Journal of Machine Learning Research 26 (2025) 1-61 Submitted 1/24; Revised 4/25; Published 8/25

Understanding Deep Representation Learning via Layerwise Feature Compression and Discrimination

Peng Wang[†,*] PENG8WANG@GMAIL.COM
Xiao Li[†,*] XLXIAO@UMICH.EDU
Can Yaras[†] CJYARAS@UMICH.EDU
Zhihui Zhu[‡] ZHU.3440@OSU.EDU
Laura Balzano[†] GIRASOLE@UMICH.EDU
Wei Hu[†] VVH@UMICH.EDU
Qing Qu[†] QINGQU@UMICH.EDU

[†] *Department of Electrical Engineering & Computer Science, University of Michigan, Ann Arbor*
[‡] *Department of Computer Science & Engineering, Ohio State University, Columbus*
[*] *These authors contributed equally.*

Editor: Mahdi Soltanolkotabi

Abstract

Over the past decade, deep learning has proven to be a highly effective tool for learning meaningful features from raw data. However, it remains an open question how deep networks perform hierarchical feature learning across layers. In this work, we attempt to unveil this mystery by investigating the structures of intermediate features. Motivated by our empirical findings that linear layers mimic the roles of deep layers in nonlinear networks for feature learning, we explore how deep linear networks transform input data into output by investigating the output (i.e., features) of each layer after training in the context of multi-class classification problems. Toward this goal, we first define metrics to measure within-class compression and between-class discrimination of intermediate features, respectively. Through theoretical analysis of these two metrics, we show that the evolution of features follows a simple and quantitative pattern from shallow to deep layers when the input data is nearly orthogonal and the network weights are minimum-norm, balanced, and approximately low-rank: each layer of the linear network progressively compresses within-class features at a geometric rate and discriminates between-class features at a linear rate with respect to the number of layers that data have passed through. To the best of our knowledge, this is the first quantitative characterization of feature evolution in hierarchical representations of deep linear networks. Moreover, our extensive experiments not only validate our theoretical results but also reveal a similar pattern in deep nonlinear networks, which aligns well with recent empirical studies. Finally, we demonstrate the practical value of our results in transfer learning. Our code is available at `https://github.com/Heimine/PNC_DLN`.

Keywords: deep representation learning, low-dimensional structures, linear discriminative representation, intermediate features, feature compression and discrimination

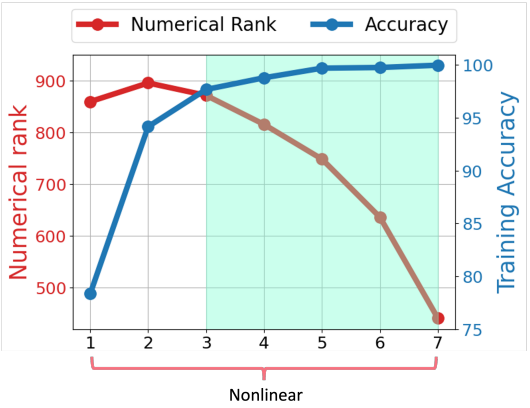
1. Introduction

In the past decade, deep learning has exhibited remarkable success across a wide range of applications in engineering and science (LeCun et al., 2015), such as computer vision (He et al., 2016; Simonyan and Zisserman, 2015), natural language processing (Sutskever et al., 2014; Vaswani et al., 2017), and health care (Esteva et al., 2019), to name a few. It is commonly believed that one major factor contributing to the success of deep learning is its ability to perform *hierarchical feature learning*: deep networks can leverage their hierarchical architectures to extract meaningful and informative features[1] from raw data (Allen-Zhu and Li, 2023; Krizhevsky et al., 2012). Despite recent efforts to understand deep networks, the underlying mechanism of how deep networks perform hierarchical feature learning across layers still remains a mystery, even for classical supervised learning problems, such as multi-class classification. Gaining deeper insight into this question will offer theoretical principles to guide the design of network architectures (He and Su, 2023), shed light on generalization and transferability (Li et al., 2024b), and facilitate network training (Xie et al., 2022).

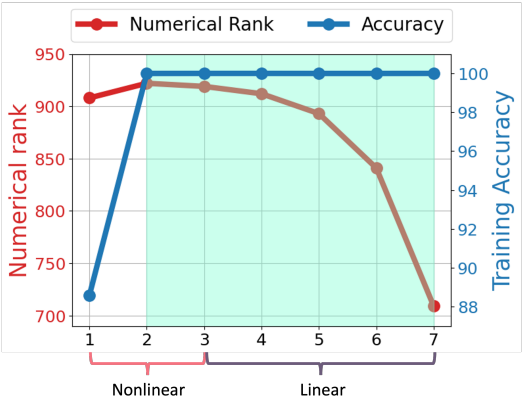
Empirical results on feature expansion and compression. Towards opening the black box of deep networks, extensive empirical research has been conducted in recent years by investigating outputs at each layer of deep networks. An intriguing line of research has empirically investigated the role of different layers in feature learning; see, e.g., Alain and Bengio (2017); Ansuini et al. (2019); Chen et al. (2022b); Masarczyk et al. (2023); Recanatesi et al. (2019); Zhang et al. (2022). In general, these empirical studies demonstrate that the initial layers expand the intrinsic dimension of features to make them linearly separable, while the subsequent layers compress the features progressively; see Figure 1. For example, in image classification tasks, Alain and Bengio (2017) observed that the features of intermediate layers are increasingly linearly separable as we reach the deeper layers. Recent works (Ansuini et al., 2019; Recanatesi et al., 2019) studied the evolution of the intrinsic dimension of intermediate features across layers in trained networks using different metrics. They both demonstrated that the dimension of the intermediate features first blows up and subsequently goes down from shallow to deep layers. More recently, Masarczyk et al. (2023) delved deeper into the role of different layers and concluded that the initial layers create linearly separable intermediate features, while later layers compress these features progressively.

Empirical results on feature compression and discrimination. Meanwhile, recent works (Papyan et al., 2020; Han et al., 2021; Fang et al., 2021; Zhu et al., 2021) have provided systematic studies on the structures of intermediate features. They revealed a fascinating phenomenon termed neural collapse (NC) during the terminal phase of training deep networks and across many different datasets and model architectures. Specifically, NC refers to a training phenomenon in which the last-layer features from the same class become nearly identical, while those from different classes become maximally linearly separable. In other words, deep networks learn within-class compressed and between-class discriminative features. Building upon these studies, a more recent line of work investigated the NC properties at each layer to understand how features are transformed from shallow to deep layers;

1. In deep networks, the (last-layer) feature typically refers to the output of the penultimate layer, which is also called *representation* in the literature.



(a) A 8-layer MLP network with ReLU



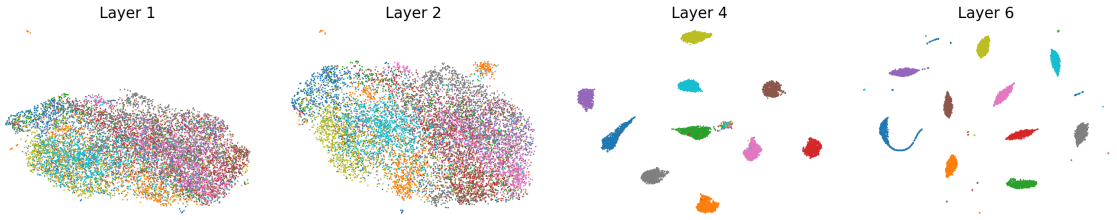
(b) A 8-layer hybrid network

Figure 1: **Illustration of numerical rank and training accuracy across layers.** We train two networks with different architectures on the CIFAR-10 dataset: (a) A 8-layer multilayer perceptron (MLP) network with ReLU activation, (b) A hybrid network consisting of a 3-layer MLP with ReLU activation followed by a 5-layer linear network. For each figure, we plot the numerical rank of the features of each layer and the training accuracy obtained by applying linear probing to the output of each layer, both against the number of layers. The green shading indicates that the features at these layers are *approximately* linearly separable, as evidenced by the near-perfect accuracy achieved by a linear classifier. The definition of numerical rank and additional experimental details are deferred to Section 5.1.1.

see, e.g., Ben-Shaul and Dekel (2022); He and Su (2023); Hui et al. (2022); Galanti et al. (2022a); Rangamani et al. (2023). In particular, He and Su (2023) empirically showed that a progressive NC phenomenon, governed by a law of data separation, occurs from shallow to deep layers. Rangamani et al. (2023) empirically showed that similar NC properties emerge in intermediate layers during training, where within-class variance decreases relative to the between-class variance as layers go deeper.

In summary, extensive empirical results demonstrate that after feature expansion by initial layers, *deep networks progressively compress features within the same class and discriminate features from different classes from shallow to deep layers*; see Figure 2. This characterization provides valuable insight into how deep networks transform data into output across layers in classification tasks. Moreover, this insight sheds light on designing more advanced network architectures, developing more efficient training strategies, and achieving better interpretability. However, to the best of our knowledge, no theoretical framework has yet been established to explain this empirical observation of progressive feature compression and discrimination. In this work, we take a first step towards bridging this gap by providing a theoretical analysis based on deep linear networks (DLNs).

Why study DLNs? Linear layers mimic deep layers in nonlinear networks for feature learning. Even though DLNs lack the strong expressive power of nonlinear networks, they possess comparable abilities for feature compression and discrimination to those



(a) A 8-layer nonlinear MLP

(b) A hybrid network with 3-layer MLP + 5-layer DLN

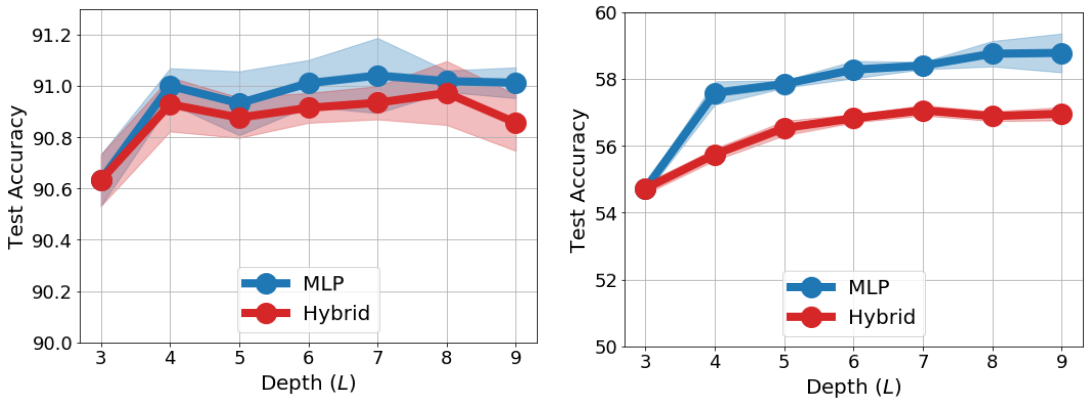
Figure 2: **Visualization of feature compression & discrimination from shallow to deep layers.** We consider the same setup as in Figure 1. For each network, we visualize the outputs of layers 1, 2, 4, and 6 on the CIFAR-10 dataset using the 2-dimensional UMAP plot (McInnes et al., 2018). Additional experimental details are deferred to Section 5.1.1.

observed in the deeper layers of nonlinear networks, as indicated in Figures 1 and 2. By evaluating the training accuracy and the numerical rank of intermediate features in both a nonlinear network and a hybrid network[2], we observe that the initial-layer features in both networks are almost linearly separable, evidenced by nearly perfect training accuracy achieved through linear probing. This phenomenon is further illustrated by the feature visualization in Figure 2. Meanwhile, the linear layers in the hybrid network mimic the role of their counterpart in the nonlinear network by performing feature compression and discrimination, as evidenced by the decreasing feature rank in Figure 1 and the increasing separation of different-class features in Figure 2 across layers in both types of networks.

Broadly speaking, DLNs have been recognized as valuable prototypes for studying nonlinear networks, as they resemble certain behaviors of their nonlinear counterparts (Alain and Bengio, 2017; Ansuini et al., 2019; Masarczyk et al., 2023; Recanatesi et al., 2019) while maintaining simplicity (Arora et al., 2018b; Gidel et al., 2019; Saxe et al., 2019). For instance, Huh et al. (2023) empirically demonstrated the presence of a low-rank bias at both initialization and after training for both linear and nonlinear networks. Saxe et al. (2019) showed that a DLN exhibits a striking hierarchical progressive differentiation of structures in its internal hidden representations, resembling patterns observed in their nonlinear counterparts.

