
&(T^*-T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\\
=&(a_k-\hat{a}_k^{(t)})[\bar{v}_k^{(t)}]^4+\hat{a}_k^{(t)}\left([\bar{v}_k^{(t)}]^4-\mathbb{E}_{k,w}^{(t)}\left\langle\bar{w}^{(t)},\bar{v}^{(t)}\right\rangle^4\right)\\
&+\sum_{i\neq k}a_i[\bar{v}_i^{(t)}]^4-\sum_{i\neq k}\hat{a}_i^{(t)}\mathbb{E}_{i,w}^{(t)}\left\langle\bar{w}^{(t)},\bar{v}^{(t)}\right\rangle^4\pm O(m\delta_1^2)
\end{aligned}$$

It's clear that $\mathbb{E}_{k,v}^{(t)} \sum_{i \neq k} a_i [\bar{v}_i^{(t)}]^4 = O(\alpha^3)$ and $\mathbb{E}_{k,v}^{(t)} \sum_{i \neq k} \hat{a}_i^{(t)} \mathbb{E}_{i,w}^{(t)} \left\langle \bar{w}^{(t)}, \bar{v}^{(t)} \right\rangle^4 = O(\alpha^3)$, so their influence can be bounded by $O(\alpha^3)$. Let's then focus on the first two terms in $(T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})$.

For the first term, we have

$$\begin{aligned}
&4\mathbb{E}_{k,v}^{(t)}(a_k - \hat{a}_k^{(t)})[\bar{v}_k^{(t)}]^4[\bar{v}_k^{(t)}]^2 - 4\mathbb{E}_{k,v}^{(t)}(a_k - \hat{a}_k^{(t)})[\bar{v}_k^{(t)}]^4 \mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 \\
=&4(a_k - \hat{a}_k^{(t)}) \left(\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^6 - \mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^4 \mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 \right) \geqslant 0.
\end{aligned}$$

Let's now turn our focus to the second term. Denote $x = \left\langle \bar{w}_{-k}^{(t)}, \bar{v}_{-k}^{(t)} \right\rangle$ and write $\left\langle \bar{w}^{(t)}, \bar{v}^{(t)} \right\rangle^4 = [\bar{w}_k^{(t)}]^4[\bar{v}_k^{(t)}]^4 + 4[\bar{w}^{(t)}]_k^3[\bar{v}_k^{(t)}]^3 x + O(x^2)$. Suppose $m = \mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2$, we know $m \in [1 - O(\alpha^2), 1]$. We also know that $[\bar{v}_k^{(t)}]^2 \in [1 - \alpha, 1]$ for every $\bar{v}^{(t)} \in S_i^{(t)}$, so we have $|[\bar{v}_k^{(t)}]^2 - m| = O(\alpha)$. We have

$$\begin{aligned}
\left| \mathbb{E}_{k,v}^{(t)} \mathbb{E}_{k,w}^{(t)} ([\bar{v}_k^{(t)}]^2 - m)[\bar{v}_k^{(t)}]^4 (1 - [\bar{w}_k^{(t)}]^4) \right| &= O(\alpha^3) \\
\left| \mathbb{E}_{k,v}^{(t)} \mathbb{E}_{k,w}^{(t)} ([\bar{v}_k^{(t)}]^2 - m)(\bar{w}_k^{(t)} \bar{v}_k^{(t)})^3 x \right| &= O(\alpha^3) \\
\mathbb{E}_{k,v}^{(t)} \mathbb{E}_{k,w}^{(t)} x^2 &= O(\alpha^4)
\end{aligned}$$

Therefore, $4\mathbb{E}_{k,v}^{(t)} \left[\hat{a}_k^{(t)} \left([\bar{v}_k^{(t)}]^4 - \mathbb{E}_{k,w}^{(t)} \left\langle \bar{w}^{(t)}, \bar{v}^{(t)} \right\rangle^4 \right) [\bar{v}_k^{(t)}]^2 \right]$

$-4\mathbb{E}_{k,v}^{(t)} \hat{a}_k^{(t)} \left([\bar{v}_k^{(t)}]^4 - \mathbb{E}_{k,w}^{(t)} \left\langle \bar{w}^{(t)}, \bar{v}^{(t)} \right\rangle^4 \right) \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^2 \geqslant -O(\hat{a}_k^{(t)} \alpha^3)$.

Combining the bounds for all four terms, we conclude that

$$4\mathbb{E}_{k,v}^{(t)} \left[(T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})[\bar{v}_k^{(t)}]^2 \right] - 4\mathbb{E}_{k,v}^{(t)} (T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4}) \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^2 \geqslant -O(\alpha^3).$$

□

Lemma B.2. *Suppose that at time t, Proposition 3.1 is true and $S_k^{(t)} \neq \varnothing$. Assuming $\delta_1^2 = O(\alpha^3/m)$, we have*

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^2 \geqslant 8\tilde{a}_k^{(t)} (1 - \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^2) - O(\alpha^3).$$

In particular, if $\lambda = \Omega(\alpha)$*, then* $\frac{\mathrm{d}}{\mathrm{d}t}\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 > 0$ *when* $\mathbb{E}_{k,v}^{(t)}[\bar{v}_k^{(t)}]^2 < 1 - \alpha^2/2$.

Proof. It suffices to combine the previous three lemmas together. □

B.1.5 Condition (c): bounds for the residual

In this section, we consider condition (c) of Proposition 3.1. Again, we need to estimate the derivative of $\tilde{a}_k^{(t)}$ when $\tilde{a}_k^{(t)}$ touches the boundary.

On the impulses Similar to the average bound in condition (b), we need to take into consideration the impulses. For the lower bound on $\tilde{a}_k^{(t)}$, we only need to consider the impulses caused by the entrance of new components since the reinitialization will only increase $\tilde{a}_k^{(t)}$. By Proposition 3.1 and Assumption B.1, the total amount of impulses is upper bounded by $m\delta_1^2$. At the beginning of epoch s, we have $\tilde{a}_k^{(t)} \geqslant \lambda/6 - (s-1)m\delta_1^2$, which is guaranteed by the induction hypothesis from the last epoch. (At the beginning of the first epoch, we have $\tilde{a}_k^{(t)} = a_k$). Thus, following Lemma B.5, it suffices to show that $\frac{\mathrm{d}}{\mathrm{d}t}\tilde{a}_k^{(t)} > 0$ when $\tilde{a}_k^{(t)} \leqslant \lambda/6$. The upper bound on $\tilde{a}_k^{(t)}$ can be proved in a similar fashion. The only difference is that now the impulses that matter are caused by the reinitialization, the total amount of which can again be bounded by $m\delta_1^2$.

Lemma B.16. *Suppose that at time* t*, Proposition 3.1 is true and no impulses happen at time* t*. Then we have*

$$\frac{1}{\hat{a}_k^{(t)}}\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} = 2\sum_{i=1}^{d} a_i \mathbb{E}_{k,v}^{(t)}[\bar{v}_i^{(t)}]^4 - 2\sum_{i=1}^{d} \hat{a}_i^{(t)} \mathbb{E}_{k,v}^{(t)}\mathbb{E}_{i,w}^{(t)}[z^{(t)}]^4 - \lambda - O(m\delta_1^2).$$

Proof. Recall that $\hat{a}_k^{(t)} = \sum_{v^{(t)} \in S_k^{(t)}} \left\|v^{(t)}\right\|^2$ and Lemma B.6 implies that

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\left\|v^{(t)}\right\|^2 = 2\sum_{i=1}^{d} a_i \left\|v^{(t)}\right\|^2 [\bar{v}_i^{(t)}]^4 - 2\sum_{i=1}^{d} \hat{a}_i^{(t)} \left\|v^{(t)}\right\|^2 \mathbb{E}_{i,w}^{(t)}\left\{[z^{(t)}]^4\right\} \\ - \lambda \left\|v^{(t)}\right\|^2 - \left\|v^{(t)}\right\|^2 O(m\delta_1^2).\end{aligned}$$

Sum both sides and we complete the proof. □

Lemma B.17. *Suppose that at time t, Proposition 3.1 is true and no impulses happen at time t. Assume $\delta_1^2 = O(\alpha^2/m)$. Then we have*

$$\frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} \leqslant 2\tilde{a}_k^{(t)} - \lambda + O(\alpha^2).$$

In particular, when $\tilde{a}_k^{(t)} \leqslant \lambda/6$, we have $\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} < 0$.

Proof. By Lemma B.16, we have

$$\frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} \leqslant 2a_k - 2\hat{a}_k^{(t)} \mathbb{E}_{k,v}^{(t)} \mathbb{E}_{k,w}^{(t)} [z^{(t)}]^4 + 2\sum_{i \neq k} a_i \mathbb{E}_{k,v}^{(t)} [\bar{v}_i^{(t)}]^4 - \lambda.$$

By Lemma B.12, we have

$$2a_k - 2\hat{a}_k^{(t)} \mathbb{E}_{k,v}^{(t)} \mathbb{E}_{k,w}^{(t)} [z^{(t)}]^4 \leqslant 2\tilde{a}_k^{(t)} + O(a_k \alpha^2)$$

For each term in the summation, we have

$$\mathbb{E}_{k,v}^{(t)} [\bar{v}_i^{(t)}]^4 \leqslant \mathbb{E}_{k,v}^{(t)} \left\{ \left(1 - [\bar{v}_k^{(t)}]^2\right)^2 \right\} \leqslant \alpha \mathbb{E}_{k,v}^{(t)} \left\{ 1 - [\bar{v}_k^{(t)}]^2 \right\} \leqslant \alpha^3.$$

Thus,

$$\begin{aligned} \frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} &\leqslant 2\tilde{a}_k^{(t)} + O(a_k \alpha^2) + 2\sum_{i \neq k} a_i^2 \alpha^3 - \lambda \\ &\leqslant 2\tilde{a}_k^{(t)} - \lambda + O(\alpha^2). \end{aligned}$$

□

Lemma B.18. *Suppose that at time t, Proposition 3.1 is true. and no impulses happen at time t. Then at time t, we have*

$$\frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} \geqslant 2\tilde{a}_k^{(t)} - \lambda - O\left(\alpha^2\right).$$

In particular, when $\tilde{a}_k^{(t)} \geqslant \lambda$, we have $\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} > 0$.

Proof. By Lemma B.16 (and the fact $\hat{a}_i^{(t)} \leqslant a_i$), we have

$$\frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} \geqslant 2a_k \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^4 - 2\hat{a}_k^{(t)} - 2 \sum_{i \neq k} a_i \mathbb{E}_{k,v}^{(t)} \mathbb{E}_{i,w}^{(t)} [z^{(t)}]^4 - \lambda - O(m\delta_1^2).$$

Note that $\mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^4 \geqslant 1 - O(\alpha^2)$, whence

$$2a_k \mathbb{E}_{k,v}^{(t)} [\bar{v}_k^{(t)}]^4 - 2\hat{a}_k^{(t)} \geqslant 2\tilde{a}_k^{(t)} - O\left(a_k \alpha^2\right).$$

For each term in the summation, by Lemma B.10, we have $\mathbb{E}_{k,v}^{(t)} \mathbb{E}_{i,w}^{(t)} [z^{(t)}]^4 \leqslant O(\alpha^3)$. Thus,

$$\frac{1}{\hat{a}_k^{(t)}} \frac{\mathrm{d}}{\mathrm{d}t} \hat{a}_k^{(t)} \geqslant 2\tilde{a}_k^{(t)} - \lambda - O\left(\alpha^2\right).$$

□

B.1.6 Counterexample

We prove Claim 3.1 as follows.

Claim 3.1. *Suppose* $T^* = e_k^{\otimes 4}$ *and* $T = v^{\otimes 4}/\|v\|^2 + w^{\otimes 4}/\|w\|^2$ *with* $\|w\|^2 + \|v\|^2 \in [2/3, 1]$. *Suppose* $\bar{v}_k^2 = 1 - \alpha$ *and* $\bar{v}_k = \bar{w}_k, \bar{v}_{-k} = -\bar{w}_{-k}$. *Assuming* $\|v\|^2 \leqslant c_1$ *and* $\alpha \leqslant c_2$ *for small enough constants* c_1, c_2, *we have* $\frac{\mathrm{d}}{\mathrm{d}t} \bar{v}_k^2 < 0$.

Proof. Similar as in Lemma B.7, we can compute $\frac{\mathrm{d}}{\mathrm{d}t} \bar{v}_k^2$ as follows,

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} \bar{v}_k^2 =& 8(1 - \bar{v}_k^2) \bar{v}_k^4 \\
&- 8(1 - \bar{v}_k^2) \left(\|v\|^2 \langle \bar{v}, \bar{v} \rangle^4 + \|w\|^2 \langle \bar{w}, \bar{v} \rangle^4 \right) \\
&+ 8 \left(\|w\|^2 \langle \bar{w}, \bar{v} \rangle^3 \langle \bar{w}_{-k}, \bar{v}_{-k} \rangle + \|v\|^2 \langle \bar{v}, \bar{v} \rangle^3 \langle \bar{v}_{-k}, \bar{v}_{-k} \rangle \right).
\end{aligned}$$

Since $\bar{v}_k^2 = 1 - \alpha, \bar{v}_k = \bar{w}_k$ and $\bar{v}_{-k} = -\bar{w}_{-k}$, we have $\langle \bar{w}, \bar{v} \rangle^4, \langle \bar{w}, \bar{v} \rangle^3 \geqslant 1 - O(\alpha)$ and $\langle \bar{w}_{-k}, \bar{v}_{-k} \rangle = -\alpha$. Therefore, we have

$$\frac{\mathrm{d}}{\mathrm{d}t} \bar{v}_k^2 \leqslant 8\alpha - 8\alpha(\|v\|^2 + \|w\|^2 (1 - O(\alpha))) - 8 \|w\|^2 (1 - O(\alpha))\alpha + 8 \|v\|^2 \alpha$$

We have

$$\frac{\mathrm{d}}{\mathrm{d}t}\bar{v}_k^2 \leqslant 8\alpha\left((1-\|w\|^2-\|v\|^2)-\|w\|^2(1-O(\alpha))+\|v\|^2\right)<0,$$

where the last inequality assumes $\|w\|^2+\|v\|^2 \in [2/3,1]$ and $\|v\|^2, \alpha$ smaller than certain constant. □

B.2 Proofs for (Re)-initialization and Phase 1

We specify the constants that will be used in the proof of initialization (Section B.2.1) and Phase 1 (Section B.2.2). We will assume it always hold in the proof of Section B.2.1 and Section B.2.2. We omit superscript s for simplicity.

Proposition B.1 (Choice of parameters)**.** *The following hold with proper choices of constants* $\gamma, c_e, c_\rho, c_{max}, c_t$

1. $t_1' := \frac{c_t d}{8\beta \log d} \leqslant t_1 \leqslant \frac{(1-\gamma)}{8\beta c_e}\cdot\frac{d}{\log d}$,
2. $\Gamma_i = \frac{1}{8a_i t_1'}$ *if* $S_i^{(s,0)} = \varnothing$, *and* $\Gamma_i = \frac{1}{8\lambda t_1'}$ *otherwise.* $\rho_i = c_\rho \Gamma_i$. $\Gamma_{max} = c_{max}\log d/d$.
3. $c_e < \frac{c_\rho c_{max}}{2(1-c_\rho)}$, $c_\rho/c_t > 4c_e$, $c_t c_{max} \geqslant 4$.
4. $c_a = (1-c_\rho)/(c_t c_{max})$

Proof. The results hold if let γ, c_e, c_ρ, c_t be small enough constant and c_{max} be large enough constant. For example, we can choose $c_e < c_\rho/4 < 0.01$, $c_t, \gamma < 0.01$ and $c_{max} > 10/c_t$. □

B.2.1 Initialization

We give a more detailed version of initialization with specified constants to fit the definition of S_{good}, S_{pot} and S_{bad}. We show that at the beginning of any epoch s, the

following conditions hold with high probability. Intuitively, it suggests all directions that we will discover satisfy $a_i = \Omega(\beta)$ as $S_{i,pot} \neq \varnothing$.

Lemma B.19 ((Re-)Initialization space)**.** *In the setting of Theorem 3.1, the following hold at the beginning of current epoch with probability* $1 - 1/poly(d)$.

1. *For all* $a_i - \hat{a}_i^{(0)} \geqslant \beta$, *we have* $S_{i,good} \neq \varnothing$.
2. *For all* $a_i - \hat{a}_i^{(0)} < \beta c_a$, *we have* $S_{i,pot} = \varnothing$.
3. $S_{bad} = \varnothing$
4. $\left\|v^{(0)}\right\|_2 = \Theta(\delta_0)$, $[\bar{v}_i^{(0)}]^2 \leqslant \Gamma_{max} = c_{max} \log d/d$
5. *For every* v, *there are at most* $O(\log d)$ *many* $i \in [d]$ *such that* $[\bar{v}_i^{(0)}]^2 \geqslant c_e \log(d)/(10d)$.
6. $|\{v | v \text{ was reinitialized in epoch } s\}| = (1 - O(1/\log^2 d))m$.

Proof. Let the constants in Lemma B.20 be $\eta = 1/c_t$, $c_i = \Gamma_i d/\log d$ and satisfy Proposition B.1, then we know at the time of (re-)initialization, all statements hold. Since we further know from Lemma 3.4 that $\|v\| = \Theta(\delta_0)$ and $\bar{v}_i^2$ will only change $o(\log d/d)$, we have at the beginning of every epoch, all statements hold. □

Lemma B.20. *There exist* $m_0 = poly(d)$ *and* $m_1 = poly(d)$ *such that if* $m \in [m_0, m_1]$ *and we random sample* m *vectors* v *from Unif*$(\mathbb{S}^{d-1})$, *with probability* $1 - 1/poly(d)$ *the following hold with proper absolute constant* η, γ, c_ρ, c_i, c_e, c_{max} *satisfying* $\eta(1-\gamma) \leqslant c_i$, $c_{max} \geqslant 4\eta$, γ, c_ρ *are small enough and* c_{max}, η *are large enough*

1. *For every* $i \in [d]$ *such that* $c_i \leqslant \eta$, *there exists* v *such that* $[\bar{v}_i^{(0)}]^2 \geqslant c_i(1 + 2c_\rho)\log d/d$ *and* $[\bar{v}_j^{(t)}]^2 \leqslant c_j(1 - 2c_\rho)\log d/d$ *for* $j \neq i$.

2. *For every v, there does not exist $i \neq j$ such that $[\bar{v}_i^{(0)}]^2 \geqslant c_i(1-2c_\rho)\log d/d$ and $[\bar{v}_j^{(0)}]^2 \geqslant c_j(1-2c_\rho)\log d/d$.*

3. *For every v and $i \in [d]$, $[\bar{v}_i^{(0)}]^2 \leqslant c_{max}\log d/2d$.*

4. *For every v, there are at most $O(\log d)$ many $i \in [d]$ such that $[\bar{v}_i^{(0)}]^2 \geqslant c_e \log(d)/11d$.*

5. *$|\{v|$there exists $i \in [d]$ such that $[\bar{v}_i^{(0)}]^2 \geqslant c_i(1-2c_\rho)\log d/d\}| \leqslant m/\log^2(d)$.*

Proof. It is equivalent to consider sample v from $\mathcal{N}(0, I)$. Let $x \in \mathbb{R}$ be a standard Gaussian variable, according to Proposition 2.1.2 in Vershynin (2018), we have for any $t > 0$

$$\left(\frac{2}{t} - \frac{2}{t^3}\right) \cdot \frac{1}{\sqrt{2\pi}} e^{-t^2/2} \leqslant \Pr\left[x^2 \geqslant t^2\right] \leqslant \frac{2}{t} \cdot \frac{1}{\sqrt{2\pi}} e^{-t^2/2}.$$

Therefore, for any $i \in [d]$, we have for any constant $c > 0$

$$\Pr\left[v_i^2 \geqslant c\log(d)\right] = \Theta(d^{-c/2}\log^{-1/2} d).$$

According to Theorem 3.1.1 in Vershynin (2018), we know with probability at least $1 - 2\exp(-\Omega(d))$, $(1-r)d \leqslant \|v\|^2 \leqslant (1+r)d$ for any constant $0 < r < 1$. Hence, we have

$$\Pr\left[\bar{v}_i^2 \geqslant \frac{c\log(d)}{d}\right] \geqslant \Theta(d^{-c(1+r)/2}\log^{-1/2} d),$$

$$\Pr\left[\bar{v}_i^2 \geqslant \frac{c\log(d)}{d}\right] \leqslant \Theta(d^{-c(1-r)/2}\log^{-1/2} d).$$

Part 1. For fixed $i \in [d]$ such that $\eta(1-\gamma) \leqslant c_i \leqslant \eta$, we have

$$\Pr\left[\bar{v}_i^2 \geqslant c_i(1+2c_\rho)\log(d)/d\right] \geqslant \Theta(d^{-c_i(1+2c_\rho)(1+r)/2}\log^{-1/2} d),$$

For a given $j \neq i$, we have

$$\begin{aligned}&\Pr\left[\bar{v}_i^2 \geqslant c_i(1+2c_\rho)\log(d)/d,\ \bar{v}_j^2 \geqslant c_j(1-2c_\rho)\log(d)/d\right]\\ &\leqslant \Theta(d^{-c_i(1+2c_\rho)(1-r)/2-c_j(1-2c_\rho)(1-r)/2}) = O(d^{-\eta(1-\gamma)(1-r)}).\end{aligned}$$

Since $c_i \leqslant \eta$, we know the desired event happens with probability $\Theta(d^{-\eta(1+2c_\rho)(1+r)/2} - d^{-\eta(1-\gamma)(1-r)+1})$. Since γ, c_ρ are small enough constant, when $m_0 \geqslant \Omega(d^{\eta(1+2c_\rho)(1+r)/2+1})$, with probability $1 - O(e^{-d})$ there exists at least one v such that $\bar{v}_i^2 \geqslant c_i(1+2c_\rho)\log(d)$ and $[\bar{v}_j^{(t)}]^2 \leqslant c_j(1-2c_\rho)\log d/d$ for $j \neq i$. Take the union bound for all $i \in [d]$, we know when $m_0 \geqslant \Omega(d^{\eta(1+2c_\rho)(1+r)/2+2})$, the desired statement holds with probability $1 - O(de^{-d})$.

Part 2. For any given $i \neq j$, we have

$$\begin{aligned}&[\Pr\left[[\bar{v}_i^{(0)}]^2 \geqslant c_i(1-2c_\rho)\log d/d,\ [\bar{v}_j^{(0)}]^2 \geqslant c_j(1-2c_\rho)\log d/d\right]\\ \leqslant &O(d^{-(c_i+c_j)(1-2c_\rho)(1-r)/2}).\end{aligned}$$

Since $\eta(1-\gamma) \leqslant c_i$, the probability that there exist $i \neq j$ such that the above happens is at most $O(d^{-\eta(1-\gamma)(1-2c_\rho)(1-r)+2})$. Thus, with $m_1 \leqslant O(d^{\eta(1-\gamma)(1-2c_\rho)(1-r)-2}/\mathrm{poly}(d))$, the desired statement holds with probability $1 - 1/\mathrm{poly}(d)$.

Part 3. We know

$$\Pr\left[\text{for all } i \in [d],\ \bar{v}_i^2 \leqslant c_{max}\log d/2d\right] \geqslant 1 - O(d^{-c_{max}(1-r)/4+1}).$$

With $m_1 \leqslant O(d^{c_{max}(1-r)/4-1}/\mathrm{poly}(d))$ the desired statement holds with probability $1 - 1/\mathrm{poly}(d)$.

Part 4. Since $m \leqslant m_1 = \mathrm{poly}(d)$, we know for any constant c_e, this statement holds with probability $1 - O(e^{-\log^2 d})$.

Part 5. We have

$$\Pr\Big[\text{there exists } i \in [d] \text{ such that } [\bar{v}_i^{(0)}]^2 \geqslant c_i(1-2c_\rho)\log d/d\Big] \leqslant O(d^{-c_i(1-2c_\rho)/2+1}).$$

Let p be the above probability and set A as the v satisfy above condition, by Chernoff's bound we have

$$\Pr\big[|A| \geqslant m/\log^2 d\big] \leqslant e^{-pm}\left(\frac{epm}{m/\log^2 d}\right)^{m/\log^2 d} = O(e^{-d}).$$

Combine all parts above, we know as long as r, γ, c_ρ are small enough, $c_{max} \geqslant 4\eta$ and η is large enough, we have when $m_0 \geqslant \Omega(d^{0.6\eta})$ and $m_1 \leqslant O(d^{0.9\eta})$, the results hold. □

B.2.2 Proof of Phase 1

In this section, we first give a proof overview of Phase 1 and then give the detailed proof for each lemma in later subsections.

Proof overview

We give the proof overview in this subsection and present the proof of Lemma 3.3 and Lemma 3.2 at the end of this subsection. We remark that the proof idea in this phase is inspired by (Li et al., 2020a).

We describe the high-level proof plan for phase 1. Recall that at the beginning of this epoch, we know $S_{bad} = \varnothing$ which implies there is at most one large coordinate for every component. Roughly speaking, we will show that for those small coordinate they will remain small in phase 1, and the only possibility for one component to have larger norm is to grow in the large direction. This intuitively suggests all components that have a relatively large norm in phase 1 are basis-like components.

We first show within $t_1' = c_t d/(8\beta\log d))$ time, there are components that can improve their correlation with some ground truth component e_i to a non-trivial

polylog$(d)/d$ correlation. This lemma suggests that there is at most one coordinate can grow above $O(\log d/d)$.

Note that we should view the analysis in this section and the analysis in Appendix B.1 as a whole induction/continuity argument. It's easy to verify that at any time $0 \leqslant t \leqslant t_1^{(s)}$, Assumption B.1 holds and Proposition 3.1 holds.

Lemma B.21. *In the setting of Lemma 3.2, suppose* $\left\|\bar{v}^{(0)}\right\|_\infty^2 \leqslant \log^4(d)/d$. *Then, for every* $k \in [d]$

1. *for* $v \notin S_{pot}$, $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for all* $i \in [d]$ *and* $t \leqslant t_1'$.
2. *if* $S_k^{(t)} = \varnothing$ *for* $t \leqslant t_1'$, *then for* $v \in S_{k,good}$, *there exists* $t \leqslant t_1'$ *such that* $[\bar{v}_k^{(t)}]^2 \geqslant \log^4(d)/d$ *and* $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$.
3. *for* $v \in S_{k,pot} \backslash (S_{good} \cup S_{bad})$, $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$ *and* $t \leqslant t_1'$.

The above lemma is in fact a direct corollary from the following lemma when considering the definition of S_{good} and S_{pot}. It says if a direction is below certain threshold, it will remain $O(\log d/d)$, while if a direction is above certain threshold and there are no basis-like components for this direction, it will grow to have a polylog(d) improvement.

Lemma B.22. *In the setting of Lemma 3.2, we have*

1. *if* $[\bar{v}_k^{(0)}]^2 \leqslant \min\{\Gamma_k - \rho_k, \Gamma_{max}\}$, *then* $[\bar{v}_k^{(t)}]^2 = O(\log(d)/d)$ *for* $t \leqslant t_1'$.
2. *if* $S_k^{(t)} = 0$ *for* $t \leqslant t_1'$, $[\bar{v}_k^{(0)}]^2 \geqslant \Gamma_k + \rho_k$, $[\bar{v}_i^{(0)}]^2 \leqslant \Gamma_i - \rho_i$ *for all* $i \neq k$ *and* $\left\|\bar{v}^{(0)}\right\|_\infty^2 \leqslant \log^4(d)/d$, *then there exists* $t \leqslant t_1'$ *such that* $[\bar{v}_k^{(t)}]^2 \geqslant \log^4(d)/d$.

The following lemma shows if $[\bar{v}_i^{(t_1')}]^2 = O(\log d/d)$ at t_1', it will remain $O(\log d/d)$ to the end of phase 1. This implies for components that are not in S_{pot}, they will not have large correlation with any ground truth component in phase 1.

Lemma B.23. *In the setting of Lemma 3.2, suppose* $[\bar{v}_i^{(t_1')}]^2 = O(\log(d)/d)$. *Then we have* $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for* $t_1' \leqslant t \leqslant t_1$.

The following two lemmas show good components (those have polylog$(d)/d$ correlation before t_1') will quickly grow to have constant correlation and δ_1 norm. Note that the following condition $a_k = \Omega(\beta)$ holds in our setting because when $a_i < \beta c_a$, we have $S_{i,good} = S_{i,pot} = \varnothing$ (this means for those small directions there are no components that can have polylog$(d)/d$ correlation as shown in Lemma B.21).

Lemma B.24 (Good component, constant correlation)**.** *In the setting of Lemma 3.2, suppose* $S_k^{(t)} = \varnothing$ *for* $t \leqslant t_1$, $a_k = \Omega(\beta)$. *If there exists* $\tau_0 \leqslant t_1$ *such that* $[\bar{v}_k^{(\tau_0)}]^2 > \log^4(d)/d$ *and* $[\bar{v}_i^{(\tau_0)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$, *then for any constant* $c \in (0,1)$ *we have* $[\bar{v}_k^{(t)}]^2 > c$ *and* $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$ *when* $\tau_0 + t_1'' \leqslant t \leqslant t_1$ *with* $t_1'' = \Theta(d/(\beta \log^3 d))$.

Lemma B.25 (Good component, norm growth)**.** *In the setting of Lemma 3.2, suppose* $S_k^{(t)} = \varnothing$ *for* $t \leqslant t_1$, $a_k = \Omega(\beta)$. *If there exists* $\tau_0' \leqslant t_1$ *such that* $[\bar{v}_k^{(\tau_0')}]^2 > c$ *and* $[\bar{v}_i^{(\tau_0')}]^2 = O(\log(d)/d)$ *for all* $i \neq k$, *then we have* $\left\|v^{(t)}\right\|_2 \geqslant \delta_1$ *for some* $\tau_0' \leqslant t \leqslant \tau_0' + t_1'''$ *with* $t_1''' = \Theta(\log(d/\alpha)/\beta)$.

Recall from Lemma B.22 we know there is at most one coordinate that can be large. Thus, intuitively we can expect if the norm is above certain threshold, the component will become basis-like, since this large direction will contribute most of the norm and other directions will remain small. In fact, we can show (1) norm of "small and dense" components (e.g., those are not in S_{pot}) is smaller than δ_1; (2) once a component reaches norm δ_1, it is a basis-like component.

Lemma B.26. *In the setting of Lemma 3.2, we have*

1. *if* $\left\|\bar{v}^{(t)}\right\|_\infty^2 \leqslant \log^4(d)/d$ *for all* $t \leqslant t_1$, *then* $\left\|v^{(t)}\right\|_2 = O(\delta_0)$ *for all* $t \leqslant t_1$.

2. *Let* $\tau_0 = \inf\{t \in [0, t_1] | \left\|\bar{v}^{(t)}\right\|_\infty^2 \geqslant \log^4 d/d\}$. *Suppose* $[\bar{v}_k^{(\tau_0)}]^2 \geqslant \log^4 d/d$ *and* $[\bar{v}_i^{(\tau_0)}]^2 = O(\log d/d)$ *for* $i \neq k$. *If there exists* τ_1 *such that* $\tau_0 < \tau_1 \leqslant t_1$ *and* $\left\|v^{(\tau_1)}\right\|_2 \geqslant \delta_1$ *for the first time, then there exists* $k \in [d]$ *such that* $[\bar{v}_k^{(\tau_1)}]^2 \geqslant 1-\alpha^2$ *if* $\hat{a}_k^{(t)} \leqslant \alpha$ *for* $t \leqslant \tau_1$ *and* $[\bar{v}_k^{(\tau_1)}]^2 \geqslant 1 - \alpha$ *otherwise.*

One might worry that a component can first exceeds the δ_1 threshold then drop below it and eventually gets re-initialized. Next, we show that re-initialization at the end of Phase 1 cannot remove all the components in $S_k^{(t_1)}$.

Lemma B.27. *If* $S_k^{(0)} = \varnothing$ *and* $S_k^{(t')} \neq \varnothing$ *for some* $t' \in (0, t_1]$, *we have* $S_k^{(t_1)} \neq \varnothing$ *and* $\hat{a}_k^{(t_1)} \geqslant \delta_1^2$.

Given above lemma, we now are ready to prove Lemma 3.3 and the main lemma for Phase 1.

Lemma 3.3. *In the setting of Lemma 3.2, for every* $i \in [d]$

1. *(Only good/potential components can become large) If* $v^{(s,t)} \notin S_{pot}^{(s)}$, $\left\|v^{(s,t)}\right\| = O(\delta_0)$ *and* $[\bar{v}_i^{(s,t)}]^2 = O(\log(d)/d)$ *for all* $i \in [d]$ *and* $t \leqslant t_1^{(s)}$.

2. *(Good components discover ground truth components) If* $S_{i,good}^{(s)} \neq \varnothing$, *there exists* $v^{(s,t_1^{(s)})}$ *such that* $\left\|v^{(s,t_1^{(s)})}\right\| \geqslant \delta_1$ *and* $S_i^{(s,t_1^{(s)})} \neq \varnothing$.

3. *(Large components are correlated with ground truth components) If* $\left\|v^{(s,t)}\right\| \geqslant \delta_1$ *for some* $t \leqslant t_1^{(s)}$, *there exists* $i \in [d]$ *such that* $v^{(s,t)} \in S_i^{(s,t)}$.

Proof. We show statements one by one.

Part 1. The statement follows from Lemma B.21, Lemma B.23 and Lemma B.26.

Part 2. Suppose $S_k^{(t)} = \varnothing$ for all $t \leqslant t_1$. By Lemma B.19 we know $S_{k,good} \neq \varnothing$. Then by Lemma B.21, Lemma B.24 and Lemma B.25, we know there exists v such that $\left\|v^{(t)}\right\|_2 \geqslant \delta_1$ within time $t_1 = t_1' + t_1'' + t_1'''$. Then by Lemma B.26 we know $[\bar{v}_k^{(t)}]^2 \geqslant 1 - \alpha$. Therefore, we know there exists $t \leqslant t_1$ such that $S_k^{(t)} \neq \varnothing$. Finally we know it will keep until t_1 by Lemma B.27.

Part 3. The statement directly follows from Lemma B.26 and Lemma B.27. □

Lemma 3.2 (Main Lemma for Phase 1)**.** *In the setting of Theorem 3.1, suppose Proposition 3.1 holds at* $(s, 0)$*. For* $t_1^{(s)} := t_1^{(s)\prime} + t_1^{(s)\prime\prime} + t_1^{(s)\prime\prime\prime}$ *with* $t_1^{(s)\prime} = \Theta(d/(\beta^{(s)} \log d))$, $t_1^{(s)\prime\prime} = \Theta(d/(\beta^{(s)} \log^3 d))$, $t_1^{(s)\prime\prime\prime} = \Theta(\log(d/\alpha)/\beta^{(s)})$*, with probability* $1 - 1/poly(d)$ *we have*

1. *Proposition 3.1 holds at* (s, t) *for any* $0 \leqslant t < t_1^{(s)}$*, and also for* $t = t_1^{(s)}$ *after reinitialization.*
2. *If* $a_k \geqslant \beta^{(s)}$ *and* $S_k^{(s,0)} = \varnothing$*, we have* $S_k^{(s,t_1^{(s)})} \neq \varnothing$ *and* $\hat{a}_k^{(s,t_1^{(s)})} \geqslant \delta_1^2$.
3. *If* $S_k^{(s,0)} = \varnothing$ *and* $S_k^{(s,t_1^{(s)})} \neq \varnothing$*, we have* $a_k \geqslant C\beta^{(s)}$ *for universal constant* $0 < C < 1$.

Proof. By Lemma B.19 we know the number of reinitialized components are always $\Theta(m)$ so Lemma B.19 holds with probability $1 - 1/\text{poly}(d)$ for every epoch. In the following assume Lemma B.19 holds. The second and third statement directly follow from Lemma B.19 and Lemma 3.3 as $S_{k,pot} = \varnothing$ when $a_k \leqslant \beta c_a$. For the first statement, combing the proof in Appendix B.1 and Lemma B.26, we know the statement holds (see also the remark at the beginning of Appendix B.1). □

Preliminary

To simplify the proof in this section, we introduce more notations and give the following lemma.

Lemma B.28. *In the setting of Lemma 3.2, we have* $T^* - T^{(t)} = \sum_{i\in[d]} \tilde{a}_i^{(t)} e_i^{\otimes 4} + \Delta^{(t)}$, *where* $\tilde{a}_i^{(t)} = a_i - \hat{a}_i^{(t)}$ *and* $\|\Delta\|_F = O(\alpha + m\delta_1^2)$. *We know* $\tilde{a}_i^{(0)} = a_i$ *if* $S_i^{(s,0)} = \varnothing$ *and* $\tilde{a}_i^{(t)} = \Theta(\lambda)$ *if* $S_i^{(s,0)} \neq \varnothing$. *That is, the residual tensor is roughly the ground truth tensor* T^* *with unfitted directions at the beginning of this epoch and plus a small perturbation* Δ.

Proof. We can decompose $T^{(t)}$ as

$$T^{(t)} = \sum_{i\in[d]} T_i^{(t)} + T_\varnothing^{(t)} = \sum_{i\in[d]} \left(\hat{a}_i^{(t)} e_i^{\otimes 4} + (T_i^{(t)} - \hat{a}_i^{(t)} e_i^{\otimes 4}) \right) + T_\varnothing^{(t)},$$

where $T_i^{(t)} = \sum_{w\in S_i^{(t)}} \|w\|^2 \bar{w}^{\otimes 4}$ and $T_\varnothing^{(t)} = \sum_{w\in S_\varnothing^{(t)}} \|w\|^2 \bar{w}^{\otimes 4}$. Note that when $S_i^{(t)} = \varnothing$, $\hat{a}_i^{(t)} = 0$ and when $S_i^{(t)} \neq \varnothing$ we have $\left\|(T_i^{(t)} - \hat{a}_i^{(t)} e_i^{\otimes 4})\right\|_F = O(\hat{a}_i^{(t)}\alpha)$ and $\left\|T_\varnothing^{(t)}\right\|_F \leqslant m\delta_1^2$. This gives the desired form of $T^* - T^{(t)}$.

□

We give the dynamic of $[\bar{v}_k^{(t)}]^2$ and $[v_k^{(t)}]^2$ here, which will be frequently used in

our analysis.

$$\begin{aligned}
\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} &= 2\bar{v}_k^{(t)} \cdot \frac{d}{dt}\frac{v_k^{(t)}}{\|v^{(t)}\|} \\
&= 2\bar{v}_k^{(t)} \cdot \frac{1}{\|v^{(t)}\|}\frac{d}{dt}v_k^{(t)} + 2[\bar{v}_k^{(t)}]^2 \cdot \frac{d}{dt}\frac{1}{\|v\|} \\
&= 2\bar{v}_k^{(t)} \cdot \frac{1}{\|v^{(t)}\|}[-\nabla L(v^{(t)})]_k - 2[\bar{v}_k^{(t)}]^2 \cdot \frac{\left\langle \bar{v}^{(t)}, -\nabla L(v^{(t)})\right\rangle}{\|v^{(t)}\|} \\
&= 2\bar{v}_k^{(t)} \cdot \frac{1}{\|v^{(t)}\|}[-(I - \bar{v}^{(t)}[\bar{v}^{(t)}]^\top)\nabla L(v^{(t)})]_k \\
&= 8\bar{v}_k^{(t)}\left[(T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 3}, I) - (T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})\bar{v}^{(t)}\right]_k \\
&= 8[\bar{v}_k^{(t)}]^2\left(\tilde{a}_k^{(t)}[\bar{v}_k^{(t)}]^2 - \sum_{i\in[d]}\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F}{|\bar{v}_k^{(t)}|}\right).
\end{aligned} \tag{B.2}$$

$$\begin{aligned}
\frac{\mathrm{d}[v_k^{(t)}]^2}{\mathrm{d}t} &= 2v_k^{(t)} \cdot \frac{\mathrm{d}v_k^{(t)}}{\mathrm{d}t} \\
&= 2v_k^{(t)} \cdot [-\nabla L(v^{(t)})]_k \\
&= 4v_k^{(t)}\left[2(T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 3}, I)\left\|v^{(t)}\right\|_2 - (T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4})v^{(t)}\right]_k \\
&= 4[v_k^{(t)}]^2\left(2\tilde{a}_k^{(t)}[\bar{v}_k^{(t)}]^2 - \sum_{i\in[d]}\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F\left\|v^{(t)}\right\|_2}{|v_k^{(t)}|}\right).
\end{aligned} \tag{B.3}$$

The following lemma allows us to ignore these already fitted direction as they will remain as small as their (re-)initialization in phase 1.

Lemma B.29. *In the setting of Lemma 3.2, if direction e_k has been fitted before current epoch (i.e., $S_k^{(s,0)} \neq \varnothing$), then for v that was reinitialized in the previous epoch, we have $[\bar{v}_k^{(t)}]^2 = O(\log(d)/d)$ for all $t \leqslant t_1$.*

Proof. Since direction e_k has been fitted before current epoch, we know $\tilde{a}_k^{(t)} = \Theta(\lambda)$. We only need to consider the time when $[\bar{v}_k^{(t)}]^2 \geqslant \log d/d$. By (B.2) we have

$$\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} = 8[\bar{v}_k^{(t)}]^2 \left(\tilde{a}_k^{(t)}[\bar{v}_k^{(t)}]^2 - \sum_{i\in[d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F}{|\bar{v}_k^{(t)}|} \right) \leqslant [\bar{v}_k^{(t)}]^2 O\left(\lambda + d\left\|\Delta^{(t)}\right\|_F\right).$$

Since λ and $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$ are small enough and $[\bar{v}_k^{(0)}]^2 = O(\log d/d)$, we know $[\bar{v}_k^{(t)}]^2 = O(\log d/d)$ for $t \leqslant t_1$.

□

Proof of Lemma B.21 and Lemma B.22

Lemma B.21 directly follows from Lemma B.22 and the definition of S_{good}, S_{pot} and S_{bad} as in Definition 3.2. We focus on Lemma B.22 in the rest of this section. We need following lemma to give the proof of Lemma B.22.

Lemma B.30. *In the setting of Lemma 3.2, if* $\left\|\bar{v}^{(t)}\right\|_\infty^2 \leqslant \log^4(d)/d$, *we have* $\sum_i [\bar{v}_i^{(t)}]^4 \leqslant c_e \log d/d$ *for all* $t \leqslant t_1$.

Proof. We claim that for all $t \leqslant t_1$, there are at most $O(\log d)$ many $i \in [d]$ such that $[\bar{v}_i^{(t)}]^2 \geqslant c_e \log(d)/2d$. Based on this claim, we know

$$\begin{aligned}\sum_{i\in[d]} [\bar{v}_i^{(t)}]^4 \leqslant & O(\log d)\frac{\log^8 d}{d^2} + \sum_{i:[\bar{v}_i^{(t)}]^2 < c_e \log(d)/2d} [\bar{v}_i^{(t)}]^4 \\ \leqslant & O\left(\frac{\log^9 d}{d^2}\right) + \frac{c_e \log(d)}{2d} \leqslant \frac{c_e \log(d)}{d},\end{aligned}$$

which gives the desired result.

In the following, we prove the above claim. From Lemma B.19, we know when $t = 0$, the claim is true. For any $[\bar{v}_k^{(0)}]^2 \leqslant c_e \log(d)/10d$, we will show $[\bar{v}_k^{(t)}]^2 \leqslant c_e \log(d)/2d$ for all $t \leqslant t_1$. By (B.2) we have

$$\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} = 8[\bar{v}_k^{(t)}]^2 \left(\tilde{a}_k^{(t)}[\bar{v}_k^{(t)}]^2 - \sum_{i\in[d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F}{|\bar{v}_k^{(t)}|} \right).$$

In fact, we only need to show that for any τ_0 such that $[\bar{v}_k^{(\tau_0)}]^2 = c_e \log(d)/10d$ and $[\bar{v}_k^{(t)}]^2 \geqslant c_e \log(d)/10d$ when $\tau_0 \leqslant t \leqslant \tau_0 + t_1$, we have $[\bar{v}_k^{(t)}]^2 \leqslant c_e \log(d)/2d$. To show this, we have

$$\begin{aligned}\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} \leqslant & 8[\bar{v}_k^{(t)}]^2 \left(\tilde{a}_k^{(t)} [\bar{v}_k^{(t)}]^2 + \frac{\left\| \Delta^{(t)} \right\|_F}{|\bar{v}_k^{(t)}|} \right) \\ \leqslant & [\bar{v}_k^{(t)}]^2 \cdot 16 \tilde{a}_k^{(t)} [\bar{v}_k^{(t)}]^2 \leqslant [\bar{v}_k^{(t)}]^2 \cdot \frac{\beta}{1-\gamma} \cdot \frac{8 c_e \log(d)}{d},\end{aligned}$$

where we use $\left\| \Delta^{(t)} \right\|_F = O(\alpha + m\delta_1^2)$ and $\tilde{a}_k^{(t)} \leqslant \beta/(1-\gamma)$. Therefore, with our choice of t_1, we know $[\bar{v}_k^{(t)}]^2 \leqslant c_e \log(d)/2d$. This finish the proof.

□

We now are ready to give the proof of Lemma B.22.

Lemma B.22. *In the setting of Lemma 3.2, we have*

1. *if* $[\bar{v}_k^{(0)}]^2 \leqslant \min\{\Gamma_k - \rho_k, \Gamma_{max}\}$, *then* $[\bar{v}_k^{(t)}]^2 = O(\log(d)/d)$ *for* $t \leqslant t_1'$.
2. *if* $S_k^{(t)} = 0$ *for* $t \leqslant t_1'$, $[\bar{v}_k^{(0)}]^2 \geqslant \Gamma_k + \rho_k$, $[\bar{v}_i^{(0)}]^2 \leqslant \Gamma_i - \rho_i$ *for all* $i \neq k$ *and* $\left\| \bar{v}^{(0)} \right\|_\infty^2 \leqslant \log^4(d)/d$, *then there exists* $t \leqslant t_1'$ *such that* $[\bar{v}_k^{(t)}]^2 \geqslant \log^4(d)/d$.

Proof. We focus on the dynamic of $[\bar{v}_k^{(t)}]^2$. For those already fitted direction e_k, we have $\Gamma_k = 1/(8\lambda t_1')$, which means $\Gamma_{max} \leqslant \Gamma_k - \rho_k$. From Lemma B.29 we know $[\bar{v}_k^{(t)}]^2 = O(\log d/d)$ for $t \leqslant t_1'$. In the rest of proof, we focus on these unfitted direction e_k. By (B.2) we have

$$\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} = 8[\bar{v}_k^{(t)}]^2 \left(\tilde{a}_k^{(t)} [\bar{v}_k^{(t)}]^2 - \sum_{i \in [d]} \tilde{a}_i^{(t)} [\bar{v}_i^{(t)}]^4 \pm \frac{\left\| \Delta^{(t)} \right\|_F}{|\bar{v}_k^{(t)}|} \right)$$

Part 1. Define the following dynamics $p^{(t)}$,

$$\frac{\mathrm{d}p^{(t)}}{\mathrm{d}t} = 8p^{(t)} \left(a_k p^{(t)} + \frac{a_k c_e \log d}{d} \right), \quad p^{(0)} = [\bar{v}_k^{(0)}]^2$$

Given that $\tilde{a}_i^{(t)} \leqslant a_i$ and $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$ is small enough, it is easy to see $[\bar{v}_k^{(t)}]^2 \leqslant \max\{\log(d)/d, p^{(t)}\}$. Then it suffices to bound $p^{(t)}$ to have a bound for $[\bar{v}_k^{(t)}]^2$. Consider the following dynamic $x^{(t)}$

$$\frac{\mathrm{d}x^{(t)}}{\mathrm{d}t} = \tau_1[x^{(t)}]^2, \quad x^{(0)} = \tau_2. \tag{B.4}$$

We know $x^{(t)} = 1/(1/\tau_2 - \tau_1 t)$. Set $\tau_1 = 8a_k$ and $\tau_2 = 1/(\tau_1 t_1') = \Gamma_k$. Then, with our choice of $\rho_k = c_\rho \Gamma_k$, we know

1. $p^{(0)} = [\bar{v}_k^{(0)}]^2 \leqslant \Gamma_k - \rho_k \leqslant \Gamma_{max}$. As long as $\rho_k \geqslant \frac{2c_e \log d}{d}$ and $x^{(0)} = p^{(0)} + \rho_k/2$, we have $p^{(t)} \leqslant x^{(t)} - \rho_k/2$ for $t \leqslant t_1'$. Therefore, $p^{(t_1')} \leqslant x^{(t_1')} \leqslant 2\Gamma_k^2/\rho_k = O(\log d/d)$.

2. $p^{(0)} = [\bar{v}_k^{(0)}]^2 \leqslant \Gamma_{max} < \Gamma_k - \rho_k$. As long as $x^{(0)} = p^{(0)} + \frac{c_e \log d}{d}$, we have $p^{(t)} \leqslant x^{(t)} - \frac{c_e \log d}{d}$ for $t \leqslant t_1'$. Therefore, $p^{(t_1')} \leqslant x^{(t_1')} = O(\log d/d)$.

Together we know $[\bar{v}_k^{(t)}]^2 = O(\log d/d)$ for $t \leqslant t_1'$.

Part 2. Define the following dynamics $q^{(t)}$,

$$\frac{\mathrm{d}q^{(t)}}{\mathrm{d}t} = 8q^{(t)}\left(a_k q^{(t)} - \frac{2\beta c_e \log d}{d}\right), \quad q^{(0)} = [\bar{v}_k^{(0)}]^2.$$

Since $S_k^{(t)} = \varnothing$, we know $\tilde{a}_k^{(t)} = a_k$. Given that $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$ and Lemma B.30, it is easy to see as long as $\left\|\bar{v}^{(t)}\right\|_\infty^2 \leqslant \log^4 d/d$, if $q^{(0)} \geqslant [\bar{v}_k^{(0)}]^2 \geqslant \Theta(\log d/d)$ and $a_k[q^{(0)}]^2 - \frac{2\beta c_e \log d}{d} > 0$, we have $[\bar{v}_k^{(t)}]^2 \geqslant q^{(t)}$. Then it suffices to bound $q^{(t)}$ to get a bound on $[v_k^{(t)}]^2$. Consider the same dynamic (B.4) with same τ_1 and τ_2, as long as $q^{(0)} = [\bar{v}_k^{(0)}]^2 \geqslant \Gamma_k + \rho_k$, $\rho_k \geqslant \frac{4\beta c_e \log d}{a_k d}$ and $x^{(0)} = q^{(0)} - \rho_k/2$, we have $q^{(t)} \geqslant x^{(t)} + \rho_k/2$ if $\left\|\bar{v}^{(t)}\right\|_\infty^2 \leqslant \log^4 d/d$ holds. We can verify that $x^{(T_1')} = +\infty$, which implies there exists $t \leqslant t_1'$ such that $\left\|\bar{v}^{(t)}\right\|_\infty^2 > \log^4 d/d$.

□

Proof of Lemma B.23

Lemma B.23. *In the setting of Lemma 3.2, suppose* $[\bar{v}_i^{(t_1')}]^2 = O(\log(d)/d)$. *Then we have* $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for* $t_1' \leqslant t \leqslant t_1$.

Proof. Recall $t_1 - t_1' = t_1'' + t_1''' = o(d/(\beta \log d))$, it suffices to show if $[\bar{v}_i^{(t_1')}]^2 = c_1 \log(d)/d$, then $[\bar{v}_i^{(t)}]^2$ will be at most $2c_1 \log(d)/d$ in $t'_{max} = o(d/(\beta \log d))$ time. Suppose there exists time $\tau_1 \leqslant t'_{max}$ such that $[\bar{v}_i^{(\tau_1)}]^2 \geqslant 2c_1 \log(d)/d$ for the first time. We only need to show if $[\bar{v}_i^{(t)}]^2 \geqslant c_1 \log(d)/d$ for $t \leqslant \tau_1$, we have $[\bar{v}_i^{(t)}]^2 < 2c_1 \log(d)/d$. We know the dynamic of $[\bar{v}_i^{(t)}]^2$

$$\frac{\mathrm{d}[\bar{v}_i^{(t)}]^2}{\mathrm{d}t} = 8[\bar{v}_i^{(t)}]^2 \left(\tilde{a}_k^{(t)} [\bar{v}_i^{(t)}]^2 - \sum_{j \in [d]} \tilde{a}_j^{(t)} [\bar{v}_j^{(t)}]^4 \pm \frac{\left\| \Delta^{(t)} \right\|_F}{|\bar{v}_i^{(t)}|} \right) \leqslant [\bar{v}_i^{(t)}]^2 O\left(\frac{\beta \log d}{d} \right),$$

where we use $\left\| \Delta^{(t)} \right\|_F = O(\alpha + m\delta_1^2)$ is small enough and $\tilde{a}_k^{(t)} \leqslant 1$. This implies $[\bar{v}_i^{(t)}]^2 \leqslant 2c_1 \log d/d$ as $t'_{max} = o(d/(\beta \log d))$. □

Proof of Lemma B.24

Lemma B.24 (Good component, constant correlation)**.** *In the setting of Lemma 3.2, suppose* $S_k^{(t)} = \varnothing$ *for* $t \leqslant t_1$, $a_k = \Omega(\beta)$. *If there exists* $\tau_0 \leqslant t_1$ *such that* $[\bar{v}_k^{(\tau_0)}]^2 > \log^4(d)/d$ *and* $[\bar{v}_i^{(\tau_0)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$, *then for any constant* $c \in (0,1)$ *we have* $[\bar{v}_k^{(t)}]^2 > c$ *and* $[\bar{v}_i^{(t)}]^2 = O(\log(d)/d)$ *for all* $i \neq k$ *when* $\tau_0 + t_1'' \leqslant t \leqslant t_1$ *with* $t_1'' = \Theta(d/(\beta \log^3 d))$.

Proof. By Lemma B.23 we know $[\bar{v}_i^{(t)}]^2$ will remain $O(\log d/d)$ for those $[\bar{v}_i^{(\tau_0)}]^2 = O(\log d/d)$.

We now show $[\bar{v}_k^{(t)}]^2$ will become constant within t_1'' time. We know $\sum_{i \neq k} \tilde{a}_i^{(t)} [\bar{v}_i^{(t)}]^4 \leqslant \beta c_1 \log d/d$ for some constant c_1. Hence, with the fact $S_k^{(t)} = \varnothing$,

$a_k = \Omega(\beta)$, $[\bar{v}_k^{(\tau_0)}]^2 > \log^4(d)/d$ and $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$,

$$\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} = 8[\bar{v}_k^{(t)}]^2 \left(\tilde{a}_k^{(t)}[\bar{v}_k^{(t)}]^2(1 - [\bar{v}_k^{(t)}]^2) - \sum_{i \neq k} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F}{|\bar{v}_k^{(t)}|} \right)$$

$$\geqslant 8(1-2c)[\bar{v}_k^{(t)}]^2 a_k [\bar{v}_k^{(t)}]^2 = [\bar{v}_k^{(t)}]^2 \Omega\left(\frac{\beta \log^4 d}{d}\right).$$

This implies that within t_1'' time, we have $[\bar{v}_k^{(t)}]^2 \geqslant c$. Since $[\bar{v}_i^{(t)}]^2$ will remain $O(\log d/d)$ for $i \neq k$ and $t \leqslant t_1$, following the same argument above, it is easy to see $\frac{\mathrm{d}[\bar{v}_k^{(t)}]^2}{\mathrm{d}t} \geqslant 0$ after $[\bar{v}_k^{(t)}]^2$ reaches c. Therefore, $[\bar{v}_k^{(t)}]^2 \geqslant c$ for $t \leqslant t_1$.

□

Proof of Lemma B.25

Lemma B.25 (Good component, norm growth)**.** *In the setting of Lemma 3.2, suppose $S_k^{(t)} = \varnothing$ for $t \leqslant t_1$, $a_k = \Omega(\beta)$. If there exists $\tau_0' \leqslant t_1$ such that $[\bar{v}_k^{(\tau_0')}]^2 > c$ and $[\bar{v}_i^{(\tau_0')}]^2 = O(\log(d)/d)$ for all $i \neq k$, then we have $\left\|v^{(t)}\right\|_2 \geqslant \delta_1$ for some $\tau_0' \leqslant t \leqslant \tau_0' + t_1'''$ with $t_1''' = \Theta(\log(d/\alpha)/\beta)$.*

Proof. For $\left\|v^{(t)}\right\|_2^2$, we have

$$\frac{\mathrm{d}\left\|v^{(t)}\right\|_2^2}{\mathrm{d}t} = \left\|v^{(t)}\right\|^2 \left(4 \sum_{i \in [d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \left\|\Delta^{(t)}\right\|_F - 2\lambda \right).$$

Given the fact $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$ and λ are small enough , it is easy to see $\left\|v^{(\tau_0')}\right\|_2 \geqslant \delta_0/2$ as $\tau_0' \leqslant t_1$. We now show that there exist time $\tau_1 \leqslant t_1' + t_1'' + t_1''' = t_1$ such that $\left\|v^{(\tau_1)}\right\|_2 \geqslant \delta_1$. By Lemma B.24 we know $[\bar{v}_k^{(t)}]^2 \geqslant c$ after time $\tau_0 + t_1' \leqslant t_1' + t_1''$. And since $S_k^{(t)} = \varnothing$, we know $\tilde{a}_k^{(t)} = a_k = \Omega(\beta)$. Then with the fact that $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$ and λ are small enough, we have

$$\frac{\mathrm{d}\left\|v^{(t)}\right\|^2}{\mathrm{d}t} \geqslant \left\|v^{(t)}\right\|^2 \Omega(\beta).$$

This implies that $\left\|v^{(\tau_1)}\right\|_2^2 \geqslant \delta_1^2$ as $t_1''' = \Theta(\log(d/\alpha)/\beta)$. □

Proof of Lemma B.26

Lemma B.26. *In the setting of Lemma 3.2, we have*

1. *if* $\left\|\bar{v}^{(t)}\right\|_\infty^2 \leqslant \log^4(d)/d$ *for all* $t \leqslant t_1$, *then* $\left\|v^{(t)}\right\|_2 = O(\delta_0)$ *for all* $t \leqslant t_1$.

2. *Let* $\tau_0 = \inf\{t \in [0, t_1] | \left\|\bar{v}^{(t)}\right\|_\infty^2 \geqslant \log^4 d/d\}$. *Suppose* $[\bar{v}_k^{(\tau_0)}]^2 \geqslant \log^4 d/d$ *and* $[\bar{v}_i^{(\tau_0)}]^2 = O(\log d/d)$ *for* $i \neq k$. *If there exists* τ_1 *such that* $\tau_0 < \tau_1 \leqslant t_1$ *and* $\left\|v^{(\tau_1)}\right\|_2 \geqslant \delta_1$ *for the first time, then there exists* $k \in [d]$ *such that* $[\bar{v}_k^{(\tau_1)}]^2 \geqslant 1-\alpha^2$ *if* $\hat{a}_k^{(t)} \leqslant \alpha$ *for* $t \leqslant \tau_1$ *and* $[\bar{v}_k^{(\tau_1)}]^2 \geqslant 1-\alpha$ *otherwise.*

Proof. For $\left\|v^{(t)}\right\|_2^2$, we have

$$\frac{\mathrm{d}\left\|v^{(t)}\right\|_2^2}{\mathrm{d}t} = \left\|v^{(t)}\right\|^2 \left(4 \sum_{i\in[d]} \tilde{a}_i^{(t)} [\bar{v}_i^{(t)}]^4 \pm \left\|\Delta^{(t)}\right\|_F - 2\lambda\right)$$

Part 1. By Lemma B.30 and $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$, we know

$$\frac{\mathrm{d}\left\|v^{(t)}\right\|^2}{\mathrm{d}t} \leqslant \left\|v^{(t)}\right\|^2 \frac{5\beta c_e \log d}{d}.$$

This implies $\left\|v^{(t)}\right\|_2^2 = O(\delta_0)$ as $t_1 = O(\frac{d}{\beta \log d})$.

Part 2. By Part 1, we know $\left\|v^{(\tau_0)}\right\|_2 = O(\delta_0)$ and $[v_i^{(\tau_0)}]^2 = O(\delta_0^2 \log d/d)$ for $i \neq k$. For $[\bar{v}_i^{(\tau_0)}]^2 = O(\log d/d)$, we know $[\bar{v}_i^{(t)}]^2 = O(\log d/d)$ for $\tau_0 \leqslant t \leqslant \tau_1$ by Lemma B.23. We consider following cases separately.

1. Case 1: Suppose $\hat{a}_k^{(t)} \leqslant \alpha$ for $t \leqslant \tau_1$. In the following we show there exists some constant C such that for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant C\delta_0^2 \log d/d$ for $\tau_0 \leqslant t \leqslant \tau_1$. Let τ_2 be the first time that the above claim is false, which means for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant C\delta_0^2 \log d/d$ when $t \leqslant \tau_2$.

For any $i \neq k$, we only need to consider the time period $t \leqslant \tau_2$ whenever $[v_i^{(t)}]^2 \geqslant \delta_0^2 \log d/d$. By Lemma B.32, we have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2 =& 4[v_i^{(t)}]^2 \left(2\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm O(\alpha + m\delta_1^2) \right. \\ & \left. \pm O\left(\frac{(\alpha^2 + d\alpha^3 + d\alpha(1-[\bar{v}_k^{(t)}]^2)^{1.5} + m\delta_1^2)\left\|v^{(t)}\right\|}{|v_i^{(t)}|} \right) \right) \\ \leqslant & [v_i^{(t)}]^2 \left(O\left(\frac{\beta \log d}{d}\right) + O\left(\frac{(\alpha^2 + \alpha(1-[\bar{v}_k^{(t)}]^2)^{1.5} + m\delta_1^2)\left\|v^{(t)}\right\|}{|v_i^{(t)}|} \right) \right).\end{aligned}$$

Since for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant C\delta_0^2 \log d/d$, we know $\sum_{i\neq k}[v_i^{(t)}]^2 = \left\|v^{(t)}\right\|^2 (1 - [\bar{v}_k^{(t)}]^2) = O(\delta_0^2 \log d)$. Together with the fact $[v_i^{(t)}]^2 \geqslant \delta_0^2 \log d/d$, we have

$$\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2 \leqslant [v_i^{(t)}]^2 O\left(\frac{\beta \log d}{d}\right).$$

Since $t_1 = O(d/(\beta \log d))$, we know if we choose large enough C, it must be $\tau_2 \geqslant \tau_1$. Therefore, we know for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant C\delta_0^2 \log d/d$ for $\tau_0 \leqslant t \leqslant \tau_1$. Then at time τ_1 when $\left\|v^{(\tau_1)}\right\|_2 \geqslant \delta_1$, it must be $[\bar{v}_k^{(t)}]^2 \geqslant 1 - \alpha^2$ since $\delta_1 = \Theta(\delta_0 \log^{1/2}(d)/\alpha)$.

2. Case 2: We do not make assumption on $\hat{a}_k^{(t)}$. In the following we show there exists some constant C such that for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant \delta_1^2\alpha/d$ for $\tau_0 \leqslant t \leqslant \tau_1$. Let τ_2 be the first time that the above claim is false, which means for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant \delta_1^2\alpha/d$ when $t \leqslant \tau_2$.

   For any $i \neq k$, we only need to consider the time period $t \leqslant \tau_2$ whenever

$[v_i^{(t)}]^2 \geqslant \delta_1^2\alpha/2d$. We have

$$\frac{\mathrm{d}[v_i^{(t)}]^2}{\mathrm{d}t} = 4[v_i^{(t)}]^2\left(2\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]}\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \frac{\left\|\Delta^{(t)}\right\|_F\left\|v^{(t)}\right\|_2}{|v_i^{(t)}|}\right)$$

$$\leqslant [v_i^{(t)}]^2\left(O\left(\frac{\beta\log d}{d}\right) + O\left(\frac{\alpha + m\delta_1^2}{\alpha^{1/2}d^{-1/2}}\right)\right).$$

Since $m\delta_1^2 = O(\alpha)$ and $t_1 = O(d/(\beta\log d))$, we know it must be $\tau_2 \geqslant \tau_1$. Therefore, we know for all $i \neq k$ $[v_i^{(t)}]^2 \leqslant \delta_1^2\alpha/d$ for $\tau_0 \leqslant t \leqslant \tau_1$. Then at time τ_1 when $\left\|v^{(\tau_1)}\right\|_2 \geqslant \delta_1$, it must be $[\bar{v}_k^{(t)}]^2 \geqslant 1-\alpha$.

□

Proof of Lemma B.27

To prove Lemma B.27, we need the following calculation on $\frac{d}{dt}\left\|v^{(t)}\right\|^2$.

Lemma B.31. *Suppose $v^{(t)} \in S_k^{(t)}$, we have*

$$\frac{d}{dt}\left\|v^{(t)}\right\|^2 = \left(4\tilde{a}_k^{(t)} - 2\lambda \pm O(\alpha + m\delta_1^2)\right)\left\|v^{(t)}\right\|^2.$$

Proof. We can write down $\frac{d}{dt}\left\|v^{(t)}\right\|^2$ as follows:

$$\frac{d}{dt}\left\|v^{(t)}\right\|^2 = \left(4(T^* - T^{(t)})([\bar{v}^{(t)}]^{\otimes 4}) - 2\lambda\right)\left\|v^{(t)}\right\|^2$$

$$= \left(4\sum_{i\in[d]}\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm \left\|\Delta^{(t)}\right\|_F - 2\lambda\right)\left\|v^{(t)}\right\|^2$$

Since $[\bar{v}_k^{(t)}]^2 \geqslant 1-\alpha$, $[\bar{v}_i^{(t)}]^2 \leqslant \alpha$ for any $i \neq k$ and $\left\|\Delta^{(t)}\right\|_F = O(\alpha + m\delta_1^2)$, we have

$$\frac{d}{dt}\left\|v^{(t)}\right\|^2 = \left(4\tilde{a}_k^{(t)} - 2\lambda \pm O(\alpha + m\delta_1^2)\right)\left\|v^{(t)}\right\|^2.$$

□

Now we are ready to prove Lemma B.27.

Lemma B.27. *If $S_k^{(0)} = \varnothing$ and $S_k^{(t')} \neq \varnothing$ for some $t' \in (0, t_1]$, we have $S_k^{(t_1)} \neq \varnothing$ and $\hat{a}_k^{(t_1)} \geqslant \delta_1^2$.*

Proof. If $\tilde{a}_k^{(t)} = \Omega(\lambda)$ through Phase 1, according to Lemma B.31, we know $\left\|v^{(t)}\right\|^2$ will never decrease for any $v^{(t)} \in S_k^{(t)}$. So, we have $S_k^{(t_1)} \neq \varnothing$ and $\hat{a}_k^{(t_1)} \geqslant \delta_1^2$.

If $\tilde{a}_k^{(t)} = O(\lambda)$ at some time in Phase 1, according to Lemma B.18, it's not hard to show at the end of Phase 1 we still have $a_k - \hat{a}_k^{(t_1)} = O(\lambda)$. This then implies $\hat{a}_k^{(t_1)} = \Omega(\frac{\epsilon}{\sqrt{d}})$. Note that we only re-initialize the components that have norm less than δ_1. As long as $\delta_1^2 = O(\frac{\epsilon}{m\sqrt{d}})$, we ensure that after the re-initialization, we still have $\hat{a}_k^{(t_1)} = \Omega(\frac{\epsilon}{\sqrt{d}})$, which of course means $S_k^{(t_1)} \neq \varnothing$. □

Technical Lemma

Lemma B.32. *In the setting of Lemma B.26, suppose $\hat{a}_k^{(t)} \leqslant \alpha$. We have for $i \neq k$*

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2 =& 4[v_i^{(t)}]^2 \left(2\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm O(\alpha + m\delta_1^2) \right. \\ & \left. \pm O\left(\frac{(\alpha^2 + \alpha(1 - [\bar{v}_k^{(t)}]^2)^{1.5} + m\delta_1^2) \left\|v^{(t)}\right\|}{|v_i^{(t)}|} \right) \right).\end{aligned}$$

Proof. In order to prove this lemma, we need a more careful analysis on $\frac{d}{dt}[v_i^{(t)}]^2$. Recall we can decompose $T^{(t)}$ as $\sum_{i\in[d]} T_i^{(t)} + T_\varnothing^{(t)}$ and further write each $T_i^{(t)}$ as $\hat{a}_i^{(t)} e_i^{\otimes 4} + (T_i^{(t)} - \hat{a}_i^{(t)} e_i^{\otimes 4})$. Note that $\left\|(T_i^{(t)} - \hat{a}_i^{(t)} e_i^{\otimes 4})\right\|_F = O(\hat{a}_i^{(t)}\alpha)$ and $\left\|T_\varnothing^{(t)}\right\|_F \leqslant m\delta_1^2$.

We can write down $\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2$ in the following form:

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2 =& 4[v_i^{(t)}]^2 \left(2a_i[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} a_i[\bar{v}_i^{(t)}]^4 \right) \\
& - 8v_i^{(t)} \left\|v^{(t)}\right\| \sum_{j\in[d]} \left[T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i - 8v_i^{(t)} \left\|v^{(t)}\right\| \left[T_\varnothing^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i \\
& + 4v_i^{(t)} \sum_{j\in[d]} \left[T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 4}) v^{(t)} \right]_i + 4v_i^{(t)} \left[(T_\varnothing^{(t)}([\bar{v}^{(t)}]^{\otimes 4}) v^{(t)} \right]_i \\
& = 4[v_i^{(t)}]^2 \left(2a_i[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} a_i[\bar{v}_i^{(t)}]^4 \right) \\
& - 8v_i^{(t)} \left\|v^{(t)}\right\| \sum_{j\in[d]} \left[T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i \pm v_i^{(t)} \left\|v^{(t)}\right\| O(m\delta_1^2) \\
& + 4[v_i^{(t)}]^2 \sum_{j\in[d]} T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 4}) \pm [v_i^{(t)}]^2 O(m\delta_1^2) \\
& = 4[v_i^{(t)}]^2 \left(2a_i[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} (a_i - \hat{a}_i)[\bar{v}_i^{(t)}]^4 \pm O(\alpha + m\delta_1^2) \right) \\
& - 8v_i^{(t)} \left\|v^{(t)}\right\| \sum_{j\in[d]} \left[T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i \pm v_i^{(t)} \left\|v^{(t)}\right\| O(m\delta_1^2).
\end{aligned}$$

We now bound the term $\left[T_j^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i$.

1. Case 1: $j = i$. If $\hat{a}_i^{(t)} = 0$, we know $T_i^{(t)} = 0$. Otherwise, denote $x = \left\langle \bar{w}_{-i}, \bar{v}_{-i}^{(t)} \right\rangle$, we have

$$\begin{aligned}
& \left[T_i^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I) \right]_i \\
& = \hat{a}_i^{(t)} \mathbb{E}_{i,w}^{(t)} \bar{w}_i \left\langle \bar{w}, \bar{v}^{(t)} \right\rangle^3 \\
& = \hat{a}_i^{(t)} \mathbb{E}_{i,w}^{(t)} \bar{w}_i \left((\bar{w}_i \bar{v}_i^{(t)})^3 + (\bar{w}_i \bar{v}_i^{(t)})^2 x + (\bar{w}_i \bar{v}_i^{(t)}) x^2 + x^3 \right) \\
& \leqslant \hat{a}_i^{(t)} [\bar{v}_i^{(t)}]^3 + \hat{a}_i^{(t)} |\bar{v}_i^{(t)}| \mathbb{E}_{i,w}^{(t)} |x| + \hat{a}_i^{(t)} |\bar{v}_i^{(t)}| \mathbb{E}_{i,w}^{(t)} x^2 + \hat{a}_i^{(t)} \mathbb{E}_{i,w}^{(t)} x^3.
\end{aligned}$$

Since $|x| \leqslant \|\bar{w}_{-1}\|$ and $\mathbb{E}_{i,w}^{(t)} \|\bar{w}_{-i}\| \leqslant (\mathbb{E}_{i,w}^{(t)} \|\bar{w}_{-i}\|^2)^{1/2} = O(\alpha)$, we have $\left[T_i^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I)\right]_i = \hat{a}_i^{(t)}[\bar{v}_i^{(t)}]^3 + \hat{a}_i^{(t)}|\bar{v}_i^{(t)}|O(\alpha) + \hat{a}_i^{(t)}O(\alpha^{2.5})$.

2. Case 2: $j = k$. We have $\left[T_k^{(t)}([\bar{v}^{(t)}]^{\otimes 3}, I)\right]_i = \hat{a}_k^{(t)}\mathbb{E}_{k,w}^{(t)}\bar{w}_i \left\langle \bar{w}, \bar{v}^{(t)} \right\rangle^3 \leqslant \hat{a}_k^{(t)}\mathbb{E}_{k,w}^{(t)}|\bar{w}_i|$ $= O(\alpha^2)$, since $\hat{a}_k^{(t)} \leqslant \alpha$ and $\mathbb{E}_{k,w}^{(t)}|\bar{w}_i| \leqslant (\mathbb{E}_{k,w}^{(t)}|\bar{w}_i|^2)^{1/2} = O(\alpha)$.

3. Case 3: $j \neq i, k$. $j \neq i, k$. If $\hat{a}_j^{(t)} = 0$, we know $T_j^{(t)} = 0$. Otherwise, we can write $T_j^{(t)}$ as $\hat{a}_j^{(t)}\mathbb{E}_{j,w}^{(t)}\bar{w}^{\otimes 4}$. So we just need to bound $\mathbb{E}_{j,w}^{(t)}\bar{w}_i \left\langle \bar{w}, \bar{v}^{(t)} \right\rangle^3$. We know $\left|\left\langle \bar{w}, \bar{v}^{(t)} \right\rangle\right| = \left|\left\langle \bar{w}_{-j}, \bar{v}_{-j}^{(t)} \right\rangle + \bar{w}_j\bar{v}_j^{(t)}\right| \leqslant \|\bar{w}_{-j}\| + \sqrt{1 - [\bar{v}_k^{(t)}]^2}$. So we have

$$\begin{aligned}\mathbb{E}_{j,w}^{(t)}\bar{w}_i \left\langle \bar{w}, \bar{v}^{(t)} \right\rangle^3 =& \mathbb{E}_{j,w}^{(t)}\bar{w}_i O\left(\|\bar{w}_{-j}\|^3 + (1 - [\bar{v}_k^{(t)}]^2)^{1.5}\right) \\ \leqslant& O\left(\alpha^3 + \alpha(1 - [\bar{v}_k^{(t)}]^2)^{1.5}\right),\end{aligned}$$

where in the lase line we use $\mathbb{E}_{j,w}^{(t)}\bar{w}_i \leqslant (\mathbb{E}_{j,w}^{(t)}\bar{w}_i^2)^{1/2} = O(\alpha)$.

Recall that $\tilde{a}_i^{(t)} = a_i - \hat{a}_i^{(t)}$. We now have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}[v_i^{(t)}]^2 =& 4[v_i^{(t)}]^2 \left(2\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^2 - \sum_{i\in[d]} \tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 \pm O(\alpha + m\delta_1^2) \right. \\ & \left. \pm O\left(\frac{(\alpha^2 + \alpha(1 - [\bar{v}_k^{(t)}]^2)^{1.5} + m\delta_1^2)\left\|v^{(t)}\right\|}{|v_i^{(t)}|} \right) \right).\end{aligned}$$

□

B.3 Proofs for Phase 2

The goal of this section is to show that all discovered directions can be fitted within time $t_2^{(s)} - t_1^{(s)}$ and the reinitialized components will not move significantly. Namely, we prove the following lemma.

Lemma 3.4 (Main Lemma for Phase 2)**.** *In the setting of Theorem 3.1, suppose Proposition 3.1 holds at* $(s, t_1^{(s)})$*, we have for* $t_2^{(s)} - t_1^{(s)} := O(\frac{\log(1/\delta_1)+\log(1/\lambda)}{\beta^{(s)}})$

1. *Proposition 3.1 holds at* (s,t) *for any* $t_1^{(s)} \leqslant t \leqslant t_2^{(s)}$.
2. *If* $S_k^{(s,t_1^{(s)})} \neq \varnothing$*, we have* $a_k - \hat{a}_k^{(s,t_2^{(s)})} \leqslant 2\lambda$.
3. *For any component* v *that was reinitialized at* $t_1^{(s)}$*, we have* $\left\|v^{(s,t_2^{(s)})}\right\|^2 = \Theta(\delta_0^2)$ *and* $\left[\bar{v}_i^{(s,t_2^{(s)})}\right]^2 = \left[\bar{v}_i^{(s,t_1^{(s)})}\right]^2 \pm o\left(\frac{\log d}{d}\right)$ *for every* $i \in [d]$.

Note that since $\delta_1^2 = \mathrm{poly}(\varepsilon)/\mathrm{poly}(d)$ and $\log(d/\varepsilon) = o(d/\log d)$, we have $t_2^{(s)} - t_1^{(s)} = \frac{o(d/\log d)}{\beta^{(s)}}$.

Notations As in Sec. B.1, to simplify the notations, we shall drop the superscript of epoch s, and write $z^{(t)} := \left\langle \bar{v}^{(t)}, \bar{w}^{(t)} \right\rangle$ and $\tilde{a}_k^{(t)} := a_k - \hat{a}_k^{(t)}$. Within this section, we write $T := t_2^{(s)} - t_1^{(s)}$.

Proof overview The first part is proved using the analysis in Appedix B.1. Note that we should view the analysis in this section and the analysis in Appendix B.1 as a whole induction/continuity argument. It's easy to verify that at any time $t_1^{(s)} \leqslant t \leqslant t_2^{(s)}$, Assumption B.1 holds and Proposition 3.1 holds.

The second part is a simple corollary of Lemma B.18 that gives a lower bound for the increasing speed of $\hat{a}_k^{(t)}$.

For the third part, we proceed as follows. At the beginning of phase 2, for any reinitialized component $v^{(t)}$, we know there exists some universal constant $C > 0$ s.t. $[\bar{v}_k^{(t)}]^2 \leqslant C \log d/d$ for all $k \in [d]$. Let T' be the minimum time needed for some $[\bar{v}_k^{(t)}]^2$ to reach $2C \log d/d$. For any $t \leqslant T' + t_1^{(s)}$, we have $[\bar{v}_k^{(t)}]^2 \leqslant 2C \log d/d$ and then we can derive an upper bound on the movement speed of $v^{(t)}$, with which we

show the change of $[\bar{v}_k^{(t)}]^2$ is $o(\log d/d)$ within time T. (Also note this automatically implies that $T' > T$.) To bound the change of the norm, we proceed in a similar way but with T' being the minimum time needed for some $\|v^{(t)}\|$ to reach $2\delta_0$. (Strictly speaking, the actual T' is the smaller one between them.)