The role of depth in DLNs: improving generalization, feature compression, and training speed. Although stacking linear layers in DLNs ultimately results in an end-to-end linear transformation from input to output, the overparameterization in these

2. For the hybrid network, we introduce nonlinearity in the first few layers and follow them with linear layers.



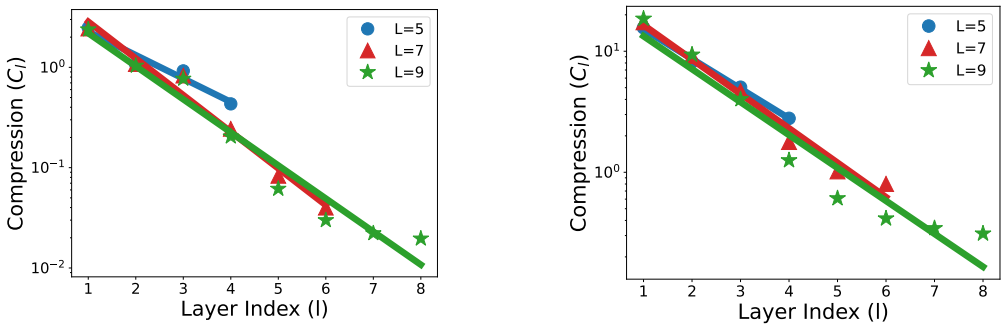
(a) FashionMNIST (b) CIFAR-10

Figure 3: **Depth of DLNs lead to better generalization performance.** We train hybrid networks consisting of a 2-layer MLP with ReLU activation followed by $(L-2)$ linear layers on the FashionMNIST and CIFAR-10 datasets, respectively. As a reference, we also train nonlinear networks comprised exclusively of MLP layers. We plot the test accuracy against the different number of layers averaged over 5 different runs. It is observed that adding either linear layers or MLP layers can improve generalization performance. More experimental details are deferred to Section 5.1.2.

DLNs distinguishes them from a basic linear operator: increasing the depth of DLNs can significantly improve their generalization capabilities, enhance feature compression, and facilitate network training. Specifically, recent works have demonstrated that linear over-parameterization by depth (i.e., expanding one linear layer into a composition of multiple linear layers) in deep nonlinear networks yields better generalization performance across different network architectures and datasets (Guo et al., 2020; Huh et al., 2023; Kwon et al., 2024). This is corroborated by our experiments in Figure 3, where increasing the depth of linear layers of a hybrid network leads to improved test accuracy. Moreover, our results in Figure 4 suggest that increasing the depth of DLN also leads to improved feature compression. We refer interested readers to Nichani (2021) for further discussion on the role of depth in DLNs.

1.1 Our Contributions

In this work, we study hierarchical representations of deep networks for multi-class classification problems. Towards this goal, we explore how a DLN transforms input data into a one-hot encoding label matrix from shallow to deep layers by investigating features at each layer. To characterize the structures of intermediate features, we define metrics to measure within-class feature compression and between-class feature discrimination at each layer, respectively (see Definition 1). We establish a unified framework to analyze these metrics and reveal a simple and quantitative pattern in the evolution of features from shallow to deep layers:



(a) FashionMNIST (b) CIFAR-10

Figure 4: **Progressive feature compression on DLNs trained with default initialization and real datasets.** Using the DLNs trained in Figure 3, we plot the within-class compression metrics C_l (see Definition 1) against the layer indices. It is observed that progressive linear decay still (approximately) happens without the orthogonal initialization and datasets described in Assumption 2.

The compression metric ***decays at a geometric rate****, while the discrimination metric* ***increases at a linear rate*** *with respect to the number of layers.*

More specifically, we rigorously prove the above claim under the following assumptions:

- *Assumption on the training data.* We study a K-class classification problem with balanced training data, where each class contains n training samples. We assume that the training samples are θ-nearly orthonormal for a constant $\theta \in [0, 1/4)$, and the feature dimension is larger than the total number of samples so that they are linearly separable (see Assumption 2).
- *Assumption on the trained weights.* We assume that an L-layer DLN is trained such that its weights are minimum-norm, δ-balanced, and ε-approximately low-rank, due to the implicit bias of gradient descent, where $\delta, \varepsilon \in (0, 1)$ are constants (see Assumption 3).

We discuss the validity of our assumptions in Section 3.2. Based upon these assumptions, we show in Theorem 4 that the ratio of the within-class feature compression metric between the $(l+1)$-th layer and the l-th layer is $O(\varepsilon^2/n^{1/L})$, implying a geometric decay from shallow to deep layers. Moreover, we show that the between-class discrimination metric increases linearly with respect to (w.r.t.) the number of layers, with a slope of $O((\theta + 4\delta)/L)$. To the best of our knowledge, this is the first quantitative characterization of feature evolution in hierarchical representations of DLNs in terms of feature compression and discrimination. Notably, this analytical framework can be extended to nonlinear networks (Jacot et al., 2025). Finally, we substantiate our theoretical findings in Section 5 on both synthetic and real-world datasets, demonstrating that our claims hold for DLNs and also manifest empirically in nonlinear networks.

Significance of our results. In recent years, there has been a growing body of literature studying hierarchical feature learning to open the black box of deep networks. These studies

include works on neural tangent kernel (Jacot et al., 2018; Huang and Yau, 2020), intermediate feature analysis (Alain and Bengio, 2017; Masarczyk et al., 2023; Rangamani et al., 2023), neural collapse (Dang et al., 2023; He and Su, 2023; Súkeník et al., 2023; Tirer and Bruna, 2022), and learning dynamic analysis (Allen-Zhu and Li, 2023; Bietti et al., 2022; Damian et al., 2022), among others. We refer readers to Section 4 for a more comprehensive discussion. Our work contributes to this emerging area by showing that each layer of deep networks plays an equally important role in hierarchical feature learning, which compresses within-class features at a geometric rate and discriminates between-class features at a linear rate w.r.t. the number of layers. This provides a simple and precise characterization of how deep networks transform data hierarchically from shallow to deep layers. It also addresses an open question about neural collapse and offers a new perspective justifying the importance of depth in feature learning; see the discussion in Section 3.1. Moreover, our result explains why projection heads (Chen et al., 2020), which usually refer to one or several MLP layers added between the feature layer and final classifier during pretraining and discarded afterwards, can improve the performance of transfer learning on downstream tasks (Li et al., 2024b; Kornblith et al., 2021; Galanti et al., 2022b). In summary, our result provides important guiding principles for deep learning practices related to interpretation, architecture design, and transfer learning.

Differences and connections to the existing literature. Finally, we highlight the differences and connections between our work and two closely related recent works (He and Su, 2023; Saxe et al., 2019) as follows:

- First, He and Su (2023) empirically showed that a *progressive NC* phenomenon governed by a law of data separation occurs from shallow to deep layers. Specifically, they observed that in trained over-parameterized nonlinear networks for classification problems, a metric of data separation decays at a geometric rate w.r.t. the number of layers. This is similar to our studied within-class feature compression at a geometric rate, albeit with different metrics (see the remark after Definition 1). However, they do not provide any theoretical explanation for the progressive NC phenomenon. While our research focuses on DLNs and may not provide a complete understanding of phenomena in nonlinear networks, our theoretical analysis offers insights into the empirical behavior of deeper nonlinear layers. This is supported by our findings presented in Figure 1, which demonstrate that linear layers can replicate the function of deep nonlinear layers in terms of feature learning.

- Second, Saxe et al. (2019) reveals that during training, nonlinear neural networks exhibit a hierarchical progressive differentiation of structure in their internal representations—a phenomenon they refer to as *progressive differentiation*. While both their study and ours reveal and justify a progressive separation phenomenon based on DLNs, the focus of each study is entirely different yet highly complimentary to each other. Specifically, we investigate how a trained neural network separates data according to their class membership from shallow to deep layers *after training*, while they investigate how weights of a neural network change w.r.t. training time and their impact on class differentiation *during training*. This distinction can be illustrated by the example in Saxe et al. (2019). Suppose a neural network is trained to classify eight items: sunfish, salmon, canary, robin, daisy, rose, oak, and pine. We study how these eight items are represented by the neural

networks from shallow to deep layers in the trained neural network. In comparison, Saxe et al. (2019) explain why animals versus plants are first distinguished at the initial stage of training, then birds versus fish, then trees versus flowers, and finally individual items, throughout the training process.

1.2 Notation and Paper Organization

Notation. Let $\mathbb{R}^n$ be the n-dimensional Euclidean space and $\|\cdot\|$ be the Euclidean norm. Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we use $\boldsymbol{a}_i$ to denote its i-th column; we use $\|\boldsymbol{A}\|_F$ to denote the Frobenius norm of $\boldsymbol{A}$; we use $\sigma_{\max}(\boldsymbol{A})$ (or $\|\boldsymbol{A}\|$), $\sigma_i(\boldsymbol{A})$, and $\sigma_{\min}(\boldsymbol{A})$ to denote the largest, the i-th largest, and the smallest singular values, respectively. We use $\boldsymbol{A}^\dagger$ to denote the pseudo-inverse of a matrix $\boldsymbol{A}$. Given $L \in \mathbb{N}$, we use $[L]$ to denote the index set $\{1, \cdots, L\}$. Let $\mathcal{O}^n = \{\boldsymbol{Z} \in \mathbb{R}^{n\times n} : \boldsymbol{Z}^T\boldsymbol{Z} = \boldsymbol{I}\}$ denote the set of all $n\times n$ orthogonal matrices. We denote the Kronecker product by $\otimes$. Given weight matrices $\boldsymbol{W}_1, \ldots, \boldsymbol{W}_L$, let $\boldsymbol{W}_{l:1} := \boldsymbol{W}_l \cdots \boldsymbol{W}_1$ denote a matrix multiplication from $\boldsymbol{W}_l$ to $\boldsymbol{W}_1$ for all $l \in [L]$.

Organization. The rest of the paper is organized as follows. In Section 2, we introduce the basic problem setup. In Section 3, we present the main results and discuss their implications, with proofs provided in Section A in the appendix. In Section 4, we discuss the connections of our results to related works. In Section 5, we validate our theoretical claims and investigate nonlinear networks through numerical experiments. Finally, we conclude and discuss future directions in Section 6. We defer all the auxiliary technical results to the appendix.

2. Preliminaries

In this section, we first formally introduce the problem of training DLNs for solving multi-class classification problems in Section 2.1, and then present the metrics for measuring within-class compression and between-class discrimination of features at each layer in Section 2.2.

2.1 Problem Setup

Multi-class classification problem. We consider a K-class classification problem with training samples and labels $\{(\boldsymbol{x}_{k,i}, \boldsymbol{y}_k)\}_{i\in[n_k],k\in[K]}$, where $\boldsymbol{x}_{k,i} \in \mathbb{R}^d$ is the i-th sample in the k-th class, and $\boldsymbol{y}_k \in \mathbb{R}^K$ is an one-hot label vector with the k-th entry being 1 and 0 elsewhere. We denote by n_k the number of samples in the k-th class for each $k \in [K]$. Here, we assume that the number of samples in each class is the same, i.e., $n_1 = \cdots = n_K = n$. This assumption is commonly used in recent studies on deep representation learning for classification problems (Zhu et al., 2021; Yaras et al., 2022; Zhou et al., 2022a,b). Moreover, we denote the total number of samples by $N = nK$. Without loss of generality, we arrange the training samples in a class-by-class manner such that

$$\boldsymbol{X} \;=\; [\boldsymbol{x}_{1,1}, \ldots, \boldsymbol{x}_{1,n_1}, \ldots, \boldsymbol{x}_{K,1}, \ldots, \boldsymbol{x}_{K,n_K}] \in \mathbb{R}^{d\times N}, \quad \boldsymbol{Y} \;=\; \boldsymbol{I}_K \otimes \boldsymbol{1}_n^T \in \mathbb{R}^{K\times N}, \tag{1}$$

where $\otimes$ denotes the Kronecker product. For convenience, we also use $\boldsymbol{x}_i$ to denote the i-th column of $\boldsymbol{X}$.