Lemma B.33. *If $S_k^{(s,t_1^{(s)})} \neq \varnothing$, then after at most $\frac{4}{a_k}\log\left(\frac{a_k}{2\delta_1^2}\right)$ time, we have $\tilde{a}_k^{(t)} \leqslant \lambda$.*

Proof. Recall that Lemma B.18 says [3]

$$\frac{1}{\hat{a}_k^{(t)}}\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} \geqslant 2\tilde{a}_k^{(t)} - \lambda - O\left(\alpha^2\right).$$

As a result, when $\tilde{a}_k^{(t)} < 2\lambda/3$, we have $\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} \geqslant \tilde{a}_k^{(t)}\hat{a}_k^{(t)}$ or, equivalently, $\frac{\mathrm{d}}{\mathrm{d}t}\tilde{a}_k^{(t)} \leqslant -\tilde{a}_k^{(t)}\hat{a}_k^{(t)}$. When $\hat{a}_k^{(t)} \leqslant a_k/2$, we have $\frac{\mathrm{d}}{\mathrm{d}t}\hat{a}_k^{(t)} \geqslant a_k\hat{a}_k^{(t)}/2$, whence it takes at most $\frac{2}{a_k}\log\left(\frac{a_k}{2\delta_1^2}\right)$ time for $\hat{a}_k^{(t)}$ to grow from δ_1^2 to $a_k/2$. When $\hat{a}_k^{(t)} \geqslant a_k/2$, we have $\frac{\mathrm{d}}{\mathrm{d}t}\tilde{a}_k^{(t)} \leqslant -a_k\tilde{a}_k^{(t)}/2$, whence it takes at most $\frac{2}{a_k}\log\left(\frac{a_k}{2\lambda}\right)$. Hence, the total amount of time is upper bounded by $\frac{2}{a_k}\left(\log\left(\frac{a_k}{2\delta_1^2}\right) + \log\left(\frac{a_k}{2\lambda}\right)\right)$. Finally, use the fact $\lambda > \delta_1^2$ to complete the proof. □

Lemma B.34. *For any $k \in [d]$ and $\bar{v}^{(t)}$ with $\|\bar{v}^{(t)}\|_\infty^2 \leqslant O(\log d/d)$, we have $\mathbb{E}_{k,w}^{(t)}[z^{(t)}]^4 = [\bar{v}_k^{(t)}]^4 \pm O\left(\frac{\log d}{d}\alpha\right)$. Meanwhile, for each $\bar{w}^{(t)} \in S_k^{(t)}$, we have $\left|z^{(t)}\right| \leqslant O\left(\sqrt{\frac{\log d}{d}}\right)$.*

Proof. For simplicity, put $x^{(t)} = \left\langle \bar{w}_{-k}^{(t)}, \bar{v}_{-k}^{(t)} \right\rangle$. Then we have

$$\mathbb{E}_{k,w}^{(t)}[z^{(t)}]^4 = \mathbb{E}_{k,w}^{(t)}\Big\{[\bar{w}_k^{(t)}]^4[\bar{v}_k^{(t)}]^4 + 4[\bar{w}_k^{(t)}]^3[\bar{v}_k^{(t)}]^3x^{(t)} + 6[\bar{w}_k^{(t)}]^2[\bar{v}_k^{(t)}]^2[x^{(t)}]^2 + 4\bar{w}_k^{(t)}\bar{v}_k^{(t)}[x^{(t)}]^3 + [x^{(t)}]^4\Big\}.$$

[3] $\alpha^2 = o(\lambda)$.

For the first term, we have $[\bar{v}_k^{(t)}]^4 \mathbb{E}_{k,w}^{(t)}[\bar{w}_k^{(t)}]^4 = [\bar{v}_k^{(t)}]^4 (1 \pm O(\alpha^2))$. To bound the rest terms, we compute

$$\mathbb{E}_{k,w}^{(t)} \left\{ [\bar{w}_k^{(t)}]^3 [\bar{v}_k^{(t)}]^3 x^{(t)} \right\} \leqslant O(1) \left(\frac{\log d}{d} \right)^{1.5} \mathbb{E}_{k,w}^{(t)} \sqrt{1 - [\bar{w}_k^{(t)}]^2} \leqslant O(1) \left(\frac{\log d}{d} \right)^{1.5} \alpha,$$

$$\mathbb{E}_{k,w}^{(t)} \left\{ [\bar{w}_k^{(t)}]^2 [\bar{v}_k^{(t)}]^2 [x^{(t)}]^2 \right\} \leqslant O(1) \frac{\log d}{d} \alpha^2$$

$$\mathbb{E}_{k,w}^{(t)} \left\{ \bar{v}_k^{(t)} [x^{(t)}]^3 \right\} \leqslant O(1) \sqrt{\frac{\log d}{d}} \alpha^{2.5}$$

$$\mathbb{E}_{k,w}^{(t)} \left\{ [x^{(t)}]^4 \right\} \leqslant O(1) \alpha^3.$$

Use the fact $\alpha \leqslant \log d / d$ and we get

$$\mathbb{E}_{k,w}^{(t)} [z^{(t)}]^4 = [\bar{v}_k^{(t)}]^4 \left(1 \pm O(\alpha^2) \right) \pm O(1) \frac{\log d}{d} \alpha = [\bar{v}_k^{(t)}]^4 \pm O\left(\frac{\log d}{d} \alpha \right).$$

For the individual bound, it suffices to note that

$$\left| z^{(t)} \right| \leqslant \left| \bar{v}_k^{(t)} \right| + \sqrt{1 - [\bar{w}_k^{(t)}]^2} \leqslant O\left(\sqrt{\frac{\log d}{d}} \right) + \sqrt{\alpha} = O\left(\sqrt{\frac{\log d}{d}} \right).$$

□

Lemma B.35 (Bound on the tangent movement)**.** *In Phase 2, for any reinitialized component $v^{(t)}$ and $k \in [d]$, we have $[\bar{v}_k^{(t_2)}]^2 = [\bar{v}_k^{(t_1)}]^2 + o(\log d / d)$.*

Proof. Recall the definition of G_1, G_2 and G_3 from Lemma B.7. By Lemma B.34, we have

$$G_1 \leqslant 8 \tilde{a}_k^{(t)} \left(1 - [\bar{v}_k^{(t)}]^2 \right) [\bar{v}_k^{(t)}]^4 + O(1) a_k \frac{\log d}{d} \alpha + 8 \hat{a}_k^{(t)} \mathbb{E}_{k,w}^{(t)} \left\{ [z^{(t)}]^3 \langle \bar{w}_{-k}, \bar{v}_{-k} \rangle \right\}$$

$$\leqslant 8 \tilde{a}_k^{(t)} \left(1 - [\bar{v}_k^{(t)}]^2 \right) [\bar{v}_k^{(t)}]^4 + O\left(a_k \frac{\log d}{d} \alpha \right),$$

where the second line comes from

$$\mathbb{E}_{k,w}^{(t)} \left\{ [z^{(t)}]^3 \langle \bar{w}_{-k}, \bar{v}_{-k} \rangle \right\} \leqslant O(1) \frac{\log d}{d} \mathbb{E}_{k,w}^{(t)} \sqrt{1 - [\bar{w}_k^{(t)}]^2} \leqslant O\left(\frac{\log d}{d} \alpha \right).$$

Similarly, we have $|G_2| \leqslant O(1) \sum_{i \neq k} a_i \frac{\log d}{d} \alpha$. For G_3, by Lemma B.34, we have

$$a_i [\bar{v}_i^{(t)}]^4 - \hat{a}_i^{(t)} \mathbb{E}_{i,w}^{(t)} \left\{ [z^{(t)}]^4 \right\} = \tilde{a}_i^{(t)} [\bar{v}_i^{(t)}]^4 \pm O\left(a_i \frac{\log d}{d} \alpha \right).$$

Therefore

$$\begin{aligned}
|G_3| &\leqslant 8 [\bar{v}_k^{(t)}]^2 \sum_{i \neq k} \left(\tilde{a}_i^{(t)} [\bar{v}_i^{(t)}]^4 \pm O\left(a_i \frac{\log d}{d} \alpha \right) \right) \\
&\leqslant 8 [\bar{v}_k^{(t)}]^2 \left(\left(\max_{i \neq k} \tilde{a}_i^{(t)} \right) O\left(\frac{\log d}{d} \right) + O\left(\frac{\log d}{d} \alpha \right) \right) \\
&\leqslant O\left(\beta^{(s)} \frac{\log^2 d}{d^2} \right).
\end{aligned}$$

Thus[4],

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} [\bar{v}_k^{(t)}]^2 &\leqslant 8 \tilde{a}_k^{(t)} [\bar{v}_k^{(t)}]^4 + O\left(\frac{\log d}{d} \alpha \right) + O\left(\beta^{(s)} \frac{\log^2 d}{d^2} \right) \\
&\leqslant O\left(\beta^{(s)} \frac{\log^2 d}{d^2} \right).
\end{aligned}$$

Integrate both sides and recall that $T = \frac{o(d/\log d)}{\beta^{(s)}}$. Thus, the change of $[\bar{v}_k^{(t)}]^2$ is $o(\log d/d)$. □

Lemma B.36 (Bound on the norm growth)**.** *In Phase 2, for any reinitialized component $v^{(t)}$ and $k \in [d]$, we have* $\left| \left\| v^{(t_2)} \right\|^2 - \left\| v^{(t_2)} \right\|^2 \right| = o(\delta_0^2)$.

[4] $\alpha \leqslant O(\beta^{(s)} \log d/d)$

Proof. By Lemma B.6 and Lemma B.34, we have

$$\begin{aligned}\frac{1}{2\left\|v^{(t)}\right\|^2}\frac{\mathrm{d}}{\mathrm{d}t}\left\|v^{(t)}\right\|^2 &\leqslant \sum_{i=1}^{d}\left(a_i[\bar{v}_i^{(t)}]^4 - \hat{a}_i^{(t)}\mathbb{E}_{i,w}^{(t)}[z^{(t)}]^4\right)\\ &\leqslant \sum_{i=1}^{d}\left(\tilde{a}_i^{(t)}[\bar{v}_i^{(t)}]^4 + a_i O\left(\frac{\log d}{d}\alpha\right)\right)\\ &\leqslant \left(\max_{i\in[d]}\tilde{a}_i^{(t)}\right)O\left(\frac{\log d}{d}\right) + O\left(\frac{\log d}{d}\alpha\right)\\ &= \left(\max_{i\in[d]}\tilde{a}_i^{(t)}\right)O\left(\frac{\log d}{d}\right).\end{aligned}$$

Recall that $\max_{i\in[d]}\tilde{a}_i^{(t)} \leqslant O(\beta^{(s)})$ and $\|v^{(t)}\| \leqslant O(\delta_0)$. Hence,

$$\frac{\mathrm{d}}{\mathrm{d}t}\left\|v^{(t)}\right\|^2 \leqslant O\left(\beta^{(s)}\frac{\log d}{d}\right)\delta_0^2.$$

Integrate both sides, use the fact $T = \frac{o(d/\log d)}{\beta^{(s)}}$, and then we complete the proof. □

Proof of Lemma 3.4. Lemma 3.4 follows by combining the above lemmas with the analysis in Appendix B.1. □

B.4 Proof for Theorem 3.1

In the section, we give a proof of Theorem 3.1.

Theorem 3.1. *For any $\epsilon \geqslant \exp(-o(d/\log d))$, there exists $\gamma = \Theta(1)$, $m = poly(d)$, $\lambda = \min\{O(\log d/d), O(\epsilon/d^{1/2})\}$, $\alpha = \min\{O(\lambda/d^{3/2}), O(\lambda^2), O(\epsilon^2/d^4)\}$, $\delta_1 = O(\alpha^{3/2}/m^{1/2})$, $\delta_0 = \Theta(\delta_1\alpha/\log^{1/2}(d))$ such that with probability $1 - 1/poly(d)$ in the (re)-initializations, Algorithm 3 terminates in $O(\log(d/\epsilon))$ epochs and returns a tensor T such that*

$$\|T - T^*\|_F \leqslant \epsilon.$$

Note that Proposition 3.1 guarantees any ground truth component with $a_i \geqslant \beta^{(s)}/(1-\gamma)$ must have been fitted before epoch s starts. When $\beta^{(s)}$ decreases below $O(\epsilon/\sqrt{d})$, all the ground truth components larger than $O(\epsilon/\sqrt{d})$ have been fitted and the residual $\|T - T^*\|_F$ must be less than ϵ. Since $\beta^{(s)}$ decreases in a constant rate, the algorithm must terminate in $O(\log(d/\epsilon))$ epochs.

Proof. According to Lemma 3.2 and Lemma 3.4, we know Proposition 3.1 holds through the algorithm. We first show that $\beta^{(s)}$ is always lower bounded by $\Omega(\epsilon/\sqrt{d})$ before the algorithm ends. For the sake of contradiction, assume $\beta^{(s)} \leqslant O(\frac{\epsilon}{\sqrt{d}})$. We show that $\left\|T^{(s,0)} - T^*\right\|_F < \epsilon$, which is a contradiction because our algorithm should have terminated before this epoch. For simplicity, we drop the superscript on epoch s in the proof.

We can upper bound $\left\|T^* - T^{(t)}\right\|_F$ by splitting T^* into $\sum_{i\in[d]} T_i^*$ and splitting $T^{(t)}$ into $\sum_{i\in[d]} T_i^{(t)} + T_\varnothing^{(t)}$. Then, we have

$$\begin{aligned}\left\|T^* - T^{(t)}\right\|_F \leqslant& \left\|\sum_{i\in d}(a_i - \hat{a}_i^{(t)})e_i^{\otimes 4}\right\|_F + \sum_{i\in[d]}\left\|T_i^{(t)} - \hat{a}_i^{(t)}e_i^{\otimes 4}\right\|_F + \left\|T_\varnothing^{(t)}\right\|_F \\ \leqslant& O\left(\sqrt{d}\max\left(\beta^{(s)}, \lambda\right)\right) + O(\alpha + m\delta_1^2),\end{aligned}$$

where the second inequality holds because $(a_i - \hat{a}_i^{(t)}) \leqslant O(\max\left(\beta^{(s)}, \lambda\right))$, $\left\|T_i^{(t)} - \hat{a}_i^{(t)}e_i^{\otimes 4}\right\|_F \leqslant O(\hat{a}_i^{(t)}\alpha)$ and $\left\|T_\varnothing^{(t)}\right\|_F \leqslant m\delta_1^2$. Choosing $\lambda, \alpha = O(\frac{\epsilon}{\sqrt{d}})$ and $\delta_1^2 = O(\frac{\epsilon}{m\sqrt{d}})$, we have

$$\left\|T^* - T^{(t)}\right\|_F < \epsilon.$$

Since $\beta^{(s)}$ starts from $O(1)$ and decreases by a constant factor at each epoch, it will decrease below $O(\frac{\epsilon}{\sqrt{d}})$ after $O(\log(d/\epsilon))$ epochs. This means our algorithm terminates in $O(\log(d/\epsilon))$ epochs. □

B.5 Experiments

In Section B.5.1, we give detailed settings for our experiments in Figure 3.1. Then, we give additional experiments on non-orthogonal tensors in Section B.5.2.

B.5.1 Experiment settings for orthogonal tensor decomposition

We chose the ground truth tensor T^* as $\sum_{i\in[5]} a_i e_i^{\otimes 4}$ with $e_i \in \mathbb{R}^{10}$ and $a_i/a_{i+1} = 1.2$. We normalized T^* so its Frobenius norm equals 1.

Our model T was over-parameterized to have 50 components. Each component $W[:, i]$ was randomly initialized from $\delta_0 \mathrm{Unif}(\mathbb{S}^{d-1})$ with $\delta_0 = 10^{-15}$.

The objective function is $\frac{1}{2}\|T - T^*\|_F^2$. We ran gradient descent with step size 0.1 for 2000 steps. We repeated the experiment from 5 different experiments and plotted the results in Figure 3.1. Our experiments was ran on a normal laptop and took a few minutes.

B.5.2 Additional results on non-orthogonal tensor decomposition

In this subsection, we give some empirical observations that suggests non-orthogonal tensor decomposition may not follow the greedy low-rank learning procedure in Li et al. (2020b).

Ground truth tensor T^*: The ground truth tensor is a $10 \times 10 \times 10 \times 10$ tensor with rank 5. It's a symmetric and non-orthogonal tensor with $\|T^*\|_F = 1$. The specific ground truth tensor we used is in the code.

Greedy low-rank learning (GLRL): We first generate the trajectory of the greedy low-rank learning. In our setting, GLRL consists of 5 epochs. At initialization, the model has no component. At each epoch, the algorithm first adds a small component (with norm 10^{-60}) that maximizes the correlation with the current residual to the model,

then runs gradient descent until convergence.

To find the component that has best correlation with residual R, we ran gradient descent on $R(w^{\otimes 4})$ and normalize w after each iteration. In other words, we ran projected gradient descent to solve $\min_{w|\|w\|=1} R(w^{\otimes 4})$. We repeated this process from 50 different initializations and chose the best component among them.

In the experiment, we chose the step size as 0.3. And at the s-th epoch, we ran $s \times 2000$ iterations to find the best rank-one approximation and also ran $s \times 2000$ iterations on our model after we included the new component. After each epoch, we saved the current tensor as a saddle point. We also included the zero tensor as a saddle point so there are 6 saddles in total.

Figure B.1 shows that the loss decreases sharply in each epoch and eventually converges to zero.

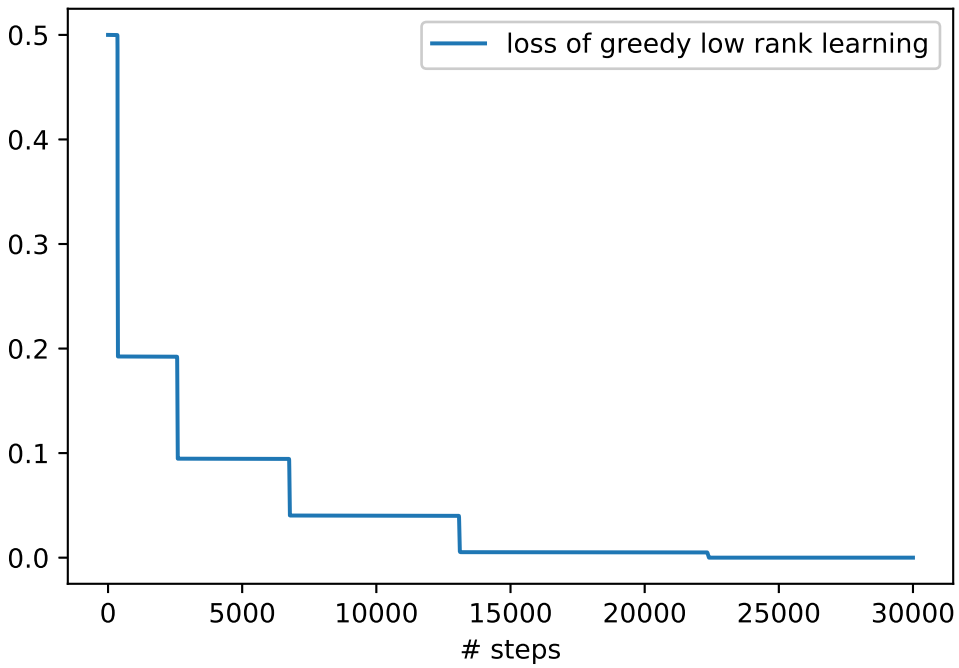

FIGURE B.1: Loss trajectory of greedy low-rank learning.

Over-parameterized gradient descent: If the over-parameterized gradient descent follows the greedy low-rank learning procedure, one should expect that the model passes the same saddles when the tensor rank increases. To verify this, we ran experiments with gradient descent and computed the distance to the closest GLRL saddles at each iteration.

Our model has 50 components and each component is initialized from $\delta_0 \text{Unif}(\mathbb{S}^{d-1})$ with $\delta_0 = 10^{-60}$. We ran gradient descent with step size 0.3 for 1000 iterations.

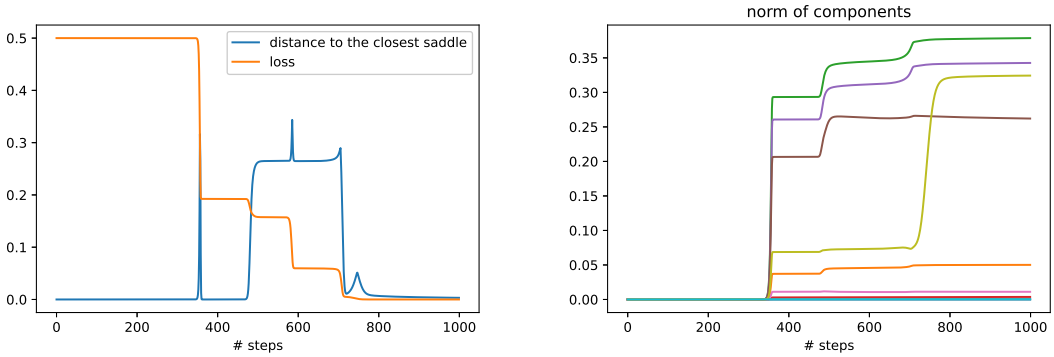

FIGURE B.2: Non-orthogonal tensor decomposition with number of components $m = 50$ and initialization scale $\delta_0 = 10^{-60}$. The left figure shows the loss trajectory and the distance to the closest GLRL saddles; the right figures shows the norm trajectory of different components.

Figure B.2 (left) shows that after fitting the first direction, over-parameterized gradient descent then has a very different trajectory from GLRL. After roughly 450 iterations, the loss continues decreasing but the distance to the closest saddle is high. After 800 iterations, gradient descent converges and the distance to the closest saddle (which is T^*) becomes low.

In Figure B.2 (right), we plotted the norm trajoeries for 10 of the components. The figure shows that some of the already large components become even larger at roughly 450 iterations, which corresponds to the second drop of the loss. We picked two of these components and found that their correlation $\langle \bar{w}, \bar{v} \rangle$ drops from 1

at the 400-th iteration to 0.48 at the 550-th iteration. This suggests that two large component in the same direction can actually split into two directions in the training.

One might suspect that this phenomenon would disappear if we use more aggressive over-parameterization and even smaller initialization. We then let our model have 1000 components and let the initialization size to be 10^{-100} and re-did the experiments. We observed almost the same behavior as before. Figure B.3 (left) shows the same pattern for the distance to closest GLRL saddles as in Figure B.2. In Figure B.3 (right), we randomly chose 10 of the 1000 components and plotted their norm change, and we again observe that one large component becomes even larger at roughly iteration 700 that corresponds to the second drop of the loss function.

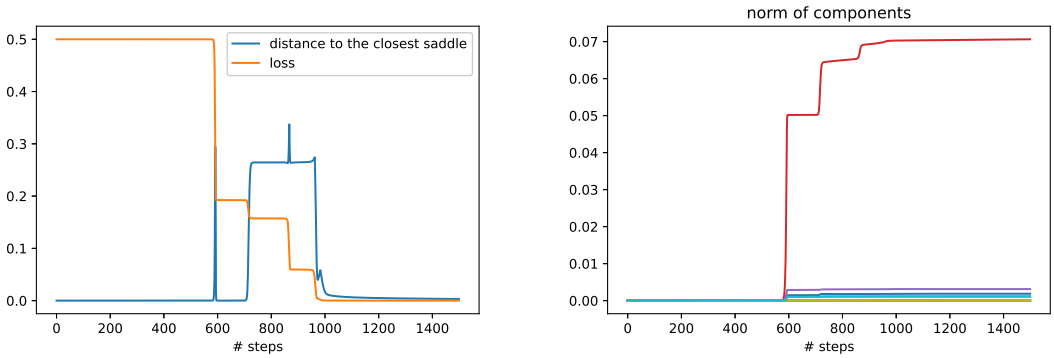

FIGURE B.3: Non-orthogonal tensor decomposition with number of components $m = 1000$ and initialization scale $\delta_0 = 10^{-100}$. The left figure shows the loss trajectory and the distance to the closest GLRL saddles; the right figures shows the norm trajectory of different components.

Appendix C

Supplementary Materials for Chapter 4

C.1 Detailed Experiment Setting

STL-10, CIFAR-10, CIFAR-100 : We use ResNet-18 (He et al., 2016) as the backbone network, a two-layer nonlinear MLP (with batch normalization, ReLU activation, hidden layer width 512, output width 128) as the projector, and a linear predictor. Unless specified otherwise, SGD is used as the optimizer with momentum 0.9, weight decay $\eta = 0.0004$ and batch size 128. The EMA parameter for the target network is set as 0.996 and the EMA parameter μ of the correlation matrix $\hat{F}$ is set as 0.5. Our code is adapted from Tian et al. (2021) [1], and we follow the same data augmentation process.

To evaluate the quality of the pre-trained representations, we follow the linear evaluation protocol. Each setting is repeated 5 times to compute the mean and standard deviation. The accuracy is reported as "mean±std". Unless explicitly specified, we use learning rate $\gamma = 0.01$, regularization $\epsilon = 0.2$ on STL-10; $\gamma = 0.02, \epsilon = 0.3$ on CIFAR-10 and $\gamma = 0.03, \epsilon = 0.3$ on CIFAR-100.

[1] Their open source code is at https://github.com/facebookresearch/luckmatters/tree/main/ssl

ImageNet : Following BYOL (Grill et al., 2020), we use ResNet-50 as the backbone and a two-layer MLP (with batch normalization, ReLU, hidden layer width 4096, output width 256) as the projector. We use LARS (You et al., 2017) optimizer and trains the model for 100 epochs, with a batch size 4096. The learning rate is 7.2, which is linearly scaled from the base learning rate 0.45 at batch size 256. Other setups such as weight decay ($\eta = 1e^{-6}$), target EMA (scheduled from 0.99 to 1), augmentation recipe (color jitters, blur, etc.), and linear evaluation protocol are the same as BYOL.

C.2 Proofs of Single-layer Linear Networks

C.2.1 Gradient Flow on Population Loss

In this section, we give the proof of Theorem 4.1, which shows that DirectSet(α) running on the population loss with infinitesimal learning rate and η weight decay can learn the projection matrix onto subspace S.

Theorem 4.1. *Suppose network architecture and data distribution follow Assumption 4.1 and Assumption 4.2, respectively. Suppose we initialize online network W as δI, and run DirectSet(α) on population loss (see Eqn. 4.1) with infinitesimal step size and η weight decay. If $\eta \in \left(\frac{1}{4(1+\sigma^2)}, \frac{1}{4}\right)$ and $\delta > \left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$, then W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$ when time goes to infinity.*

As we already mentioned in the main text, Theorem 4.1 is proved by analyzing each eigenvalue of W separately. We show that the eigenvalues in the B subspace converge to zero, and the eigenvalues in the S subspace converge to the same positive number, which immediately implies that W converges to a scaling of the projection matrix P_S.

Proof of Theorem 4.1. We can compute the gradient in terms of W as follows,

$$\begin{aligned}\nabla L(W) =&\mathbb{E}_{x_1,x_2} W_p^\top \left(W_p W x_1 - W_a x_2\right) x_1^\top \\ =&W_p^\top \left(W_p W \mathbb{E}_{x_1} x_1 x_1^\top - W_a \mathbb{E}_{x_1,x_2} x_2 x_1^\top\right).\end{aligned}$$

Note that the two augmented views x_1, x_2 are sampled by first sampling input x from $\mathbb{N}(0, I_d)$, and then independently sampling x_1, x_2 from $\mathbb{N}(x, \sigma^2 P_B)$. Therefore, we know $\mathbb{E}_{x_1} x_1 x_1^\top = I + \sigma^2 P_B$ and $\mathbb{E}_{x_1,x_2} x_2 x_1^\top = I$. Recall that we run gradient flow on W with weight decay η, so the dynamics on W is as follows:

$$\dot{W} = W_p^\top(-W_p W(I + \sigma^2 P_B) + W_a) - \eta W,$$

where the first term comes from the gradient and the second term is due to weight decay.

Since W is initialized as δI, and $W_a = W, W_p = (WW^\top)^\alpha$, so we know initially W, W_p, W_a, I and P_B are all simultaneously diagonalizable, which then implies $\dot{W}$ is simultaneously diagonalizable with W. This argument can continue to show that at any time point, W, W_p, W_a, I and P_B are all simultaneously diagonalizable. Since W is always a real symmetric matrix, we have $W_p = (WW^\top)^\alpha = |W|^{2\alpha}$. The dynamics on W can then be written as

$$\begin{aligned}\dot{W} =& |W|^{2\alpha} \left(-|W|^{2\alpha} W(I + \sigma^2 P_B) + W\right) - \eta W \\ =&W\left(-(I + \sigma^2 P_B)|W|^{4\alpha} + |W|^{2\alpha} - \eta\right).\end{aligned}$$

Let the eigenvalue decomposition of W be $\sum_{i=1}^d \lambda_i u_i u_i^\top$, with span$(\{u_{d-r+1}, \cdots, u_d\})$ equals to subspace B. We can separately analyze the dynamics of each λ_i. Furthermore, we know $\lambda_1, \cdots, \lambda_r$ have the same value λ_S and $\lambda_{d-r+1}, \cdots, \lambda_d$ have the same value λ_B. Next, we separately show that λ_B converge to zero and λ_S converges to a positive value.

Dynamics for λ_B: We can write down the dynamics for λ_B as follows:

$$\dot{\lambda}_B = \lambda_B\left[-(1 + \sigma^2)|\lambda_B|^{4\alpha} + |\lambda_B|^{2\alpha} - \eta\right]$$

Similar as the analysis in Tian et al. (2021), when $\eta > \frac{1}{4(1+\sigma^2)}$, we know $\dot{\lambda}_B < 0$ for any $\lambda_B > 0$ and $\lambda_B = 0$ is a critical point. This means, as long as $\eta > \frac{1}{4(1+\sigma^2)}$, λ_B must converge to zero.

Dynamics for λ_S: We can write down the dynamics for λ_S as follows:

$$\dot{\lambda}_S = \lambda_S \left[- |\lambda_S|^{4\alpha} + |\lambda_S|^{2\alpha} - \eta \right].$$

When $0 < \eta < \frac{1}{4}$, we know $\dot{\lambda}_S > 0$ for $\lambda_S^{2\alpha} \in \left(\frac{1-\sqrt{1-4\eta}}{2}, \frac{1+\sqrt{1-4\eta}}{2} \right)$ and $\dot{\lambda}_S < 0$ for $\lambda_S^{2\alpha} \in \left(\frac{1+\sqrt{1-4\eta}}{2}, \infty \right)$. Furthermore, we know $\dot{\lambda}_S = 0$ when $\lambda_S^{2\alpha} = \frac{1+\sqrt{1-4\eta}}{2}$. Therefore, as long as $0 < \eta < \frac{1}{4}$ and initialization $\delta^{2\alpha} > \frac{1-\sqrt{1-4\eta}}{2}$, we know $\lambda_S^{2\alpha}$ converges to $\frac{1+\sqrt{1-4\eta}}{2}$.

Overall, we know when $\frac{1}{4(1+\sigma^2)} < \eta < \frac{1}{4}$ and $\delta > \left(\frac{1-\sqrt{1-4\eta}}{2} \right)^{1/(2\alpha)}$, we have λ_B converge to zero and λ_S converge to $\left(\frac{1+\sqrt{1-4\eta}}{2} \right)^{1/(2\alpha)}$. That is, matrix W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2} \right)^{1/(2\alpha)} P_S$. □

C.2.2 Gradient Descent on Empirical Loss

In this section, we prove that DirectCopy successfully learns the projection matrix given polynomial number of samples.

Theorem 4.2. *Suppose network architecture and data distribution are as defined in Assumption 4.1 and Assumption 4.2, respectively. Suppose we initialize online network as δI, and run DirectCopy on empirical loss (see Eqn. 4.2) with γ step size and η weight decay. Assume $\sigma^2 = \Theta(1), \eta \in \left(\frac{1+\sigma^2/4}{4(1+\sigma^2)}, \frac{1+3\sigma^2/4}{4(1+\sigma^2)} \right), \delta \in (1/2, O(1))$ and $\gamma = \Theta(1)$. For any accuracy $\hat{\epsilon} > 0$, given $n \geqslant poly(d, 1/\hat{\epsilon})$ number of samples, with*

probability at least 0.99 *there exists* $t = O(\log(1/\hat{\epsilon}))$ *such that*

$$\left\| \widetilde{W}_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| \leqslant \hat{\epsilon},$$

where $\widetilde{W}_t$ *is the online network weights at the* t*-th step.*

When running gradient descent on the empirical loss, the eigenspace of $\widetilde{W}_t$ can shift and become no longer simultaneously diagonalizable with P_B. So we cannot independently analyze each eigenvalue of $\widetilde{W}_t$ as before, which brings significant challenge into the analysis. Instead of directly analyzing the dynamics of $\widetilde{W}_t$, we first show that the gradient descent iterates W_t on the population loss converges to P_S in linear rate, and then show that $\widetilde{W}_t$ stays close to W_t within certain iterations.

Lemma C.1. *In the setting of Theorem 4.2, let* W_t *be the gradient descent iterations on the population loss* L*. Given any accuracy* $\hat{\epsilon} > 0$*, for any* $t \geqslant C\log(1/\hat{\epsilon})$*, we have*

$$\left\| W_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| \leqslant \hat{\epsilon},$$

where C *is a positive constant.*

The proof of Lemma C.1 is similar as the gradient flow analysis in Section 4.3.2. Next, we show that the gradient descent trajectory on the empirical loss stays close to the gradient descent trajectory on the population loss within $O(\log(1/\hat{\epsilon}))$ iterations.

Lemma C.2. *In the setting of Theorem 4.2, let* W_t *be the gradient descent iterations on the population loss and let* $\widetilde{W}_t$ *be the gradient descent iterations on the empirical loss. For any accuracy* $\hat{\epsilon} > 0$*, given* $n \geqslant \mathrm{poly}(d, 1/\hat{\epsilon})$ *number of samples, with probability at least* 0.99*, for any* $t \leqslant C\log(1/\hat{\epsilon})$*, we have*

$$\left\| \widetilde{W}_t - W_t \right\| \leqslant \hat{\epsilon},$$

where the constant C *comes from Lemma C.1.*

Then the proof of Theorem 4.2 directly follows from Lemma C.1 and Lemma C.2.

Proof of Theorem 4.2. According to Lemma C.1, we know given any accuracy $\hat{\epsilon}'$, for $t = C\log(1/\hat{\epsilon})$, we have

$$\left\| W_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| \leqslant \hat{\epsilon}',$$

where C is a positive constant.

According to Lemma C.2, we know given $n \geqslant \mathrm{poly}(d, 1/\hat{\epsilon}')$ number of samples, with probability at least 0.99,

$$\left\| \widetilde{W}_t - W_t \right\| \leqslant \hat{\epsilon}'.$$

Therefore, we have

$$\left\| \widetilde{W}_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| \leqslant \left\| W_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}} P_S \right\| + \left\| \widetilde{W}_t - W_t \right\| \leqslant 2\hat{\epsilon}'.$$

Replacing $\hat{\epsilon}'$ by $\hat{\epsilon}/2$ finishes the proof. □

In section C.2.2, we give the proof of Lemma C.1 and Lemma C.2. Proofs of some technical lemmas are left in Appendix C.2.5.

Proofs for Lemma C.1 and Lemma C.2

Proof of Lemma C.1. Similar as in Theorem 4.1, we can show that at any step t, W_t is simultaneously diagonalizable with $W_{a,t}, W_{p,t}, I$ and P_B. The update on W_t is as follows,

$$W_{t+1} = W_t + \gamma W_t \left(-(I + \sigma^2 P_B) W_t^4 + W_t^2 - \eta \right).$$

Let the eigenvalue decomposition of W_t be $\sum_{i=1}^{d} \lambda_{i,t} u_i u_i^\top$, with $\mathrm{span}(\{u_{d-r+1}, \cdots, u_d\})$ equals to subspace B. We can separately analyze the dynamics of each $\lambda_{i,t}$. Furthermore, we know $\lambda_{1,t}, \cdots, \lambda_{r,t}$ have the same value $\lambda_{S,t}$

and $\lambda_{d-r+1,t}, \cdots, \lambda_{d,t}$ have the same value $\lambda_{B,t}$. Next, we separately show that $\lambda_{B,t}$ converge to zero and $\lambda_{S,t}$ converges to a positive value in linear rate.

Dynamics of $\lambda_{B,t}$: We show that

$$0 \leqslant \lambda_{B,t} \leqslant (1 - \gamma C_1)^t \delta$$

for any step size $\gamma \leqslant C_2$, where C_1, C_2 are two positive constants.

According to the gradient update, we have

$$\lambda_{B,t+1} = \lambda_{B,t} + \gamma \lambda_{B,t} \left[-(1+\sigma^2)\lambda_{B,t}^4 + \lambda_{B,t}^2 - \eta \right].$$

We only need to prove that for any $\lambda_{B,t} \in [0, \delta]$, we have

$$-(1+\sigma^2)\lambda_{B,t}^4 + \lambda_{B,t}^2 - \eta = -\Theta(1).$$

This is true since $\eta \in \left(\frac{1+\sigma^2/4}{4(1+\sigma^2)}, \frac{1+3\sigma^2/4}{4(1+\sigma^2)} \right)$ and σ^2, δ are two positive constants.

Dynamics of λ_S: We show that

$$0 \leqslant \left| \lambda_{S,t}^2 - \frac{1+\sqrt{1-4\eta}}{2} \right| \leqslant (1 - \gamma C_3)^t \left| \delta^2 - \frac{1+\sqrt{1-4\eta}}{2} \right|$$

for any step size $\gamma \leqslant C_4$, where C_3, C_4 are two positive constants.

There are two cases to consider: when the initialization scale $\delta^2 \in [1/2, \frac{1+\sqrt{1-4\eta}}{2}]$, we prove

$$0 \leqslant \frac{1+\sqrt{1-4\eta}}{2} - \lambda_{B,t}^2 \leqslant (1 - \gamma C_3)^t \left(\frac{1+\sqrt{1-4\eta}}{2} - \delta^2 \right);$$

when the initialization scale $\delta^2 > \frac{1+\sqrt{1-4\eta}}{2}$, we prove

$$0 \leqslant \lambda_{B,t}^2 - \frac{1+\sqrt{1-4\eta}}{2} \leqslant (1 - \gamma C_3)^t \left(\delta^2 - \frac{1+\sqrt{1-4\eta}}{2} \right).$$

We focus on the second case; the proof for the first case is similar.

According to the gradient update, we have

$$\begin{aligned}\lambda_{S,t+1} =&\lambda_{S,t} + \gamma\lambda_{S,t}\left[-\lambda_{S,t}^4 + \lambda_{S,t}^2 - \eta\right]\\ =&\lambda_{S,t} - \gamma\lambda_{S,t}\left(\lambda_{S,t}^2 - \frac{1-\sqrt{1-4\eta}}{2}\right)\left(\lambda_{S,t}^2 - \frac{1+\sqrt{1-4\eta}}{2}\right)\end{aligned}$$

We only need to show that $\lambda_{S,t}\left(\lambda_{S,t}^2 - \frac{1-\sqrt{1-4\eta}}{2}\right) = \Theta(1)$ for any $\lambda_{S,t}^2 \in [\frac{1+\sqrt{1-4\eta}}{2}, \delta]$. This is true because $\eta \in \left(\frac{1+\sigma^2/4}{4(1+\sigma^2)}, \frac{1+3\sigma^2/4}{4(1+\sigma^2)}\right)$ and σ^2, δ are two positive constants.

Overall, we know that there exists constant step size such that after $t = O(\log(1/\hat{\epsilon}))$ steps, we have

$$0 \leqslant \lambda_{B,t} \leqslant \hat{\epsilon} \text{ and } \left|\lambda_{S,t} - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}}\right| \leqslant \hat{\epsilon}.$$

This then implies,

$$\left\|W_t - \sqrt{\frac{1+\sqrt{1-4\eta}}{2}}P_S\right\| \leqslant \hat{\epsilon}.$$

□

Proof of Lemma C.2. We know the update on $\widetilde{W}_t$ is

$$\widetilde{W}_{t+1} - \widetilde{W}_t = \gamma\widetilde{W}_{p,t}^\top\left(-\widetilde{W}_{p,t}\widetilde{W}_t\left(\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_1^{(i)}]^\top\right) + \widetilde{W}_{a,t}\left(\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_2^{(i)}]^\top\right)\right) - \gamma\eta\widetilde{W}_t,$$

and the update on W_t is

$$W_{t+1} - W_t = \gamma W_{p,t}^\top\left(-W_{p,t}W_t\left(I + \sigma^2 P_B\right) + W_{a,t}\right) - \gamma\eta W_t.$$

Next, we bound $\left\|\widetilde{W}_{t+1} - \widetilde{W}_t - (W_{t+1} - W_t)\right\|$. According to Lemma C.3, we know with probability at least $1 - O(d^2)\exp\left(-\Omega(\hat{\epsilon}'^2 n/d^2)\right)$,

$$\left\|\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_1^{(i)}]^\top - I - \sigma^2 P_B\right\|, \left\|\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_2^{(i)}]^\top - I\right\|, \left\|\frac{1}{n}\sum_{i=1}^n x^{(i)}[x^{(i)}]^\top - I\right\| \leqslant \hat{\epsilon}'.$$

Recall that we set $\widetilde{W}_{a,t} = \widetilde{W}_t$ and set $W_{a,t}$ as W_t, so we have $\left\|\widetilde{W}_{a,t} - W_{a,t}\right\| = \left\|\widetilde{W}_t - W_t\right\|$. Also since we set $\widetilde{W}_{p,t} = \widetilde{W}_t \left(\frac{1}{n}\sum_{i=1}^n x^{(i)}[x^{(i)}]^\top\right)\widetilde{W}_t^\top$ and set $W_{p,t} = W_t W_t^\top$, we have $\left\|\widetilde{W}_{p,t} - W_{p,t}\right\| = O\left(\left\|\widetilde{W}_t - W_t\right\| + \hat{\epsilon}'\right)$ since $\|W_t\| = O(1)$.

Combing the above bounds and recall γ is a constant, we have

$$\left\|\widetilde{W}_{t+1} - \widetilde{W}_t - (W_{t+1} - W_t)\right\| = O\left(\left\|\widetilde{W}_t - W_t\right\| + \hat{\epsilon}'\right).$$

Therefore,

$$\left\|\widetilde{W}_t - W_t\right\| \leqslant C_1^t \hat{\epsilon}',$$

where C_1 is a constant larger than 1. So for any $t \leqslant C\log(1/\hat{\epsilon})$, we have

$$\left\|\widetilde{W}_t - W_t\right\| \leqslant C_1^{C\log(1/\hat{\epsilon})}\hat{\epsilon}' \leqslant (1/\hat{\epsilon})^{C_2}\hat{\epsilon}',$$

for some positive constant C_2. Choosing $\hat{\epsilon}' = \hat{\epsilon}^{C_2+1}$, we know as long as $n \geqslant \mathrm{poly}(d, 1/\hat{\epsilon})$, with probability at least 0.99, for any $t \leqslant C\log(1/\hat{\epsilon})$, we have

$$\left\|\widetilde{W}_t - W_t\right\| \leqslant \hat{\epsilon}.$$

□

C.2.3 Sample Complexity on Down-stream Tasks

In this section, we give a proof for Theorem 4.3, which shows that the learned representations can indeed reduce sample complexity in downstream tasks.

Theorem 4.3. *Suppose the downstream data distribution is as defined in Assumption 4.4. Suppose* $\left\|\hat{P} - P\right\|_F \leqslant \hat{\epsilon}$ *with* $\hat{\epsilon} < 1$. *Choose the regularizer coefficient* $\rho = \hat{\epsilon}^{1/3}$. *For any* $\zeta < 1/2$, *given* $n \geqslant O(r + \log(1/\zeta))$ *number of samples, with probability at least* $1 - \zeta$, *the training loss minimizer* $\hat{w}$ *satisfies*

$$\left\|\hat{P}\hat{w} - w^*\right\| \leqslant O\left(\hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right).$$

Suppose $\{(z^{(i)}, y^{(i)})\}_{i=1}^n$ are n training samples in the downstream task, let $Z \in \mathbb{R}^{n\times d}$ be the data matrix with its i-th row equal to $z^{(i)}$. Denote $y \in \mathbb{R}^n$ as the label vector with its i-th entry as $y^{(i)}$. Each input $z^{(i)}$ is transformed by a matrix $\hat{P} \in \mathbb{R}^{d\times d}$ to get its representation $\hat{P}z^{(i)}$. The regularized loss can be written as

$$L(w) := \frac{1}{2n}\left\|Z\hat{P}w - y\right\|^2 + \frac{\rho}{2}\|w\|^2.$$

This is the ridge regression problem on inputs $\{(\hat{P}z^{(i)}, y^{(i)})\}_{i=1}^n$, and the unique global minimizer $\hat{w}$ has the following close form:

$$\hat{w} = \left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1}\frac{1}{n}\hat{P}^\top Z^\top y \tag{C.1}$$

With the above closed form of $\hat{w}$, the proof of Theorem 4.3 follows by bounding the difference between $\hat{P}\hat{w}$ and w^* by matrix concentration inequalities and matrix perturbation bounds. Some proofs of technical lemmas are left in Appendix C.2.5.

Proof of Theorem 4.3. Denoting $\hat{P}$ as $P + \Delta$, we know $\|\Delta\|_F \leqslant \hat{\epsilon}$ by assumption. We can also write y as $Zw^* + \xi$ where $\xi \in \mathbb{R}^n$ is the noise vector with its i-th entry equal to $\xi^{(i)}$. Then, we can divide $\hat{w}$ into two terms,

$$\begin{aligned}
\hat{w} &= \left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1}\frac{1}{n}\hat{P}^\top Z^\top y \\
&= \left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1}\frac{1}{n}P^\top Z^\top (Zw^* + \xi) \\
&\quad + \left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1}\frac{1}{n}\Delta^\top Z^\top (Zw^* + \xi)
\end{aligned}$$

Let's first give an upper bound for the second term that comes from the error term $\Delta^\top$.

Upper bounding $\left\|\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1}\frac{1}{n}\Delta^\top Z^\top (Zw^* + \xi)\right\|$ We first bound the norm of $\frac{1}{n}\Delta^\top Z^\top Zw^*$. According to Lemma C.5, we know with probability at least $1 -$

$\exp(-\Omega(n))$, $\left\|\frac{1}{\sqrt{n}}\Delta^\top Z^\top\right\|_F \leqslant O(\hat{\epsilon})$. Since Zw^* is a standard Gaussian vector with dimension n, according to Lemma C.8, with probability at least $1-\exp(-\Omega(n))$, $\left\|\frac{1}{\sqrt{n}}Zw^*\right\| \leqslant O(1)$. Therefore, we have $\left\|\frac{1}{n}\Delta^\top Z^\top Zw^*\right\| \leqslant O(\hat{\epsilon})$.

Then we bound the norm of $\frac{1}{n}\Delta^\top Z^\top\xi$. According to Lemma C.8, we know with probability at least $1-\exp(-\Omega(n))$, $\left\|\frac{1}{\sqrt{n}}\xi\right\| \leqslant O(\beta)$. According to Lemma C.6, we know with probability at least $1-\zeta/3$, $\left\|\Delta^\top Z^\top\bar{\xi}\right\| \leqslant O\left(\hat{\epsilon}\sqrt{\log(1/\zeta)}\right)$. Therefore, we have $\left\|\frac{1}{n}\Delta^\top Z^\top\xi\right\| \leqslant O\left(\frac{\beta\hat{\epsilon}\sqrt{\log(1/\zeta)}}{\sqrt{n}}\right)$.

Since $\lambda_{\min}\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P}+\rho I\right) \geqslant \rho$, we have $\left\|\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P}+\rho I\right)^{-1}\right\| \leqslant \frac{1}{\rho}$. Combining with above bound on $\left\|\frac{1}{n}\Delta^\top Z^\top\left(Zw^*+\xi\right)\right\|$, we know with probability at least $1-\exp\left(-\Omega(n)\right)-\zeta/3$,

$$\left\|\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P}+\rho I\right)^{-1}\frac{1}{n}\Delta^\top Z^\top\left(Zw^*+\xi\right)\right\| \leqslant O\left(\frac{\hat{\epsilon}}{\rho}+\frac{\beta\hat{\epsilon}\sqrt{\log(1/\zeta)}}{\rho\sqrt{n}}\right).$$

Analyzing $\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P}+\rho I\right)^{-1}\frac{1}{n}P^\top Z^\top\left(Zw^*+\xi\right)$ We can write $\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P}$ as $\frac{1}{n}P^\top Z^\top ZP+E$, where

$$E=\frac{1}{n}\Delta^\top Z^\top ZP+\frac{1}{n}P^\top Z^\top Z\Delta+\frac{1}{n}\Delta^\top Z^\top Z\Delta.$$

Let's first bound the spectral norm of ZP. Since P is a projection matrix on an r-dimensional subspace S, we can write P as $UU^\top$, where $U\in\mathbb{R}^{d\times r}$ has columns as an orthonormal basis of subspace S. According to Lemma C.4, we know with probability at least $1-\exp\left(-\Omega(n)\right)$,

$$\Omega(1) \leqslant \sigma_{\min}\left(\frac{1}{\sqrt{n}}ZU\right) \leqslant \sigma_{\max}\left(\frac{1}{\sqrt{n}}ZU\right) \leqslant O(1).$$

Since $\|U\| \leqslant 1$, we have $\left\|\frac{1}{\sqrt{n}} ZP\right\| = \left\|\frac{1}{\sqrt{n}} ZUU^\top\right\| \leqslant O(1)$.

According to Lemma C.5, we know with probability at least $1 - \exp(-\Omega(n))$,

$$\left\|\frac{1}{\sqrt{n}} Z\Delta\right\|_F \leqslant O(\hat{\epsilon}).$$

So overall, we know $\|E\| \leqslant \|E\|_F \leqslant O(\hat{\epsilon})$.

Then, we can write

$$\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1} = \left(\frac{1}{n}P^\top Z^\top ZP + \rho I\right)^{-1} + F.$$

According to the perturbation bound for matrix inverse (Lemma C.11), we have $\|F\| \leqslant O(\frac{\hat{\epsilon}}{\rho^2})$. Then, we have

$$\begin{aligned}
&\left(\frac{1}{n}\hat{P}^\top Z^\top Z\hat{P} + \rho I\right)^{-1} \frac{1}{n}P^\top Z^\top (Zw^* + \xi) \\
=&\left(\frac{1}{n}P^\top Z^\top ZP + \rho I\right)^{-1} \frac{1}{n}P^\top Z^\top Zw^* \\
&+ F\frac{1}{n}P^\top Z^\top Zw^* \\
&+ \left(\left(\frac{1}{n}P^\top Z^\top ZP + \rho I\right)^{-1} + F\right)\frac{1}{n}P^\top Z^\top \xi
\end{aligned}$$

We first show that the first term is close to w^*. Let the eigenvalue decomposition of $\frac{1}{n}P^\top Z^\top ZP$ be $V\Sigma V^\top$, where V's columns are an orthonormal basis for subspace S. Here $\Sigma \in \mathbb{R}^{r\times r}$ is the diagonal matrix that contains all the eigenvalues of $\frac{1}{n}P^\top Z^\top ZP$. According to Lemma C.4, we know that with probability at least $1 - \exp(-\Omega(n))$, all the non-zero eigenvalues of $\frac{1}{n}P^\top Z^\top ZP$ are $\Theta(1)$.

Then, it's not hard to show that

$$\left\|\left(\frac{1}{n}P^\top Z^\top ZP + \rho I\right)^{-1} \frac{1}{n}P^\top Z^\top ZP - P\right\| \leqslant O(\rho).$$

This immediately implies that

$$\left\| \left(\frac{1}{n} P^\top Z^\top Z P + \rho I \right)^{-1} \frac{1}{n} P^\top Z^\top Z w^* - w^* \right\| \leqslant O(\rho)$$

Next, we bound the norm of the second term $F \frac{1}{n} P^\top Z^\top Z w^*$. Similar as before, we know with probability at least $1 - \exp(-\Omega(n))$, $\left\| \frac{1}{\sqrt{n}} Z w^* \right\| \leqslant O(1)$ and $\left\| \frac{1}{\sqrt{n}} P^\top Z^\top \right\| \leqslant O(1)$. Therefore, we have

$$\left\| F \frac{1}{n} P^\top Z^\top Z w^* \right\| \leqslant \|F\| \left\| \frac{1}{\sqrt{n}} P^\top Z^\top \right\| \left\| \frac{1}{\sqrt{n}} Z w^* \right\| \leqslant O\left(\frac{\hat{\epsilon}}{\rho^2} \right).$$

Finally, let's bound the third term $\left(\left(\frac{1}{n} P^\top Z^\top Z P + \rho I \right)^{-1} + F \right) \frac{1}{n} P^\top Z^\top \xi$. We first bound the norm of $\frac{1}{n} P^\top Z^\top \xi$. with probability at least $1 - \exp(-\Omega(n))$, we know $\|\xi\| \leqslant 2\beta\sqrt{n}$. Therefore, we know $\left\| \frac{1}{n} P^\top Z^\top \xi \right\| \leqslant O(\beta/\sqrt{n}) \left\| P^\top Z^\top \bar{\xi} \right\|$, where $\bar{\xi} = \xi / \|\xi\|$. According to Lemma C.7, with probability at least $1 - \zeta/3$, we have $\left\| P^\top Z^\top \bar{\xi} \right\| \leqslant \sqrt{r} + O(\sqrt{\log(1/\zeta)})$. Overall, with probability at least $1 - \exp(-\Omega(n)) - \zeta/3$,

$$\left\| \frac{1}{n} P^\top Z^\top \xi \right\| \leqslant O\left(\frac{\sqrt{r}\beta + \sqrt{\log(1/\zeta)}\beta}{\sqrt{n}} \right).$$

It's not hard to verify that for any vector $v \in \mathbb{R}^d$ in the subspace S, we have $\left\| \left(\left(\frac{1}{n} P^\top Z^\top Z P + \rho I \right)^{-1} + F \right) v \right\| \leqslant O(\|v\|)$. Since $\frac{1}{n} P^\top Z^\top \xi$ lies on subspace S, we have

$$\left\| \left(\left(\frac{1}{n} P^\top Z^\top Z P + \rho I \right)^{-1} + F \right) \frac{1}{n} P^\top Z^\top \xi \right\| \leqslant O\left(\frac{\sqrt{r}\beta + \sqrt{\log(1/\zeta)}\beta}{\sqrt{n}} \right).$$

Combining the above analysis and taking a union bound over all the events, we know with probability at least $1 - \exp(-\Omega(n)) - 2\zeta/3$,

$$\|\hat{w} - w^*\| = O\left(\rho + \frac{\hat{\epsilon}}{\rho} + \frac{\hat{\epsilon}}{\rho^2} + \frac{\beta\hat{\epsilon}\sqrt{\log(1/\zeta)}}{\rho\sqrt{n}} + \frac{\sqrt{r}\beta + \sqrt{\log(1/\zeta)}\beta}{\sqrt{n}} \right)$$

Suppose $n \geqslant O(\log(1/\zeta))$ and setting $\rho = \hat{\epsilon}^{1/3}$, we further have with probability at least $1-\zeta$,

$$\begin{aligned}\|\hat{w} - w^*\| =& O\left(\hat{\epsilon}^{1/3} + \frac{\beta\hat{\epsilon}^{2/3}\sqrt{\log(1/\zeta)}}{\sqrt{n}} + \frac{\sqrt{r}\beta + \sqrt{\log(1/\zeta)}\beta}{\sqrt{n}}\right) \\ \leqslant& O\left(\hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right),\end{aligned}$$

where the last inequality assumes $\hat{\epsilon} < 1$.

We can also bound $\left\|\hat{P}\hat{w} - w^*\right\|$ as follows,

$$\begin{aligned}\left\|\hat{P}\hat{w} - w^*\right\| =& \left\|\hat{P}\hat{w} - P\hat{w} + P\hat{w} - Pw^*\right\| \\ \leqslant& \left\|\hat{P}\hat{w} - P\hat{w}\right\| + \|P\hat{w} - Pw^*\| \\ \leqslant& \left\|\hat{P} - P\right\| \|\hat{w}\| + \|P\| \|\hat{w} - w^*\| \\ \leqslant& \hat{\epsilon}O\left(1 + \hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right) + O\left(\hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right) \\ \leqslant& O\left(\hat{\epsilon}^{1/3} + \beta\frac{\sqrt{r} + \sqrt{\log(1/\zeta)}}{\sqrt{n}}\right)\end{aligned}$$

□

C.2.4 Analysis with $W_p := (W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha$

In this section, we prove that DirectSet(α) can also learn the projection matrix when we set $W_p := (W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha$. For the network architecture and data distribution, we follow exactly the same setting as in Section 4.3.2. Therefore, we know $W_p := (W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha = (W(I + \sigma^2 P_B)W^\top)^\alpha$.

Theorem C.1. *Suppose network architecture and data distribution are as defined in Assumption 4.1 and Assumption 4.2, respectively. Suppose we initialize online*

network W as δI, and run DirectPred(α) on population loss (see Eqn. 4.1) with infinitesimal step size and η weight decay. Suppose we set $W_a = W$ and $W_p = (W\mathbb{E}_{x_1} x_1 x_1^\top W^\top)^\alpha$. Assuming the weight decay coefficient $\eta \in \left(\frac{1}{4(1+\sigma^2)^{1+2\alpha}}, \frac{1}{4}\right)$ and initialization scale $\delta > \left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$, we know W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$ when time goes to infinity.

The only difference from Theorem C.1 is that now the initialization δ is only required to be larger than $\frac{1}{4(1+\sigma^2)^{1+2\alpha}}$. The proof is almost the same as in Theorem 4.1.

Proof of Theorem C.1. Similar as in the proof of Theorem 4.1, we can write the dynamics on W is as follows:

$$\begin{aligned}
\dot{W} =& W_p^\top(-W_p W(I+\sigma^2 P_B) + W_a) - \eta W \\
=& \left|W^2(I+\sigma^2 P_B)\right|^\alpha \left(-\left|W^2(I+\sigma^2 P_B)\right|^\alpha W(I+\sigma^2 P_B) + W\right) - \eta W \\
=& W\left(-(I+\sigma^2 P_B)^{1+2\alpha}|W|^{4\alpha} + |W|^{2\alpha} - \eta\right).
\end{aligned}$$

Dynamics for λ_B: We can write down the dynamics for λ_B as follows:

$$\dot{\lambda}_B = \lambda_B\left[-(1+\sigma^2)^{1+2\alpha}|\lambda_B|^{4\alpha} + |\lambda_B|^{2\alpha} - \eta\right]$$

When $\eta > \frac{1}{4(1+\sigma^2)^{1+2\alpha}}$, we know $\dot{\lambda}_B < 0$ for any $\lambda_B > 0$ and $\lambda_B = 0$ is a critical point. This means, as long as $\eta > \frac{1}{4(1+\sigma^2)^{1+2\alpha}}$, λ_B must converge to zero.

Dynamics for λ_S: The dynamics is same as when setting $W_p = (WW^\top)^\alpha$,

$$\dot{\lambda}_S = \lambda_S\left[-|\lambda_S|^{4\alpha} + |\lambda_S|^{2\alpha} - \eta\right].$$

so when $0 < \eta < \frac{1}{4}$ and initialization $\delta^{2\alpha} > \frac{1-\sqrt{1-4\eta}}{2}$, we know $\lambda_S^{2\alpha}$ converges to $\frac{1+\sqrt{1-4\eta}}{2}$.

Overall, we know when $\frac{1}{4(1+\sigma^2)^{1+2\alpha}} < \eta < \frac{1}{4}$ and $\delta > \left(\frac{1-\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$, we have λ_B converge to zero and λ_S converge to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)}$. That is, matrix W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2}\right)^{1/(2\alpha)} P_S$. □

C.2.5 Technical Lemmas

Lemma C.3. *Suppose $\{x^{(i)}, x_1^{(i)}, x_2^{(i)}\}_{i=1}^n$ are sampled as decribed in Section 4.3. Suppose $n \geqslant O(d/\hat{\epsilon}^2)$, with probability at least $1 - O(d^2)\exp\left(-\Omega(\hat{\epsilon}^2 n/d^2)\right)$, we have*

$$\left\|\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_1^{(i)}]^\top - I - \sigma^2 P_B\right\|, \left\|\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_2^{(i)}]^\top - I\right\|, \left\|\frac{1}{n}\sum_{i=1}^n x^{(i)}[x^{(i)}]^\top - I\right\| \leqslant \hat{\epsilon}.$$

Proof of Lemma C.3. For each $x_1^{(i)}$, we can write it as $x^{(i)} + z_1^{(i)}$ where $x^{(i)} \sim \mathcal{N}(0, I)$ and $z_1^{(i)} \sim \mathcal{N}(0, \sigma^2 P_B)$. So we have

$$\frac{1}{n}\sum_{i=1}^n x_1^{(i)}[x_1^{(i)}]^\top = \frac{1}{n}\sum_{i=1}^n \left(x^{(i)}[x^{(i)}]^\top + z_1^{(i)}[z_1^{(i)}]^\top + x^{(i)}[z_1^{(i)}]^\top + z_1^{(i)}[x^{(i)}]^\top\right).$$

According to Lemma C.9, we know as long as $n \geqslant O(d/\hat{\epsilon}^2)$, with probability at least $1 - \exp(-\Omega(\hat{\epsilon}^2 n))$,

$$\left\|\frac{1}{n}\sum_{i=1}^n x^{(i)}[x^{(i)}]^\top - I\right\| \leqslant \hat{\epsilon}.$$

Similarly, with probability at least $1 - \exp(-\Omega(\hat{\epsilon}^2 n))$,

$$\left\|\frac{1}{n}\sum_{i=1}^n z_1^{(i)}[z_1^{(i)}]^\top - \sigma^2 P_B\right\| \leqslant \hat{\epsilon}.$$

Next we bound $\left\|\frac{1}{n}\sum_{i=1}^n x^{(i)}[z_1^{(i)}]^\top\right\|$. We know each entry in matrix $\frac{1}{n}\sum_{i=1}^n x^{(i)}[z_1^{(i)}]^\top$ is the average of n zero-mean $O(1)$-subexponential independent

random variables. Therefore, according to the Bernstein's inequality, for any fixed entry (k, l), with probability at least $1 - \exp\left(-\hat{\epsilon}^2 n/d^2\right)$,

$$\left| \left[\frac{1}{n} \sum_{i=1}^{n} x^{(i)} [z_1^{(i)}]^\top \right]_{k,l} \right| \leqslant \hat{\epsilon}/d.$$

Taking a union bound over all the entries, we know with probability at least $1 - d^2 \exp\left(-\hat{\epsilon}^2 n/d^2\right)$,

$$\left\| \frac{1}{n} \sum_{i=1}^{n} x^{(i)} [z_1^{(i)}]^\top \right\| \leqslant \left\| \frac{1}{n} \sum_{i=1}^{n} x^{(i)} [z_1^{(i)}]^\top \right\|_F \leqslant \hat{\epsilon}.$$

The same analysis also applies to $\left\| \frac{1}{n} \sum_{i=1}^{n} z_1^{(i)} [x^{(i)}]^\top \right\|$. Combing all the bounds, we know with probability at least $1 - O(d^2) \exp\left(-\Omega(\hat{\epsilon}^2 n/d^2)\right)$,

$$\left\| \frac{1}{n} \sum_{i=1}^{n} x_1^{(i)} [x_1^{(i)}]^\top - I - \sigma^2 P_B \right\| \leqslant 4\hat{\epsilon}.$$

Similarly, we can prove that with probability at least $1 - O(d^2) \exp\left(-\Omega(\hat{\epsilon}^2 n/d^2)\right)$,

$$\left\| \frac{1}{n} \sum_{i=1}^{n} x_1^{(i)} [x_2^{(i)}]^\top - I \right\| \leqslant 4\hat{\epsilon}.$$

Changing $\hat{\epsilon}$ to $\hat{\epsilon}'/4$ finishes the proof. □

Lemma C.4. *Let $X \in \mathbb{R}^{n \times d}$ be a standard Gaussian matrix, and let $U \in \mathbb{R}^{d \times r}$ be a matrix with orthonormal columns. Suppose $n \geqslant 2r$, with probability at least $1 - \exp(-\Omega(n))$, we know*

$$\Omega(1) \leqslant \lambda_{\min}\left(\frac{1}{n} U^\top X^\top X U\right) \leqslant \lambda_{\max}\left(\frac{1}{n} U^\top X^\top X U\right) \leqslant O(1).$$

Proof of Lemma C.4. Since U has orthonormal columns, we know XU is a $n \times r$ matrix with each entry independently sampled from $\mathcal{N}(0,1)$. According to Lemma C.9, we know when $n \geqslant 2r$, with probability at least $1 - \exp(-\Omega(n))$,

$$\Omega(1) \leqslant \sigma_{\min}\left(\frac{1}{\sqrt{n}}XU\right) \leqslant \sigma_{\max}\left(\frac{1}{\sqrt{n}}XU\right) \leqslant O(1).$$

This immediately implies that

$$\Omega(1) \leqslant \lambda_{\min}\left(\frac{1}{n}U^\top X^\top XU\right) \leqslant \lambda_{\max}\left(\frac{1}{n}U^\top X^\top XU\right) \leqslant O(1).$$

□

Lemma C.5. *Let Δ be a $d \times d$ matrix with Frobenius norm $\hat{\epsilon}$, and let X be a $n \times d$ standard Gaussian matrix. We know with probability at least $1 - \exp(-\Omega(n))$,*

$$\left\|\frac{1}{\sqrt{n}}X\Delta\right\|_F \leqslant O(\hat{\epsilon}).$$

Proof of Lemma C.5. Let the singular value decomposition of Δ be $U\Sigma V^\top$, where U, V have orthonormal columns and Σ is a diagonal matrix with diagonals equal to singular values σ_i's. Since $\|\Delta\|_F = \hat{\epsilon}$, we know $\sum_{i=1}^d \sigma_i^2 = \hat{\epsilon}^2$.

Since U is an orthonormal matrix, we know $\hat{X} := XU$ is still an $n \times d$ standard Gaussian matrix. Next, we bound the Frobenius norm of $\widetilde{X} := \hat{X}\Sigma$. It's not hard to verify that all the entries in $\widetilde{X}$ are independent Gaussian variables and $\widetilde{X}_{ij} \sim \mathcal{N}(0, \sigma_j^2)$. According to the Bernstein's inequality for sum of independent and sub-exponential random variables, we have for every $t > 0$,

$$\Pr\left[\left|\sum_{i\in[n],j\in[d]} \widetilde{X}_{ij}^2 - n\hat{\epsilon}^2\right| \geqslant t\right] \leqslant 2\exp\left[-c\min\left(\frac{t^2}{\sum_{i\in[n],j\in[d]}\sigma_j^4}, \frac{t}{\max_{j\in[d]}\sigma_j^2}\right)\right].$$

Since $\sum_{j=1}^{d} \sigma_j^2 = \|\Delta\|_F^2 = \hat{\epsilon}^2$, we know $\max_{j\in[d]} \sigma_j^2 \leqslant \hat{\epsilon}^2$. We also have $\sum_{j\in[d]} \sigma_j^4 \leqslant \left(\sum_{j\in[d]} \sigma_j^2\right)^2 = \hat{\epsilon}^4$. Therefore, we have

$$\Pr\left[\left|\sum_{i\in[n],j\in[d]} \widetilde{X}_{ij}^2 - n\hat{\epsilon}^2\right| \geqslant t\right] \leqslant 2\exp\left[-c\min\left(\frac{t^2}{n\hat{\epsilon}^4}, \frac{t}{\hat{\epsilon}^2}\right)\right].$$

Replacing t by $n\hat{\epsilon}^2$, we concluded that with probability at least $1 - \exp(-\Omega(n))$, $\left\|\widetilde{X}\right\|_F^2 \leqslant 2n\hat{\epsilon}^2$. Furthermore, since $\left\|V^\top\right\| = 1$, we have

$$\left\|\frac{1}{\sqrt{n}} X\Delta\right\|_F = \left\|\frac{1}{\sqrt{n}}\widetilde{X}V^\top\right\|_F \leqslant \left\|\frac{1}{\sqrt{n}}\widetilde{X}\right\|_F \|V\| \leqslant O(\hat{\epsilon}).$$

□

Lemma C.6. *Let $\Delta^\top$ be a $d \times d$ matrix with Frebenius norm $\hat{\epsilon}$ and let $X^\top$ be a $d \times n$ standard Gaussian matrix. Let $\bar{\xi}$ be a unit vector with dimension n. We know with probability at least $1 - \zeta/3$,*

$$\left\|\Delta^\top X^\top \bar{\xi}\right\| \leqslant O(\hat{\epsilon}\sqrt{\log(1/\zeta)}).$$

Proof of Lemma C.6. Let the sigular value decomposition of $\Delta^\top$ be $U\Sigma V^\top$. We know $X^\top\bar{\xi}$ is a d-dimensional standard Gaussian vector. Further, we know $V^\top X^\top \bar{\xi}$ is also a d-dimensional standard Gaussian vector. So $\Sigma V^\top X^\top \bar{\xi}$ has independent Gaussian entries with its i-th entry distributed as $\mathcal{N}(0, \sigma_i^2)$. According to the Bernstein's inequality for sum of independent and sub-exponential random variables, we have for every $t > 0$,

$$\Pr\left[\left|\left\|\Sigma V^\top X^\top \bar{\xi}\right\|^2 - \hat{\epsilon}^2\right| \geqslant t\right] \leqslant 2\exp\left[-c\min\left(\frac{t^2}{\hat{\epsilon}^4}, \frac{t}{\hat{\epsilon}^2}\right)\right].$$

Choosing t as $O(\hat{\epsilon}^2\log(1/\zeta))$, we know with probability at least $1 - \zeta/3$, we have

$$\left\|\Sigma V^\top X^\top \bar{\xi}\right\|^2 \leqslant O\left(\hat{\epsilon}^2\log(1/\zeta)\right).$$

Since $\|U\| = 1$, we further have

$$\left\|\Delta^\top X^\top \bar{\xi}\right\| = \left\|U\Sigma V^\top X^\top \bar{\xi}\right\| \leqslant \|U\| \left\|\Sigma V^\top X^\top \bar{\xi}\right\| \leqslant O\left(\hat{\epsilon}\sqrt{\log(1/\zeta)}\right)$$

□

Lemma C.7. *Let $P \in \mathbb{R}^{d\times d}$ be a projection matrix on a r-dimensional subspace, and let $\bar{\xi}$ be a unit vector in $\mathbb{R}^d$. Let $X^\top$ be a $d \times n$ standard Gaussian matrix that is independent with P and ξ. With probability at least $1 - \zeta/3$, we have*

$$\left\|P^\top X^\top \bar{\xi}\right\| \leqslant \sqrt{r} + O(\sqrt{\log(1/\zeta)}).$$

Proof of Lemma C.7. Since P is a projection matrix on an r-dimensional subspace, we can write P as $UU^\top$, where $U \in \mathbb{R}^{d\times r}$ has orthonormal columns. We know $U^\top X^\top$ is still a standard Gaussian matrix with dimension $r \times n$. Furthermore, $U^\top X^\top \bar{\xi}$ is an r-dimensional standard Gaussian vector. According to Lemma C.8, with probability at least $1 - \zeta/3$, we have

$$\left\|U^\top X^\top \bar{\xi}\right\| \leqslant \sqrt{r} + O(\sqrt{\log(1/\zeta)}).$$

Since $\|U\| = 1$, we further have

$$\left\|P^\top X^\top \bar{\xi}\right\| = \left\|UU^\top X^\top \bar{\xi}\right\| \leqslant \|U\| \left\|U^\top X^\top \bar{\xi}\right\| \leqslant \sqrt{r} + O(\sqrt{\log(1/\zeta)}).$$

□

C.3 Analysis of Deep Linear Networks

In this section, we extend the analysis in Section 4.3.2 to deep linear networks. We consider the same data distribution as defined in Assumption 4.2. We consider the following network,

Assumption C.1 (Deep linear network)**.** *The online network is an l-layer linear networks $W_l W_{l-1} \cdots W_1$ with each $W_i \in \mathbb{R}^{d\times d}$. The target network has the same architecture with weight matrices $W_{a,l} W_{a,l-1} \cdots W_{a,1}$. For convenience, we denote W as $W_l W_{l-1} \cdots W_1$ and denote W_a as $W_{a,l} W_{a,l-1} \cdots W_{a,1}$.*

Training procedure: At the initialization, we initialize each W_i as $\delta^{1/l} I_d$. Through the training, we fix W_p as $\left(WW^\top\right)^\alpha$ and fix each $W_{a,i}$ as W_i. We run gradient flow on every W_i with weight decay η. The population loss is

$$\begin{aligned}&L(\{W_i\}, W_p, \{W_{a,i}\})\\ :=&\frac{1}{2}\mathbb{E}_{x_1,x_2}\left\|W_p W_l W_{l-1}\cdots W_1 x_1 - \mathrm{StopGrad}(W_{a,l} W_{a,l-1}\cdots W_{a,1} x_2)\right\|^2.\end{aligned}$$

Theorem C.2. *Suppose the data distribution and network architecture satisfies Assumption 4.2 and Assumption C.1, respectively. Suppose we train the network as described above. Assuming the weight decay coefficient $\eta \in \left(\frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}(1+\sigma^2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}, \frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}}\right)$, and initialization scale $\delta \geqslant \left(\frac{2\alpha l+2l-2}{4\alpha l+2l-2}\right)^{\frac{1}{2\alpha}}$, we know W converges to cP_S as time goes to infinity, where c is a positive number within $\left(\left(\frac{2\alpha l+2l-2}{4\alpha l+2l-2}\right)^{\frac{1}{2\alpha}}, 1\right)$.*

Similar as in the setting of single-layer linear networks, we prove Theorem C.2 by analyzing the dynamics of the eigenvalues of W. Note that with constant α, the upper/lower bounds for η and scalar c in the Theorem are always constants no matter how large l is.

Proof of Theorem C.2. For $j \geqslant i$, we use $W_{[j:i]}$ to denote $W_j W_{j-1}\cdots W_i$ and for $j < i$ have $W_{[j:i]} = I$. We use similar notations for $W_{a,[j:i]}$. For each W_i, we can compute its dynamics as follows:

$$\begin{aligned}\dot{W}_i =& -\left(W_p W_{[l:i+1]}\right)^\top \left(W_p W(I+\sigma^2 P_B)\right)\left(W_{[i-1:1]}\right)^\top\\ &+\left(W_p W_{a,[l:i+1]}\right)^\top W_a \left(W_{a,[i-1:1]}\right)^\top - \eta W_i.\end{aligned}$$

It's clear that through the training all W_i's remains the same and they are simultaneously diagonalizable with W_p, I and P_B. We also have $W_a = W$ and $W_p = |W|^{2\alpha}$. Since we will ensure that W is always positive semi-definite so $W_p = |W|^{2\alpha} = W^{2\alpha} = W_i^{2\alpha l}$. So the dynamics for each W_i can be simplified as follows:

$$\dot{W}_i = -W_i^{4\alpha l+2l-1}(I + \sigma^2 P_B) + W_i^{2\alpha l+2l-1} - \eta W_i.$$

Let the eigenvalue decomposition of W_i be $\sum_{i=1}^{d} \nu_i u_i u_i^\top$, with span$(\{u_{d-r+1}, \cdots, u_d\})$ equals to subspace B. We can separately analyze the dynamics of each ν_i. Furthermore, we know $\nu_1, \cdots, \nu_r$ have the same value ν_S and $\nu_{d-r+1}, \cdots, \nu_d$ have the same value ν_B. We can write down the dynamics for ν_S and ν_B as follows,

$$\begin{aligned}
\dot{\nu}_S =& -\nu_S^{4\alpha l+2l-1} + \nu_S^{2\alpha l+2l-1} - \eta\nu_S, \\
\dot{\nu}_B =& -\nu_B^{4\alpha l+2l-1}(1+\sigma^2) + \nu_B^{2\alpha l+2l-1} - \eta\nu_B.
\end{aligned}$$

Let λ_S be the eigenvalue of W corresponding to eigen-directions $u_1, \cdots, u_r$, and let λ_B be the eigenvalue of W corresponding to eigen-directions $u_{d-r+1}, \cdots, u_d$. We know $\lambda_S = \nu_S^l$ and $\lambda_B = \nu_B^l$. So we can write down the dynamics for λ_B as follows,

$$\begin{aligned}
\dot{\lambda}_B = l\nu_B^{l-1}\dot{\nu}_B =& -l\nu_B^{4\alpha l+3l-2}(1+\sigma^2) + l\nu_B^{2\alpha l+3l-2} - l\eta\nu_B^l \\
=& -l\lambda_B^{4\alpha+3-2/l}(1+\sigma^2) + l\lambda_B^{2\alpha+3-2/l} - l\eta\lambda_B,
\end{aligned}$$

and similarly for λ_S we have

$$\dot{\lambda}_S = -l\lambda_S^{4\alpha+3-2/l} + l\lambda_S^{2\alpha+3-2/l} - l\eta\lambda_S.$$

Dynamics for λ_B: We can write the dynamics on λ_B as follows,

$$\dot{\lambda}_B = l\lambda_B g(\lambda_B),$$

where $g(\lambda_B) := -\lambda_B^{4\alpha+2-2/l}(1+\sigma^2) + \lambda_B^{2\alpha+2-2/l} - \eta$. We show that when η is large enough, $g(\lambda_B)$ is negative for any positive λ_B. We compute the maximum value of

$g(\lambda_B)$ for $\lambda_B > 0$. We first compute the derivative of g as follows:

$$\begin{aligned} g'(\lambda_B) =& -(4\alpha + 2 - 2/l)(1+\sigma^2)\lambda_B^{4\alpha+1-2/l} + (2\alpha + 2 - 2/l)\lambda_B^{2\alpha+1-2/l} \\ =& \lambda_B^{2\alpha+1-2/l}\left(-(4\alpha + 2 - 2/l)(1+\sigma^2)\lambda_B^{2\alpha} + (2\alpha + 2 - 2/l)\right). \end{aligned}$$

It's clear that $g'(\lambda_B) > 0$ for $\lambda_B^{2\alpha} \in (0, \frac{2\alpha l+2l-2}{(4\alpha l+2l-2)(1+\sigma^2)})$ and $g'(\lambda_B) < 0$ for $\lambda_B^{2\alpha} \in (\frac{2\alpha l+2l-2}{(4\alpha l+2l-2)(1+\sigma^2)}, +\infty)$. Therefore, the maximum value of $g(\lambda_B)$ for positive λ_B takes at $\lambda_B^* = \left(\frac{2\alpha l+2l-2}{(4\alpha l+2l-2)(1+\sigma^2)}\right)^{\frac{1}{2\alpha}}$ and

$$\begin{aligned} g(\lambda_B^*) =& -\left(\frac{2\alpha l + 2l - 2}{(4\alpha l + 2l - 2)(1+\sigma^2)}\right)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}(1+\sigma^2) \\ &+ \left(\frac{2\alpha l + 2l - 2}{(4\alpha l + 2l - 2)(1+\sigma^2)}\right)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}} - \eta \\ =& \frac{2\alpha l(2\alpha l + 2l - 2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l + 2l - 2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}(1+\sigma^2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}} - \eta. \end{aligned}$$

As long as $\eta > \frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}(1+\sigma^2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}$, we know $g(\lambda_B) < 0$ for any $\lambda_B > 0$, which further implies that $\dot{\lambda}_B < 0$ for any $\lambda_B > 0$. So λ_B converges to zero.