DLNs for classification problems. In this work, we consider an L-layer ($L \geq 2$) linear network $f_{\boldsymbol{\Theta}}(\cdot) : \mathbb{R}^d \to \mathbb{R}^K$, parameterized by $\boldsymbol{\Theta} = \{\boldsymbol{W}_l\}_{l=1}^L$ with input $\boldsymbol{x} \in \mathbb{R}^d$, i.e.,

$$f_{\boldsymbol{\Theta}}(\boldsymbol{x}) \;:=\; \boldsymbol{W}_L \cdots \boldsymbol{W}_1 \boldsymbol{x} \;=\; \boldsymbol{W}_{L:1}\boldsymbol{x}, \tag{2}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{d_1 \times d}$, $\boldsymbol{W}_l \in \mathbb{R}^{d_l \times d_{l-1}}$ for $l = 2, \ldots, L-1$, and $\boldsymbol{W}_L \in \mathbb{R}^{K \times d_{L-1}}$ are the weight matrices. The last-layer weight $\boldsymbol{W}_L$ is referred to as the linear *classifier* and the output of the l-th layer as the l-th layer *feature* for all $l \in [L-1]$. As discussed in Section 1, DLNs are often used as prototypes for studying practical deep networks (Hardt and Ma, 2016; Kawaguchi, 2016; Laurent and Brecht, 2018; Lu and Kawaguchi, 2017). We train an L-layer linear network to learn weights $\boldsymbol{\Theta} = \{\boldsymbol{W}_l\}_{l=1}^L$ via minimizing the mean squared error (MSE) loss between $f_{\boldsymbol{\Theta}}(\boldsymbol{x})$ and $\boldsymbol{y}$ over the training data, i.e.,

$$\min_{\boldsymbol{\Theta}} \ell(\boldsymbol{\Theta}) \;=\; \frac{1}{2}\sum_{k=1}^{K}\sum_{i=1}^{n_k} \|\boldsymbol{W}_{L:1}\boldsymbol{x}_{k,i} - \boldsymbol{y}_k\|^2 \;=\; \frac{1}{2}\|\boldsymbol{W}_{L:1}\boldsymbol{X} - \boldsymbol{Y}\|_F^2. \tag{3}$$

Before we proceed, we make some remarks on this problem.

- First, the cross-entropy (CE) loss is arguably the most popular loss function used to train neural networks for classification problems (Zhu et al., 2021). However, recent studies (Hui and Belkin, 2020; Zhou et al., 2022a) have demonstrated through extensive experiments that the MSE loss achieves performance comparable to or even superior to that of the CE loss across various tasks.

- Second, while DLNs may appear simple, the loss landscape of the objective function as defined in (3) is highly nonconvex, leading to highly nonlinear learning dynamics in gradient descent (GD) (Chen et al., 2025; Achour et al., 2024). Notably, the nonlinear GD dynamics in DLNs closely mirror that in their nonlinear counterparts (Lampinen and Ganguli, 2018; Saxe et al., 2019).

- Finally, DLNs tend to be over-parameterized, with width of networks d_l and feature dimensions d exceeding the training samples N. In this setting, Problem (3) has infinitely many solutions that can achieve zero training loss, i.e., $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. Nevertheless, many studies have examined the convergence behavior of GD by closely examining its learning trajectory, revealing that GD—when initialized appropriately—exhibits an implicit bias towards *minimum norm solutions* (Bartlett et al., 2020; Min et al., 2021) with approximately *balanced* and *low-rank* weights (Arora et al., 2018a; Min et al., 2021), which will be discussed further in Section 3.2.

2.2 The Metrics of Feature Compression and Discrimination

In this work, we focus on studying feature structures at each layer in a trained DLN. Given the weights $\boldsymbol{\Theta} = \{\boldsymbol{W}_l\}_{l=1}^L$ satisfying $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$, the weights $\{\boldsymbol{W}_l\}_{l=1}^L$ of the DLN transform the input data $\boldsymbol{X}$ into the membership matrix $\boldsymbol{Y}$ at the final layer. However, the hierarchical structure of the DLN prevents us from gaining insight into the underlying mechanism of how it transforms the input data into output from shallow to deep layers. To

unravel this puzzle, we probe the features learned at intermediate layers. In our setting, we write the l-th layer's feature of an input sample $\boldsymbol{x}_{k,i}$ as

$$\boldsymbol{z}_{k,i}^l = \boldsymbol{W}_l \dots \boldsymbol{W}_1 \boldsymbol{x}_{k,i} = \boldsymbol{W}_{l:1} \boldsymbol{x}_{k,i}, \ \forall l = 1, \dots, L-1, \tag{4}$$

and we denote $\boldsymbol{z}_{k,i}^0 = \boldsymbol{x}_{k,i}$. For $l = 0, 1, \dots, L-1$, let $\boldsymbol{\Sigma}_W^l$ and $\boldsymbol{\Sigma}_B^l$ respectively denote the sum of the within-class and between-class covariance matrices for the l-th layer, i.e.,

$$\boldsymbol{\Sigma}_W^l := \frac{1}{N} \sum_{k=1}^{K} \sum_{i=1}^{n_k} \left(\boldsymbol{z}_{k,i}^l - \boldsymbol{\mu}_k^l \right) \left(\boldsymbol{z}_{k,i}^l - \boldsymbol{\mu}_k^l \right)^T, \tag{5}$$

$$\boldsymbol{\Sigma}_B^l := \frac{1}{N} \sum_{k=1}^{K} n_k \left(\boldsymbol{\mu}_k^l - \boldsymbol{\mu}^l \right) \left(\boldsymbol{\mu}_k^l - \boldsymbol{\mu}^l \right)^T, \tag{6}$$

where

$$\boldsymbol{\mu}_k^l := \frac{1}{n_k} \sum_{i=1}^{n_k} \boldsymbol{z}_{k,i}^l, \quad \boldsymbol{\mu}^l := \frac{1}{K} \sum_{k=1}^{K} \boldsymbol{\mu}_k^l \tag{7}$$

denote the mean of the l-th layer's features in the k-th class and the global mean of the l-th layer's features, respectively. Equipped with the above setup, we can measure the compression of features within the same class and the discrimination of features between different classes using the following metrics.

Definition 1 (Intermediate layer-wise feature compression and discrimination) *For all $l = 0, 1, \dots, L-1$, we say that*

$$C_l = \frac{\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)}{\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)} \quad \textit{and} \quad D_l = 1 - \max_{k \neq k'} \frac{\langle \boldsymbol{\mu}_k^l, \boldsymbol{\mu}_{k'}^l \rangle}{\|\boldsymbol{\mu}_k^l\| \|\boldsymbol{\mu}_{k'}^l\|} \tag{8}$$

are the metrics of within-class compression and between-class discrimination of intermediate features at the l-th layer, respectively.

From now on, we will use these two metrics to study the evolution of features across layers. Intuitively, the features in the same class at the l-th layer are more compressed if C_l decreases, while the features from different classes are more discriminative if D_l increases. Before we proceed, let us delve deeper into the rationale behind each metric.

Discussion on the metric of feature compression. The study of feature compression has recently caught great attention in both supervised (Yu et al., 2020; Fang et al., 2021; Papyan et al., 2020) and unsupervised (Shwartz Ziv and LeCun, 2024) deep learning. For our definition of feature compression in (8), the numerator $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$ of the metric C_l measures how well the features from the same class are compressed towards the class mean at the l-th layer. More precisely, the features of each class are more compressed around their respective means as $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$ decreases. The denominator $\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)$ prevents reporting spuriously small values of C_l in near-collapse cases where all features approach a single mean, while simultaneously serving as a normalization factor that renders the metric invariant to feature

scaling. Specifically, given some weights $\boldsymbol{W}_1, \ldots, \boldsymbol{W}_l$ with $C_l = \mathrm{Tr}(\boldsymbol{\Sigma}_W^l)/\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)$, if we scale them to $t\boldsymbol{W}_1, \ldots, t\boldsymbol{W}_l$ for some $t > 0$, then C_l does not change, while the corresponding numerator becomes $t^l \mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$. It should be noted that this metric and similar ones have been studied in recent works (Kothapalli et al., 2023; Rangamani et al., 2023; Tirer et al., 2023; Yaras et al., 2023). For instance, Tirer et al. (2023) employed this metric to measure the variability of within-class features to simplify theoretical analysis. Moreover, a similar metric $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l \boldsymbol{\Sigma}_B^{l\dagger})$ has been used to characterize within-class variability collapse in recent studies on neural collapse in terms of last-layer features (Fang et al., 2021; Papyan et al., 2020; Rangamani et al., 2023; Yaras et al., 2022; Zhu et al., 2021). Our studied metric C_l can be viewed as its simplification. Additionally, He and Su (2023) employed $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l \boldsymbol{\Sigma}_B^{l\dagger})$ to measure how well the data are separated across intermediate layers. Due to their similarity, C_l can also serve as a metric for measuring data separation.

Discussion on the metric of feature discrimination. It is worth pointing out that learning discriminative features has a long history, tracing back to unsupervised dictionary learning (Donoho and Elad, 2003; Arora et al., 2014; Sun et al., 2016; Qu et al., 2020; Zhai et al., 2020). For our definition of feature discrimination, $\arccos\left(\langle \boldsymbol{\mu}_k^l, \boldsymbol{\mu}_{k'}^l \rangle / (\|\boldsymbol{\mu}_k^l\| \|\boldsymbol{\mu}_{k'}^l\|)\right)$ computes the angle between class means $\boldsymbol{\mu}_k^l$ and $\boldsymbol{\mu}_{k'}^l$. This, together with (8), indicates that D_l measures the feature discrimination by calculating the smallest angles among feature means of all pairs. Moreover, we can equivalently rewrite D_l in (8) as

$$D_l = \frac{1}{2} \min_{k \neq k'} \left\| \frac{\boldsymbol{\mu}_k^l}{\|\boldsymbol{\mu}_k^l\|} - \frac{\boldsymbol{\mu}_{k'}^l}{\|\boldsymbol{\mu}_{k'}^l\|} \right\|^2 .$$

This indicates that D_l computes the smallest distance between normalized feature means. According to these two interpretations, features between classes become more discriminative as D_l increases. Recently, Masarczyk et al. (2023) considered a variant of the inter-class variance $\sum_{k=1}^K \sum_{k' \neq k} \|\boldsymbol{\mu}_k - \boldsymbol{\mu}_{k'}\|^2$ to measure linear separability of representations of deep networks.

3. Main Results

In this section, we present our main theoretical results based on Section 2, first describing the theorem in Section 3.1 and then discussing the assumptions in Section 3.2.

3.1 Main Theorem

Before we present the main theorem, we make the following assumptions on the input data and the weights of the DLN in (2).

Assumption 2 *For the data matrix $\boldsymbol{X} \in \mathbb{R}^{d \times N}$, the data dimension is no smaller than the number of samples, i.e., $d \geq N$. Moreover, the data is θ-nearly orthonormal, i.e., there exists an $\theta \in [0, 1/4)$ such that*

$$\left| \|\boldsymbol{x}_i\|^2 - 1 \right| \leq \frac{\theta}{N}, \ |\langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle| \leq \frac{\theta}{N}, \ \textit{for all } 1 \leq i \neq j \leq N, \tag{9}$$

where $\boldsymbol{x}_i$ denotes the i-th column of $\boldsymbol{X}$.

Discussion on Assumption 2. We make this assumption primarily to simplify our analysis. It can be relaxed with a more refined analysis, and in practice, it may even be violated in empirical scenarios. Here, the condition $d \geq N$ guarantees that $\boldsymbol{X}$ is linearly separable in the sense that there exists a linear classifier $\boldsymbol{W} \in \mathbb{R}^{K \times d}$ such that $\boldsymbol{W}\boldsymbol{x}_{k,i} = \boldsymbol{y}_k$ for all i, k. We should point out that the same condition has been studied in (Chatterji and Long, 2023; Chatterji et al., 2022; Frei et al., 2023), and similar linear separability conditions have been widely used for studying implicit bias of gradient descent (Nacson et al., 2019; Phuong and Lampert, 2020; Soudry et al., 2018). Notably, this condition also holds for nonlinear networks in the sense that the intermediate features generated by the initial layers exhibit linear separability as shown in Alain and Bengio (2017); Ansuini et al. (2019); Masarczyk et al. (2023); Recanatesi et al. (2019) (see Figure 1, where the near-perfect accuracy at intermediate layers shows that the features at that layer are already linearly separable.). In addition, nearly orthonormal data is commonly used in the theoretical analysis of learning dynamics for training neural networks; see, e.g., Boursier et al. (2022); Frei et al. (2023); Phuong and Lampert (2020). In particular, this condition holds with high probability for well-conditioned Gaussian distributions (Frei et al., 2023), and it generally applies to a broad class of subgaussian distributions, as demonstrated in Claim 3.1 of Hu et al. (2020).