Dynamics for λ_S : We can write down the dynamics on λ_S as follows,

$$\dot{\lambda}_S = l\lambda_S h(\lambda_S),$$

where $h(\lambda_S) = -\lambda_S^{4\alpha+2-2/l} + \lambda_S^{2\alpha+2-2/l} - \eta$. We compute the derivative of h as follows:

$$h'(\lambda_S) = \lambda_S^{2\alpha+1-2/l}\left(-(4\alpha + 2 - 2/l)\lambda_S^{2\alpha} + (2\alpha + 2 - 2/l)\right).$$

So $h(\lambda_S)$ is increasing in $(0, \left(\frac{2\alpha l+2l-2}{4\alpha l+2l-2}\right)^{\frac{1}{2\alpha}})$ and is decreasing in $(\left(\frac{2\alpha l+2l-2}{4\alpha l+2l-2}\right)^{\frac{1}{2\alpha}}, \infty)$. The maximum value of h for positive λ_S takes at $\lambda_S^* = \left(\frac{2\alpha l+2l-2}{4\alpha l+2l-2}\right)^{\frac{1}{2\alpha}}$ and we have

$$h(\lambda_S^*) = \frac{2\alpha l(2\alpha l + 2l - 2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l + 2l - 2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}} - \eta.$$

As long as $\eta < \frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}}$, we have $h(\lambda_S^*) > 0$. Furthermore, since h is increasing in $(0, \lambda_S^*)$ and is decreasing in (λ_S^*, ∞) and $h(0), h(\infty) < 0$, we know there exists $\lambda_S^- \in (0, \lambda_S^*), \lambda_S^+ \in (\lambda_S^*, \infty)$ such that $h(\lambda_S) < 0$ in $(0, \lambda_S^-)$, $h(\lambda_S) > 0$ in $(\lambda_S^-, \lambda_S^+)$ and $h(\lambda_S) < 0$ in (λ_S^+, ∞). Therefore, as long as $\delta \geqslant \lambda_S^* > \lambda_S^-$, we have λ_S converges to λ_S^+. Since $h(1) < 0$, we know $\lambda_S^+ \in ((\frac{2\alpha l+2l-2}{4\alpha l+2l-2})^{\frac{1}{2\alpha}}, 1)$.

Overall as long as $\eta \in \left(\frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}(1+\sigma^2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}, \frac{2\alpha l(2\alpha l+2l-2)^{1+\frac{1}{\alpha}-\frac{1}{\alpha l}}}{(4\alpha l+2l-2)^{2+\frac{1}{\alpha}-\frac{1}{\alpha l}}} \right)$, we know W converges to cP_S, where c is a positive number within $((\frac{2\alpha l+2l-2}{4\alpha l+2l-2})^{\frac{1}{2\alpha}}, 1)$. □

C.4 Analysis of Predictor Regularization.

In this section, we study the influence of predictor regularization in a simple linear setting. In particular, we consider the same setting as in Section 4.3.2 except that we set $W_p := (WW^\top)^\alpha + \epsilon I$.

Theorem C.3. *In the setting of Theorem 4.1 except that we set $W_p = (WW^\top)^\alpha + \epsilon I$. We have*

- *when $\epsilon \in [0, \frac{1+\sqrt{1-4\eta}}{2})$, as long as $\delta > \left(\max\left(\frac{1-\sqrt{1-4\eta}}{2} - \epsilon, 0\right)\right)^{\frac{1}{2\alpha}}$, we have W converges to $\left(\frac{1+\sqrt{1-4\eta}}{2} - \epsilon\right)^{\frac{1}{2\alpha}} P_S$;*
- *when $\epsilon \geqslant \frac{1+\sqrt{1-4\eta}}{2}$, W always converges to zero.*

Proof of Theorem C.3. We can write the dynamics of W as follows,

$$\begin{aligned}\dot{W} =& W_p^\top(-W_p W(I + \sigma^2 P_B) + W_a) - \eta W \\ =& W\left(-(I + \sigma^2 P_B)\left(|W|^{2\alpha} + \epsilon I\right)^2 + \left(|W|^{2\alpha} + \epsilon I\right) - \eta\right).\end{aligned}$$

Let the eigenvalue decomposition of W be $\sum_{i=1}^d \lambda_i u_i u_i^\top$, with $\mathrm{span}(\{u_{d-r+1}, \cdots, u_d\})$ equals to subspace B. We can separately analyze the dy-

namics of each λ_i. Furthermore, we know $\lambda_1, \cdots, \lambda_r$ have the same value λ_S and $\lambda_{d-r+1}, \cdots, \lambda_d$ have the same value λ_B.

Dynamics for λ_B: We can write down the dynamics for λ_B as follows:

$$\dot{\lambda}_B = \lambda_B \left[-(1+\sigma^2) \left(|\lambda_B|^{2\alpha} + \epsilon \right)^2 + \left(|\lambda_B|^{2\alpha} + \epsilon \right) - \eta \right]$$

When $\eta > \frac{1}{4(1+\sigma^2)}$, we still know $\dot{\lambda}_B < 0$ for any $\lambda_B > 0$ and $\lambda_B = 0$ is a critical point. So λ_B converges to zero.

Dynamics for λ_S: We can write down the dynamics for λ_S as follows:

$$\begin{aligned}
\dot{\lambda}_S =& \lambda_S \left[- \left(|\lambda_S|^{2\alpha} + \epsilon \right)^2 + \left(|\lambda_S|^{2\alpha} + \epsilon \right) - \eta \right] \\
=& - \lambda_S \left(|\lambda_S|^{2\alpha} + \epsilon - \frac{1 - \sqrt{1-4\eta}}{2} \right) \left(|\lambda_S|^{2\alpha} + \epsilon - \frac{1 + \sqrt{1-4\eta}}{2} \right),
\end{aligned}$$

where the second inequality assumes $0 < \eta < \frac{1}{4}$. We have

- when $\epsilon \in [0, \frac{1+\sqrt{1-4\eta}}{2})$, as long as $\delta > \left(\max \left(\frac{1-\sqrt{1-4\eta}}{2} - \epsilon, 0 \right) \right)^{\frac{1}{2\alpha}}$, we have λ_S converges to $\left(\frac{1+\sqrt{1-4\eta}}{2} - \epsilon \right)^{\frac{1}{2\alpha}} > 0$;
- when $\epsilon \geqslant \frac{1+\sqrt{1-4\eta}}{2}$, λ_S always converges to zero.

□

C.5 Technical Tools

C.5.1 Norm of Random Vectors

The following lemma shows that a standard Gaussian vector with dimension n has ℓ_2 norm concentrated at $\sqrt{n}$.

Lemma C.8 (Theorem 3.1.1 in Vershynin (2018)). *Let $X = (X_1, X_2, \cdots, X_n) \in \mathbb{R}^n$ be a random vector with each entry independently sampled from $\mathcal{N}(0,1)$. Then*

$$\Pr[\big|\|x\| - \sqrt{n}\big| \geqslant t] \leqslant 2\exp(-t^2/C^2),$$

where C is an absolute constant.

C.5.2 Singular Values of Gaussian Matrices

The following lemma shows a tall random Gaussian matrix is well-conditioned with high probability.

Lemma C.9 (Corollary 5.35 in Vershynin (2010)). *Let A be an $N \times n$ matrix whose entries are independent standard normal random variables. Then for every $t \geqslant 0$ with probability at least $1 - 2\exp(-t^2/2)$ one has*

$$\sqrt{N} - \sqrt{n} - t \leqslant s_{\min}(A) \leqslant s_{\max}(A) \leqslant \sqrt{N} + \sqrt{n} + t$$

C.5.3 Perturbation Bound for Matrix Pseudo-inverse

With a lowerbound on $\sigma_{\min}(A)$, we can get bounds for the perturbation of pseudo-inverse.

Lemma C.10 (Theorem 3.4 in Stewart (1977)). *Consider the perturbation of a matrix $A \in \mathbb{R}^{m\times n} : B = A + E$. Assume that $rank(A) = rank(B) = n$, then*

$$\left\|B^\dagger - A^\dagger\right\| \leqslant \sqrt{2}\left\|A^\dagger\right\|\left\|B^\dagger\right\|\|E\|.$$

The following corollary is particularly useful for us.

Lemma C.11 (Lemma G.8 in Ge et al. (2015b)). *Consider the perturbation of a matrix $A \in \mathbb{R}^{m\times n} : B = A + E$ where $\|E\| \leqslant \sigma_{\min}(A)/2$. Assume that $rank(A) = rank(B) = n$, then*

$$\left\|B^\dagger - A^\dagger\right\| \leqslant 2\sqrt{2}\|E\|/\sigma_{\min}(A)^2.$$

Appendix D

Supplementary Materials for Chapter 5

D.1 Examples for the disconnection between linear interpolation shape and optimization difficulty

We give two examples that illustrates the disconnection between the linear interpolation shape and the optimization difficulty. In Section D.1.1, we show a function that is NP-hard to optimize, but has a convex and monotonically decreasing loss interpolation. Then in Section D.1.2, we give a function that is easy to optimize, but has a non-monotonic loss interpolation.

D.1.1 Hard function with convex loss interpolation

For any symmetric third-order tensor $T \in \mathbb{R}^{d\times d\times d}$, our goal is to minimize

$$f(x, z) = -T(x, x, x) + \|x\|^4 + z^4 \tag{D.1}$$

where $x \in \mathbb{R}^d$ and $z \in \mathbb{R}$.

It's known that finding the spectral norm of a symmetric third-order tensor (that is, $\max_{v\in\mathbb{R}^d,\|v\|=1} T(v, v, v)$) is NP-hard (Hillar and Lim, 2013). We prove that minimizing $f(x, z)$ is also NP-hard by reducing the tensor spectral norm problem to it.

Proposition D.1. *Minimizing $f(x, z)$ as defined in Eqn. D.1 is NP-hard.*

Proof. For any non-zero tensor T, let (x^*, z^*) be one minimizer of $f(x, z)$, it's easy to verify that $T(x^*, x^*, x^*) > 0$. We show that $\bar{x}^* := x^*/\|x^*\|$ must be a solution to $\max_{v\in\mathbb{R}^d, \|v\|=1} T(v, v, v)$.

For the sake of contradiction, assume there exists v^* with unit norm such that $T(v^*, v^*, v^*) > T(\bar{x}^*, \bar{x}^*, \bar{x}^*)$. It's easy to verify that $f(\|x^*\| v^*, z^*) < f(x^*, z^*)$, which however contradicts the optimality of (x^*, z^*). □

Next, we prove that start from certain initialization, the loss along the linear interpolation path is convex and monotonically decreasing. Note that assuming the unit Frobenius norm of T does not hurt the NP-hardness of the problem. And our initialization is oblivious of the tensor T.

Proposition D.2. *Assume $\|T\|_F = 1$. Suppose we start from initialization (x_0, z_0) with $x_0 = 0$ and $|z_0| > \frac{3\sqrt{2}}{4}$. Let (x^*, z^*) be a minimizer of $f(x, z)$ as defined in Eqn. D.1. We know the loss interpolation curve*

$$\gamma(\alpha) := f\left((1-\alpha)x_0 + \alpha x^*, (1-\alpha)z_0 + \alpha z^*\right)$$

is convex and monotonically decreasing for $\alpha \in [0, 1]$.

Proof. We first prove that at any minimizer (x^*, z^*), we must have $z^* = 0$. Otherwise, we can set z as zero to further decrease the loss. Starting from an initialization $(x^{(0)}, z^{(0)})$ with $x^{(0)} = 0$, we know at each interpolation point $x^{[\alpha]} = \alpha x^*, z^{[\alpha]} = (1-\alpha)z^{(0)}$. Therefore, we have

$$\begin{aligned}
\gamma(\alpha) = f(x^{[\alpha]}, z^{[\alpha]}) =& -T(x^{[\alpha]}, x^{[\alpha]}, x^{[\alpha]}) + \left\|x^{[\alpha]}\right\|^4 + \left[z^{[\alpha]}\right]^4 \\
=& -T(\alpha x^*, \alpha x^*, \alpha x^*) + \|\alpha x^*\|^4 + \left[(1-\alpha)z^{(0)}\right]^4 \\
=& -\alpha^3 T(x^*, x^*, x^*) + \alpha^4 \|x^*\|^4 + (1-\alpha)^4 \left[z^{(0)}\right]^4.
\end{aligned}$$

To prove the convexity of $\gamma(\alpha)$ for $\alpha \in [0,1]$, we only need to prove $\gamma''(\alpha) > 0$ for $\alpha \in [0,1]$. We have

$$\begin{aligned}\gamma''(\alpha) =& -6\alpha T(x^*, x^*, x^*) + 12\alpha^2 \|x^*\|^4 + 12(1-\alpha)^2 \left[z^{(0)}\right]^4 \\ =& -6\alpha \|x^*\|^3 T(\bar{x}^*, \bar{x}^*, \bar{x}^*) + 12\alpha^2 \|x^*\|^4 + 12(1-\alpha)^2 \left[z^{(0)}\right]^4.\end{aligned}$$

Since the formula for $\gamma''(\alpha)$ involves both $T(\bar{x}^*, \bar{x}^*, \bar{x}^*)$ and $\|x^*\|$, we first figure out the relation between these two quantities. Suppose $T(\bar{x}^*, \bar{x}^*, \bar{x}^*) = p > 0$, it's not hard to find $\|x^*\|$ must be equal to $\frac{3p}{4}$. This is because $-\|x^*\|^3 p + \|x^*\|^4$ is minimized when $\|x^*\| = \frac{3p}{4}$. Next, we prove $\gamma''(\alpha) > 0$ for $\alpha \in (2/3, 1]$ and $\alpha \in [0, 2/3]$ separetely.

When $\alpha \in (2/3, 1]$, we have

$$12\alpha^2 \|x^*\|^4 > 6p\alpha \|x^*\|^3 = 6\alpha \|x^*\|^3 T(\bar{x}^*, \bar{x}^*, \bar{x}^*).$$

Therefore, we know $\gamma''(\alpha) > 0$.

When $\alpha \in [0, 2/3]$, we know

$$12(1-\alpha)^2 \left[z^{(0)}\right]^4 \geqslant \frac{4}{3} \left[z^{(0)}\right]^4.$$

Since $\|T\|_F = 1$, we know $T(\bar{x}^*, \bar{x}^*, \bar{x}^*) \leqslant 1$ and $\|x^*\| \leqslant 3/4$. Therefore, we have

$$6\alpha \|x^*\|^3 T(\bar{x}^*, \bar{x}^*, \bar{x}^*) \leqslant 6 \cdot \frac{2}{3} \cdot \left(\frac{3}{4}\right)^3 \cdot 1 = \frac{27}{16}.$$

Then, we know that if $\left|z^{(0)}\right| > \frac{3\sqrt{2}}{4}$, we have $\gamma''(\alpha) > 0$. □

D.1.2 Easy function with non-monotonic loss interpolation

In this section, we give an easy-to-optimize function that however has a non-monotonic loss interpolation curve. We consider the following loss function

$$f(x,y) = \begin{cases} 0 & \text{if } x = y = 0 \\ \left(1 - \frac{y}{\sqrt[3]{x^2+y^2}}\right)\left(\left(x^2+y^2\right)^2 - 2\left(x^2+y^2\right)\right) & \text{otherwise,} \end{cases} \tag{D.2}$$

where $x, y \in \mathbb{R}$. We can also re-parameterize $f(x, y)$ using angle $\theta \in [0, 2\pi)$ and length $r \in [0, \infty)$ as $h(\theta, r) = \left(1 - \frac{\sin(\theta)}{3}\right)(r^4 - 2r^2)$.

Next, we prove that starting from any non-zero point, gradient flow converges to the global minimizer.

Proposition D.3. *Starting from any non-zero initialization, gradient flow on $f(x, y)$ as defined in Eqn. D.2 converges to the global minimizer $(0, -1)$.*

Proof. We know the unique minimizer of $f(x, y)$ is $(0, -1)$ by considering its equivalent form $h(\theta, r)$. For $h(\theta, r) = \left(1 - \frac{\sin(\theta)}{3}\right)(r^4 - 2r^2)$, we know $(r^4 - 2r^2)$ is minimized at $r = 1$ and $\left(1 - \frac{\sin(\theta)}{3}\right)$ is maximized at $\theta = \frac{3\pi}{2}$.

Besides the minimizer $(0, -1)$, the other stationary point is at $(0, 0)$. For any point (x, y) different from $(0, -1)$ and $(0, 0)$, if $x^2 + y^2 \neq 1$, the gradient along the radial direction is non-zero; if $\frac{y}{\sqrt{x^2+y^2}} \neq -1$, the gradient along the tangent direction is non-zero. It's also easy to verify that starting from a non-zero point, gradient flow does not converge to $(0, 0)$, so it must converge to $(0, -1)$ □

It's also very easy to prove that gradient descent with appropriate step size converges to an ϵ-neighborhood of the global minimizer within $\text{poly}(1/\epsilon)$ number of iterations. This is because the gradient is at least $\text{poly}(\epsilon)$ for any non-zero point outside of the ϵ-neighborhood of the global minimizer. Starting from an initialization (x, y) with $x^2 + y^2 = \Theta(1)$, the smoothness along the training is also bounded by a constant.

Next, we prove that starting from certain initialization [1], the loss interpolation between the initialization and the global minimizer is non-monotonic. We prove this

[1] Note the initialization condition in Prop. D.4 is satisfied with constant probability for a reasonable initialization scheme. For example, if we uniformly sample (x, y) from the set $S = \{(x, y) \in \mathbb{R}^2 | x^2 + y^2 \leqslant R\}$ with $R \geqslant 2$, the condition is satisfied with constant probability.

by identifying two points along the interpolation path such that the point closer to minimizer has a higher loss compared with the point further to the minimizer.

Proposition D.4. *Suppose we start from an initialization* $(x_0, y_0) = (r\sin(\beta), r\cos(\beta))$ *with* $r \geqslant 1$ *and* $\beta \in [-\pi/3, \pi/3]$. *Consider the loss interpolation curve* $\gamma(\alpha) = f((1-\alpha)x_0 + \alpha x^*, (1-\alpha)y_0 + \alpha y^*)$ *with* $(x^*, y^*) = (0, -1)$ *and* $f(\cdot, \cdot)$ *defined in Eqn. D.2. We know there exist* $0 \leqslant \alpha_1 < \alpha_2 \leqslant 1$ *such that*

$$\gamma(\alpha_2) - \gamma(\alpha_1) \geqslant \frac{5}{32}.$$

Proof. We prove for any $\beta \in [-\pi/3, \pi/3]$ and any $r \geqslant 1$, the loss interpolation between $(r\sin(\beta), r\cos(\beta))$ to $(0, -1)$ is non-monotonic. In particular, we show there are two points along the linear interpolation satisfying

$$f\left(\sin\left(\beta/2\right)\cos\left(\beta/2\right), -\sin\left(\beta/2\right)\sin\left(\beta/2\right)\right) - f\left(\sin(\beta), \cos(\beta)\right) \geqslant 1/12,$$

where $\left(\sin\left(\beta/2\right)\cos\left(\beta/2\right), -\sin\left(\beta/2\right)\sin\left(\beta/2\right)\right)$ is the middle point between $(\sin(\beta), \cos(\beta))$ and $(0, -1)$.

Next, we separately upper bound $f\left(\sin(\beta), \cos(\beta)\right)$ and lower bound $f\left(\sin\left(\beta/2\right)\cos\left(\beta/2\right), -\sin\left(\beta/2\right)\sin\left(\beta/2\right)\right)$. We have

$$\max_{\beta \in [-\pi/3, \pi/3]} f\left(\sin(\beta), \cos(\beta)\right) \leqslant f(0, 1) = -\frac{2}{3}$$

and

$$\begin{aligned}
&\min_{\beta \in [-\pi/3, \pi/3]} f\left(\sin\left(\beta/2\right)\cos\left(\beta/2\right), -\sin\left(\beta/2\right)\sin\left(\beta/2\right)\right) \\
\geqslant & f\left(\sin\left(\pi/6\right)\cos\left(\pi/6\right), -\sin\left(\pi/6\right)\sin\left(\pi/6\right)\right) \\
= & \left(1 + \frac{1}{2}\cdot\frac{1}{3}\right)\left(\left(\frac{1}{2}\right)^4 - 2\left(\frac{1}{2}\right)^2\right) \\
= & -\frac{49}{96}.
\end{aligned}$$

Therefore, we have $f\left(\sin\left(\beta/2\right)\cos\left(\beta/2\right), -\sin\left(\beta/2\right)\sin\left(\beta/2\right)\right) - f\left(\sin(\beta), \cos(\beta)\right) \geqslant \frac{5}{32}$. □

D.2 Proof for plateau and monotonicity

We first consider the r-homogeneous-weight model. We prove the plateau and monotonicity properties for the error interpolation (Theorem D.1) in Section D.2.1. We then prove the plateau and monotonicity properties for the loss interpolation (Theorem D.2) in Section D.2.2. Theorem 5.3 is a simple combination of Theorem D.1 and Theorem D.2. Finally, we give the plateau analysis for the fully-connected neural networks (Theorem 5.1) in Section D.2.3.

D.2.1 Error interpolation for r-homogeneous-weight model

Theorem D.1 (Error Interpolation)**.** *Suppose the network at initialization and after training satisfy the properties described in Theorem 5.2 and Induction Hypothesis 5.1. Suppose* $\delta \leqslant \min(O(1), O(R_{\min}^{\frac{1}{r-1}}\Delta_{\min}^{1/r}), O((\frac{W_{\min}}{W_{\max}})^{\frac{2r}{r-2}}))$. *There exist* $\alpha_1 = \frac{\delta}{\Delta_{\min}}$ *and* $\alpha_2 = (\frac{1}{1+O(\sqrt{\delta})})^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}}$, *such that*

1. *for all* $\alpha \in [\alpha_1, \alpha_2]$, *the error is* $1 - 1/k$;
2. *for all* $\alpha \in [\alpha_1, 1]$, *the error is non-increasing.*

Proof of Theorem D.1. This theorem directly follows from Lemma D.1 and Lemma D.2. □

Next, we separately prove the initial plateau in Lemma D.1 and the monotonicity in Lemma D.2.

Lemma D.1 (Error Plateau)**.** *In the same setting as in Theorem D.1, there exists* $\alpha_1 = \frac{\delta}{\Delta_{\min}}$ *and* $\alpha_2 = \left(\frac{1}{1+O(\sqrt{\delta})}\right)^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}}$, *such that for any interpolation point with*

$\alpha \in [\alpha_1, \alpha_2]$, *the error is* $1 - 1/k$. *Moreover, we have* $f_i^{[\alpha]}(e_j) < f_k^{[\alpha]}(e_j)$ *for all* $j \in [k]$ *and all* $i \neq k$.

In the proof of Lemma D.1, we show that for interpolation point $\alpha \in [\alpha_1, \alpha_2]$, the bias term dominates and all samples are classified as class k that has the largest bias.

Proof of Lemma D.1. We only need to show that for all $\alpha \in [\alpha_1, \alpha_2]$, we have

$$f_i^{[\alpha]}(x) < f_k^{[\alpha]}(x)$$

for all $x \in \mathcal{S}$ and all $i \neq k$, which immediately implies the error is $1 - 1/k$. Without loss of generality, assume $x \in \mathcal{S}_j$ where j may equal i or k.

For $\alpha \in \Big[\alpha_1, \frac{\sqrt{\delta}}{W_{\min}}\Big)$. If $\alpha_1 = \frac{\delta}{\Delta_{\min}} \geqslant \frac{\sqrt{\delta}}{W_{\min}}$, we only need to consider the case when $\alpha \in \Big[\frac{\sqrt{\delta}}{W_{\min}}, \alpha_2\Big]$. So here we assume $\frac{\delta}{\Delta_{\min}} < \frac{\sqrt{\delta}}{W_{\min}}$. We can lower bound $f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x)$ as

$$\begin{aligned}
&f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x) \\
=&\Big[\Big\langle W_{k,:}^{[\alpha]}, x\Big\rangle\Big]^r + b_k^{[\alpha]} - \Big[\Big\langle W_{i,:}^{[\alpha]}, x\Big\rangle\Big]^r - b_i^{[\alpha]} \\
=&\Big[\Big\langle W_{k,:}^{[\alpha]}, x\Big\rangle\Big]^r + b_k^{[\alpha]} - \Big[W_{i,j}^{[\alpha]} \pm O(\delta)\Big]^r - b_i^{[\alpha]} \\
\geqslant&\alpha\Delta_i - \Big[W_{i,j}^{(0)} + \alpha W_{i,j}^{(T)} + O(\delta)\Big]^r,
\end{aligned}$$

where the second equality uses $\Big|\Big\langle W_{i,:}^{[\alpha]}, \xi_x\Big\rangle\Big| \leqslant O(\delta)$ and the inequality uses $\Big\langle W_{k,:}^{[\alpha]}, x\Big\rangle \geqslant 0$.

To prove $f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x) > 0$ for $\alpha \in \Big[\frac{\delta}{\Delta_{\min}}, \frac{\sqrt{\delta}}{W_{\min}}\Big]$, we only need to prove $\frac{\delta}{\Delta_{\min}}\Delta_i - \Big[W_{i,j}^{(0)} + \frac{\sqrt{\delta}}{W_{\min}} W_{i,j}^{(T)} + O(\delta)\Big]^r > 0$. Since $\Delta_i \geqslant \Delta_{\min}$, we know $\frac{\delta}{\Delta_{\min}}\Delta_i \geqslant \delta$. Due to full accuracy, we know $\Big\langle W_{i,:}^{(T)}, x\Big\rangle \geqslant \Delta_i^{1/r}$ for $x \in \mathcal{S}_i$, which then implies

$W_{i,i}^{(T)} \geqslant \Omega\left(\Delta_i^{1/r}\right)$ because $\Delta_i \geqslant \Omega(1)$ and $\left\langle W_{i,:}^{(T)}, \xi_x \right\rangle \leqslant O(\delta) \leqslant O(1)$. Since $W_{i,i}^{(T)} \geqslant \Omega\left(\Delta_i^{1/r}\right)$ and $W_{i,j}^{(T)} \leqslant O(\delta)$ for $i \neq j$, so we have $W_{i,j}^{(T)} \leqslant W_{i,i}^{(T)} \leqslant W_{\max}$ as long as $\delta \leqslant O(\Delta_{\min}^{1/r})$. So we can upper bound $\left[W_{i,j}^{(0)} + \frac{\sqrt{\delta}}{W_{\min}} W_{i,j}^{(T)} + O(\delta)\right]^r$ as follows,

$$\begin{aligned}
\left[W_{i,j}^{(0)} + \frac{\sqrt{\delta}}{W_{\min}} W_{i,j}^{(T)} + O(\delta)\right]^r &\leqslant \left[O(\delta) + \frac{\sqrt{\delta} W_{\max}}{W_{\min}}\right]^r \\
&\leqslant \left[\frac{\sqrt{\delta} W_{\max}}{r W_{\min}} + \frac{\sqrt{\delta} W_{\max}}{W_{\min}}\right]^r \\
&\leqslant e\left(\frac{\sqrt{\delta} W_{\max}}{W_{\min}}\right)^r,
\end{aligned}$$

where the second inequality assumes $\delta \leqslant O\left(\frac{W_{\max}^2}{W_{\min}^2}\right)$. Therefore, to prove $\frac{\delta}{\Delta_{\min}}\Delta_i - \left[W_{i,j}^{(0)} + \frac{\sqrt{\delta}}{W_{\min}} W_{i,j}^{(T)} + O(\delta)\right]^r > 0$ we only need

$$\delta - e\left(\frac{\sqrt{\delta} W_{\max}}{W_{\min}}\right)^r > 0,$$

which holds as long as $\delta < \left[\frac{1}{e}\left(\frac{W_{\min}}{W_{\max}}\right)^r\right]^{\frac{2}{r-2}}$.

For $\alpha \in \left[\frac{\sqrt{\delta}}{W_{\min}}, \alpha_2\right]$. Similar as above, we only need to show that $\alpha\Delta_i - \left[W_{i,j}^{(0)} + \alpha W_{i,j}^{(T)} + O(\delta)\right]^r > 0$ for $i \neq k$ and $j \in [k]$. Since $W_{i,j}^{(0)} \leqslant O(\delta)$ and $\alpha \geqslant \sqrt{\delta}/W_{\min}$, we have $W_{i,j}^{(0)} \leqslant O(\sqrt{\delta}\alpha W_{\min})$. Therefore, we have $W_{i,j}^{(0)} + \alpha W_{i,j}^{(T)} + O(\delta) \leqslant \left(1 + O(\sqrt{\delta})\right) \alpha W_{i,i}^{(T)}$. Therefore, we have

$$\alpha\Delta_i - \left[W_{i,j}^{(0)} + \alpha W_{i,j}^{(T)} + O(\delta)\right]^r \geqslant \alpha\Delta_i - \left(1 + O(\sqrt{\delta})\right)^r \alpha^r \left[W_{i,i}^{(T)}\right]^r > 0,$$

where the last inequality assumes $\alpha \leqslant \alpha_2 := \left(\frac{1}{1+O(\sqrt{\delta})}\right)^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}}$ where $R_{\min} = \min_{i \in [k-1]} \Delta_i / [W_{i,i}^{(T)}]^r$. □

Next, we show that the error is non-increasing for $\alpha \in [\alpha_1, 1]$ by proving that once a sample is classified correctly it will remain so.

Lemma D.2 (Error Monotonicity)**.** *In the same setting as in Theorem D.1, there exists* $\delta_1 = \frac{\delta}{\Delta_{\min}}$ *such that the error is non-increasing for* $\alpha \in [\alpha_1, 1]$.

Proof of Lemma D.2. We first show that sample e_k is correctly classified for the whole range $[\alpha_1, 1]$. Second, we show for any other sample once it become classified right it will remain so. Combining these two cases, we prove the monotonicity of the error rate.

Class k. We first show that every $x \in \mathcal{S}_k$ is classified correctly for any $\alpha \in [\alpha_1, 1]$. According to Lemma D.1, we know that

$$f_k^{[\alpha_1]}(x) > f_i^{[\alpha_1]}(x)$$

for any $i \neq k$. We only need to prove that $f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x)$ is increasing for $\alpha \in [\alpha_1, 1]$. Expanding $f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x)$, we have

$$\begin{aligned}
&f_k^{[\alpha]}(x) - f_i^{[\alpha]}(x) \\
=&\left[(1-\alpha)\left\langle W_{k,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{k,:}^{(T)}, x\right\rangle\right)\right]^r - \left[(1-\alpha)\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^r \\
&+ \alpha\left(b_k^{(T)} - b_i^{(T)}\right),
\end{aligned}$$

which is increasing since $\left|\left\langle W_{k,:}^{(T)}, x\right\rangle\right|, \left|\left\langle W_{k,:}^{(0)}, x\right\rangle\right|, \left|\left\langle W_{i,:}^{(T)}, x\right\rangle\right|, \left|\left\langle W_{i,:}^{(0)}, x\right\rangle\right| \leqslant O(\delta)$ and $b_k^{(T)} - b_i^{(T)} > \Omega(1)$.

Other classes. For any class $i \neq k$, from Lemma D.1, we know that it is classified incorrectly for $\alpha \in [\alpha_1, \alpha_2]$. We prove that once it become classified correctly at some $\alpha' \in (\alpha_2, 1]$, it remains so for $\alpha \in [\alpha', 1]$.

We show that at α, for any $x \in \mathcal{S}_i$, if $f_i^{[\alpha]}(x) > f_j^{[\alpha]}(x)$ for all $j \neq i$, we have $\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)\right) > 0$. Expanding $f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)$, we have

$$\begin{aligned}
&f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x) \\
=&\left[\left\langle W_{i,:}^{[\alpha]}, x\right\rangle\right]^r + b_i^{[\alpha]} - \left[\left\langle W_{j,:}^{[\alpha]}, x\right\rangle\right]^r - b_j^{[\alpha]} \\
=&\left[\left\langle W_{i,:}^{[\alpha]}, x\right\rangle\right]^r - \left[\left\langle W_{j,:}^{[\alpha]}, x\right\rangle\right]^r - \alpha\left(b_j^{(T)} - b_i^{(T)}\right).
\end{aligned}$$

Since $f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x) > 0$, we have

$$\left[\left\langle W_{i,:}^{[\alpha]}, x\right\rangle\right]^r > \alpha\left(b_j^{(T)} - b_i^{(T)}\right),$$

where we use $\left\langle W_{j,:}^{[\alpha]}, x\right\rangle \geqslant 0$. Computing $\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)\right)$, we have

$$\begin{aligned}
&\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)\right) \\
=&\frac{\partial}{\partial \alpha}\left(\left[(1-\alpha)\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^r - \left[(1-\alpha)\left\langle W_{j,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{j,:}^{(T)}, x\right\rangle\right]^r\right) \\
&+ \frac{\partial}{\partial \alpha}\left(\alpha\left(b_i^{(T)} - b_j^{(T)}\right)\right) \\
\geqslant& r\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]^{r-1}\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right) \\
&- \left(b_j^{(T)} - b_i^{(T)}\right) - O(\delta^r),
\end{aligned}$$

where the inequality uses $\left|\left\langle W_{j,:}^{(0)}, x\right\rangle\right|, \left|\left\langle W_{j,:}^{(T)}, x\right\rangle\right| \leqslant O(\delta)$.

If $b_j^{(T)} - b_i^{(T)} \leqslant 0$, we only need to prove

$$\begin{aligned}
&r\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]^{r-1}\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right) - O(\delta^r) \\
&> 0,
\end{aligned}$$

which holds since $\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right), \left\langle W_{i,:}^{[\alpha]}, x\right\rangle \geqslant \Omega(1)$.

If $b_j^{(T)} - b_i^{(T)} > 0$, we have

$$
\begin{aligned}
&\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)\right) \\
=&r\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]^{r-1}\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right) \\
&-\left(b_j^{(T)} - b_i^{(T)}\right) - O(\delta^r) \\
>&\frac{(1-O(\delta^r))\, r\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)}{\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]} \cdot \alpha\left(b_j^{(T)} - b_i^{(T)}\right) - \left(b_j^{(T)} - b_i^{(T)}\right),
\end{aligned}
$$

where the last inequality uses $\left[\left\langle W_{i,:}^{[\alpha]}, x\right\rangle\right]^r > \alpha\left(b_j^{(T)} - b_i^{(T)}\right)$. Therefore, to prove $\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(e_i) - f_j^{[\alpha]}(e_i)\right) > 0$, we only need to prove $\frac{(1-O(\delta^r))r\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)}{\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]} \geqslant \frac{1}{\alpha}$. We have

$$
\begin{aligned}
&\frac{(1-O(\delta^r))\, r\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)}{\left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)\right]} \\
\geqslant&\frac{(1-O(\delta^r))\, r\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)}{2\alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)} \\
\geqslant&\frac{1}{\alpha}.
\end{aligned}
$$

The first inequality requires $\left\langle W_{i,:}^{(0)}, x\right\rangle \leqslant \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)$ and the second inequality uses $r \geqslant 3, (1 - O(\delta^r)) \geqslant 2/3$. To prove $\left\langle W_{i,:}^{(0)}, x\right\rangle \leqslant \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)$, it's equivalent to show $\left\langle W_{i,:}^{(0)}, x\right\rangle \leqslant \frac{\alpha}{1+\alpha}\left\langle W_{i,:}^{(T)}, x\right\rangle$. Since $\alpha \geqslant \alpha_2 = \left(\frac{1}{1+O(\sqrt{\delta})}\right)^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}}$, we can lower

bound $\frac{\alpha}{1+\alpha}$ as follows,

$$\begin{aligned}\frac{\alpha}{1+\alpha} \geqslant & \frac{1}{2}\left(\frac{1}{1+O(\sqrt{\delta})}\right)^{\frac{r}{r-1}} R_{\min}^{\frac{1}{r-1}} \\ \geqslant & \frac{1}{8} R_{\min}^{\frac{1}{r-1}},\end{aligned}$$

where the first inequality uses $\alpha \leqslant 1$ and the second inequality uses $1+O(\sqrt{\delta}) \leqslant 2, r \geqslant 2$. So we have $\frac{\alpha}{1+\alpha}\left\langle W_{i,:}^{(T)}, x\right\rangle \geqslant \frac{1}{8} R_{\min}^{\frac{1}{r-1}} \Delta_{\min}^{1/r}$. Therefore, we only need $\delta \leqslant O\left(R_{\min}^{\frac{1}{r-1}} \Delta_{\min}^{1/r}\right)$ to ensure that $\left\langle W_{i,:}^{(0)}, x\right\rangle \leqslant \alpha\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right)$. □

D.2.2 Loss interpolation for r-homogeneous-weight model

In this section, we give a proof of Theorem D.2.

Theorem D.2 (Loss Interpolation)**.** *Suppose the network at initialization and after training satisfy the properties described in Theorem 5.2 and Induction Hypothesis 5.1. For any $\epsilon \in (0,1)$, suppose $\delta \leqslant O(\epsilon^{1/r})$, there exist $\alpha_3 = \frac{\epsilon^{1/r}}{W_{\max}}$ and $\alpha_4 = (1+O(\delta))^{\frac{1}{r-1}}\left(\frac{R_{\max}}{r}\right)^{\frac{1}{r-1}}$ such that*

1. *for all $\alpha \in [0, \alpha_3]$, we have $\log k - e\epsilon \leqslant \frac{1}{N} L(W^{[\alpha]}, b^{[\alpha]}) \leqslant \log k + \alpha\Delta_{\max} + e\epsilon$;*
2. *for all $\alpha \in [\alpha_4, 1]$, the loss is monotonically decreasing.*

Proof of Theorem D.2. This theorem directly follows from Lemma D.3 and Lemma D.4. □

Next, we prove the initial loss plateau in Lemma D.3 and the monotonicity in Lemma D.4.

Lemma D.3 (Loss Plateau)**.** *In the same setting as in Theorem D.2, for any $\epsilon > 0$, there exists $\alpha_3 = \frac{\epsilon^{1/r}}{W_{\max}}$ such that for all $\alpha \in [0, \alpha_3]$*

$$N\left(\log k - e\epsilon\right) \leqslant L(W^{[\alpha]}, b^{[\alpha]}) \leqslant N\left(\log k + \alpha\Delta_{\max} + e\epsilon\right).$$

We show that for $\alpha \in [0, \alpha_3]$, the weights $W^{[\alpha]}$ is negligible and the bias dominates, which then gives a lower bound and an upper bound of the loss.

Proof of Lemma D.3. Since $\alpha \leqslant \alpha_3 = \frac{\epsilon^{1/r}}{W_{\max}}$ and $\delta \leqslant O\left(\epsilon^{1/r}\right)$, we have

$$\begin{aligned}\left[\left\langle W_{i,:}^{[\alpha]}, x\right\rangle\right]^r &= \left[\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle)\right]^r \\ &\leqslant \left[\left(1+\frac{1}{r}\right)\epsilon^{1/r}\right]^r \\ &\leqslant e\epsilon,\end{aligned}$$

for all $i \in [k], x \in \mathcal{S}$.

We can divide the dataset $\mathcal{S}$ into N/k disjoint subsets $\{P_l\}_{l=1}^{N/k}$ where each P_l contains exactly one sample from each class. Next, we bound the total loss of each subset P_l. Without loss of generality, let's consider subset P_1 and suppose $x^{(i)}$ is the i-th class sample in this subset. For convenience, we denote the total loss of samples in P_1 as $L_1(W^{[\alpha]}, b^{[\alpha]})$.