Implicit bias of GD. Since the DLN in Problem (3) is over-parameterized, it has infinitely many solutions satisfying $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. However, GD for training networks typically has an implicit bias towards certain solutions with benign properties (Arora et al., 2019; Gunasekar et al., 2017; Ji and Telgarsky, 2019; Min et al., 2021; Shah et al., 2020; Soudry et al., 2018). In particular, prior work has demonstrated that, with assumptions on network initialization and the dataset, gradient flow tends to favor solutions with minimum norms and balanced weights; see, e.g., Min et al. (2021); Chatterji and Long (2023); Arora et al. (2018b); Du et al. (2018). Recent studies also reveal that GD primarily updates a minimal invariant subspace of the weight matrices, thereby preserving the approximate low-rankness of the weights across all layers (Huh et al., 2023; Yaras et al., 2023). Based on these findings, we assume that the trained weights $\boldsymbol{\Theta}$ satisfy the following benign properties to investigate how trained deep networks hierarchically transform input data into labels.

Assumption 3 *For an L-layer DLN with weights $\boldsymbol{\Theta} = \{\boldsymbol{W}_l\}_{l=1}^{L}$ described in (2) with $d_l = d > 2K$ for all $l \in [L-1]$, the weights $\boldsymbol{\Theta}$ satisfy*
(i) Minimum-norm solution:

$$\boldsymbol{W}_{L:1} = \boldsymbol{Y}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T. \tag{10}$$

(ii) δ-Balancedness: *There exists a constant $\delta > 0$ such that*

$$\boldsymbol{W}_{l+1}^T\boldsymbol{W}_{l+1} = \boldsymbol{W}_l\boldsymbol{W}_l^T, \forall l \in [L-2],\ \|\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\|_F \leq \delta. \tag{11}$$

(iii) ε-Approximate low-rank weights: *There exist positive constants $\varepsilon \in (0,1)$ and $\rho \in [0, \varepsilon)$ such that for all $l \in [L-1]$,*

$$\varepsilon - \rho \leq \sigma_i(\boldsymbol{W}_l) \leq \varepsilon,\ \forall i = K+1, \ldots, d-K. \tag{12}$$

We defer the discussion of Assumption 3 to Section 3.2, where we provide both theoretical and empirical evidence in support of it. It is worth noting that Assumption 3 serves as an initial framework for understanding progressive feature behavior and can be relaxed to more general conditions. For example, a follow-up work (Jacot et al., 2025, Theorem 3.1) proves related but narrower results under weakened versions of the conditions in Assumption 3 (e.g., (11) holds only approximately). Building upon the above assumptions, we now present our main theorem on hierarchical representations in terms of feature compression and discrimination.

Theorem 4 *Consider a K-class classification problem on the training data $(\boldsymbol{X}, \boldsymbol{Y}) \in \mathbb{R}^{d\times N} \times \mathbb{R}^{K\times N}$, where the matrix $\boldsymbol{X}$ satisfies Assumption 2 with parameter θ. Suppose that we train an L-layer DLN with weights $\boldsymbol{\Theta} = \{\boldsymbol{W}_l\}_{l=1}^{L}$ such that $\boldsymbol{\Theta}$ satisfies Assumption 3, with the parameters $(\delta, \rho, \varepsilon)$ of weight balancedness and low-rankness satisfying*

$$\delta \le \min\left\{\frac{(2n)^{1/L}}{30L^2}, \frac{n^{1/L}}{128\sqrt{K}}, \frac{1}{16\sqrt{K}}\right\},\ \varepsilon \le \min\left\{\frac{n^{\frac{1}{2L}}}{4}, 1\right\},\ \rho \le \frac{\varepsilon}{2L\sqrt{n}}. \tag{13}$$

(i) Progressive within-class feature compression: *For C_l in Definition 1, it holds that*

$$\frac{c\varepsilon^2}{\kappa(2n)^{1/L}} \le \frac{C_1}{C_0} \le \frac{2\kappa\varepsilon^2}{c(n/2)^{1/L}}, \tag{14}$$

$$\frac{c\varepsilon^2}{\kappa(2n)^{1/L}} \le \frac{C_{l+1}}{C_l} \le \frac{\kappa\varepsilon^2}{c(n/2)^{1/L}},\ \forall l \in [L-2], \tag{15}$$

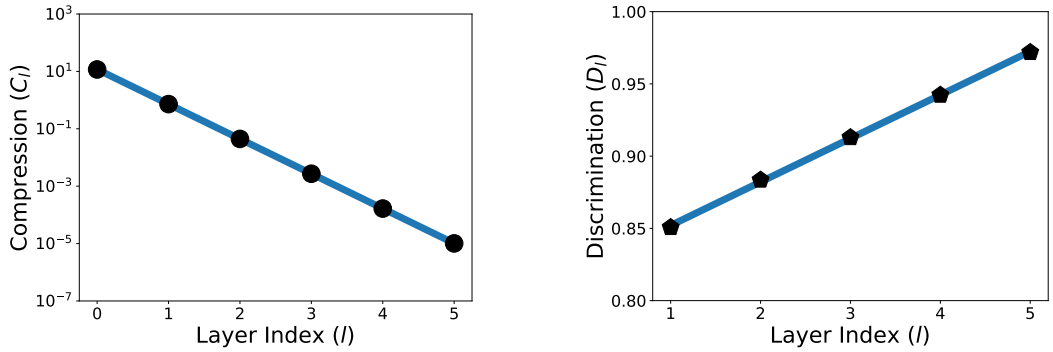
where

$$c = \frac{(n-3)K-1}{(n-1)K+1},\ \kappa = \frac{1+n^{-\Omega(1)}}{1-n^{-\Omega(1)}}.$$

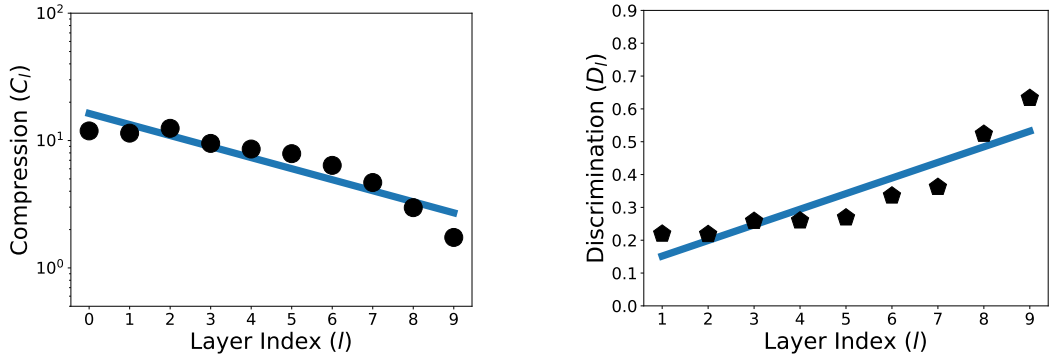
(ii) Progressive between-class feature discrimination: *For D_l in Definition 1, for all $l \in [L-1]$, we have*

$$D_l \ge 1 - 32\,(\theta + 4\delta)\left(2 - \frac{l+1}{L}\right) - n^{-\Omega(1)}. \tag{16}$$

We defer the detailed proof to Appendix A. As we discussed in Section 1, numerous empirical studies have been conducted to explore the feature structures of intermediate and final layers in deep networks, particularly concerning feature compression and discrimination; see, e.g., Alain and Bengio (2017); Ansuini et al. (2019); Ben-Shaul and Dekel (2022); Chen et al. (2022b); Galanti et al. (2022a); He and Su (2023); Hui et al. (2022); Masarczyk et al. (2023); Rangamani et al. (2023); Recanatesi et al. (2019); Zhang et al. (2022). However, there remains a scarcity of theoretical analysis to elucidate their observations. Despite the acknowledged limitations of DLNs, our work takes the first step towards rigorously developing a unified framework to quantify feature compression and discrimination across different layers. Specifically, Theorem 4 shows that, given input data satisfying Assumption 2 and a trained DLN with weights satisfying Assumption 3, features within the same class are compressed at a *geometric rate* on the order of $\varepsilon^2/n^{1/L}$ when n is sufficiently large, while



(a) A 6-layer **linear** network

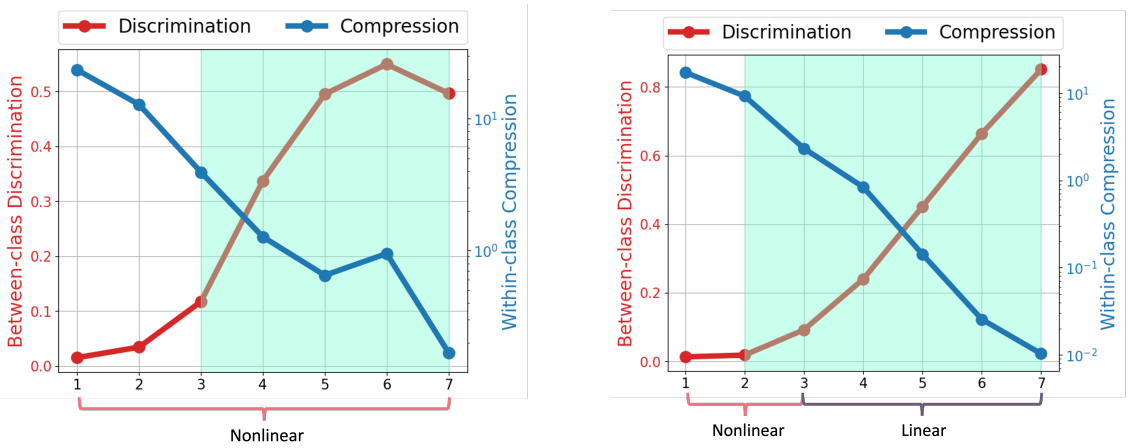


(b) A 10-layer **nonlinear** network

Figure 5: **Progressive feature compression and discrimination on both linear and nonlinear networks.** We plot the feature compression and discrimination metrics defined in (8) for $l = 1, \ldots, L-1$ on both the linear network (top row) and nonlinear network (bottom row). We train both networks using a nearly orthogonal dataset as described in Assumption 2, initializing the network weights satisfying (18), with an initialization scale of $\xi = 0.3$. We train both networks via gradient descent until convergence. In each figure, the x-axis denotes the number of layers from shallow to deep, with layer-0 denoting the inputs. In the left figures, the y-axis denotes the compression measure C_l in the logarithmic scale; In the right figures, the y-axis denotes the discrimination measure D_l. More experimental details can be found in Section 5.2.2.

the features between classes are discriminated at a *linear rate* on the order of $1/L$ across layers. In Figure 5, we also show empirical support for these findings in Figure 5 (top figures), where we simulate data and train a linear network according to our assumptions and plot the metrics at each layer. This phenomenon also appears in nonlinear networks, as shown in Figure 5 (bottom figures), where we train a nonlinear MLP with the same synthetic training data and plot the metrics at each layer. In the following, we discuss the implications of our main result.

From linear to nonlinear networks. Although our result is rooted in DLNs, it provides valuable insights into the feature evolution in nonlinear networks. Specifically, the linear separability of features learned by initial layers in nonlinear networks (see Figure 1) allows



(a) A 8-layer MLP network with ReLU

(b) A 8-layer hybrid network

Figure 6: **Within-class compression and between-class discrimination of features of two 8-layer networks.** We use the same setup as in Figure 1 and plot the metrics of within-class compression and between-class discrimination (see Definition 1) of the two networks across layers after training, respectively. Additional experimental details are deferred to Section 5.1.1.

later layers to be effectively replaced by linear counterparts for compressing within-class features and discriminating between-class features. We empirically support this claim in Figure 6, where we observe that both the linear and subsequent nonlinear layers exhibit consistent trends of feature compression and discrimination with respect to depth. Therefore, studying DLNs helps us understand the role of the nonlinear layers after the initial layers in nonlinear networks for learning features. This understanding also sheds light on the pattern where within-class features compress at a geometric rate and between-class features discriminate at a linear rate in nonlinear networks, as illustrated in Figure 5 (bottom figures). A natural direction is to extend our current analysis framework to nonlinear networks, especially homogeneous neural networks (Lyu and Li, 2020).

Neural collapse beyond the unconstrained feature model. One important implication of our result is that it addresses an open problem about NC. Specifically, almost all existing works assume the *unconstrained features model* (Mixon et al., 2020; Papyan et al., 2020) to analyze the NC phenomenon, where the last-layer features of the deep network are treated as free optimization variables to simplify interactions across layers; see, e.g., Zhu et al. (2021); Wang et al. (2022); Zhou et al. (2022b); Yaras et al. (2022); Zhou et al. (2022a); Li et al. (2024a). However, a major drawback of this model is that it overlooks the hierarchical structure of deep networks as well as the structure of the input data. In this work, we address this issue without assuming the unconstrained feature model. Specifically, according to Theorem 4, for a sufficiently deep linear network trained on nearly orthogonal data, the last-layer features within the same class concentrate around their class means, while the last-layer features from different classes are nearly orthogonal to each other. This

directly implies that the last-layer features exhibit variability collapse and convergence to simplex equiangular right frame approximately after centralization.

Since the submission of our manuscript, several follow-up works have investigated NC beyond the unconstrained feature model in different settings; see, e.g., Hong and Ling (2024); Wang et al. (2024); Jacot et al. (2025); Kothapalli and Tirer (2025); Xu et al. (2023). In particular, Hong and Ling (2024) studied 2-layer and 3-layer ReLU neural networks and investigated the impact of network width, depth, data dimension, and statistical properties of input data on NC. Wang et al. (2024) characterized the geometric properties of intermediate layers of ResNet and proposed a conjecture on progressive feedforward collapse. Recently, Jacot et al. (2025) studied a hybrid network with nonlinear layers followed by at least two linear layers and provided generic guarantees on progressive NC. It is worth noting that their derived conditions extend those in Assumption 3 and are slightly more general.

Guidance on network architecture design. Progressive feature compression and discrimination in Theorem 4 provide guiding principles for network architecture design. Specifically, according to (14), (15), and (16), features are more compressed within the same class and more discriminated between different classes, improving the separability of input data as the depth of the network increases. This is also supported by our experiments in Figure 4. This indicates that the network should be deep enough for effective data separation in classification problems. However, it is worth noting that the belief that deeper networks are better is not always true. Indeed, it becomes increasingly challenging to train a neural network as it gets deeper, especially for DLNs (Glorot and Bengio, 2010). Moreover, over-compressed features may degrade the out-of-distribution performance of deep networks as shown in Masarczyk et al. (2023). This, together with our result, indicates that neural networks should not be too deep for improved out-of-distribution generalization performance.

Understanding projection heads for transfer learning. In contrastive learning (Chen et al., 2020; Chen and He, 2021), a successful empirical approach to improving transfer learning performance in downstream tasks involves the use of projection heads (Li et al., 2024b). These projection heads, typically consisting of one or several MLP layers, are added between the feature extractor and the final classifier layers during pre-training. For downstream tasks, the projection head is discarded and only the features learned by the feature extractor are utilized for transfer learning. Recent works (Li et al., 2024b; Kornblith et al., 2021; Galanti et al., 2022b; Xie et al., 2022) have established an empirical correlation between the degree of feature collapse during pre-training and downstream performance: less feature collapse leads to more transferable models. However, it remains unclear why features prior to projection heads exhibit less collapse and greater transferability. Our study addresses this question and offers a theoretical insight into the utilization of projection heads. According to Theorem 4, it becomes apparent that features from the final layers tend to be more collapsed than the features before the projection head. This, together with the empirical correlation between feature collapse and transferability, implies that leveraging features before projection heads improves transferability. We provide further empirical evidence with experiments in Section 5.3.2. Moreover, the progressive compression pattern in Theorem 4 also provides insight into the phenomenon studied in Yosinski et al. (2014),

which suggests that deeper layers in a neural network become excessively specialized for the pre-training task, consequently limiting their effectiveness in transfer learning.

3.2 Discussions on Assumption 3

In this subsection, we justify the properties of trained network weights in Assumption 3 using both theoretical and experimental findings. While the solution described in Assumption 3 is only one of infinitely many global optima for Problem (3), the implicit bias of GD ensures that, with proper initialization, iterations almost always converge to the desired solution. In the following discussion, we delve into this phenomenon in greater depth.

GD with orthogonal initialization. We consider training a DLN for solving Problem (3) by GD, i.e., for all $l \in [L]$,

$$\boldsymbol{W}_l(t+1) = \boldsymbol{W}_l(t) - \eta \frac{\partial \ell(\boldsymbol{\Theta}(t))}{\partial \boldsymbol{W}_l},\ t \geq 0. \tag{17}$$

Notably, when the learning rate is infinitesimally small, i.e., $\eta \to 0$, GD in (17) reduces to gradient flow. Moreover, we initialize the weight matrices $\boldsymbol{W}_l$ for all $l \in [L]$ using ξ-scaled orthogonal matrices for a constant $\xi > 0$, i.e.,

$$\boldsymbol{W}_l(0)^T \boldsymbol{W}_l(0) = \xi^2 \boldsymbol{I}_d, \forall l \in [L-1], \quad \boldsymbol{W}_L(0) = [\xi \boldsymbol{U}\ \boldsymbol{0}], \tag{18}$$

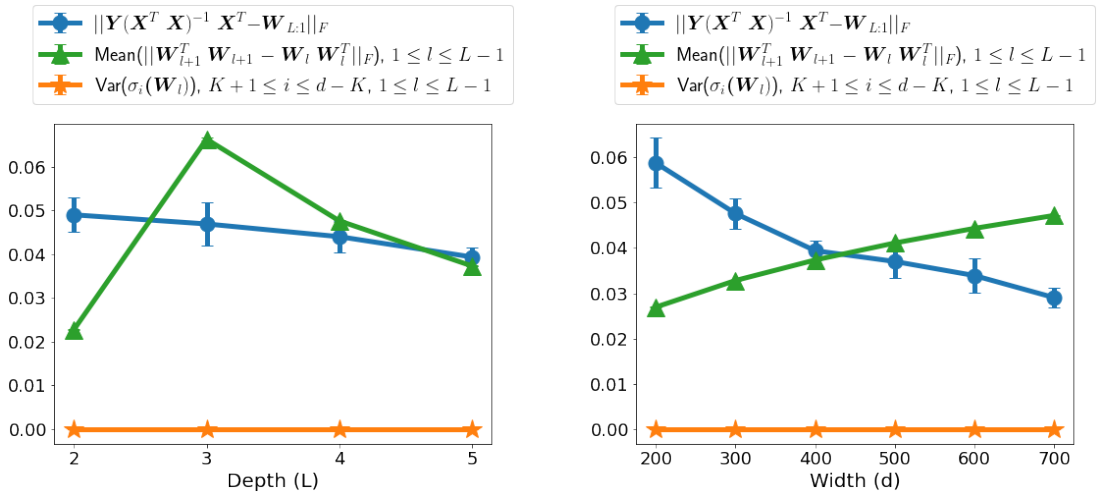
where $\boldsymbol{U} \in \mathcal{O}^K$. It is worth noting that orthogonal initialization is widely used to train deep networks, which can speed up the convergence of GD; see, e.g., Pennington et al. (2018); Xiao et al. (2018); Hu et al. (2019).

Theoretical justification of Assumption 3. Theoretically, we can prove the conditions outlined in Assumption 3 when the gradient flow is trained on a square and orthogonal data matrix using the results in Arora et al. (2018b); Yaras et al. (2023, 2024); Kwon et al. (2024).

Proposition 5 *Suppose that the data matrix $\boldsymbol{X} \in \mathbb{R}^{d \times N}$ is square, i.e., $d = N$, and satisfies Assumption 2 with $\theta = 0$. Suppose in addition that we apply GD (17) to solve Problem (3) with $\eta \to 0$ and the initialization in (18). If GD converges to a global optimal solution, then (10), (11), and (12) hold with $\delta = \xi^2\sqrt{d-K}$, $\varepsilon = \xi$, and $\rho = 0$.*

The proof of this proposition is deferred to Appendix A.5. Although we can only prove Assumption 3 in this restrictive setting, we see empirically that it seems to hold in more general settings. We substantiate this claim by drawing support from empirical evidence and existing findings in the literature, as elaborated in the following.

Empirical justifications of Assumption 3. Here, we run GD (17) using the initiation (18) on nearly orthogonal data satisfying Assumption 2. We refer the reader to Section 5.2 for the experimental setup. After training, we plot the metrics of minimum norm residual, balancedness residual, and variance of singular values (see the legends in Figure 7) against depth (resp. width) in Figure 7(a) (resp. Figure 7(b)). It is observed that the magnitudes of these metrics are very small for different depths and widths of neural networks. This indicates that Assumption 3 approximately holds for GD in a broader setting.



(a) Fix width $d = 400$, change depth L

(b) Fix depth $L = 5$, change width d

Figure 7: **Assumption 3 holds approximately among different network configurations.** We train DLNs using the orthogonal initialization with varying widths and depths. For each network width d and depth L, we show the minimum-norm residual, the average weight balancedness residual of all weights, and the variance of the singular values of all weights (see Equation (20) for details) for the DLNs that have been trained until convergence. Each data point represents the mean value of 10 runs with different random seeds, with error bars indicating standard deviation. Here, we plot these metrics in (a) with fixed network width d and varying depth L and (b) with fixed network depth $L = 5$ and varying network width d. More experimental details are deferred to Section 5.2.1.

Prior arts on implicit bias support Assumption 3. Moreover, Assumption 3 is also well supported by many related results in the literature.

- **Minimum-norm solutions and balanced weights.** It has been extensively studied in the literature that GD with proper initialization converges to minimum-norm and balanced solutions, especially in the setting of gradient flow (Arora et al., 2018b; Chatterji and Long, 2023; Du et al., 2018; Min et al., 2021). Specifically, Du et al. (2018); Arora et al. (2018b) proved that for linear networks, the iterates of gradient flow satisfy

$$\frac{d}{dt}\left(\boldsymbol{W}_l(t)\boldsymbol{W}_l(t)^T - \boldsymbol{W}_{l+1}(t)^T\boldsymbol{W}_{l+1}(t)\right) = \boldsymbol{0}, \ \forall l \in [L-1], \tag{19}$$

  where t denotes a continuous time index. This, together with (18), implies that (11) holds with $\delta = \xi\sqrt{d-K}$. Moreover, using the result in Min et al. (2021), if we initialize the weights of two-layer networks as in (18) with $\boldsymbol{W}_2(0)\boldsymbol{W}_1(0) \in \text{span}(\boldsymbol{X})$, gradient flow always yields the minimum-norm solution upon convergence. However, the conditions (10) and (11) are rarely studied in the context of GD for training deep networks. In this context, we empirically verify these two conditions as shown in Figure 7 under different network configurations.

- **Approximate low-rankness of the weights.** Recently, numerous studies have demonstrated that GD exhibits a bias towards low-rank weights (Arora et al., 2019; Gunasekar et al., 2017; Yaras et al., 2023; Kwon et al., 2024). In particular, Yaras et al. (2023); Kwon et al. (2024) showed that when the input data $\boldsymbol{X}$ is orthonormal, learning dynamics of GD for DLNs in (17), with the initialization in (18), only updates an invariant subspace of dimension $2K$ for each weight matrix across all layers, where the invariant subspace is spanned by the singular vectors associated with the K-largest and K-smallest singular values, and the rest singular values remain unchanged. More experimental demonstration can be found in Appendix B.

4. Relationship to Prior Arts

In this section, we discuss the relationship between our results and prior works on the empirical and theoretical study of deep networks.