Lower bounding $L_1(W^{[\alpha]}, b^{[\alpha]})$. We have

$$\begin{aligned}L_1(W^{[\alpha]}, b^{[\alpha]}) &= \sum_{i\in[k]} \log\left(\frac{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)}{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}\right) \\ &= \log\left(\prod_{i\in[k]}\frac{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)}{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}\right) \\ &\geqslant \log\left(\left(\frac{k}{\sum_{i\in[k]}\frac{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)}}\right)^k\right),\end{aligned}$$

where the last inequality uses the HM-GM inequality. We can then upper bound $\sum_{i\in[k]} \frac{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)}$ as follows,

$$\begin{aligned}
\sum_{i\in[k]} \frac{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)} &= \sum_{i\in[k]} \frac{\exp\left(\left[\left\langle W_{i,:}^{[\alpha]}, x^{(i)}\right\rangle\right]^r + \alpha b_i\right)}{\sum_{j\in[k]}\exp\left(\left[\left\langle W_{j,:}^{[\alpha]}, x^{(i)}\right\rangle\right]^r + \alpha b_j\right)} \\
&\leqslant \sum_{i\in[k]} \frac{\exp\left(\alpha b_i + e\epsilon\right)}{\sum_{j\in[k]}\exp\left(\alpha b_j\right)} \\
&= \frac{\sum_{i\in[k]}\exp\left(\alpha b_i\right)}{\sum_{j\in[k]}\exp\left(\alpha b_j\right)}\exp\left(e\epsilon\right) \\
&= \exp\left(e\epsilon\right).
\end{aligned}$$

Plugging back to the lower bound of $L_1(W^{[\alpha]}, b^{[\alpha]})$, we have

$$L_1(W^{[\alpha]}, b^{[\alpha]}) \geqslant k\log\left(\frac{k}{\exp\left(e\epsilon\right)}\right) = k\left(\log k - e\epsilon\right).$$

Upper bounding $L_1(W^{[\alpha]}, b^{[\alpha]})$. We have

$$\begin{aligned}
L_1(W^{[\alpha]}, b^{[\alpha]}) &= \sum_{i\in[k]} \log\left(\frac{\sum_{j\in[k]}\exp\left(f_j^{[\alpha]}(x^{(i)})\right)}{\exp\left(f_i^{[\alpha]}(x^{(i)})\right)}\right) \\
&\leqslant \sum_{i\in[k]} \log\left(\frac{\sum_{j\in[k]}\exp\left(\alpha b_j + e\epsilon\right)}{\exp\left(\alpha b_i\right)}\right) \\
&\leqslant k\log\left(k\exp\left(\alpha\Delta_{\max} + e\epsilon\right)\right) \\
&\leqslant k\left(\log k + \alpha\Delta_{\max} + e\epsilon\right)
\end{aligned}$$

The above analysis applies for every subset P_l, so we have

$$N\left(\log k - e\epsilon\right) \leqslant L(W^{[\alpha]}, b^{[\alpha]}) \leqslant N\left(\log k + \alpha\Delta_{\max} + e\epsilon\right).$$

□

Next we show that when α is reasonably large, we have $f_i^{[\alpha]}(e_i) - f_j^{[\alpha]}(e_i)$ increasing for all $i \neq j$, which then implies that the loss is decreasing.

Lemma D.4 (Loss Monotonicity)**.** *In the same setting as in Theorem D.2, there exists* $\alpha_4 = (1+O(\delta))^{\frac{1}{r-1}} \left(\frac{R_{\max}}{r}\right)^{\frac{1}{r-1}}$ *such that the loss is monotonically decreasing for* $\alpha \in [\alpha_4, 1]$.

Proof of Lemma D.4. To prove that the loss is monotonically decreasing, we only need to show that for any $i \in [k]$ and any $x \in \mathcal{S}_i$, $f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)$ is monotonically increasing for $j \neq i$.

Same as in Lemma D.2, it's easy to prove that for $x \in \mathcal{S}_k$, $f_k(x) - f_j(x)$ with $j \neq k$ monotonically increases for $\alpha \in [0,1]$. So we focus on other classes.

For $i \neq k$, we show that $\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(x) - f_j^{[\alpha]}(x)\right) > 0$ for $x \in \mathcal{S}_i$ when $\alpha \geqslant \alpha_4$,

$$
\begin{aligned}
&\frac{\partial}{\partial \alpha}\left(f_i^{[\alpha]}(e_i) - f_j^{[\alpha]}(e_i)\right) \\
=&\frac{\partial}{\partial \alpha}\left(\left[(1-\alpha)\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^r - \left[(1-\alpha)\left\langle W_{j,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{j,:}^{(T)}, x\right\rangle\right]^r\right) \\
&+ \frac{\partial}{\partial \alpha}\left(\alpha\left(b_i^{(T)} - b_j^{(T)}\right)\right) \\
\geqslant& r\left[(1-\alpha)\left\langle W_{i,:}^{(0)}, x\right\rangle + \alpha\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^{r-1}\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right) \\
&- \left(b_k^{(T)} - b_i^{(T)}\right) - O(\delta^r) \\
\geqslant& r\alpha^{r-1}\left[\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^{r-1}\left(\left\langle W_{i,:}^{(T)}, x\right\rangle - \left\langle W_{i,:}^{(0)}, x\right\rangle\right) - \left(b_k^{(T)} - b_i^{(T)}\right) - O(\delta^r) \\
\geqslant& r\alpha^{r-1}\left(1 - O\left(\frac{\delta}{\Delta_{\min}^{1/r}}\right)\right)\left[\left\langle W_{i,:}^{(T)}, x\right\rangle\right]^r - \left(b_k^{(T)} - b_i^{(T)}\right)(1+O(\delta^r)) \\
>& 0,
\end{aligned}
$$

where the second last inequality uses $\left\langle W_{i,:}^{(0)}, x\right\rangle / \left\langle W_{i,:}^{(T)}, x\right\rangle \leqslant O\left(\delta/\Delta_{\min}^{1/r}\right)$. The last inequality requires

$$
r\alpha^{r-1} \geqslant (1+O(\delta))\frac{b_k^{(T)} - b_i^{(T)}}{\left[W_{i,i}^{(T)}\right]^r}
$$

which is satisfied as long as $\alpha \geqslant (1+O(\delta))^{\frac{1}{r-1}} \left(\frac{R_{\max}}{r}\right)^{\frac{1}{r-1}}$ where

$R_{\max} = \max_{i\in[k-1]} \Delta_i/[W_{i,i}^{(T)}]^r$.. □

D.2.3 Plateau for deep fully-connected networks

In this section, we consider fully-connected neural networks as defined in Section 5.3 and prove that both the error and loss curves have plateau. We restate Theorem 5.1 as follows.

Theorem 5.1. *Suppose the network is defined as in Equation* (5.1) *and suppose the weights satisfy* $\left\|V_i^{(0)}\right\| \leqslant \delta, \left\|V_i^{(T)}\right\| \leqslant V_{\max}$ *for all layers* $i \in [r]$. *On a* k*-class balanced dataset whose inputs have* ℓ_2 *norm at most* 1, *if Assumption 5.1 holds, for any* $\epsilon > 0$, *as long as* $\delta < \min\left(\frac{\epsilon^{1/r}}{r}, \frac{1}{r^2}, \left(\frac{1}{2e}\right)^{\frac{2}{r-2}}\right)$, *there exist* $\alpha_1 = \frac{\delta}{\Delta_{\min}}, \alpha_2 = \left(\frac{1}{1+\sqrt{\delta}}\right)^{\frac{r}{r-1}} \left(\frac{\Delta_{\min}}{2V_{\max}^r}\right)^{\frac{1}{r-1}}$ *and* $\alpha_3 = \frac{\epsilon^{1/r}}{V_{\max}}$ *such that*

1. *for all* $\alpha \in [\alpha_1, \alpha_2]$, *the error is* $1 - 1/k$;
2. *for all* $\alpha \in [0, \alpha_3]$, *we have*

$$\log k - 2e\epsilon \leqslant \frac{1}{N} L\left(\left\{V_i^{[\alpha]}\right\}, b^{[\alpha]}\right) \leqslant \log k + \alpha\Delta_{\max} + 2e\epsilon,$$

where N *is the number of training examples.*

Proof of Theorem 5.1. This theorem directly follows from Lemma D.5 and Lemma D.6. □

We separately prove the plateau of error interpolation in Lemma D.5 and the plateau of loss interpolation in Lemma D.6. Then, Theorem 5.1 is simply a combination of Lemma D.5 and Lemma D.6. For convenience, we denote $h(x) := V_r\sigma\left(V_{r-1}\cdots\sigma(V_1x)\cdots\right)$ in the proof.

Lemma D.5. *In the setting of Theorem 5.1, there exist* $\alpha_1 = \frac{\delta}{\Delta_{\min}}$ *and* $\alpha_2 = \left(\frac{1}{1+\sqrt{\delta}}\right)^{\frac{r}{r-1}} \left(\frac{\Delta_{\min}}{2V_{\max}^r}\right)^{\frac{1}{r-1}}$ *such that the error is* $1-1/k$ *for any interpolation point* $\alpha \in [\alpha_1, \alpha_2]$.

Proof of Lemma D.5. Recall that the network output under input x is $g(x) := V_r\sigma\left(V_{r-1}\cdots\sigma(V_1x)\cdots\right)+b$. Similar as in the proof of Lemma D.1, we only need to show that for all $\alpha \in [\alpha_1, \alpha_2]$, we have

$$g_i^{[\alpha]}(x) < g_k^{[\alpha]}(x)$$

for all $i \neq k$ and all samples x, which immediately implies the error is $1-1/k$.

For $\alpha \in \left[\alpha_1, \frac{\sqrt{\delta}}{V_{\max}}\right)$. If $\alpha_1 = \frac{\delta}{\Delta_{\min}} \geqslant \frac{\sqrt{\delta}}{V_{\max}}$, we only need to consider the case when $\alpha \in \left[\frac{\sqrt{\delta}}{V_{\max}}, \alpha_2\right]$. So here we assume $\frac{\delta}{\Delta_{\min}} < \frac{\sqrt{\delta}}{V_{\max}}$. We can lower bound $g_k^{[\alpha]}(x) - g_i^{[\alpha]}(x)$ as

$$\begin{aligned} g_k^{[\alpha]}(x) - g_i^{[\alpha]}(x) &= h_k^{[\alpha]}(x) + b_k^{[\alpha]} - h_i^{[\alpha]}(x) - b_i^{[\alpha]} \\ &\geqslant \alpha\Delta_{\min} - 2\prod_{j\in[r]}\left\|(1-\alpha)V_j^{(0)} + \alpha V_j^{(T)}\right\| \\ &\geqslant \alpha\Delta_{\min} - 2\left(\delta + \alpha V_{\max}\right)^r, \end{aligned}$$

where the first inequality holds because $b_k^{[\alpha]} - b_i^{[\alpha]} \geqslant \alpha\Delta_{\min}$ and $\left|h_k^{[\alpha]}(x)\right|, \left|h_i^{[\alpha]}(x)\right| \leqslant \prod_{j\in[r]}\left\|(1-\alpha)V_j^{(0)} + \alpha V_j^{(T)}\right\|$. The second inequality uses $\left\|(1-\alpha)V_j^{(0)} + \alpha V_j^{(T)}\right\| \leqslant (1-\alpha)\left\|V_j^{(0)}\right\| + \alpha\left\|V_j^{(T)}\right\| \leqslant \delta + \alpha V_{\max}$.

Since $\alpha \in \left[\frac{\delta}{\Delta_{\min}}, \frac{\sqrt{\delta}}{V_{\max}}\right)$, we have

$$\begin{aligned} g_k^{[\alpha]}(x) - g_i^{[\alpha]}(x) \geqslant & \frac{\delta}{\Delta_{\min}}\Delta_{\min} - 2\left(\delta + \frac{\sqrt{\delta}}{V_{\max}}V_{\max}\right)^r, \\ \geqslant & \delta - 2\left(\left(1+\frac{1}{r}\right)\sqrt{\delta}\right)^r, \\ \geqslant & \delta - 2e\delta^{r/2} \\ > & 0, \end{aligned}$$

where the second inequality assumes $\delta \leqslant 1/r^2$ and the last inequality assumes $\delta < \left(\frac{1}{2e}\right)^{\frac{2}{r-2}}$.

For $\alpha \in \left[\frac{\sqrt{\delta}}{V_{\max}}, \alpha_2\right]$. Similar as above, we only need to show that $\alpha\Delta_{\min} - 2\left(\delta + \alpha V_{\max}\right)^r > 0$. Since $\alpha \geqslant \frac{\sqrt{\delta}}{V_{\max}}$, we have $\delta \leqslant \sqrt{\delta}\alpha V_{\max}$. Therefore, we have

$$\alpha\Delta_{\min} - 2\left(\delta + \alpha V_{\max}\right)^r \geqslant \alpha\Delta_{\min} - 2\left(\left(1+\sqrt{\delta}\right)\alpha V_{\max}\right)^r > 0,$$

where the second inequality holds as long as $\alpha \leqslant \alpha_2 := \left(\frac{1}{2}\right)^{\frac{1}{r-1}}\left(\frac{1}{1+\sqrt{\delta}}\right)^{\frac{r}{r-1}}\left(\frac{\Delta_{\min}}{V_{\max}^r}\right)^{\frac{1}{r-1}}$.

□

Next, we show that for $\alpha \in [0, \frac{\epsilon^{1/r}}{V_{\max}}]$, the loss cannot decrease by much. Similar as in Lemma D.3, we prove that the signal is very small and the logit is dominated by the bias term. This then gives a lower and upper bounds for the loss.

Lemma D.6. *In the setting of Theorem 5.1, there exists* $\alpha_3 = \frac{\epsilon^{1/r}}{V_{\max}}$ *such that for all* $\alpha \in [0, \alpha_3]$

$$\log k - 2e\epsilon \leqslant \frac{1}{N}L\left(\left\{V_i^{[\alpha]}\right\}, b^{[\alpha]}\right) \leqslant \log k + \alpha\Delta_{\max} + 2e\epsilon,$$

where N *is the number of samples.*

Proof of Lemma D.6. Since $\alpha \leqslant \alpha_3 = \frac{\epsilon^{1/r}}{V_{\max}}$ and $\delta \leqslant \frac{\epsilon^{1/r}}{r}$, we have

$$\left\|h^{[\alpha]}(x)\right\| \leqslant (\delta + \alpha V_{\max})^r \leqslant e\epsilon$$

for all input x.

Similar as in the proof of Lemma D.3, we can show that

$$\log k - 2e\epsilon \leqslant \frac{1}{N} L\left(\left\{V_i^{[\alpha]}\right\}, b^{[\alpha]}\right) \leqslant \log k + \alpha\Delta_{\max} + 2e\epsilon,$$

where we have an additional factor of 2 before $e\epsilon$ because now the signal can be positive or negative. Here N is the number of samples. □

D.3 Proof of training dynamics

In this section, we give the complete proof of Theorem 5.2.

Theorem 5.2. *Suppose the neural network, dataset and optimization procedure are as defined in Section 5.2. Suppose initialization scale $\delta \leqslant \Theta(1)$, noise level $\sigma \leqslant \widetilde{\Theta}(1)$, dimension $d \geqslant \widetilde{\Theta}(1/\delta^{2r-2})$ and number of samples $N \geqslant \widetilde{\Theta}(1/\delta^{r-1})$, with probability at least 0.99 in the initialization, there exists time $T = \Theta(\log(1/\delta)/\delta^{r-2})$ such that we have*

1. *zero error: for all different $i, j \in [k]$ and for all $x \in \mathcal{S}_i$, $f_i^{(T)}(x) \geqslant f_j^{(T)}(x) + \Omega(1)$;*
2. *bias gap: $b_{i^*}^{(T)} - \max_{i \neq i^*} b_i^{(T)} \geqslant \Omega(1)$ with $i^* = \arg\max_{i \in [k]} b_i^{(T)}$.*

Proof of Theorem 5.2. This theorem directly follows from Proposition 5.1. □

Next, we prove Proposition 5.1 while leaving the proof of supporting lemmas into Section D.3.1. Through the proof of Proposition 5.1, we restate the lemmas when we use it for the convenience of readers.

Proposition 5.1 (Induction Hypothesis)**.** *In the same setting of Theorem 5.2, with probability at least 0.99 in initialization, there exist time points $0 =: s_1 < t_1 < s_2 <$*

$t_2 < \cdots < s_{k-1} < t_{k-1} < s_k := T$ *with* $t_i - s_i = \Theta(\log(1/\delta)/\delta^{r-2})$ *and* $s_{i+1} - t_i = \Theta(1)$ *for* $i \in [k-1]$ *such that for any* $t \in [s_i, s_{i+1}]$,

1. ***(classes not yet learned)*** *for any class* $j, j' \geqslant i+1$, *we have (1)* $b_j^{(t)} \geqslant \max_{i' \in [k]} b_{i'}^{(t)} - O(\delta^r)$, *(2)* $\left| b_j^{(t)} - b_{j'}^{(t)} \right| \leqslant O(\delta^r)$ *and (3)* $W_{j,j}^{(t)} \leqslant O(\delta)$;

2. ***(classes already learned)*** *for any class* $j \leqslant i-1$, *we have (1)* $b_j^{(t)} \leqslant \max_{i' \in [k]} b_{i'}^{(t)} - \Omega(1)$, *(2)* $f_j^{(t)}(x) \geqslant f_{i'}^{(t)}(x) + \Omega(1)$ *for* $i' \neq j, x \in \mathcal{S}_j$ *and (3)* $W_{j,j}^{(t)} \geqslant \Omega(1)$;

3. ***(parameters movement)*** *(1) for any* $j \in [k], \Theta(\delta) = W_{j,j}^{(0)} < W_{j,j}^{(t)}$, *(2) for any distinct* $j, j' \in [k], 0 < W_{j,j'}^{(t)} \leqslant O(\delta)$ *and (3) for any* $j, j' \in [k]$ *and any* $x \in \mathcal{S}_{j'}, \left| \left\langle W_{j,:}^{(t)}, \xi_x \right\rangle \right| \leqslant \min\left(O(\delta), W_{j,j'}^{(t)} \right)$.

Proof of Proposition 5.1. Through the proof, we assume the conditions in Theorem 5.2 hold in all the lemmas without explicitly stated. At the initialization, we have the following properties with probability at least 0.99.

Lemma D.7 (Initialization)**.** *With probability at least* 0.99 *in the initialization, we have*

1. *for all* $j, j' \in [k]$, $W_{j,j'}^{(0)} = \Theta(\delta)$;

2. *for all distinct* $j, j' \in [k]$, $\left| W_{j,j}^{(0)} - W_{j',j'}^{(0)} \right| = \Theta(\delta)$;

3. *for all* $x \in \mathcal{S}$, $\|\xi_x\| \leqslant O(\sigma)$;

4. *for all distinct* $x, x' \in \mathcal{S}$, $\left| \left\langle \bar{\xi}_x, \bar{\xi}_{x'} \right\rangle \right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}} \right)$.

5. *for all* $j \in [k]$ *and all* $x \in \mathcal{S}$, $\left| \left\langle \bar{\xi}_x, e_j \right\rangle \right|, \left| \left\langle \bar{\xi}_x, \bar{W}_{j,:}^{(0)} \right\rangle \right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}} \right)$.

Without loss of generality, we assume $W_{1,1}^{(0)} > W_{2,2}^{(0)} > \cdots > W_{k,k}^{(0)}$.

It's not hard to verify that the induction hypothesis holds at the initialization [2]. For any $i \in [k-1]$, assuming the induction hypothesis holds for time $[0, s_i]$, now we prove that it continues to hold in $[s_i, s_{i+1}]$. Next, we first prove the first two properties in the Proposition 5.1 and leave the last one at the end.

The learning of $x \in \mathcal{S}_i$ can be divided into four stages:

1. **Stage 0** for $t \in [s_i, t_i]$ with $t_i - s_i = O(\log(1/\delta)/\delta^{r-2})$. During this stage, $W_{i,i}^{(t)}$ grows to a small constant μ_0.

2. **Stage 1** for $t \in [t_i, t_i^{(w)}]$ with $t_i^{(w)} - t_i = O(1)$. In this stage, $W_{i,i}^{(t)}$ grows from μ_0 to a large constant μ_1.

3. **Stage 2** for $t \in [t_i^{(w)}, t_i^{(u)}]$ with $t_i^{(u)} - t_i^{(w)} = O(1)$. At the end of this stage, we have $\min_{x \in \mathcal{S}_i} u_i^{(t)}(x) \geqslant 1 - \mu_2$ for a small constant μ_2.

4. **Stage 3** for $t \in [t_i^{(u)}, t_i^{(b)}]$ with $t_i^{(b)} - t_i^{(u)} = O(1)$, where $t_i^{(b)} = s_{i+1}$. During this stage, we have $b_i^{(t)} - b_k^{(t)}$ decreases to $-\mu_3$ with μ_3 a positive constant.

Next, we consider these four stages one by one.

Stage 0. We show that $W_{i,i}^{(t)}$ increases faster than $W_{i+1,i+1}^{(t)}$ so that $W_{i,i}^{(t)}$ reaches a constant while $W_{i+1,i+1}^{(t)}$ is still $O(\delta)$. We use the following lemma to characterize the increasing rate of $W_{i+1,i+1}^{(t)}$ and $W_{i,i}^{(t)}$.

Lemma D.8. *For any $j \in [k]$, we have*

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \geqslant -O\left(\frac{\delta^{r-1}\sqrt{\log N}\sigma}{\sqrt{d}}\right).$$

[2] We will maintain a stronger bound on $\left|\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle\right|$ by proving $\left|\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle\right| \leqslant O(\sqrt{\log N}\sigma\delta)$, which implies $\left|\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle\right| \leqslant O(\delta)$ as long as $\sigma \leqslant O(1/\sqrt{\log N})$.

If $\min_{x\in\mathcal{S}_j}(1-u_j(x)) \geqslant \Omega(1)$*, we further have*

$$\left(1-O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,rW_{j,j}^{r-1} \leqslant \frac{\partial W_{j,j}^{(t)}}{\partial t}.$$

and

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \leqslant \left(1+O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,rW_{j,j}^{r-1}.$$

It's not hard to verify that $\min_{x\in\mathcal{S}_i}\left(1-u_i^{(t)}(x)\right), \min_{x\in\mathcal{S}_{i+1}}\left(1-u_{i+1}^{(t)}(x)\right) \geqslant \Omega(1)$, so we have

$$\frac{\partial W_{i,i}^{(t)}}{\partial t} \geqslant \left(1-O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_i}\left(1-u_i^{(t)}(x)\right)r\left[W_{i,i}^{(t)}\right]^{r-1},$$

$$\frac{\partial W_{i+1,i+1}^{(t)}}{\partial t} \leqslant \left(1+O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_{i+1}}\left(1-u_{i+1}^{(t)}(x)\right)r\left[W_{i+1,i+1}^{(t)}\right]^{r-1}.$$

We can upper bound $1-u_{i+1}^{(t)}(x)$ for any $x\in\mathcal{S}_{i+1}$ as follows,

$$\begin{aligned}1-u_{i+1}^{(t)}(x) =& \frac{\sum_{i'\in[k]}\exp\left(f_{i'}^{(t)}(x)\right)-\exp\left(f_{i+1}^{(t)}(x)\right)}{\sum_{i'\in[k]}\exp\left(f_{i'}^{(t)}(x)\right)}\\ \leqslant& \frac{\sum_{i'\in[k]}\exp\left(b_{i'}^{(t)}\right)-\exp\left(b_{i+1}^{(t)}\right)}{\sum_{i'\in[k]}\exp\left(b_{i'}^{(t)}\right)}\left(1+O(\delta^r)\right),\end{aligned}$$

where the inequality uses $\left|\left\langle W_{i',:}^{(t)}, x\right\rangle\right| \leqslant O(\delta)$ for every $i'\in[k]$.

We can lower bound $1 - u_i^{(t)}(x)$ for $x \in \mathcal{S}_i$ as follows,

$$\begin{aligned}
1 - u_i^{(t)}(x) =& \frac{\sum_{i' \in [k]} \exp\left(f_{i'}^{(t)}(x)\right) - \exp\left(f_i^{(t)}(x)\right)}{\sum_{i' \in [k]} \exp\left(f_{i'}^{(t)}(x)\right)} \\
\geqslant& \frac{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right) - \exp\left(b_i^{(t)}\right)}{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right)} \left(1 - O(\mu_0^r)\right) \\
\geqslant& \frac{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right) - \exp\left(b_{i+1}^{(t)}\right)}{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right)} \left(1 - O(\mu_0^r) - O(\delta^r)\right).
\end{aligned}$$

The first inequality uses $\left|\left\langle W_{i',:}^{(t)}, x \right\rangle\right| \leqslant O(\mu_0)$ for every $i' \in [k]$. The second inequality uses $b_i^{(t)} \leqslant b_{i+1}^{(t)} + O(\delta^r)$, which is guaranteed by the following lemma.

Lemma 5.2 (Bias Gap Control I)**.** *For any different $j', j \in [k]$, if $W_{j',j'} \geqslant W_{j,j}, W_{j,j} \leqslant O(\delta)$ and $b_{j'} - b_j \geqslant O(\delta^r), b_j \geqslant \max_{i' \in [k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j < 0$.*

According to Lemma D.7, we know there exists constant $C > 1$ such that $W_{i,i}^{(0)} \geqslant CW_{i+1,i+1}^{(0)}$. Choose constant S such that $S^{\frac{1}{r-2}} = \sqrt{C}$ and $W_{i,i}^{(0)} \geqslant S^{\frac{1}{r-2}}\sqrt{C}W_{i+1,i+1}^{(0)}$. Choosing μ_0 as small constants and $\sigma \leqslant O(1/\sqrt{\log N}), \delta \leqslant O(1)$, we have

$$\left(1 + O(\sqrt{\log N}\sigma)\right) \max_{x \in \mathcal{S}_{i+1}} \left(1 - u_{i+1}^{(t)}(x)\right) \leqslant S \left(1 - O(\sqrt{\log N}\sigma)\right) \min_{x \in \mathcal{S}_i} \left(1 - u_i^{(t)}(x)\right).$$

We can also lower bound $1 - u_i^{(t)}(x)$ for $x \in \mathcal{S}_i$ by a constant,

$$\begin{aligned}
1 - u_i^{(t)}(x) \geqslant& \frac{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right) - \exp\left(b_i^{(t)}\right)}{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right)} \left(1 - O(\mu_0^r)\right) \\
\geqslant& \frac{\exp\left(b_{i+1}^{(t)}\right)}{\sum_{i' \in [k]} \exp\left(b_{i'}^{(t)}\right)} \left(1 - O(\mu_0^r)\right) \\
\geqslant& \Omega(1),
\end{aligned}$$

where the last inequality holds because μ_0 is a small constant and $b_{i+1}^{(t)} \geqslant \max_{i' \in [k]} b_{i'}^{(t)} - O(\delta^r)$.

Lemma D.9 (Adapted from Lemma C.19 in Allen-Zhu and Li (2020b)). *Let $r \geqslant 3$ be a constant and let $\{W_{i,i}^{(t)}, W_{j,j}^{(t)}\}_{t \geqslant 0}$ be two positive sequences updated as*

$$\frac{\partial W_{i,i}^{(t)}}{\partial t} \geqslant C_t \left[W_{i,i}^{(t)}\right]^{r-1} \text{ for some } C_t = \Theta(1),$$

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \leqslant S C_t \left[W_{j,j}^{(t)}\right]^{r-1} \text{ for some } S = \Theta(1).$$

Suppose $W_{i,i}^{(0)} \geqslant W_{j,j}^{(0)} S^{\frac{1}{r-2}} \left(1 + \Omega(1)\right)$, then we must have for every $A = O(1)$, let t_i be the first time such that $W_{i,i}^{(t_i)} \geqslant A$, then

$$W_{j,j}^{(t_i)} \leqslant O(W_{j,j}^{(0)}).$$

Then, according to Lemma D.9, we know that there exists $t_i = O(\log(1/\delta)/\delta^{r-2})$ such that $W_{i,i}^{(t_i)} = \mu_0$ and $W_{i+1,i+1}^{(t_i)} \leqslant O(\delta)$. By similar argument, we also know $W_{j,j}^{(t_i)} \leqslant O(\delta)$ for any $j \geqslant i+1$.

Stage 1. In this stage, we show that $W_{i,i}^{(t)}$ grows to a large constant μ_1 within constant time. Since $W_{i,i}^{(t)} \leqslant \mu_1$ and $b_{i,i}^{(t)} \leqslant b_{i+1,i+1}^{(t)} + O(\delta^r)$, we have

$$1 - u_i^{(t)}(x) \geqslant \Omega(1),$$

for all $x \in \mathcal{S}_i$. This further implies,

$$\begin{aligned}\frac{\partial W_{i,i}^{(t)}}{\partial t} \geqslant & \left(1 - O(\sqrt{\log N}\sigma)\right) \frac{k}{N} \sum_{x \in \mathcal{S}_i} \left(1 - u_i^{(t)}(x)\right) r \left[W_{i,i}^{(t)}\right]^{r-1} \\ \geqslant & \Omega(1),\end{aligned}$$

where the inequality also uses $W_{i,i}^{(t)} \geqslant \mu_0$. Since the increasing rate is at least a constant, we know $W_{i,i}^{(t)}$ grows to μ_1 in constant time. For any $j \geqslant i+1$, since the

increasing rate of $W_{j,j}^{(t)}$ is merely $O(\delta^{r-1})$, we know $W_{j,j}^{(t)}$ remains as $O(\delta)$ through Stage 1.

Stage 2. In this stage, we prove that $u_i^{(t)}(x)$ for any $x \in \mathcal{S}_i$ grows to $1-\mu_2$ with μ_2 a small constant. We use the following lemma to characterize the increasing rate of $f_i^{(t)}(x) - f_j^{(t)}(x)$.

Lemma D.10. *For any $x \in \mathcal{S}_i$ and any $j \neq i$, if $1 - u_i(x) \geqslant \Omega(1)$, we have*

$$\frac{\partial}{\partial t}\left(f_i(x) - f_j(x)\right) \geqslant \Omega(W_{i,i}^{2r-2}) - O(1).$$

Since $u_i^{(t)}(x) \leqslant 1-\mu_2$, we know $1 - u_i^{(t)}(x) \geqslant \Omega(1)$. For any $j \neq i$, we have

$$\begin{aligned}\frac{\partial}{\partial t}\left(f_i^{(t)}(x) - f_j^{(t)}(x)\right) \geqslant & \Omega\left(\left[W_{i,i}^{(t)}\right]^{2r-2}\right) - O(1) \\ \geqslant & \Omega(1),\end{aligned}$$

where the last inequality holds because $W_{i,i}^{(t)} \geqslant \mu_1$ with μ_1 a large enough constant.

The next lemma guarantees that at the beginning of Stage 2, we have $b_i^{(t)} - b_j^{(t)} \geqslant -O(1)$, which then implies $f_i^{(t)}(x) - f_j^{(t)}(x) \geqslant -O(1)$.

Lemma D.11 (Bias Gap Control III)**.** *For any different $i, j \in [k]$, if $W_{i,i} \leqslant O(1)$, $W_{j,j} \leqslant O(\delta)$ and $b_i - b_j \leqslant -O(1)$, we have*

$$\dot{b}_i - \dot{b}_j > 0.$$

Let C be a constant such that $f_i^{(t)}(x) - f_j^{(t)}(x) \geqslant C$ for every $j \neq i$ implies $u_i(x) \geqslant 1-\mu_2$. Since at the at the beginning of Stage 2, we have $f_i^{(t)}(x) - f_j^{(t)}(x) \geqslant -O(1)$, within constant time, we have $f_i^{(t)}(x) - f_j^{(t)}(x) \geqslant C$ for every $j \neq i$ and $u_i(x) \geqslant 1-\mu_2$.

Lemma D.12 (Accuracy Monotonicity)**.** *Given any positive constant C_2, there exists positive constant C_1 such that for all different $i, j \in [k]$, as long as $W_{i,i} \geqslant C_1$ and $f_i(x) - f_j(x) \leqslant C_2$ for any $x \in \mathcal{S}_i$, we have $\frac{\partial(f_i(x) - f_j(x))}{\partial t} > 0$.*

According to Lemma D.12, by choosing large enough μ_1, we can ensure that $f_i^{(t)}(e_i) - f_j^{(t)}(e_i) \geqslant C$ and $u_i(e_i) \geqslant 1 - \mu_2$ throughout the training. Note that once $W_{i,i}^{(t)}$ rises to μ_1, it will stay at least $\mu_1 - O(\delta)$ throughout the training, according to the gradient lower bound in Lemma D.8.

Stage 3. In this stage, we prove that within constant time we have $b_i^{(t)} - b_k^{(t)} \leqslant -\mu_3$. The following lemma shows that $b_i^{(t)} - b_k^{(t)}$ decreases in at least a constant rate.

Lemma 5.3 (Bias Gap Control II)**.** *There exist small positive constants C_1, C_2 such that for any $j \in [k-1]$ and any $x \in \mathcal{S}_j$, if $1 - u_j(x) \leqslant C_1, W_{k,k} \leqslant O(\delta)$ and $b_j - b_k \geqslant -C_2$, we have $\dot{b}_j - \dot{b}_k < -\Omega(1)$.*

Choosing $\mu_2 = C_1, \mu_3 = C_2$ where C_1, C_2 are from Lemma 5.3, so we know that $b_i^{(t)} - b_k^{(t)}$ decreases at a constant rate until $b_i^{(t)} - b_k^{(t)} \leqslant -\mu_3$. At time $t_i^{(u)}$, we know that $b_i^{t_i^{(u)}} - b_k^{t_i^{(u)}} \leqslant O(\delta^r)$. So within constant time, we have $b_i^{s_{i+1}} - b_k^{s_{i+1}} = -\mu_3$. By Lemma 5.3, we also know that for any $t \geqslant s_{i+1}$, we have $b_i^{(t)} - b_k^{(t)} \leqslant -\mu_3$.

The following lemma shows that $b_k^{(t)}$ is close to the maximum bias.

Lemma 5.1 (Coupling Biases)**.** *Assuming $W_{j',j'}, W_{j,j} \leqslant O(\delta)$ and $b_{j'}, b_j \geqslant \max_{i' \in [k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j > 0$ if $b_{j'} - b_j \leqslant -\mu\delta^r$, and $\dot{b}_{j'} - \dot{b}_j < 0$ if $b_{j'} - b_j \geqslant +\mu\delta^r$ for some positive constant μ.*

Combining Lemma 5.1 and Lemma 5.2, we know that throughout the training $b_k^{(t)} \geqslant \max_{i' \in [k]} b_{i'}^{(t)} - O(\delta^r)$. Therefore, we have $b_i^{(t)} - \max_{i' \in [k]} b_{i'}^{(t)} \leqslant -\Omega(1)$ for $t \geqslant s_{i+1}$.

Finally, let's bound the movement of different parameters.

Monotonicity of diagonal terms: For $j \in [k-1]$, according to Lemma D.8 we know $W_{j,j}^{(t)}$ can only start decreasing when it exceeds a large constant and can only decrease by at most $O(\delta)$ through the algorithm. By choosing $\delta \leqslant O(1)$, we can ensure that $W_{j,j}^{(0)} < W_{j,j}^{(t)}$ for any t. For $W_{k,k}^{(t)}$, we know it monotonically increases

since we always have $1 - u_k^{(t)}(x) \geqslant \Omega(1)$ for $x \in \mathcal{S}_k$. This is because $W_{k,k}^{(t)}$ remains as small as $O(\delta)$ through the algorithm and $b_k^{(t)} - b_{k-1}^{(t)} \leqslant O(1)$.

Bounding non-diagonal terms: We use the following lemma to prove that $\widetilde{\Omega}(\delta) < W_{j,j'}^{(t)} \leqslant O(\delta)$ for $j \neq j'$.

Lemma D.13. *For any $j \neq j'$, we have $T\dot{W}_{j,j'} \leqslant O(\delta)$. Furthermore, there exists absolute constant $\mu > 0$ such that if $0 < W_{j,j'} < \frac{\mu\delta}{\log^{\frac{1}{r-2}}(1/\delta)}$, we have $T\dot{W}_{j,j'} \geqslant -\frac{\mu\delta}{2\log^{\frac{1}{r-2}}(1/\delta)}$.*

The first property in Lemma D.13 guarantees that the increasing rate is so small that that the total increase within T time is only $O(\delta)$, which then implies that $W_{j,j'}^{(t)} \leqslant O(\delta)$ through the training. The second property in Lemma D.13 guarantees that once $W_{j,j'}^{(t)}$ falls below $\frac{\mu\delta}{\log^{\frac{1}{r-2}}(1/\delta)}$, its decreasing rate is so small that that the total decrease within T time is only $\frac{\mu\delta}{2\log^{\frac{1}{r-2}}(1/\delta)}$, which then implies that $W_{j,j'}^{(t)} > \widetilde{\Omega}(\delta)$ through the training.

Bounding noise correlations: The following lemma shows that the total change of $\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle$ within T time is only $O(\sqrt{\log N}\sigma\delta)$. Since at initialization, we know $\left|\left\langle W_{j,:}^{(0)}, \xi_x \right\rangle\right| \leqslant O(\sqrt{\log N}\sigma\delta)$, we conclude that $\left|\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle\right| \leqslant O(\sqrt{\log N}\sigma\delta)$ throughout the training. Since $W_{j,j'}^{(t)} \geqslant \widetilde{\Omega}(\delta)$, as long as $\sigma \leqslant \widetilde{O}(1)$, we also have $\left|\left\langle W_{j,:}^{(t)}, \xi_x \right\rangle\right| \leqslant W_{j,j'}$ for $x \in \mathcal{S}_{j'}$.

Lemma D.14. *For every $j \in [k]$ and every $x \in \mathcal{S}$, we have*

$$\left|\left\langle \dot{W}_{j,:}, \xi_x \right\rangle\right| \cdot T \leqslant O\left(\sqrt{\log N}\sigma\delta\right)$$

□

D.3.1 Proof of lemmas

Lemma D.7 (Initialization)**.** *With probability at least* 0.99 *in the initialization, we have*

1. *for all* $j, j' \in [k]$, $W^{(0)}_{j,j'} = \Theta(\delta)$;
2. *for all distinct* $j, j' \in [k]$, $\left|W^{(0)}_{j,j} - W^{(0)}_{j',j'}\right| = \Theta(\delta)$;
3. *for all* $x \in \mathcal{S}$, $\|\xi_x\| \leqslant O(\sigma)$;
4. *for all distinct* $x, x' \in \mathcal{S}$, $\left|\left\langle \bar{\xi}_x, \bar{\xi}_{x'} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}}\right)$.
5. *for all* $j \in [k]$ *and all* $x \in \mathcal{S}$, $\left|\left\langle \bar{\xi}_x, e_j \right\rangle\right|, \left|\left\langle \bar{\xi}_x, \bar{W}^{(0)}_{j,:} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}}\right)$.

Without loss of generality, we assume $W^{(0)}_{1,1} > W^{(0)}_{2,2} > \cdots > W^{(0)}_{k,k}$.

Proof of Lemma D.7. Recall that each $W^{(0)}_{j,j'}$ is independently sampled from $\mathcal{N}(0, \delta^2)$ before taking the absolute value. By standard Gaussian concentration inequality, we know for any $j, j' \in [k]$, with probability at least $1 - \frac{1}{1000k^2}$,

$$W^{(0)}_{j,j'} \leqslant O(\delta).$$

By anti-concentration inequality of Gaussian polynomials, we know for any $j, j' \in [k]$, with probability at least $1 - \frac{1}{1000k^2}$,

$$W^{(0)}_{j,j'} \geqslant \Omega(\delta).$$

Also by anti-concentration inequality of Gaussian polynomials, we know for any distinct $j, j' \in [k]$, with probability at least $1 - \frac{1}{1000k^2}$,

$$\left|\left[W^{(0)}_{j,j}\right]^2 - \left[W^{(0)}_{j',j'}\right]^2\right| \geqslant \Omega(\delta^2),$$

which implies $\left|W_{j,j}^{(0)} - W_{j',j'}^{(0)}\right| \geqslant \Omega(\delta)$ assuming $W_{j,j}^{(0)}, W_{j',j'}^{(0)} = \Theta(\delta)$.

By the norm concentration of random vectors with independent Gaussian entries, for each $x \in \mathcal{S}$, we have with probability at least $1 - \frac{1}{1000N^2}$,

$$\|\xi_x\| \leqslant O(\sigma)$$

as long as $d \geqslant O(\log N)$.