Hierarchical feature learning in deep networks. Deep networks, organized in hierarchical layers, can perform effective and automatic feature learning (Allen-Zhu and Li, 2023), where the layers learn useful representations of the data. To better understand hierarchical feature learning, plenty of studies have been conducted to investigate the structures of features learned at intermediate layers. One line of these works is to investigate the neural collapse (NC) properties at intermediate layers. For example, Tirer and Bruna (2022) extended the study of neural collapse to three-layer nonlinear networks with the MSE loss, showing that the features of each layer exhibit neural collapse. Dang et al. (2023) generalized the study of neural collapse for DLNs with imbalanced training data, and they drew a similar conclusion to Tirer and Bruna (2022) that the features of each layer are collapsed. However, these results are based on the unconstrained features model and thus cannot capture the input-output relationship of the network. Moreover, the conclusion they draw on the collapse of intermediate features is far from what we observe in practice. Indeed, it is seen for both linear and nonlinear networks in Figure 5 that the intermediate features are progressively compressed across layers rather than exactly collapsing to their means for each layer. In comparison, under the assumption that the input $\boldsymbol{X}$ is nearly orthogonal in Assumption 2, our result characterizes the progressive compression and discrimination phenomenon to capture the training phenomena on practical networks as demonstrated in Figure 5.

On the other hand, another line of work (Yu et al., 2020; Chan et al., 2022; Ma et al., 2022; Yu et al., 2024) argues that deep networks prevent within-class feature compression while promoting between-class feature discrimination, contrasting with our study where features are compressed across layers. Such a difference can be attributed to the choice of loss function. In our work, we focused on the study of the commonly used MSE loss, with the goal of understanding a prevalent phenomenon in classical training of deep networks. It has been empirically shown that increasing feature compression is beneficial for improving in distribution generalization and robustness (Papyan et al., 2020; Li et al., 2024b; Ben-Shaul and Dekel, 2022; He and Su, 2023; Chen et al., 2022a). In comparison, Yu et al. (2020) introduced a new maximum coding rate reduction loss that is intentionally designed to prevent feature compression.

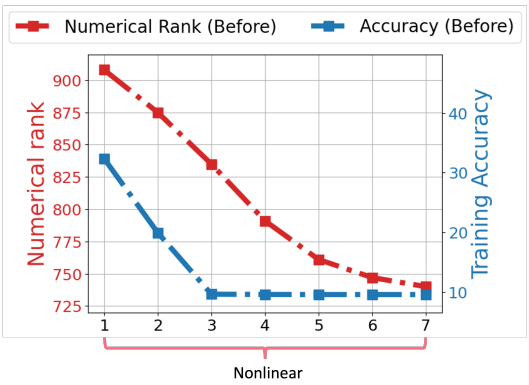
Learning dynamics and implicit bias of GD for training deep networks. Our main result in Theorem 4 is based on Assumption 3 for the trained weights. As discussed in Section 3.2, Assumption 3 holds as a consequence of results in recent works on analyzing the learning dynamics and implicit bias of gradient flow or GD in training deep networks. We briefly review the related results as follows. For training DLNs, Arora et al. (2018a) established linear convergence of GD based upon whitened data and with a similar setup to ours. For Gaussian initialization, Du and Hu (2019) showed that GD also converges globally at a linear rate when the width of hidden layers is larger than the depth, whereas Hu et al. (2019) demonstrated the advantage of orthogonal initialization over random initialization by showing that linear convergence of GD with orthogonal initialization is independent of the depth. More recent developments can be found in Gidel et al. (2019); Nguegnang et al. (2021); Shin (2022) for studying GD dynamics. On the other hand, another line of works focused on studying gradient flow for learning DLNs due to its simplicity (Bah et al., 2022; Eftekhari, 2020; Min et al., 2021; Tarmoun et al., 2021; Bah et al., 2022), by analyzing its convergence behavior.

Numerous studies have shown that the effectiveness of deep learning is partially due to the implicit bias of its learning dynamics, which favors some particular solutions that generalize exceptionally well without overfitting in the over-parameterized setting (Belkin et al., 2019; Huh et al., 2023; Neyshabur, 2017). To gain insight into the implicit bias of GD for training deep networks, a line of recent work has shown that GD tends to learn simple functions (Cao et al., 2023; Gunasekar et al., 2017; Ji and Telgarsky, 2018; Kunin et al., 2022; Shah et al., 2020; Valle-Perez et al., 2018). For instance, some studies have shown that GD is biased towards max-margin solutions in linear networks trained for binary classification via separable data (Soudry et al., 2018; Kunin et al., 2022). In addition to the simplicity bias, another line of work showed that deep networks trained by GD exhibit a bias towards low-rank solutions (Gunasekar et al., 2017; Huh et al., 2023; Yaras et al., 2023). The works (Gidel et al., 2019; Arora et al., 2019) demonstrated that adding depth to matrix factorization enhances an implicit tendency towards low-rank solutions, leading to more accurate recovery.

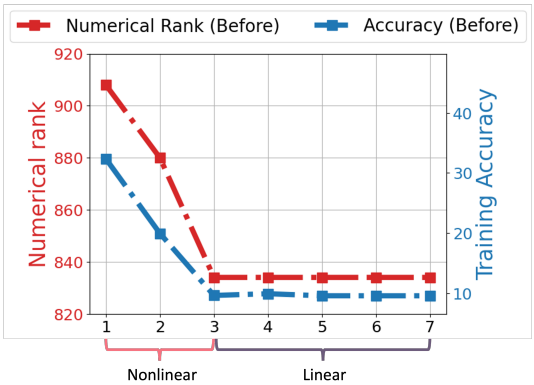
5. Experimental Results

In this section, we conduct various numerical experiments to validate our assumptions, verify our theoretical results, and investigate the implications of our results on both synthetic and real data sets. All of our experiments are conducted on a PC with 8GB memory and an Intel(R) Core i5 1.4GHz CPU, except for those involving large datasets such as CIFAR and FashionMNIST, which are conducted on a server equipped with NVIDIA A40 GPUs. Our code is implemented in Python and made available at `https://github.com/Heimine/PNC_DLN`. Throughout this section, we will repeatedly use multilayer perception (MLP) networks, where each layer of MLP networks consists of a linear layer and a batch norm layer (Ioffe and Szegedy, 2015) followed by ReLU activation. In addition, in accordance with our theoretical analysis, the compression metric is plotted for layers $0, 1, \ldots, L-1$, while the discrimination metric is plotted for layers $1, \ldots, L-1$.

The remaining sections are organized as follows. In Section 5.1, we provide detailed experimental setups for the results discussed in Section 1 and shown in Figures 6, 8, and

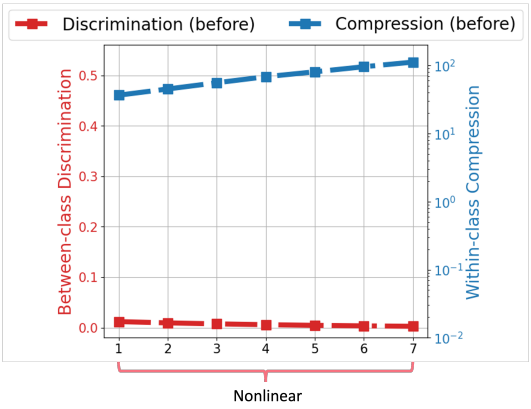


(a) A 8-layer MLP network with ReLU

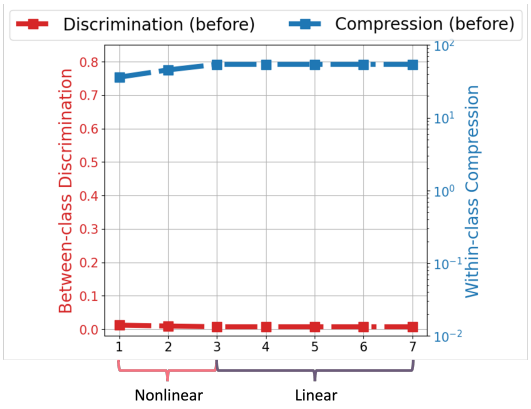


(b) A 8-layer hybrid network

Figure 8: **Accuracy and numerical rank of two 8-layer networks before training.** We use the same setup as in Figures 1 and 6 but instead plot the numerical rank and training accuracy of the two networks across layers **before** training.



(a) A 8-layer MLP network with ReLU



(b) A 8-layer hybrid network

Figure 9: **Within-class compression and Between-class discrimination of two 8-layer networks before training.** We use the same setup as in Figure 1 and Figure 6 but instead plot the compression and discrimination statistics of the two networks across layers **before** training.

9. In Section 5.2, we provide experimental results to support Theorem 4 in Section 3 and validate Assumption 3 discussed in Section 3.2. Finally, we empirically explore the implications of our results in Section 5.3 beyond our assumptions.

5.1 Significance of Studying DLNs in Feature Learning

In this subsection, our goal is to empirically demonstrate the significance of studying DLNs for feature learning as discussed in Section 1. To achieve this, we investigate (i) the roles of linear layers and MLP layers at deep layers in feature learning and (ii) the importance of the depth of DLNs for generalization.

5.1.1 Linear Layers Mimic Deep Layers in MLPs for Feature Learning

In this subsection, we study the roles of linear layers and MLP layers in deep layers for feature learning. To begin, we provide the experimental setup for our experiments.

Network architectures. In these experiments, we construct two 8-layer networks: (a) a nonlinear MLP network and (b) a hybrid network formed by a 3-layer MLP followed by a 5-layer linear network. In both cases, the final layer (L-th layer) serves as a linear classifier. We set the hidden dimension $d = 1024$ for all linear layers.

Training dataset and training methods. We employ the SGD optimizer to train the networks by minimizing the MSE loss on the CIFAR-10 dataset (Krizhevsky and Hinton, 2009). For the settings of the SGD optimizer, we use a momentum of 0.9, a weight decay of 10^{-4}, and a dynamically adaptive learning rate ranging from 10^{-3} to 10^{-5}, modulated by a CosineAnnealing learning rate scheduler as detailed in Loshchilov and Hutter (2017). We use the orthogonal initialization in (18) with $\xi = 0.1$ to initialize the network weights. The neural networks are trained for 400 epochs with a batch size of 128.

Experiments and observations. Now, we elaborate on the tasks conducted in Figures 1, 2, 6, 8, and 9 and draw conclusions from our observations.

- In Figure 1 (resp. Figure 8), we plot the numerical rank and training accuracy of the features against the layer index after training (resp. before training). Here, the numerical rank of a matrix is defined as the number of top singular values whose sum collectively accounts for more than 95% of its nuclear norm. In terms of training accuracy, we add a linear classification layer to a given layer l of the neural network and train this added layer using the cross-entropy loss to compute the classification accuracy. It is observed in Figure 1 that the training accuracy rapidly increases in the initial layers and nearly saturates in the deep layers, whereas the numerical rank increases in the initial layers and then decreases in the deep layers progressively in the trained networks. This observation suggests that the initial layers of a network create linearly separable features that can achieve accurate classification, while the subsequent layers further compress these features progressively.

- In Figure 2, we employ a 2D UMAP plot (McInnes et al., 2018) with the default settings in the UMAP Python package to visualize the evolution of features from shallow to deep layers. It is observed that in the first layer, features do not exhibit obvious structures in terms of feature compression and discrimination; the features from the same class are more and more compressed, while the features from different classes become more and more separable from layer 2 to layer 4; this within-class compression and between-class discrimination pattern is strengthened from layer 4 to layer 6. This visualization demonstrates that features in the same class are compressed, while the features from different classes are discriminated progressively.

- In Figure 6 (resp. Figure 9), we plot the metrics of within-class feature compression C_l and between-class feature discrimination D_l defined in Definition 1 against the layer index after training (resp. before training). According to Figure 6, we observe a consistent trend of decreasing C_l and increasing D_l against the layer index after training for both

networks, albeit the hybrid network exhibits a smoother transition in C_l and D_l compared to the fully nonlinear network. This observation supports our result in Theorem 4.

The role of linear layers in feature learning. Comparing Figure 1(a) to Figure 1(b) and Figure 6(a) to Figure 6(b), we conclude that linear layers play the same role as MLP layers in the deep layers of nonlinear networks in feature learning, compressing within-class features and discriminating between-class features progressively. Intuitively, this is because the representations from the initial layers are already linearly separable, hence we can replace the deeper nonlinear layers with DLNs to achieve the same functionality without sacrificing the training performance. Moreover, comparing the results after training (Figures 1 and 6) with those before training (Figures 8 and 9), we conclude that neural networks trained with SGD do not operate in the lazy-training regime described by the neural tangent kernel (NTK) theory, where network parameters remain close to their initialization and training behaves nearly linearly. This observation is consistent with findings from recent studies (Bietti et al., 2022; Damian et al., 2022).