By the concentration of standard Gaussian variable, for any distinct $x, x' \in \mathcal{S}$, we have with probability at least $1 - \frac{1}{1000N^2}$,

$$\left|\left\langle \bar{\xi}_x, \bar{\xi}_{x'} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log N}}{\sqrt{d}}\right).$$

Similarly, for any x and any e_j, we have with probability at least $1 - \frac{1}{1000kN}$,

$$\left|\left\langle \bar{\xi}_x, e_j \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log N}}{\sqrt{d}}\right);$$

for any x and any $\bar{W}_{j,:}^{(0)}$, we have with probability at least $1 - \frac{1}{1000kN}$,

$$\left|\left\langle \bar{\xi}_x, \bar{W}_{j,:}^{(0)} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log N}}{\sqrt{d}}\right);$$

Taking a union bound over all these events, we know with probability at least 0.99 in the initialization, we have

1. for all $j, j' \in [k]$, $W_{i,j}^{(0)} = \Theta(\delta)$;
2. for all distinct $j, j' \in [k]$, $\left|W_{j,j}^{(0)} - W_{j',j'}^{(0)}\right| = \Theta(\delta)$;
3. for all $x \in \mathcal{S}$, $\|\xi_x\| \leqslant O(\sigma)$;
4. for all distinct $x, x' \in \mathcal{S}$, $\left|\left\langle \bar{\xi}_x, \bar{\xi}_{x'} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}}\right)$.
5. for all $j \in [k]$ and all $x \in \mathcal{S}$, $\left|\left\langle \bar{\xi}_x, e_j \right\rangle\right|, \left|\left\langle \bar{\xi}_x, \bar{W}_{j,:}^{(0)} \right\rangle\right| \leqslant O\left(\frac{\sqrt{\log(N)}}{\sqrt{d}}\right)$.

□

Lemma D.9 (Adapted from Lemma C.19 in Allen-Zhu and Li (2020b)). *Let $r \geqslant 3$ be a constant and let $\{W_{i,i}^{(t)}, W_{j,j}^{(t)}\}_{t\geqslant 0}$ be two positive sequences updated as*

$$\frac{\partial W_{i,i}^{(t)}}{\partial t} \geqslant C_t \left[W_{i,i}^{(t)}\right]^{r-1} \text{ for some } C_t = \Theta(1),$$

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \leqslant S C_t \left[W_{j,j}^{(t)}\right]^{r-1} \text{ for some } S = \Theta(1).$$

Suppose $W_{i,i}^{(0)} \geqslant W_{j,j}^{(0)} S^{\frac{1}{r-2}} (1 + \Omega(1))$, then we must have for every $A = O(1)$, let t_i be the first time such that $W_{i,i}^{(t_i)} \geqslant A$, then

$$W_{j,j}^{(t_i)} \leqslant O(W_{j,j}^{(0)}).$$

Proof of Lemma D.9. This lemma directly follows from Lemma C.19 in Allen-Zhu and Li (2020b) by taking the continuous time limit and setting k as a constant. □

Lemma 5.1 (Coupling Biases). *Assuming $W_{j',j'}, W_{j,j} \leqslant O(\delta)$ and $b_{j'}, b_j \geqslant \max_{i'\in[k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j > 0$ if $b_{j'} - b_j \leqslant -\mu\delta^r$, and $\dot{b}_{j'} - \dot{b}_j < 0$ if $b_{j'} - b_j \geqslant +\mu\delta^r$ for some positive constant μ.*

Proof of Lemma 5.1. Let's first write down the time derivative on $b_{j'}$,

$$\begin{aligned}\dot{b}_{j'} =& 1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_{j'}(x) \\ =& 1 - \frac{k}{N}\sum_{x\in\mathcal{S}} \frac{\exp\left(\langle W_{j',:}, x\rangle^r + b_{j'}\right)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)}\end{aligned}$$

For any $x \in \mathcal{S}$, we can bound $\frac{\exp\left(\langle W_{j',:}, x\rangle^r + b_{j'}\right)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)}$ as follows,

$$\left|\frac{\exp\left(\langle W_{j',:}, x\rangle^r + b_{j'}\right)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)} - \frac{\exp\left(b_{j'}\right)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)}\right| \leqslant O(\delta^r),$$

where we uses $|\langle W_{j',:}, x\rangle| \leqslant O(\delta) + O(\sqrt{\log N}\sigma\delta) \leqslant O(\delta)$ assuming $\sigma \leqslant 1/\sqrt{\log N}$. The similar bound also holds for $\frac{\exp(\langle W_{j,:},x\rangle^r + b_j)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:},x\rangle^r + b_{i'}\right)}$

If $b_{j'} - b_j \geqslant \mu\delta^r$, we can now upper bound $\dot{b}_{j'} - \dot{b}_j$ as follows,

$$\begin{aligned}
\dot{b}_{j'} - \dot{b}_j \leqslant & \frac{k}{N}\sum_{x\in\mathcal{S}} \frac{\exp(b_j) - \exp(b_{j'})}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)} + O(\delta^r) \\
\leqslant & \frac{k}{N}\sum_{x\in\mathcal{S}_j\cup\mathcal{S}_{j'}} \frac{\exp(b_j) - \exp(b_{j'})}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)} + O(\delta^r) \\
\leqslant & -\Omega(\mu\delta^r)\cdot\frac{k}{N}\sum_{x\in\mathcal{S}_j\cup\mathcal{S}_{j'}} \frac{\exp(b_j)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)} + O(\delta^r)
\end{aligned}$$

When $x \in \mathcal{S}_j \cup \mathcal{S}_{j'}$, we can lower bound $\frac{\exp(b_j)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:},x\rangle^r + b_{i'}\right)}$ as follows,

$$\begin{aligned}
\frac{\exp(b_j)}{\sum_{i'\in[k]}\exp\left(\langle W_{i',:}, x\rangle^r + b_{i'}\right)} = & \frac{\exp(b_j)}{\sum_{i'\in[k]}\exp(b_{i'})\exp\left(\langle W_{i',:}, x\rangle^r\right)} \\
\geqslant & \frac{\exp(b_j)}{\sum_{i'\in[k]}\exp(b_{i'})}\cdot\frac{1}{1+O(\delta^r)} \\
\geqslant & \Omega(1),
\end{aligned}$$

where the first inequality uses $|\langle W_{i',:}, x\rangle| \leqslant \delta$ and the second inequality assumes $b_j \geqslant \max_{i'\in[k]} b_{i'} - O(\delta^r)$ and δ is at most some small constant.

Therefore, if $b_{j'} - b_j \geqslant \mu\delta^r$, we have

$$\dot{b}_{j'} - \dot{b}_j \leqslant -\Omega(\mu\delta^r) + O(\delta^r) < 0,$$

where the second inequality chooses μ as a large enough constant. Similarly, we can prove that if $b_{j'} - b_j \leqslant -\mu\delta^r$, we have

$$\dot{b}_{j'} - \dot{b}_j \geqslant \Omega(\mu\delta^r) - O(\delta^r) > 0.$$

□

Lemma 5.2 (Bias Gap Control I)**.** *For any different $j', j \in [k]$, if $W_{j',j'} \geqslant W_{j,j}, W_{j,j} \leqslant O(\delta)$ and $b_{j'} - b_j \geqslant O(\delta^r), b_j \geqslant \max_{i'\in[k]} b_{i'} - O(\delta^r)$, we have $\dot{b}_{j'} - \dot{b}_j < 0$.*

Proof of Lemma 5.2. We can write down $\dot{b}_{j'} - \dot{b}_j$ as follows,

$$\begin{aligned}\dot{b}_{j'} - \dot{b}_j =& \left(1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_{j'}(x)\right) - \left(1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_j(x)\right) \\ =& \frac{k}{N}\sum_{x\in\mathcal{S}_{j'}} (u_j(x) - u_{j'}(x)) + \frac{k}{N}\sum_{x\in\mathcal{S}\backslash\mathcal{S}_{j'}} (u_j(x) - u_{j'}(x)).\end{aligned}$$

We first prove that for any $x \in \mathcal{S}_{j'}$, we have $u_j(x) - u_{j'}(x) \leqslant 0$. We can upper bound $f_j(x)$ and lower bound $f_{j'}(x)$ as follows,

$$\begin{aligned}f_j(x) =& \langle W_{j,:}, x\rangle^r + b_j \leqslant O(\delta^r) + b_j \\ f_{j'}(x) =& \langle W_{j',:}, x\rangle^r + b_{j'} \geqslant b_j.\end{aligned}$$

The bound on $f_j(x)$ holds because $\langle W_{j,:}, x\rangle = W_{j,j'} + \langle W_{j,:}, \xi_x\rangle \leqslant O(\delta) + O(\sqrt{\log N}\sigma\delta)$ $\leqslant O(\delta)$. The bound on $f_{j'}(x)$ holds because $\langle W_{j',:}, x\rangle = W_{j',j'} + \langle W_{j',:}, \xi_x\rangle \geqslant \Omega(\delta) - O(\sqrt{\log N}\sigma\delta) > 0$. With the above two bounds, we know that $u_j(x) - u_{j'}(x) \leqslant 0$ as long as $b_{j'} - b_j \geqslant O(\delta^r)$.

Same as in the proof of Lemma 5.1, for each $x \in \mathcal{S}\backslash\mathcal{S}_{j'}$, we can bound $u_{j'}(x), u_j(x)$ as follows,

$$\frac{\exp(b_{j'})}{\sum_{i'\in[k]}\exp(f_{i'}(x))} - O(\delta^r) \leqslant u_{j'}(x) \leqslant \frac{\exp(b_{j'})}{\sum_{i'\in[k]}\exp(f_{i'}(x))} + O(\delta^r),$$

$$\frac{\exp(b_j)}{\sum_{i'\in[k]}\exp(f_{i'}(x))} - O(\delta^r) \leqslant u_j(x) \leqslant \frac{\exp(b_j)}{\sum_{i'\in[k]}\exp(f_{i'}(x))} + O(\delta^r).$$

Therefore, if $b_{j'} - b_j \geqslant \mu\delta^r$, we can further upper bound $\dot{b}_{j'} - \dot{b}_j$ as follows,

$$\begin{aligned}
\dot{b}_{j'} - \dot{b}_j \leqslant & \frac{k}{N} \sum_{x \in \mathcal{S} \setminus \mathcal{S}_{j'}} \left(u_j(x) - u_{j'}(x)\right). \\
\leqslant & \frac{k}{N} \sum_{x \in \mathcal{S} \setminus \mathcal{S}_{j'}} \frac{\exp\left(b_j\right) - \exp\left(b_{j'}\right)}{\sum_{i' \in [k]} \exp\left(f_{i'}(x)\right)} + O(\delta^r) \\
\leqslant & \frac{k}{N} \sum_{x \in \mathcal{S}_j} \frac{\exp\left(b_j\right) - \exp\left(b_{j'}\right)}{\sum_{i' \in [k]} \exp\left(f_{i'}(x)\right)} + O(\delta^r) \\
\leqslant & -\Omega(\mu\delta^r) \frac{k}{N} \sum_{x \in \mathcal{S}_j} \frac{\exp\left(b_j\right)}{\sum_{i' \in [k]} \exp\left(f_{i'}(x)\right)} + O(\delta^r).
\end{aligned}$$

Similar as in Lemma 5.1, we can show that $\frac{\exp(b_j)}{\sum_{i' \in [k]} \exp(f_{i'}(x))} \geqslant \Omega(1)$ due to $W_{j,j} \leqslant O(\delta)$ and $b_j \geqslant \max_{i' \in [k]} b_{i'} - O(\delta^r)$. So, finally we have

$$\dot{b}_{j'} - \dot{b}_j \leqslant -\Omega(\mu\delta^r) + O(\delta^r) < 0,$$

where the last inequality chooses μ as a large enough constant. □

Lemma D.12 (Accuracy Monotonicity)**.** *Given any positive constant C_2, there exists positive constant C_1 such that for all different $i, j \in [k]$, as long as $W_{i,i} \geqslant C_1$ and $f_i(x) - f_j(x) \leqslant C_2$ for any $x \in \mathcal{S}_i$, we have $\frac{\partial(f_i(x) - f_j(x))}{\partial t} > 0$.*

Proof of Lemma D.12. Since $f_i(x) - f_j(x) \leqslant C_2$, we know $1 - u_i(x) \geqslant \Omega(1)$. This immediately implies $\min_{x' \in \mathcal{S}_i} (1 - u_i(x')) \geqslant \Omega(1)$ since $|u_i(x) - u_i(x')| \leqslant O(\delta)$. According to Lemma D.10, we can bound $\frac{\partial(f_i(e_i) - f_j(e_i))}{\partial t}$ as follows,

$$\frac{\partial \left(f_i(x) - f_j(x)\right)}{\partial t} \geqslant \Omega(W_{i,i}^{2r-2}) - O(1) > 0$$

where the second inequality holds because $W_{i,i} \geqslant C_1$ with C_1 a large enough constant.

□

Lemma 5.3 (Bias Gap Control II)**.** *There exist small positive constants C_1, C_2 such that for any $j \in [k-1]$ and any $x \in \mathcal{S}_j$, if $1 - u_j(x) \leqslant C_1, W_{k,k} \leqslant O(\delta)$ and $b_j - b_k \geqslant -C_2$, we have $\dot{b}_j - \dot{b}_k < -\Omega(1)$.*

Proof of Lemma 5.3. Since $1 - u_j(x) \leqslant C_1$ for some $x \in \mathcal{S}_j$, we know $1 - u_j(x') \leqslant C_1 + O(\delta)$ for every $x' \in \mathcal{S}_j$. We can write down $\dot{b}_j - \dot{b}_k$ as follows,

$$\begin{aligned}\dot{b}_j - \dot{b}_k =& \left(1 - \frac{k}{N}\sum_{x' \in \mathcal{S}} u_j(x')\right) - \left(1 - \frac{k}{N}\sum_{x' \in \mathcal{S}} u_k(x')\right) \\ =& \frac{k}{N}\sum_{x' \in \mathcal{S}_j}\left(u_k(x') - u_j(x')\right) + \frac{k}{N}\sum_{x' \in \mathcal{S}\backslash\mathcal{S}_j}\left(u_k(x') - u_j(x')\right).\end{aligned}$$

First, we upper bound $u_k(x') - u_j(x')$ for every $x' \in \mathcal{S}_j$ as follows,

$$u_k(x') - u_j(x') \leqslant 1 - u_j(x') - u_j(x') = -1 + 2\left(1 - u_j(x')\right) \leqslant 2C_1 - 1 + O(\delta).$$

Same as in the proof of Lemma 5.1, for each $x' \in \mathcal{S}\backslash\mathcal{S}_j$, we can bound $u_j(x'), u_k(x')$ as follows,

$$\frac{\exp\left(b_j\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} - O(\delta^r) \leqslant u_j(x') \leqslant \frac{\exp\left(b_j\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} + O(\delta^r),$$

$$\frac{\exp\left(b_k\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} - O(\delta^r) \leqslant u_k(x') \leqslant \frac{\exp\left(b_k\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} + O(\delta^r).$$

Therefore, we can upper bound $u_k(x') - u_j(x')$ as follows,

$$\begin{aligned}u_k(x') - u_j(x') \leqslant& \frac{\exp\left(b_k\right) - \exp\left(b_j\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} + O(\delta) \\ =& \frac{\exp\left(b_j\right)}{\sum_{i' \in [k]}\exp\left(f_{i'}(x')\right)} \cdot \left(\exp(b_k - b_j) - 1\right) + O(\delta^r) \\ \leqslant& O(C_2) + O(\delta^r),\end{aligned}$$

where the last inequality uses $b_k - b_j \leqslant C_2$.

Above all, we can upper bound $\dot{b}_j - \dot{b}_k$ as follows,

$$\begin{aligned}\dot{b}_j - \dot{b}_k \leqslant & -1 + 2C_1 + O(C_2) + O(\delta^r) \\ < & -\Omega(1),\end{aligned}$$

where the second inequality holds as long as C_1, C_2, δ are at most some small constants. □

Lemma D.11 (Bias Gap Control III)**.** *For any different $i, j \in [k]$, if $W_{i,i} \leqslant O(1)$, $W_{j,j} \leqslant O(\delta)$ and $b_i - b_j \leqslant -O(1)$, we have*

$$\dot{b}_i - \dot{b}_j > 0.$$

Proof of Lemma D.11. We can write down $\dot{b}_i - \dot{b}_j$ as follows,

$$\begin{aligned}\dot{b}_i - \dot{b}_j = & \left(1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_i(x)\right) - \left(1 - \frac{k}{N}\sum_{x\in\mathcal{S}} u_j(x)\right) \\ = & \frac{k}{N}\sum_{x\in\mathcal{S}} \left(u_j(x) - u_i(x)\right)\end{aligned}$$

Next, we lower bound $u_j(x) - u_i(x)$ for every $x \in \mathcal{S}$,

$$\begin{aligned}u_j(x) - u_i(x) = & \frac{\exp\left(\langle W_{j,:}, x\rangle^r + b_j\right) - \exp\left(\langle W_{i,:}, x\rangle^r + b_i\right)}{\sum_{i'\in[k]} f_{i'}(x)} \\ \geqslant & \frac{\exp\left(O(\delta^r) + b_j\right) - \exp\left(O(1) + b_i\right)}{\sum_{i'\in[k]} f_{i'}(x)}\end{aligned}$$

So as long as $b_j - b_i > O(1)$, we have $u_j(x) - u_i(x) > 0$ for all $x \in \mathcal{S}$, which then implies $\dot{b}_i - \dot{b}_j > 0$. □

Lemma D.14. *For every $j \in [k]$ and every $x \in \mathcal{S}$, we have*

$$\left|\left\langle \dot{W}_{j,:}, \xi_x \right\rangle\right| \cdot T \leqslant O\left(\sqrt{\log N}\sigma\delta\right)$$

Proof of Lemma D.14. For each $j \in [k]$, we have

$$\dot{W}_{j,:} = \frac{k}{N}\left(\sum_{x'\in\mathcal{S}_j}(1-u_j(x'))\, r\left\langle W_{j,:}, x'\right\rangle^{r-1} x' - \sum_{x'\in\mathcal{S}\backslash\mathcal{S}_j} u_j(x') r\left\langle W_{j,:}, x'\right\rangle^{r-1} x'\right)$$

and

$$\begin{aligned}&\left\langle \dot{W}_{j,:}, \bar{\xi}_x\right\rangle \\ =&\frac{k}{N}\left(\sum_{x'\in\mathcal{S}_j}(1-u_j(x'))\, r\left\langle W_{j,:}, x'\right\rangle^{r-1}\left\langle x', \bar{\xi}_x\right\rangle - \sum_{x'\in\mathcal{S}\backslash\mathcal{S}_j} u_j(x') r\left\langle W_{j,:}, x'\right\rangle^{r-1}\left\langle x', \bar{\xi}_x\right\rangle\right)\end{aligned}$$

We know that $\left|\left\langle x, \bar{\xi}_x\right\rangle\right| \leqslant O(\sigma + \sqrt{\log N}/\sqrt{d})$. For any $x' \neq x$, we have $\left|\left\langle x', \bar{\xi}_x\right\rangle\right| \leqslant O((\sigma\sqrt{\log N})/\sqrt{d} + \sqrt{\log N}/\sqrt{d}) \leqslant O(\sqrt{\log N}/\sqrt{d})$ as long as $\sigma \leqslant 1$.

According to Lemma D.15, we know that for $x' \in \mathcal{S}_j$, we have $\left|(1-u_j(x'))\left\langle W_{j,:}, x'\right\rangle^{r-1}\right| \leqslant O(1)$. For $x' \in \mathcal{S}\backslash\mathcal{S}_i$, we have $\left|u_i(x')\left\langle W_{j,:}, x'\right\rangle^{r-1}\right| \leqslant O(\delta^{r-1})$ since $\left|\left\langle W_{j,:}, x'\right\rangle\right| \leqslant O(\delta) + O(\sqrt{\log N}\delta\sigma) \leqslant O(\delta)$ assuming $\sigma \leqslant 1/\sqrt{\log N}$.

Therefore, we can bound $\left|\left\langle \dot{W}_{j,:}, \bar{\xi}_x\right\rangle\right|$ as follows,

$$\left|\left\langle \dot{W}_{j,:}, \bar{\xi}_x\right\rangle\right| \leqslant O\left(\frac{\sigma}{N} + \frac{\sqrt{\log N}}{\sqrt{d}}\right)$$

Since $T \leqslant O(\log(1/\delta)/\delta^{r-2})$, $N \geqslant \log(1/\delta)/\delta^{r-1}$ and $d \geqslant \log^2(1/\delta)/\delta^{2r-2}$, we know

$$\left|\left\langle \dot{W}_{j,:}, \bar{\xi}_x\right\rangle\right| \cdot T \leqslant O(\sqrt{\log N}\delta).$$

□

Lemma D.15. *For any $i \in [k]$ and $x \in \mathcal{S}_i$, if $(1-u_i(x))\left\langle W_{i,:}, x\right\rangle^{r-1} \geqslant \Theta(1)$, we have*

$$\frac{d}{dt}\left((1-u_i(x))\left\langle W_{i,:}, x\right\rangle^{r-1}\right) < 0.$$

Proof of Lemma D.15. We can write $1-u_i(x)$ as $\frac{\sum_{j\in[k],j\neq i}\exp(f_j(x))}{\sum_{j\in[k],j\neq i}\exp(f_j(x))+\exp(f_i(x))}$. Next, we prove that for any $j'\neq i$, we have

$$\frac{d}{dt}\left(\frac{\exp\left(f_{j'}(x)\right)}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)\right)+\exp\left(f_i(x)\right)}\left\langle W_{i,:},x\right\rangle^{r-1}\right)<0.$$

This derivative can be written the sum of two terms:

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{\exp\left(f_{j'}(x)\right)}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)\right)+\exp\left(f_i(x)\right)}\left\langle W_{i,:},x\right\rangle^{r-1}\right)\\
=&\frac{1}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)}\frac{d}{dt}\left(\left\langle W_{i,:},x\right\rangle^{r-1}\right)\\
&+\frac{d}{dt}\left(\frac{1}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)}\right)\left\langle W_{i,:},x\right\rangle^{r-1}.
\end{aligned}$$

For the first term, we have

$$\begin{aligned}
&\frac{1}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)}\frac{d}{dt}\left(\left\langle W_{i,:},x\right\rangle^{r-1}\right)\\
=&\frac{1}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)}(r-1)\left\langle W_{i,:},x\right\rangle^{r-2}\left\langle \dot{W}_{i,:},x\right\rangle\\
\leqslant&\frac{1}{\exp\left(f_i(x)-f_{j'}(x)\right)}(r-1)\left\langle W_{i,:},x\right\rangle^{r-2}\left\langle \dot{W}_{i,:},x\right\rangle.
\end{aligned}$$

For the second term, we have

$$\begin{aligned}
&\frac{d}{dt}\left(\frac{1}{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)}\right)\left\langle W_{i,:},x\right\rangle^{r-1}\\
=&-\frac{\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)\left(\dot{f}_j(x)-\dot{f}_{j'}(x)\right)}{\left(\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)\right)^2}\left\langle W_{i,:},x\right\rangle^{r-1}\\
&-\frac{\exp\left(f_i(x)-f_{j'}(x)\right)\left(\dot{f}_i(x)-\dot{f}_{j'}(x)\right)}{\left(\sum_{j\in[k],j\neq i}\exp\left(f_j(x)-f_{j'}(x)\right)+\exp\left(f_i(x)-f_{j'}(x)\right)\right)^2}\left\langle W_{i,:},x\right\rangle^{r-1}\\
\leqslant&-\frac{1}{2}\cdot\frac{r\left\langle W_{i,:},x\right\rangle^{r-1}\left\langle \dot{W}_{i,:},x\right\rangle}{\exp\left(f_i(x)-f_{j'}(x)\right)}\left\langle W_{i,:},x\right\rangle^{r-1},
\end{aligned}$$

where the last inequality uses $f_i(x)-f_j(x)\geqslant\Omega(1)$, $\left|\dot{f}_j(x)-\dot{f}_{j'}(x)\right|\leqslant O(1)$ and $\dot{f}_i(x)-\dot{f}_{j'}(x)\geqslant r\left\langle W_{i,:},x\right\rangle^{r-1}\left\langle \dot{W}_{i,:},x\right\rangle-O(1)\geqslant\Omega(1)$.

Combining the bounds on both terms, as long as $\left\langle W_{i,:},x\right\rangle$ is larger than certain constant (which is guaranteed by $(1-u_i(x))\left\langle W_{i,:},x\right\rangle^{r-1}\geqslant\Theta(1)$), we know $\frac{d}{dt}\left((1-u_i(x))\left\langle W_{i,:},x\right\rangle^{r-1}\right)<0$. □

Lemma D.13. *For any $j\neq j'$, we have $T\dot{W}_{j,j'}\leqslant O(\delta)$. Furthermore, there exists absolute constant $\mu>0$ such that if $0<W_{j,j'}<\frac{\mu\delta}{\log^{\frac{1}{r-2}}(1/\delta)}$, we have $T\dot{W}_{j,j'}\geqslant-\frac{\mu\delta}{2\log^{\frac{1}{r-2}}(1/\delta)}$.*

Proof of Lemma D.13. We can write down the derivative of $W_{j,j'}$ as follows,

$$\begin{aligned}
&\dot{W}_{j,j'} \\
=&\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,r\left\langle W_{j,:},x\right\rangle^{r-1}\left\langle e_{j'},x\right\rangle \\
&-\frac{k}{N}\sum_{x\in\mathcal{S}_{j'}}u_j(x)r\left\langle W_{j,:},x\right\rangle^{r-1}\left\langle e_{j'},x\right\rangle \\
&-\frac{k}{N}\sum_{x\in\mathcal{S}\backslash(\mathcal{S}_j\cup\mathcal{S}_{j'})}u_j(x)r\left\langle W_{j,:},x\right\rangle^{r-1}\left\langle e_{j'},x\right\rangle \\
=&\pm O\left(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}\right)-O\left(\left(W_{j,j'}\pm O\left(\sqrt{\log N}\delta\sigma\right)\right)^{r-1}\right)\pm O\left(\delta^{r-1}\frac{\sigma\sqrt{\log N}}{\sqrt{d}}\right).
\end{aligned}$$

The bound on the first term relies on $(1-u_j(x))\left\langle W_{j,:},x\right\rangle^{r-1}\leqslant O(1)$ and $\left\langle e_{j'},x\right\rangle=\pm O\left(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}\right)$ for $x\in\mathcal{S}_j$, where $(1-u_j(x))\left\langle W_{j,:},x\right\rangle^{r-1}\leqslant O(1)$ is guaranteed by Lemma D.15. The bound on the second term uses $\left\langle W_{j,:},x\right\rangle=W_{j,j'}\pm O\left(\sqrt{\log N}\delta\sigma\right)$ and $\left\langle e_{j'},x\right\rangle=1\pm O\left(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}\right)$ for $x\in\mathcal{S}_{j'}$. The bound on the third term uses $\left\langle W_{j,:},x\right\rangle=O(\delta)$ and $\left\langle e_{j'},x\right\rangle=\pm O\left(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}\right)$ for $x\in\mathcal{S}\backslash(\mathcal{S}_j\cup\mathcal{S}_{j'})$.

To prove the upper bound of the derivative, we have

$$\dot{W}_{j,j'}\leqslant O(\frac{\sigma\sqrt{\log N}}{\sqrt{d}})$$

where we use $W_{j,j'}\pm O\left(\sqrt{\log N}\delta\sigma\right)\geqslant 0$. Since $T=O(\log(1/\delta)/\delta^{r-2})$, we have

$$T\dot{W}_{j,j'}\leqslant O(\delta),$$

as long as $d\geqslant O(\frac{\log N\log^2(1/\delta)}{\delta^{2r-2}})$.

We show that there exists absolute constant $\mu>0$ such that if $0<W_{j,j'}<\frac{\mu\delta}{\log^{\frac{1}{r-2}}(1/\delta)}$, we have $T\dot{W}_{j,j'}\geqslant-\frac{\mu\delta}{2\log^{\frac{1}{r-2}}(1/\delta)}$, which holds as long as

$\dot{W}_{j,j'} \geqslant -O\left(\frac{\mu\delta^{r-1}}{\log^{\frac{r-1}{r-2}}(1/\delta)}\right)$. We have

$$\begin{aligned}
&\dot{W}_{j,j'} \\
&= \pm O(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}) - O\left(\left(W_{j,j'} \pm O\left(\sqrt{\log N}\delta\sigma\right)\right)^{r-1}\right) \\
&\geqslant - O(\frac{\sigma\sqrt{\log N}}{\sqrt{d}}) - O\left(\frac{\mu^{r-1}\delta^{r-1}}{\log^{\frac{r-1}{r-2}}(1/\delta)}\right) \\
&\geqslant - O\left(\frac{\mu^{r-1}\delta^{r-1}}{\log^{\frac{r-1}{r-2}}(1/\delta)}\right) \\
&\geqslant - O\left(\frac{\mu\delta^{r-1}}{\log^{\frac{r-1}{r-2}}(1/\delta)}\right).
\end{aligned}$$

The first inequality assumes $\sigma \leqslant O\left(\frac{\mu}{\sqrt{\log N}\log^{\frac{1}{r-2}}(1/\delta)}\right)$. The second inequality assumes $d \geqslant O(\frac{\log N \log^{\frac{2r-2}{r-2}}(1/\delta)}{\mu^{2r-2}\delta^{2r-2}})$. The third inequality chooses μ as a small enough constant. □

Lemma D.8. *For any $j \in [k]$, we have*

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \geqslant -O\left(\frac{\delta^{r-1}\sqrt{\log N}\sigma}{\sqrt{d}}\right).$$

If $\min_{x\in\mathcal{S}_j}(1-u_j(x)) \geqslant \Omega(1)$, we further have

$$\left(1 - O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\, rW_{j,j}^{r-1} \leqslant \frac{\partial W_{j,j}^{(t)}}{\partial t}.$$

and

$$\frac{\partial W_{j,j}^{(t)}}{\partial t} \leqslant \left(1 + O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\, rW_{j,j}^{r-1}.$$

Proof of Lemma D.8. We have

$$\begin{aligned}\dot{W}_{j,j} =&\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,r\langle W_{j,:},x\rangle^{r-1}\langle x,e_j\rangle-\frac{k}{N}\sum_{x\in\mathcal{S}\setminus\mathcal{S}_j}u_j(x)r\langle W_{j,:},x\rangle^{r-1}\langle x,e_j\rangle\\ =&\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,r\left(W_{j,j}\pm O\left(\sqrt{\log N}\delta\sigma\right)\right)^{r-1}\left(1\pm O\left(\frac{\sqrt{\log N}\sigma}{\sqrt{d}}\right)\right)\\ &-\frac{k}{N}\sum_{x\in\mathcal{S}\setminus\mathcal{S}_j}u_j(x)r\left(O(\delta)\pm O\left(\sqrt{\log N}\delta\sigma\right)\right)^{r-1}\left(\pm O\left(\frac{\sqrt{\log N}\sigma}{\sqrt{d}}\right)\right)\\ =&\left(1\pm O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,rW_{j,j}^{r-1}\pm O\left(\frac{\delta^{r-1}\sqrt{\log N}\sigma}{\sqrt{d}}\right),\end{aligned}$$

where the second equality uses $|\langle W_{j,:},\xi_x\rangle|\leqslant O(\sqrt{\log N}\delta\sigma), |\langle\xi_x,e_j\rangle|\leqslant O\left(\frac{\sqrt{\log N}\sigma}{\sqrt{d}}\right)$ and $W_{j,j'}\leqslant O(\delta)$ for $j\neq j'$.

Therefore, if $\min_{x\in\mathcal{S}_j}(1-u_j(x))\geqslant\Omega(1)$, we know

$$\left(1-O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,rW_{j,j}^{r-1}\leqslant\dot{W}_{j,j}.$$

and

$$\dot{W}_{j,j}\leqslant\left(1+O(\sqrt{\log N}\sigma)\right)\frac{k}{N}\sum_{x\in\mathcal{S}_j}(1-u_j(x))\,rW_{j,j}^{r-1}.$$

And we always have

$$\dot{W}_{j,j}\geqslant-O\left(\frac{\delta^{r-1}\sqrt{\log N}\sigma}{\sqrt{d}}\right).$$

□

Lemma D.10. *For any $x\in\mathcal{S}_i$ and any $j\neq i$, if $1-u_i(x)\geqslant\Omega(1)$, we have*

$$\frac{\partial}{\partial t}(f_i(x)-f_j(x))\geqslant\Omega(W_{i,i}^{2r-2})-O(1).$$

Proof of Lemma D.10. Recall that $f_i(x) = \langle W_{i,:}, x\rangle^r + b_i$, so we have

$$\begin{aligned}\dot{f}_i(x) =& r\langle W_{i,:}, x\rangle^{r-1}\left\langle \dot{W}_{i,:}, x\right\rangle + \dot{b}_i \\ =& r\left(W_{i,i} \pm \sqrt{\log N}\sigma\delta\right)^{r-1}\left(\dot{W}_{i,i} + \left\langle \dot{W}_{i,:}, \xi_x\right\rangle\right) + \dot{b}_i \\ \geqslant& \Omega(W_{i,i}^{2r-2}) - O(1),\end{aligned}$$

where in the last inequality we uses $\dot{W}_{i,i} \geqslant \Omega(W_{i,i}^{r-1}) \geqslant \Omega(\delta^r)$ and $\left|\left\langle \dot{W}_{i,:}, \xi_x\right\rangle\right| \leqslant \frac{\sigma\sqrt{\log N}\delta^{r-1}}{\log(1/\delta)} \leqslant O(\delta^{r-1})$.

We also have

$$\begin{aligned}\dot{f}_j(x) =& r\langle W_{j,:}, x\rangle^{r-1}\left\langle \dot{W}_{j,:}, x\right\rangle + \dot{b}_j \\ =& r\left(W_{j,i} + \sqrt{\log N}\sigma\delta\right)^{r-1}\left(\dot{W}_{j,i} \pm \left\langle \dot{W}_{j,:}, \xi_x\right\rangle\right) + \dot{b}_j \\ \leqslant& O(1),\end{aligned}$$

where we uses $|W_{j,i}| \leqslant O(\delta)$, $\left|\dot{W}_{j,i}\right| \leqslant O(\delta^{r-1})$ and $\left|\left\langle \dot{W}_{j,:}, \xi_x\right\rangle\right| \leqslant O(\delta^{r-1})$.

Therefore, we have

$$\frac{dt}{d}\left(f_i(x) - f_j(x)\right) \geqslant \Omega(W_{i,i}^{2r-2}) - O(1).$$

□

D.4 Additional experiments

In this section, we describe the detailed setting of our experiments and also include additional experiment results.

MNIST & Fashion-MNIST. Unless specified otherwise, we use a depth-10 and width-1024 fully-connected ReLU neural network (FCN10) for MNIST and Fashion-MNIST. We use Kaiming initialization for the weights and set all bias terms as zero.

We use a small initialization by scaling the weights of each layer by $(0.001)^{1/h}$ so the output is scaled by 0.001, where h is the network depth. We train the network using SGD with learning rate 0.01 and momentum 0.9 for 100 epochs.

CIFAR-10 & CIFAR-100 We use VGG-16 (without batch normalization) for CIFAR-10 and CIFAR-100. We use Kaiming initialization for the weights and set all bias terms as zero. We run SGD with momentum 0.9 and weight decay 1e−4 for 100 epochs. For the learning rate, we start from 0.01 and reduce it by a factor of 0.1 at the 60-th epoch and 90-th epoch.

We linearly interpolate using 50 evenly spaced points between the network at initialization and the network at the end of training. We evaluate error and loss on the train set. For each setting, we repeat the experiments three times from different random seeds and plot the mean and deviation.

Note in Figure 5.1, to contrast the convex curve and plateau curve, we have used FCN4 with standard initialization on MNIST, and VGG-16 with 0.001 initialization on CIFAR-10.

Our code is based on the implementation from Lucas et al. (2021). Each trial of our experiment can be finished on an Nvidia Tesla P100 within one hour.

D.4.1 All bias v.s. last bias v.s. no bias

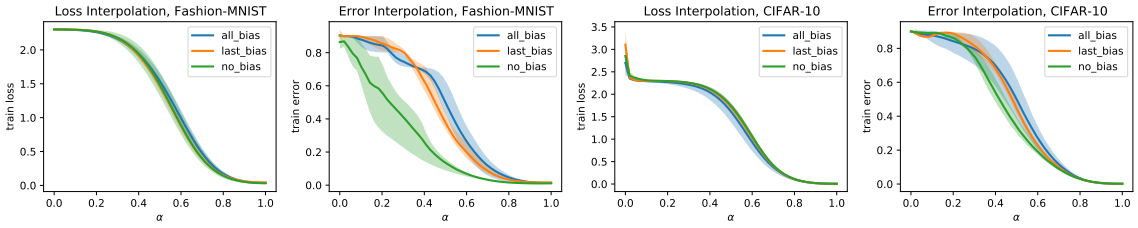

FIGURE D.1: Comparison between networks with all bias, last bias and no bias on Fashion-MNIST and CIFAR-10.

Figure D.1 shows that on both Fashion-MNIST and CIFAR-10, having bias on the last layer or on all layers can create longer plateau in error curve, while does not significantly affect the loss curve.

D.4.2 Normal interpolation v.s. homogeneous interpolation.

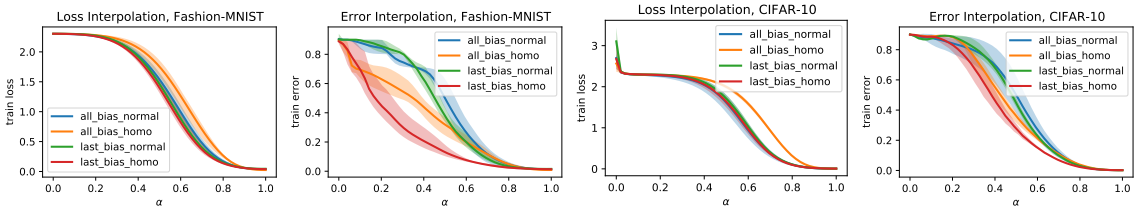

FIGURE D.2: Comparison between networks with normal interpolation and homogeneous interpolation on bias on Fashion-MNIST and CIFAR-10.

Figure D.2 shows that on both Fashion-MNIST and CIFAR-10, applying homogeneous interpolation on biases can significantly reduce the plateau on error interpolation curve.

D.4.3 Different initializations

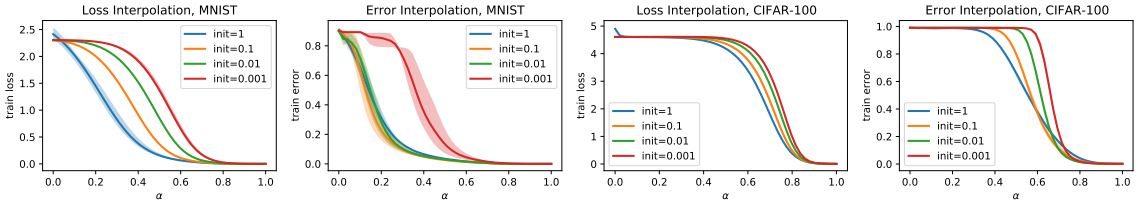

FIGURE D.3: Comparison between networks with different initialization scales on MNIST and CIFAR-100 with last bias.

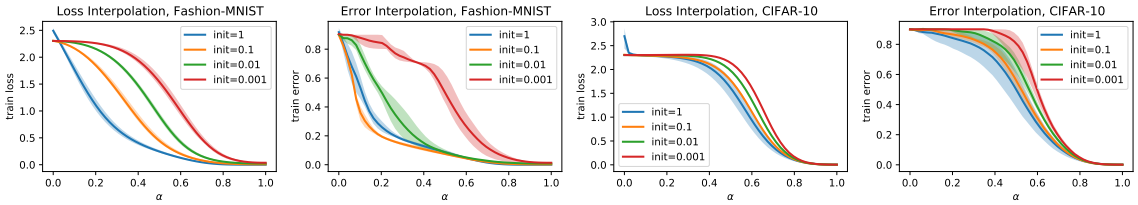

FIGURE D.4: Comparison between networks with different initialization scales on Fashion-MNIST and CIFAR-10 with all bias.

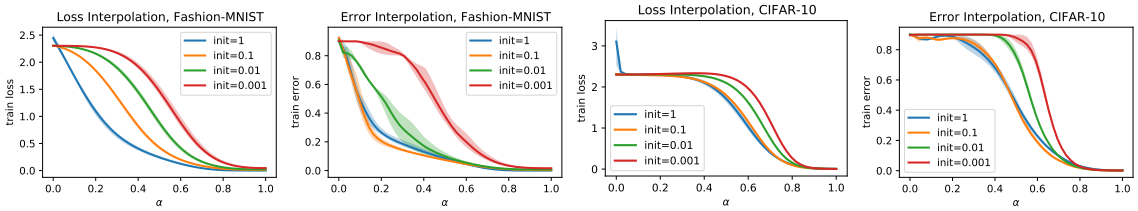

FIGURE D.5: Comparison between networks with different initialization scales on Fashion-MNIST and CIFAR-10 with last bias.

Smaller initialization creates longer plateau in both error and loss curves. See Figure D.3 for MNIST, CIFAR-100 with last bias; see Figure D.4 for Fashion-MNIST, CIFAR-10 with all bias; see Figure D.5 for Fashion-MNIST, CIFAR-10 with last bias.

D.4.4 Different depths

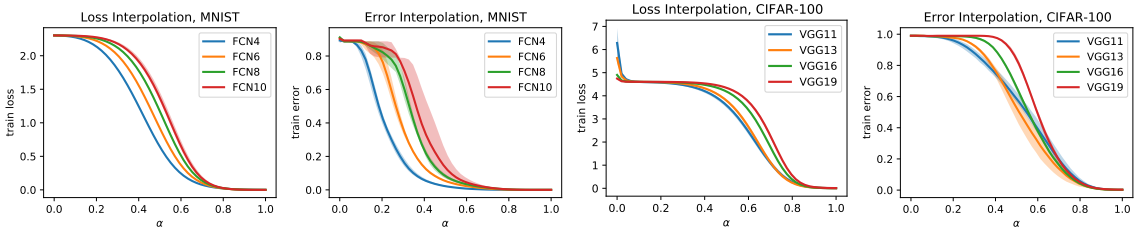

FIGURE D.6: Comparison between networks with different depth on MNIST and CIFAR-100 with last bias.

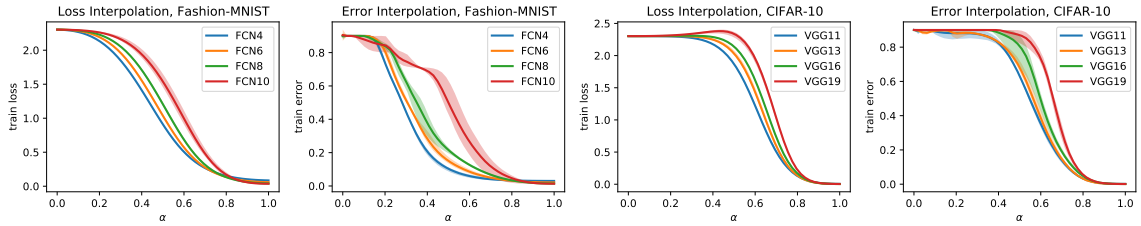

FIGURE D.7: Comparison between networks with different depth on Fashion-MNIST and CIFAR-10 with all bias. We use 0.001 initialization scale for VGG-16 on CIFAR-10.

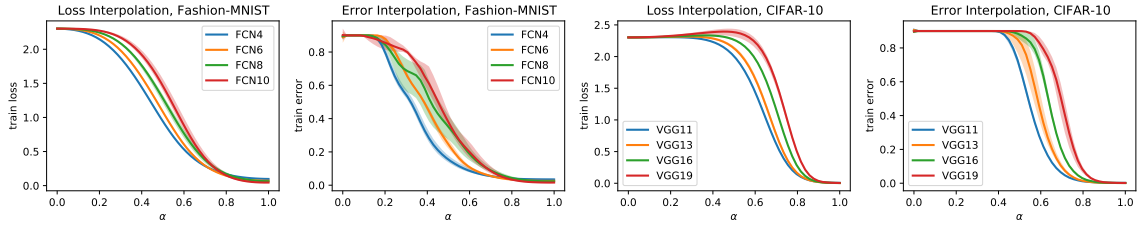

FIGURE D.8: Comparison between networks with different depth on Fashion-MNIST and CIFAR-10 with last bias. We use 0.001 initialization scale for VGG-16 on CIFAR-10.

Deeper networks create longer plateau in both error and loss curves. See Figure D.6 for MNIST, CIFAR-100 with last bias; see Figure D.7 for Fashion-MNIST, CIFAR-10 with all bias; see Figure D.8 for Fashion-MNIST, CIFAR-10 with last bias.

D.4.5 Bias dynamics

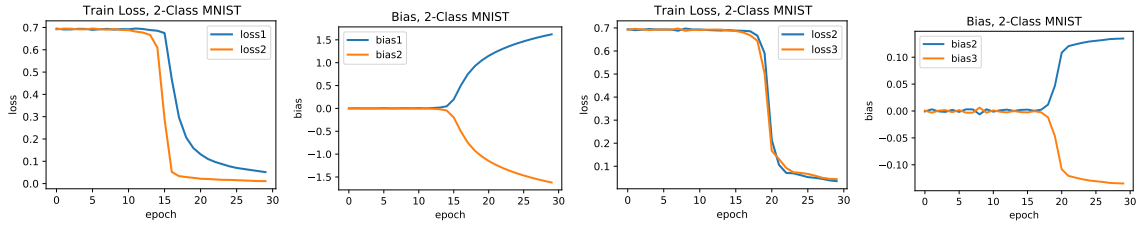

FIGURE D.9: Train loss for each class and bias term dynamics on MNIST$\{1, 2\}$ and MNIST$\{2, 3\}$.

In Figure D.9, we give two more examples on two-class MNIST in which the later learned class has larger bias.

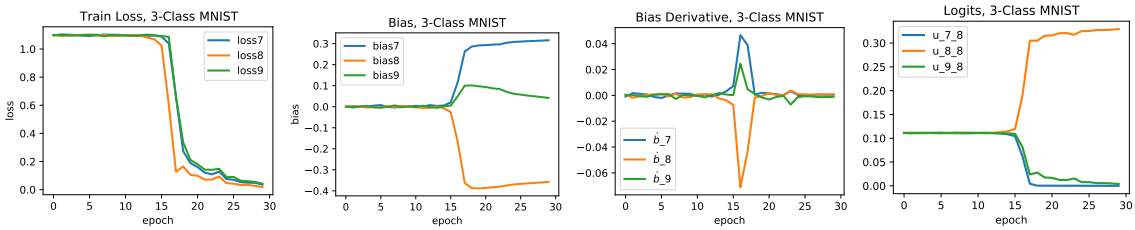

FIGURE D.10: Train loss for each class and bias term dynamics on MNIST$\{7, 8, 9\}$.

In Figure D.10, although class 9 is learned last, class 7 gets the largest bias after training. Let S be the set of all samples for number 7,8,9 and let S_7, S_9, S_9 be the set of samples for each class. For convenience, we use $u_{i,j}$ to denote $\frac{1}{|S|}\sum_{x\in S_j} u_i(x)$, where $u_i(x)$ is the softmax output for class i under input x. Then, we can write down the derivative on three bias terms:

$$\dot{b}_7 = \frac{1}{3} - u_{7,7} - u_{7,8} - u_{7,9}$$

$$\dot{b}_8 = \frac{1}{3} - u_{8,7} - u_{8,8} - u_{8,9}$$

$$\dot{b}_9 = \frac{1}{3} - u_{9,7} - u_{9,8} - u_{9,9}.$$

According to the per-class loss, we know that $\sum_{x\in S_7} -\log\left(u_7(x)\right) < \sum_{x\in S_9} -\log\left(u_9(x)\right)$, which intuitively implies that $\sum_{x\in S_7} u_7(x) > \sum_{x\in S_9} u_9(x)$ that is $u_{7,7} > u_{9,9}$. This tends to drive b_7 smaller than b_9. However, because $u_{9,8} > u_{7,8}$, we actually have $\dot{b}_9 < \dot{b}_7$. So eventually b_9 becomes smaller than b_7. Intuitively, class 9 is more correlated with class 8, so $u_{9,8} > u_{7,8}$.

Bibliography

Allen-Zhu, Z. and Li, Y. (2019), "What Can ResNet Learn Efficiently, Going Beyond Kernels?" *arXiv preprint arXiv:1905.10337*.

Allen-Zhu, Z. and Li, Y. (2020a), "Backward feature correction: How deep learning performs deep learning," *arXiv preprint arXiv:2001.04413*.

Allen-Zhu, Z. and Li, Y. (2020b), "Towards understanding ensemble, knowledge distillation and self-distillation in deep learning," *arXiv preprint arXiv:2012.09816*.

Allen-Zhu, Z., Li, Y., and Song, Z. (2018a), "A convergence theory for deep learning via over-parameterization," *arXiv preprint arXiv:1811.03962*.

Allen-Zhu, Z., Li, Y., and Liang, Y. (2018b), "Learning and generalization in overparameterized neural networks, going beyond two layers," *arXiv preprint arXiv:1811.04918*.

Allen-Zhu, Z., Li, Y., and Song, Z. (2019), "A convergence theory for deep learning via over-parameterization," in *International Conference on Machine Learning*, pp. 242–252, PMLR.

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. (2014), "Tensor decompositions for learning latent variable models," *Journal of machine learning research*, 15, 2773–2832.

Andoni, A., Panigrahy, R., Valiant, G., and Zhang, L. (2014), "Learning polynomials with neural networks," in *International conference on machine learning*, pp. 1908–1916.

Araújo, D., Oliveira, R. I., and Yukimura, D. (2019), "A mean-field limit for certain deep neural networks," *arXiv preprint arXiv:1906.00193*.

Arora, S., Cohen, N., and Hazan, E. (2018), "On the Optimization of Deep Networks: Implicit Acceleration by Overparameterization," in *International Conference on Machine Learning*, pp. 244–253.

Arora, S., Cohen, N., Golowich, N., and Hu, W. (2019a), "A Convergence Analysis of Gradient Descent for Deep Linear Neural Networks," in *ICLR*.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. (2019b), "Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks," *arXiv preprint arXiv:1901.08584*.

Arora, S., Cohen, N., Hu, W., and Luo, Y. (2019c), "Implicit regularization in deep matrix factorization," *Advances in Neural Information Processing Systems*, 32.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. (2019d), "On exact computation with an infinitely wide neural net," *arXiv preprint arXiv:1904.11955*.

Bachman, P., Hjelm, R. D., and Buchwalter, W. (2019), "Learning representations by maximizing mutual information across views," *arXiv preprint arXiv:1906.00910*.

Bai, Y. and Lee, J. D. (2019), "Beyond Linearization: On Quadratic and Higher-Order Approximation of Wide Neural Networks," *arXiv preprint arXiv:1910.01619*.

Bai, Y., Krause, B., Wang, H., Xiong, C., and Socher, R. (2020), "Taylorized Training: Towards Better Approximation of Neural Network Training at Finite Width," *arXiv preprint arXiv:2002.04010*.

Bardes, A., Ponce, J., and LeCun, Y. (2021), "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," *arXiv preprint arXiv:2105.04906*.

Bhojanapalli, S., Neyshabur, B., and Srebro, N. (2016), "Global optimality of local search for low rank matrix recovery," *Advances in Neural Information Processing Systems*, 29.

Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. (1994), "Signature verification using a" siamese" time delay neural network," *NeurIPS*.

Brutzkus, A. and Globerson, A. (2017), "Globally optimal gradient descent for a convnet with gaussian inputs," in *International conference on machine learning*, pp. 605–614, PMLR.

Carbery, A. and Wright, J. (2001), "Distributional and L^q norm inequalities for polynomials over convex bodies in R^n," *Mathematical research letters*, 8, 233–248.

Cardoso, J.-F. (1991), "Super-symmetric decomposition of the fourth-order cumulant tensor. Blind identification of more sources than sensors," in *International Conference on Acoustics, Speech, & Signal Processing, Icassp*.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. (2020), "Unsupervised learning of visual features by contrasting cluster assignments," *NeurIPS*.

Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., and Joulin, A. (2021), "Emerging properties in self-supervised vision transformers," *arXiv preprint arXiv:2104.14294*.

Chen, M., Bai, Y., Lee, J. D., Zhao, T., Wang, H., Xiong, C., and Socher, R. (2020a), "Towards understanding hierarchical learning: Benefits of neural representations," *arXiv preprint arXiv:2006.13436*.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020b), "A simple framework for contrastive learning of visual representations," *arXiv preprint arXiv:2002.05709*.

Chen, X. and He, K. (2020), "Exploring Simple Siamese Representation Learning," *arXiv preprint arXiv:2011.10566*.

Chidambaram, M., Wang, X., Hu, Y., Wu, C., and Ge, R. (2021), "Towards Understanding the Data Dependency of Mixup-style Training," in *International Conference on Learning Representations*.

Chidambaram, M., Wang, X., Wu, C., and Ge, R. (2022), "Provably Learning Diverse Features in Multi-View Data with Midpoint Mixup," *arXiv preprint arXiv:2210.13512*.

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

Chizat, L. and Bach, F. (2018a), "A note on lazy training in supervised differentiable programming," *arXiv preprint arXiv:1812.07956*, 8.

Chizat, L. and Bach, F. (2018b), "On the global convergence of gradient descent for over-parameterized models using optimal transport," in *Advances in neural information processing systems*, pp. 3040–3050.

Chizat, L. and Bach, F. (2020), "Implicit bias of gradient descent for wide two-layer neural networks trained with the logistic loss," in *Conference on Learning Theory*, pp. 1305–1338, PMLR.

Chizat, L., Oyallon, E., and Bach, F. (2019), "On lazy training in differentiable programming," in *Advances in Neural Information Processing Systems*, pp. 2933–2943.

Coates, A., Ng, A., and Lee, H. (2011), "An analysis of single-layer networks in unsupervised feature learning," in *International conference on artificial intelligence and statistics*.

Daniely, A. (2019), "Neural Networks Learning and Memorization with (almost) no Over-Parameterization," *arXiv preprint arXiv:1911.09873*.

Dasgupta, S. and Gupta, A. (2003), "An elementary proof of a theorem of Johnson and Lindenstrauss," *Random Structures & Algorithms*, 22, 60–65.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009), "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR*.

Deng, L. (2012), "The mnist database of handwritten digit images for machine learning research," *IEEE Signal Processing Magazine*, 29, 141–142.

Ding, T., Li, D., and Sun, R. (2019), "Sub-optimal local minima exist for neural networks with almost all non-linear activations," *arXiv preprint arXiv:1911.01413*.

Ding, Z., Chen, S., Li, Q., and Wright, S. J. (2022), "Overparameterization of Deep ResNet: Zero Loss and Mean-field Analysis," *Journal of Machine Learning Research*, 23, 1–65.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. A. (2018), "Essentially no barriers in neural network energy landscape," *arXiv preprint arXiv:1803.00885*.

Du, S., Lee, J., Tian, Y., Singh, A., and Poczos, B. (2018a), "Gradient descent learns one-hidden-layer cnn: Don't be afraid of spurious local minima," in *International Conference on Machine Learning*, pp. 1339–1348, PMLR.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. (2019), "Gradient descent finds global minima of deep neural networks," in *International Conference on Machine Learning*, pp. 1675–1685, PMLR.

Du, S. S. and Hu, W. (2019), "Width Provably Matters in Optimization for Deep Linear Neural Networks," *arXiv preprint arXiv:1901.08572*.

Du, S. S., Lee, J. D., Li, H., Wang, L., and Zhai, X. (2018b), "Gradient descent finds global minima of deep neural networks," *arXiv preprint arXiv:1811.03804*.

Du, S. S., Lee, J. D., and Tian, Y. (2018c), "When is a Convolutional Filter Easy to Learn?" in *International Conference on Learning Representations*.

Dyer, E. and Gur-Ari, G. (2019), "Asymptotics of wide networks from feynman diagrams," *arXiv preprint arXiv:1909.11304*.

Ermolov, A., Siarohin, A., Sangineto, E., and Sebe, N. (2021), "Whitening for self-supervised representation learning," in *International Conference on Machine Learning*, pp. 3015–3024, PMLR.

Fang, C., Lee, J., Yang, P., and Zhang, T. (2021), "Modeling from features: a mean-field framework for over-parameterized deep neural networks," in *Conference on learning theory*, pp. 1887–1936, PMLR.

Fort, S., Dziugaite, G. K., Paul, M., Kharaghani, S., Roy, D. M., and Ganguli, S. (2020), "Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel," *Advances in Neural Information Processing Systems*, 33, 5850–5861.

Frankle, J. (2020), "Revisiting" Qualitatively Characterizing Neural Network Optimization Problems"," *arXiv preprint arXiv:2012.06898*.

Frankle, J., Dziugaite, G. K., Roy, D., and Carbin, M. (2020), "Linear mode connectivity and the lottery ticket hypothesis," in *International Conference on Machine Learning*, pp. 3259–3269, PMLR.

Freeman, C. D. and Bruna, J. (2016), "Topology and Geometry of Half-Rectified Network Optimization," *arXiv preprint arXiv:1611.01540*.

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. (2018), "Loss surfaces, mode connectivity, and fast ensembling of dnns," in *NeurIPS*.

Ge, R., Huang, F., Jin, C., and Yuan, Y. (2015a), "Escaping from saddle points—online stochastic gradient for tensor decomposition," in *Conference on learning theory*, pp. 797–842, PMLR.

Ge, R., Huang, Q., and Kakade, S. M. (2015b), "Learning mixtures of gaussians in high dimensions," in *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pp. 761–770, ACM.

Ge, R., Lee, J. D., and Ma, T. (2016), "Matrix completion has no spurious local minimum," *Advances in neural information processing systems*, 29.

Ge, R., Lee, J. D., and Ma, T. (2018a), "Learning one-hidden-layer neural networks with landscape design," in *6th International Conference on Learning Representations, ICLR 2018*.

Ge, R., Lee, J. D., and Ma, T. (2018b), "Learning One-hidden-layer Neural Networks with Landscape Design," in *International Conference on Learning Representations*.

Ge, R., Kuditipudi, R., Li, Z., and Wang, X. (2019a), "Learning Two-Layer Neural Networks with Symmetric Inputs," in *International Conference on Learning Representations*.

Ge, R., Li, Z., Wang, W., and Wang, X. (2019b), "Stabilized SVRG: Simple variance reduction for nonconvex optimization," in *Conference on learning theory*, pp. 1394–1448, PMLR.

Ge, R., Ren, Y., Wang, X., and Zhou, M. (2021), "Understanding Deflation Process in Over-parametrized Tensor Decomposition," *Advances in Neural Information Processing Systems*, 34, 1299–1311.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. (2019), "Limitations of Lazy Training of Two-layers Neural Networks," *arXiv preprint arXiv:1906.08899*.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. (2020), "When do neural networks outperform kernel methods?" *arXiv preprint arXiv:2006.13409*.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. (2021), "Linearized two-layers neural networks in high dimension," *The Annals of Statistics*, 49, 1029–1054.

Gidel, G., Bach, F., and Lacoste-Julien, S. (2019), "Implicit regularization of discrete gradient dynamics in linear neural networks," *Advances in Neural Information Processing Systems*, 32.

Gissin, D., Shalev-Shwartz, S., and Daniely, A. (2019), "The Implicit Bias of Depth: How Incremental Learning Drives Generalization," in *International Conference on Learning Representations*.

Glorot, X. and Bengio, Y. (2010), "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256.

Goodfellow, I. J., Vinyals, O., and Saxe, A. M. (2014), "Qualitatively characterizing neural network optimization problems," *arXiv preprint arXiv:1412.6544*.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. (2020), "Bootstrap your own latent: A new approach to self-supervised learning," *arXiv preprint arXiv:2006.07733*.

Gunasekar, S., Woodworth, B. E., Bhojanapalli, S., Neyshabur, B., and Srebro, N. (2017), "Implicit regularization in matrix factorization," in *Advances in Neural Information Processing Systems*, pp. 6151–6159.

Gunasekar, S., Lee, J., Soudry, D., and Srebro, N. (2018a), "Characterizing implicit bias in terms of optimization geometry," *arXiv preprint arXiv:1802.08246*.

Gunasekar, S., Lee, J. D., Soudry, D., and Srebro, N. (2018b), "Implicit bias of gradient descent on linear convolutional networks," *Advances in Neural Information Processing Systems*, 31.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. (2021), "Provable Guarantees for Self-Supervised Deep Learning with Spectral Contrastive Loss," *arXiv preprint arXiv:2106.04156*.

Hardt, M. and Ma, T. (2016), "Identity matters in deep learning," *arXiv preprint arXiv:1611.04231*.

Harshman, R. (1970), "Foundations of the PARAFAC procedure: Model and conditions for an explanatory factor analysis," *Technical Report UCLA Working Papers in Phonetics 16, University of California, Los Angeles, Los Angeles, CA*.

He, K., Zhang, X., Ren, S., and Sun, J. (2015), "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034.

He, K., Zhang, X., Ren, S., and Sun, J. (2016), "Deep residual learning for image recognition," in *CVPR*.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020), "Momentum contrast for unsupervised visual representation learning," in *CVPR*.

Hillar, C. J. and Lim, L.-H. (2013), "Most tensor problems are NP-hard," *Journal of the ACM (JACM)*, 60, 1–39.

Hua, T., Wang, W., Xue, Z., Wang, Y., Ren, S., and Zhao, H. (2021), "On Feature Decorrelation in Self-Supervised Learning," *ICCV*.

Huang, J. and Yau, H.-T. (2019), "Dynamics of deep neural networks and neural tangent hierarchy," *arXiv preprint arXiv:1909.08156*.

Huang, J. and Yau, H.-T. (2020), "Dynamics of deep neural networks and neural tangent hierarchy," in *International Conference on Machine Learning*, pp. 4542–4551, PMLR.

Jacot, A., Gabriel, F., and Hongler, C. (2018), "Neural tangent kernel: Convergence and generalization in neural networks," in *Advances in neural information processing systems*, pp. 8571–8580.

Ji, Z. and Telgarsky, M. (2018a), "Gradient descent aligns the layers of deep linear networks," *arXiv preprint arXiv:1810.02032*.

Ji, Z. and Telgarsky, M. (2018b), "Risk and parameter convergence of logistic regression," *arXiv preprint arXiv:1803.07300*.

Ji, Z. and Telgarsky, M. (2019a), "Gradient descent aligns the layers of deep linear networks," in *7th International Conference on Learning Representations, ICLR 2019*.

Ji, Z. and Telgarsky, M. (2019b), "The implicit bias of gradient descent on nonseparable data," in *Conference on Learning Theory*, pp. 1772–1798, PMLR.

Ji, Z. and Telgarsky, M. (2019c), "A refined primal-dual analysis of the implicit bias," *arXiv preprint arXiv:1906.04540*.

Ji, Z. and Telgarsky, M. (2020), "Directional convergence and alignment in deep learning," *arXiv preprint arXiv:2006.06657*.

Ji, Z. and Telgarsky, M. (2021), "Characterizing the implicit bias via a primal-dual analysis," in *Algorithmic Learning Theory*, pp. 772–804, PMLR.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., and Jordan, M. I. (2017), "How to escape saddle points efficiently," in *International Conference on Machine Learning*, pp. 1724–1732, PMLR.

Jin, C., Netrapalli, P., and Jordan, M. I. (2018), "Accelerated gradient descent escapes saddle points faster than gradient descent," in *Conference On Learning Theory*, pp. 1042–1085, PMLR.

Kawaguchi, K. (2016), "Deep learning without poor local minima," *Advances in neural information processing systems*, 29.

Kawaguchi, K. and Bengio, Y. (2019), "Depth with nonlinearity creates no bad local minima in ResNets," *Neural Networks*, 118, 167–174.

Kawaguchi, K. and Kaelbling, L. (2020), "Elimination of all bad local minima in deep learning," in *International Conference on Artificial Intelligence and Statistics*, pp. 853–863, PMLR.

Krizhevsky, A. and Hinton, G. (2009), "Learning multiple layers of features from tiny images," Tech. rep., Citeseer.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012), "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, eds. F. Pereira, C. Burges, L. Bottou, and K. Weinberger, vol. 25, Curran Associates, Inc.

Kuditipudi, R., Wang, X., Lee, H., Zhang, Y., Li, Z., Hu, W., Ge, R., and Arora, S. (2019), "Explaining landscape connectivity of low-cost solutions for multilayer nets," *Advances in neural information processing systems*, 32.

Lakshmikantham, V., Bainov, D., and Simeonov, P. S. (1989), *Theory of impulsive differential equations*, World Scientific.

Lampinen, A. K. and Ganguli, S. (2018), "An analytic theory of generalization dynamics and transfer learning in deep linear networks," in *ICLR*.

Laurent, T. and Brecht, J. (2018), "Deep linear networks with arbitrary loss: All local minima are global," in *ICML*, pp. 2902–2907, PMLR.

Lee, J., Xiao, L., Schoenholz, S. S., Bahri, Y., Sohl-Dickstein, J., and Pennington, J. (2019), "Wide Neural Networks of Any Depth Evolve as Linear Models Under Gradient Descent," *arXiv preprint arXiv:1902.06720*.

Lee, J. D., Lei, Q., Saunshi, N., and Zhuo, J. (2020), "Predicting What You Already Know Helps: Provable Self-Supervised Learning," *arXiv preprint arXiv:2008.01064*.

Li, D., Ding, T., and Sun, R. (2018), "On the benefit of width for neural networks: Disappearance of bad basins," *arXiv preprint arXiv:1812.11039*.

Li, Y. and Liang, Y. (2018), "Learning Overparameterized Neural Networks via Stochastic Gradient Descent on Structured Data," *arXiv preprint arXiv:1808.01204*.

Li, Y., Ma, T., and Zhang, H. R. (2020a), "Learning over-parametrized two-layer neural networks beyond ntk," in *Conference on Learning Theory*, pp. 2613–2682, PMLR.

Li, Z., Luo, Y., and Lyu, K. (2020b), "Towards Resolving the Implicit Bias of Gradient Descent for Matrix Factorization: Greedy Low-Rank Learning," in *International Conference on Learning Representations*.

Liang, S., Sun, R., Lee, J. D., and Srikant, R. (2018), "Adding one neuron can eliminate all bad local minima," *Advances in Neural Information Processing Systems*, 31.

Livni, R., Shalev-Shwartz, S., and Shamir, O. (2013), "An algorithm for training polynomial networks," *arXiv preprint arXiv:1304.7045*.

Lu, Y., Ma, C., Lu, Y., Lu, J., and Ying, L. (2020), "A mean field analysis of deep resnet and beyond: Towards provably optimization via overparameterization from depth," in *International Conference on Machine Learning*, pp. 6426–6436, PMLR.

Lucas, J., Bae, J., Zhang, M. R., Fort, S., Zemel, R., and Grosse, R. (2021), "Analyzing monotonic linear interpolation in neural network loss landscapes," *arXiv preprint arXiv:2104.11044*.

Lyu, K. and Li, J. (2019), "Gradient Descent Maximizes the Margin of Homogeneous Neural Networks," in *International Conference on Learning Representations*.

Ma, T., Shi, J., and Steurer, D. (2016), "Polynomial-time tensor decompositions with sum-of-squares," in *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 438–446, IEEE.

Mei, S., Montanari, A., and Nguyen, P.-M. (2018), "A mean field view of the landscape of two-layer neural networks," *Proceedings of the National Academy of Sciences*, 115, E7665–E7671.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015), "Human-level control through deep reinforcement learning," *nature*, 518, 529–533.

Moroshko, E., Gunasekar, S., Woodworth, B., Lee, J. D., Srebro, N., and Soudry, D. (2020), "Implicit bias in deep linear classification: Initialization scale vs training accuracy," *arXiv preprint arXiv:2007.06738*.

Nacson, M. S., Lee, J., Gunasekar, S., Savarese, P. H. P., Srebro, N., and Soudry, D. (2019a), "Convergence of gradient descent on separable data," in *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 3420–3428, PMLR.

Nacson, M. S., Gunasekar, S., Lee, J., Srebro, N., and Soudry, D. (2019b), "Lexicographic and depth-sensitive margins in homogeneous and non-homogeneous deep models," in *International Conference on Machine Learning*, pp. 4683–4692, PMLR.

Nakkiran, P., Kaplun, G., Kalimeris, D., Yang, T., Edelman, B. L., Zhang, F., and Barak, B. (2019), "SGD on neural networks learns functions of increasing complexity," in *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pp. 3496–3506.

Nguyen, P.-M. and Pham, H. T. (2020), "A rigorous framework for the mean field limit of multilayer neural networks," *arXiv preprint arXiv:2001.11443*.

Nguyen, Q. (2019), "On connected sublevel sets in deep learning," in *International conference on machine learning*, pp. 4790–4799, PMLR.

Nguyen, Q. (2021), "A note on connectivity of sublevel sets in deep learning," *arXiv preprint arXiv:2101.08576*.

Nguyen, Q. and Hein, M. (2017), "The loss surface of deep and wide neural networks," in *International conference on machine learning*, pp. 2603–2612, PMLR.

Nguyen, Q. and Hein, M. (2018), "Optimization landscape and expressivity of deep CNNs," in *International conference on machine learning*, pp. 3730–3739, PMLR.

Nguyen, Q., Mukkamala, M. C., and Hein, M. (2018), "On the loss landscape of a class of deep neural networks with no bad local valleys," in *International Conference on Learning Representations*.

Nguyen, Q. N., Bréchet, P., and Mondelli, M. (2021), "When Are Solutions Connected in Deep Networks?" *Advances in Neural Information Processing Systems*, 34.

Nitanda, A. and Suzuki, T. (2017), "Stochastic particle gradient descent for infinite ensembles," *arXiv preprint arXiv:1712.05438*.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018), "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*.

Oymak, S. and Soltanolkotabi, M. (2020), "Towards moderate overparameterization: global convergence guarantees for training shallow neural networks," *IEEE Journal on Selected Areas in Information Theory*.

Pham, H. T. and Nguyen, P.-M. (2020), "Global Convergence of Three-layer Neural Networks in the Mean Field Regime," in *International Conference on Learning Representations*.

Razin, N. and Cohen, N. (2020), "Implicit regularization in deep learning may not be explainable by norms," *Advances in neural information processing systems*, 33, 21174–21187.

Razin, N., Maman, A., and Cohen, N. (2021), "Implicit regularization in tensor factorization," in *International Conference on Machine Learning*, pp. 8913–8924, PMLR.

Rotskoff, G. M. and Vanden-Eijnden, E. (2018a), "Neural networks as Interacting Particle Systems: Asymptotic convexity of the Loss Landscape and Universal Scaling of the Approximation Error," *arXiv preprint arXiv:1805.00915*.

Rotskoff, G. M. and Vanden-Eijnden, E. (2018b), "Trainability and accuracy of neural networks: An interacting particle system approach," *arXiv preprint arXiv:1805.00915*.

Safran, I. and Shamir, O. (2017), "Spurious Local Minima are Common in Two-Layer ReLU Neural Networks," *arXiv preprint arXiv:1712.08968*.

Safran, I. and Shamir, O. (2018), "Spurious Local Minima are Common in Two-Layer ReLU Neural Networks," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, eds. J. G. Dy and A. Krause, vol. 80 of *Proceedings of Machine Learning Research*, pp. 4430–4438, PMLR.

Saxe, A. M., McClelland, J. L., and Ganguli, S. (2013), "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," *arXiv preprint arXiv:1312.6120*.

Saxe, A. M., Mcclelland, J. L., and Ganguli, S. (2014), "Exact solutions to the nonlinear dynamics of learning in deep linear neural network," in *In International Conference on Learning Representations*.

Saxe, A. M., McClelland, J. L., and Ganguli, S. (2019), "A mathematical theory of semantic development in deep neural networks," *Proc. Natl. Acad. Sci. U. S. A.*

Shevchenko, A. and Mondelli, M. (2020), "Landscape connectivity and dropout stability of sgd solutions for over-parameterized neural networks," in *International Conference on Machine Learning*, pp. 8773–8784, PMLR.

Simonyan, K. and Zisserman, A. (2014), "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*.

Sirignano, J. and Spiliopoulos, K. (2018), "Mean Field Analysis of Neural Networks," *arXiv preprint arXiv:1805.01053*.

Sirignano, J. and Spiliopoulos, K. (2020), "Mean field analysis of neural networks: A central limit theorem," *Stochastic Processes and their Applications*, 130, 1820–1852.

Sirignano, J. and Spiliopoulos, K. (2022), "Mean field analysis of deep neural networks," *Mathematics of Operations Research*, 47, 120–152.

Soltanolkotabi, M. (2017), "Learning relus via gradient descent," *Advances in neural information processing systems*, 30.

Soudry, D. and Carmon, Y. (2016), "No bad local minima: Data independent training error guarantees for multilayer neural networks," *arXiv preprint arXiv:1605.08361*.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. (2018), "The implicit bias of gradient descent on separable data," *The Journal of Machine Learning Research*, 19, 2822–2878.

Stewart, G. W. (1977), "On the perturbation of pseudo-inverses, projections and linear least squares problems," *SIAM review*, 19, 634–662.

Sun, J., Qu, Q., and Wright, J. (2015), "Complete dictionary recovery using nonconvex optimization," in *International Conference on Machine Learning*, pp. 2351–2360, PMLR.

Sun, J., Qu, Q., and Wright, J. (2018), "A geometric analysis of phase retrieval," *Foundations of Computational Mathematics*, 18, 1131–1198.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014), "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, 27.

Tao, T. (2006), *Nonlinear dispersive equations: local and global analysis*, American Mathematical Society.

Tian, Y. (2017), "An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis," in *International conference on machine learning*, pp. 3404–3413, PMLR.

Tian, Y., Krishnan, D., and Isola, P. (2019), "Contrastive multiview coding," *arXiv preprint arXiv:1906.05849*.

Tian, Y., Yu, L., Chen, X., and Ganguli, S. (2020a), "Understanding self-supervised learning with dual deep networks," *arXiv preprint arXiv:2010.00578*.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. (2020b), "What makes for good views for contrastive learning?" *arXiv preprint arXiv:2005.10243*.

Tian, Y., Chen, X., and Ganguli, S. (2021), "Understanding self-supervised learning dynamics without contrastive pairs," *arXiv preprint arXiv:2102.06810*.

Tosh, C., Krishnamurthy, A., and Hsu, D. (2020), "Contrastive learning, multi-view redundancy, and linear models," *arXiv preprint arXiv:2008.10150*.

Tosh, C., Krishnamurthy, A., and Hsu, D. (2021), "Contrastive learning, multi-view redundancy, and linear models," in *Algorithmic Learning Theory*, pp. 1179–1206, PMLR.

Tsai, Y.-H. H., Wu, Y., Salakhutdinov, R., and Morency, L.-P. (2020), "Self-supervised learning from a multi-view perspective," *arXiv preprint arXiv:2006.05576*.

Venturi, L., Bandeira, A., and Bruna, J. (2018a), "Neural Networks with Finite Intrinsic Dimension have no Spurious Valleys," *arXiv preprint arXiv:1802.06384*.

Venturi, L., Bandeira, A. S., and Bruna, J. (2018b), "Spurious valleys in two-layer neural network optimization landscapes," *arXiv preprint arXiv:1802.06384*.

Vershynin, R. (2010), "Introduction to the non-asymptotic analysis of random matrices," *arXiv preprint arXiv:1011.3027*.

Vershynin, R. (2018), *High-dimensional probability: An introduction with applications in data science*, vol. 47, Cambridge University Press.

Vignat, C. and Bhatnagar, S. (2008), "An extension of Wick's theorem," *Statistics & probability letters*, 78, 2404–2407.

Wainwright, M. J. (2019), *High-dimensional statistics: A non-asymptotic viewpoint*, vol. 48, Cambridge University Press.

Wang, T. and Isola, P. (2020), "Understanding contrastive representation learning through alignment and uniformity on the hypersphere," in *International Conference on Machine Learning*, pp. 9929–9939, PMLR.

Wang, X., Wu, C., Lee, J. D., Ma, T., and Ge, R. (2020), "Beyond lazy training for over-parameterized tensor decomposition," *Advances in Neural Information Processing Systems*, 33, 21934–21944.

Wang, X., Yuan, S., Wu, C., and Ge, R. (2021a), "Guarantees for tuning the step size using a learning-to-learn approach," in *International Conference on Machine Learning*, pp. 10981–10990, PMLR.

Wang, X., Chen, X., Du, S. S., and Tian, Y. (2021b), "Towards demystifying representation learning with non-contrastive self-supervision," *arXiv preprint arXiv:2110.04947*.

Wang, X., Wang, A. N., Zhou, M., and Ge, R. (2022), "Plateau in Monotonic Linear Interpolation — A 'Biased' View of Loss Landscape for Deep Networks," *arXiv preprint arXiv:2210.01019*.

Wei, C., Lee, J. D., Liu, Q., and Ma, T. (2018), "On the margin theory of feedforward neural networks," *arXiv preprint arXiv:1810.05369*.

Wei, C., Lee Jason, D., Liu, Q., and Ma, T. (2019), "Regularization matters: Generalization and optimization of neural nets vs their induced kernel," *arXiv preprint arXiv:1810.05369*.

Wen, Z. and Li, Y. (2021), "Toward Understanding the Feature Learning Process of Self-supervised Contrastive Learning," *arXiv preprint arXiv:2105.15134*.

Woodworth, B., Gunasekar, S., Lee, J. D., Moroshko, E., Savarese, P., Golan, I., Soudry, D., and Srebro, N. (2020), "Kernel and rich regimes in overparametrized models," *arXiv preprint arXiv:2002.09277*.

Xiao, H., Rasul, K., and Vollgraf, R. (2017), "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*.

Xu, Z.-Q. J., Zhang, Y., Luo, T., Xiao, Y., and Ma, Z. (2019), "Frequency principle: Fourier analysis sheds light on deep neural networks," *arXiv preprint arXiv:1901.06523*.

Yehudai, G. and Shamir, O. (2019), "On the power and limitations of random features for understanding neural networks," *arXiv preprint arXiv:1904.00687*.

You, Y., Gitman, I., and Ginsburg, B. (2017), "Large batch training of convolutional networks," *arXiv:1708.03888*.

Yun, C., Sra, S., and Jadbabaie, A. (2018), "Global Optimality Conditions for Deep Neural Networks," in *International Conference on Learning Representations*.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. (2021), "Barlow twins: Self-supervised learning via redundancy reduction," *ICML*.

Zhang, X., Yu, Y., Wang, L., and Gu, Q. (2019), "Learning one-hidden-layer relu networks via gradient descent," in *The 22nd international conference on artificial intelligence and statistics*, pp. 1524–1534, PMLR.

Zhong, K., Song, Z., and Dhillon, I. S. (2017a), "Learning non-overlapping convolutional neural networks with multiple kernels," *arXiv preprint arXiv:1711.03440*.

Zhong, K., Song, Z., Jain, P., Bartlett, P. L., and Dhillon, I. S. (2017b), "Recovery guarantees for one-hidden-layer neural networks," in *International conference on machine learning*, pp. 4140–4149, PMLR.

Zhou, Y. and Liang, Y. (2017), "Critical points of neural networks: Analytical forms and landscape properties," *arXiv preprint arXiv:1710.11205*.

Zhu, X., Wang, Z., Wang, X., Zhou, M., and Ge, R. (2022), "Understanding Edge-of-Stability Training Dynamics with a Minimalist Example," *arXiv preprint arXiv:2210.03294*.

Zou, D. and Gu, Q. (2019), "An improved analysis of training over-parameterized deep neural networks," in *Advances in Neural Information Processing Systems*, pp. 2053–2062.

Zou, D., Cao, Y., Zhou, D., and Gu, Q. (2020), "Gradient descent optimizes over-parameterized deep ReLU networks," *Machine Learning*, 109, 467–492.

Biography

Xiang Wang is a Ph.D. candidate in Computer Science at Duke University. He is advised by Rong Ge. He received his bachelor's degree from Shanghai Jiao Tong University in 2017. He received the Outstanding Research Initiation Project Award from Duke Computer Science Department in 2019.