5.1.2 Effects of Depth in Linear Layers for Improving Generalization

In this subsection, we study the impact of depth in deep networks for generalization. To begin, we provide the experimental setup for our experiments.

Network architectures. We consider a 2-layer MLP network as our base network and construct networks of varying depths from 3 to 9 by adding linear or MLP layers to this base network. We use the PyTorch default initialization (He et al., 2015) to initiate all weights.

Training dataset and training methods. We train these networks on the FashionMNIST (Xiao et al., 2017) and CIFAR-10 (Krizhevsky and Hinton, 2009) datasets, using the same training approach in Section 5.1.1, except that the initial learning rate is set as 10^{-2}. We use 5 different random seeds to initialize the network weights and report the average test accuracy.

Experiments and observations. After training, we report the average test accuracy throughout training against the number of layers of constructed networks in Figure 3. Notably, we observe a consistent improvement in test accuracy as the networks grow deeper, regardless of whether the added layers are linear or MLP layers. This observation provides further evidence for the resemblance between DLNs and the deeper layers in MLPs. Additionally, it shows that DLNs can benefit from increased depth, similar to their counterparts in traditional nonlinear networks.

5.2 Experimental Verification of Assumption 3 and Theorem 4

In this subsection, we conduct numerical experiments to verify our assumptions and theorem presented in Section 3. Unless otherwise specified, we use the following experimental setup.

Network architectures. We study DLNs and MLPs with ReLU activation with different widths and depths specified in each figure.

Training dataset. In all experiments, we fix the number of classes $K = 3$ and the number of training samples $N = 30$, with varying hidden dimensions $d \geq N$ as specified in each figure. To generate input data $\boldsymbol{X} \in \mathbb{R}^{d\times N}$ that satisfies Assumption 2, we first generate a matrix $\boldsymbol{A} \in \mathbb{R}^{d\times N}$ with entries i.i.d. sampled from the standard normal distribution and a matrix $\boldsymbol{B} \in \mathbb{R}^{d\times N}$ with entries i.i.d. sampled from the uniform distribution on $[0, 1]$. Then, we apply a compact SVD to $\boldsymbol{A}$ and set $\boldsymbol{X}_0 \in \mathbb{R}^{d\times N}$ as the left singular matrix. Next, we obtain a matrix $\boldsymbol{N}$ by normalizing $\boldsymbol{B}$ such that its Frobenius norm is 1. Finally, we generate $\boldsymbol{X}$ via $\boldsymbol{X} = \boldsymbol{X}_0 + \boldsymbol{N}$ such that $\boldsymbol{X}$ is nearly orthogonal satisfying Assumption 2.

Training method and weight initialization. Unless otherwise specified, we train networks using full batch GD in (17), with a fixed learning rate $\eta = 0.1$. We use orthogonal weight initialization as described in (18) with varying initialization scaling ξ. This initialization ensures that the weights at initialization meet the condition (11).

5.2.1 Experiments for Verifying Assumption 3

In this subsection, we conduct experiments to verify Assumption 3 and corroborate the discussions in Section 3.2 based on the above experimental setup. Specifically, we train DLNs using orthogonal initialization in (18) with $\xi = 0.1$ for various network depths and widths. Given a DLN, we train it 10 times with different random seeds. In each run, we terminate training once the training loss is less than 10^{-11}. We plot the following metrics over 10 runs against depth and width in Figure 7(a) and Figure 7(b), respectively:

$$\begin{aligned}
&\text{minimum-norm residual:} \quad ||\boldsymbol{Y}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T - \boldsymbol{W}_{L:1}||_F, \\
&\text{averaged balancedness:} \quad \frac{1}{L-1}\sum_{l=1}^{L-1} ||\boldsymbol{W}_{l+1}^T\boldsymbol{W}_{l+1} - \boldsymbol{W}_l\boldsymbol{W}_l^T||_F, \\
&\text{variance of singular values of all weights:} \quad \frac{1}{(L-1)(d-2K)}\sum_{l=1}^{L-1}\sum_{i=K+1}^{d-K}\left(\sigma_i(\boldsymbol{W}_l) - \mu\right)^2,
\end{aligned} \tag{20}$$

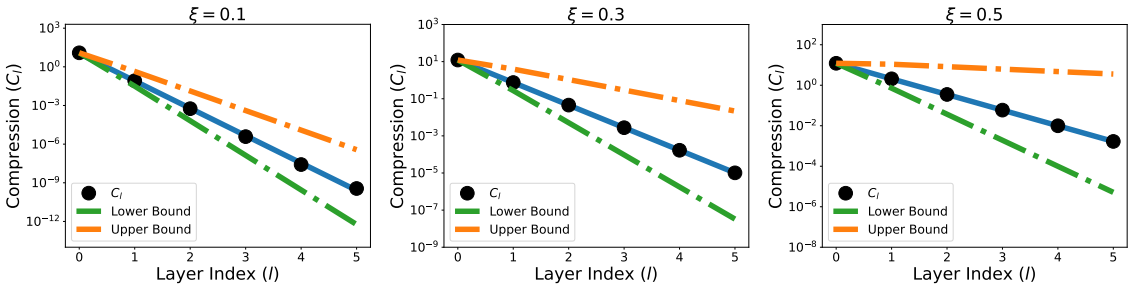
where

$$\mu = \frac{1}{(L-1)(d-2K)}\sum_{l=1}^{L-1}\sum_{i=K+1}^{d-K}\sigma_i(\boldsymbol{W}_l).$$

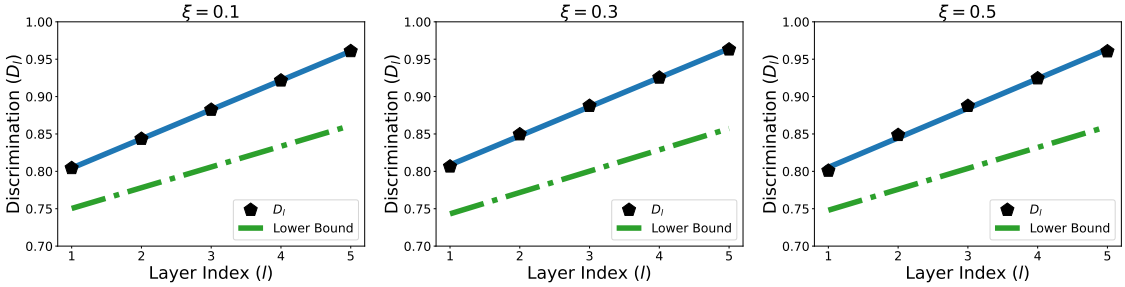
It can be observed from Figure 7 that the weights of a DLN with different depths and widths trained under our settings approximately satisfy Assumption 3. This supports our discussions in Section 3.2.

5.2.2 Experiments for Verifying Theorem 4

Now, we conduct experiments to validate Theorem 4. Specifically, we train 6-layer DLNs using GD with varying initialization scales $\xi = 0.1, 0.3, 0.5$, respectively. Then, we plot the metrics of within-class compression C_l and between-class discrimination D_l against layer index in Figure 10(a) and Figure 10(b), respectively. It is observed that the feature compression metric C_l decreases exponentially, while the feature discrimination metric D_l increases linearly w.r.t. the layer index. In the top row of Figure 10, the solid blue line



(a) The metric of within-class compression C_l



(b) The metric of between-class discrimination D_l

Figure 10: **Visualization of the metrics of within-class compression C_l and between-class discrimination across layers.** We train 6-layer DLNs with $\xi = 0.1, 0.3, 0.5$, respectively. In the top figures, we plot the metric of within-class compression C_l along with the corresponding theoretical upper and lower bounds given in (15). In the bottom figures, we plot the metric of between-class discrimination given in (16).

is plotted by fitting the values of C_l at different layers, and the green and orange dash-dotted lines are plotted according to the lower and upper bounds in (15), respectively. We can observe that the solid blue line is tightly sandwiched between the green and orange dash-dotted lines. This indicates that (14) and (15) provide a valid and tight bound on the decay rate of the feature compression metric. According to the bottom row of Figure 10, we conclude that (16) provides a valid bound on the growth rate of the feature discrimination metric.

Moreover, we respectively train 6-layer DLN and 10-layer MLP networks with hidden dimension $d = 50$ via GD with an initialization scale $\xi = 0.3$. In Figure 5, we plot the feature compression and discrimination metrics on both DLN and MLP networks, respectively. We observe that the exponential decay of feature compression and the linear increase of feature discrimination hold exactly in DLNs and approximately in nonlinear networks.

5.3 Exploratory Experiments

5.3.1 Empirical Results Beyond Theory

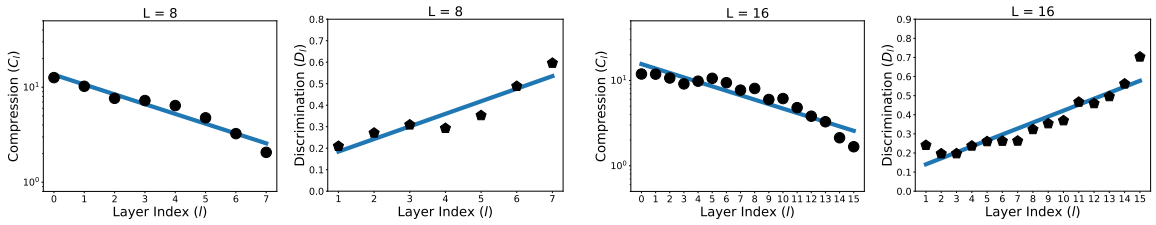
In this subsection, we conduct exploratory experiments to demonstrate the universality of our result. Unless otherwise specified, we use the experimental setup outlined at the beginning in Section 5.2 in these experiments. Here are our observed findings:

- **Progressive feature compression and discrimination in nonlinear deep networks.** We train 8-layer and 16-layer MLP networks, respectively. After training, we plot the metrics of feature compression and discrimination defined in (1) in Figure 11. It is observed that the feature compression metric C_l decays at an *approximate* geometric rate in nonlinear networks, while the feature discrimination metric D_l increases at an *approximate* linear rate. Additionally, it is worth mentioning that a similar "law of separation" phenomenon has been reported in He and Su (2023); Li et al. (2024b) on nonlinear networks, but their results are based upon a different metric of data separation.

- **Progressive feature compression and discrimination on DLNs with generic initialization.** In most of our experiments and discussion in Section 3.2, we mainly focused on orthogonal initialization, which simplifies our analysis due to induced weight balancedness across layers. To demonstrate the generality of our results, we also test the default initialization in the PyTorch package and train the DLN. As shown in Figure 12, we can observe that the compression metric C_l decays from shallow to deep layers at an *approximate* geometric rate with different network depth L. Additionally, the discrimination metric D_l increases at an *approximate* linear rate.

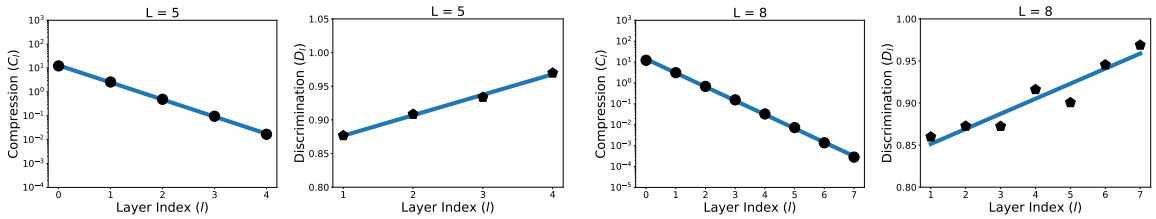
- **Progressive feature compression on real datasets.** We train a hybrid network on FashionMNIST and CIFAR datasets using the network architectures and training methods in Section 5.1.2 and plot the metric of within-class compression against layer index in Figure 4. Although we use real datasets and PyTorch default initialization, we can still observe that the within-class compression metric C_l decays progressively at an approximate geometric rate. This further demonstrates the universality of our studied phenomenon. Moreover, the results in Figure 4 also illustrate the role of depth, where the decay rates are approximately the same across all settings, and thus deeper networks lead to more feature compression.

Beyond these findings, we conduct additional experiments on the SST-5 text dataset (Socher et al., 2013) and the ImageNet (Deng et al., 2009) dataset to further support our claims. These results are reported in Appendix B.2.



(a) An 8-layer MLP. (b) A 16-layer MLP.

Figure 11: **Progressive feature compression and discrimination on nonlinear networks.** We train MLP networks using the default orthogonal weight initialization and plot the metrics C_l and D_l against layer index.



(a) A 5-layer DLN with Kaiming Init.

(b) An 8-layer DLN with Kaiming Init.

Figure 12: **Progressive feature compression and discrimination on DLNs with default PyTorch initialization.** We train DLNs with uniform weight initialization and plot the dynamics of C_l and D_l. We can observe progressive linear decay of C_l still happens without the orthogonal initialization while the expanding pattern of D_l disappears.

5.3.2 Implication on Transfer Learning

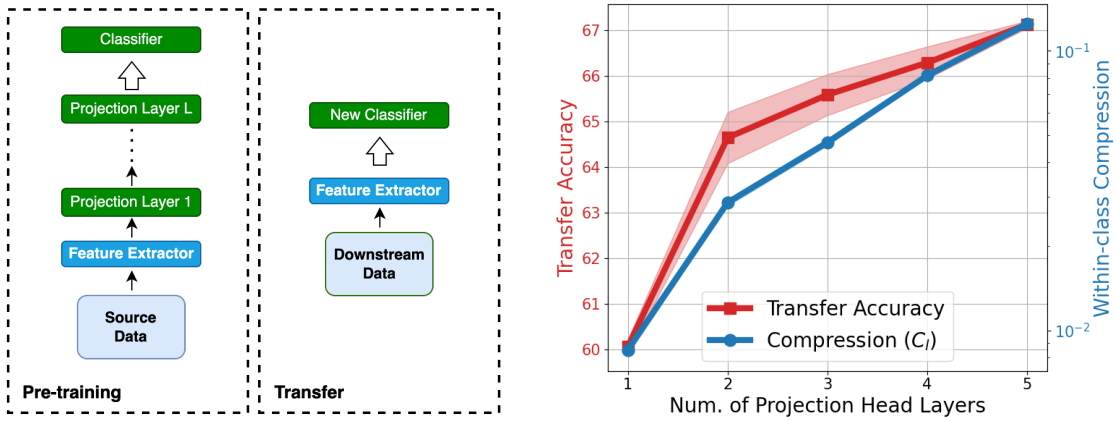
In this subsection, we experimentally substantiate our claims on transfer learning in Section 3 using practical nonlinear networks and real datasets. The results demonstrate that features before projection heads are less collapsed and exhibit better transferability. To support our argument, we conduct our experiments based on the following setup.

Network architectures. We employ a ResNet18 backbone architecture (He et al., 2016), incorporating $t \in \{1, 2, 3, 4, 5\}$ layers of projection heads between the feature extractor and the final classifier, respectively. Here, one layer of the projection head consists of a linear layer followed by a ReLU activation layer. These projection layers are only used in the pre-training phase. On the downstream tasks, they are discarded, and a new linear classifier is trained on the downstream dataset.

Training datasets and training methods. We use the CIFAR-100 and CIFAR-10 dataset in the pre-training and fine-tuning tasks, respectively. We train the networks using the Rescaled-MSE loss (Hui and Belkin, 2020), with hyperparameters set to $k = 5$ and $M = 20$ for 200 epochs. During pre-training, we employed the SGD optimizer with a momentum of 0.9, a weight decay of 5×10^{-4}, and a dynamically adaptive learning rate ranging from 10^{-2} to 10^{-5}, modulated by a CosineAnnealing learning rate scheduler (Loshchilov and Hutter, 2017). During the fine-tuning phase, we freeze all the parameters of the pre-trained model and only conduct linear probing. In other words, we only train a linear classifier on the downstream data for an additional 200 epochs. We run each experiment with 3 different random seeds.

Experiments and observations. Figure 13 illustrates the relationship between the number of layers in a projection head and two distinct metrics: (i) the compression metric C_l of the learn features on the pre-trained dataset (depicted by the blue curve), and (ii) the transfer accuracy of pre-trained models on downstream tasks (depicted by the red curve). It is observed from Figure 13(b) that an increase in the number of layers in the projection head leads to decreased feature compression and better transfer accuracy.

These observations confirm our theoretical understanding that feature compression occurs progressively through the layers from shallow to deep and that the use of projection



(a) Illustration on the usage of projection heads

(b) Transfer learning results

Figure 13: **Adding projection heads mitigates feature collapse and improves transfer accuracy.** We pre-train ResNet18 backbones with 1 to 5 layers of projection heads on the CIFAR-100 dataset. After pre-training, we drop the projection heads and compute the metric C_l on the learned features. Finally, we linearly probe the pre-trained models on the CIFAR-10 dataset and plot the evolution of C_l alongside the corresponding transfer learning performance against the number of projection head layers. The shaded area represents the standard deviation of 3 random seeds. The plot shows a clear trend that adding more layers of projection heads mitigates feature collapse and improves transfer learning results.

heads during the pre-training phase helps to prevent feature collapse at the feature extractor layer, thereby improving the model's transfer accuracy on new downstream data. Furthermore, adding more layers in the projection head tends to preserve diverse features of the pre-trained model, resulting in improved transfer learning performance.

6. Conclusion

In this work, we studied hierarchical representations of deep networks by analyzing their intermediate features. In the context of training DLNs for solving multi-class classification problems, we examined how features evolve across layers through within-class feature compression and between-class feature discrimination. We showed that under mild assumptions on the input data and trained weights, each layer of a DLN progressively compresses within-class features at a geometric rate and discriminates between-class features at a linear rate w.r.t. the layer index. Moreover, we discussed the implications of our results for deep learning practices, including interpretation, architecture design, and transfer learning. Our extensive experimental results on synthetic and real data sets not only support our theoretical findings but also highlight their relevance to deep nonlinear networks.

Our work opens several interesting directions for future work. First, our extensive experiments have demonstrated analogous patterns of progressive compression and discrimination phenomena in deep nonlinear networks. Extending our analysis to deep nonlinear networks emerges as a natural and promising direction for future research. Second, Yu et al. (2024);

Geshkovski et al. (2023) demonstrated that the outputs of self-attention layers in transformers exhibit a similar progressive compression phenomenon. It would be interesting to study this phenomenon in transformers based on our proposed framework.

Acknowledgments

XL, CY, PW, and QQ acknowledge support from NSF CAREER CCF-2143904, NSF CCF-2212066, NSF CCF-2212326, NSF IIS 2312842, ONR N00014-22-1-2529, and an AWS AI Award. QQ also acknowledges the Google Research Scholar Award. PW and LB acknowledge support from DoE award DE-SC0022186, ARO YIP W911NF1910027, and NSF CAREER CCF-1845076. ZZ acknowledges support from NSF grants CCF-2240708, IIS-2312840, and IIS-2402952. WH acknowledges support from the Google Research Scholar Program. Results presented in this paper were obtained using CloudBank, which is supported by the NSF under Award #1925001. The authors acknowledge valuable discussions with Dr. Weijie Su (Upenn), Dr. John Wright (Columbia), Dr. Rene Vidal (Upenn), Dr. Hangfeng He (U. Rochester), Mr. Zekai Zhang (U. M), Dr. Yaodong Yu (UC Berkeley), and Dr. Yuexiang Zhai (UC Berkeley) at different stages of the work.

Appendix A. Proofs of Main Results in Section 3

In this section, we provide formal proofs to show our main result in Theorem 4 concerning the behavior of within-class and between-class features. This involves establishing some key properties of the weight matrices of each layer under Assumptions 2 and 3. The rest of this section is organized as follows: In Appendix A.1, we establish some preliminary results for weight matrices under Assumptions 2 and 3. In Appendix A.2, we prove that within-class features are compressed across layers at a geometric rate. In Appendix A.3, we prove that between-class features are discriminated across layers at a linear rate. Finally, we finish the proofs of Theorem 4 and Proposition 5 in Appendix A.4 and A.5, respectively.

Before we proceed, let us introduce some further notation that will be used throughout this section. Using (10), $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \mathbf{1}_n^T$, and Assumption 2, we show that the rank of $\boldsymbol{W}_l$ is at least K for each $l \in [L]$; see Lemma 8. For all $l \in [L-1]$, let

$$\boldsymbol{W}_l = \boldsymbol{U}_l \boldsymbol{\Sigma}_l \boldsymbol{V}_l^T = \begin{bmatrix} \boldsymbol{U}_{l,1} & \boldsymbol{U}_{l,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_{l,1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{l,2} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_{l,1}^T \\ \boldsymbol{V}_{l,2}^T \end{bmatrix} = \boldsymbol{U}_{l,1} \boldsymbol{\Sigma}_{l,1} \boldsymbol{V}_{l,1}^T + \boldsymbol{U}_{l,2} \boldsymbol{\Sigma}_{l,2} \boldsymbol{V}_{l,2}^T, \tag{21}$$

be a singular value decomposition (SVD) of $\boldsymbol{W}_l$, where $\boldsymbol{\Sigma}_l \in \mathbb{R}^{d \times d}$ is diagonal, $\boldsymbol{\Sigma}_{l,1} = \operatorname{diag}(\sigma_{l,1}, \ldots, \sigma_{l,K})$ with $\sigma_{l,1} \geq \cdots \geq \sigma_{l,K} > 0$ being the first K leading singular values of $\boldsymbol{W}_l$, and $\boldsymbol{\Sigma}_{l,2} = \operatorname{diag}(\sigma_{l,K+1}, \ldots, \sigma_{l,d})$ with $\sigma_{l,K+1} \geq \cdots \geq \sigma_{l,d} \geq 0$ being the remaining singular values; $\boldsymbol{U}_l \in \mathcal{O}^d$ with $\boldsymbol{U}_{l,1} \in \mathbb{R}^{d \times K}$, $\boldsymbol{U}_{l,2} \in \mathbb{R}^{d \times (d-K)}$; $\boldsymbol{V}_l \in \mathcal{O}^d$ with $\boldsymbol{V}_{l,1} \in \mathbb{R}^{d \times K}$, $\boldsymbol{V}_{l,2} \in \mathbb{R}^{d \times (d-K)}$. Noting that $\boldsymbol{W}_L \in \mathbb{R}^{K \times d}$, let

$$\boldsymbol{W}_L = \boldsymbol{U}_L \boldsymbol{\Sigma}_L \boldsymbol{V}_L^T = \boldsymbol{U}_L \begin{bmatrix} \boldsymbol{\Sigma}_{L,1} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_{L,1}^T \\ \boldsymbol{V}_{L,2}^T \end{bmatrix} = \boldsymbol{U}_L \boldsymbol{\Sigma}_{L,1} \boldsymbol{V}_{L,1}^T, \tag{22}$$

be an SVD of $\boldsymbol{W}_L$, where $\boldsymbol{\Sigma}_L \in \mathbb{R}^{K \times d}$, and $\boldsymbol{\Sigma}_{L,1} = \operatorname{diag}(\sigma_{L,1}, \ldots, \sigma_{L,K})$ with $\sigma_{L,1} \geq \cdots \geq \sigma_{L,K} > 0$ being the singular values; $\boldsymbol{U}_L \in \mathcal{O}^K$, and $\boldsymbol{V}_L \in \mathcal{O}^d$ with $\boldsymbol{V}_{L,1} \in \mathbb{R}^{d \times K}$, $\boldsymbol{V}_{L,2} \in \mathbb{R}^{d \times (d-K)}$. Moreover, we respectively denote the k-th class mean and global mean by

$$\bar{\boldsymbol{x}}_k = \frac{1}{n} \sum_{i=1}^{N} \boldsymbol{x}_{k,i}, \quad \bar{\boldsymbol{x}} = \frac{1}{K} \sum_{k=1}^{K} \bar{\boldsymbol{x}}_k, \tag{23}$$

and let

$$\bar{\boldsymbol{X}} = [\bar{\boldsymbol{x}}_1 \ \bar{\boldsymbol{x}}_2 \ \ldots \ \bar{\boldsymbol{x}}_K] \otimes \mathbf{1}_n^T \in \mathbb{R}^{d \times N}. \tag{24}$$

Note that the above notation will be used consistently throughout the rest of this paper.

A.1 Properties of Weight Matrices

In this subsection, we establish some properties of weight matrices $\{\boldsymbol{W}_l\}_{l=1}^L$ under Assumptions 2 and 3. Towards this goal, we first prove some inequalities on the spectrum of the data matrix $\boldsymbol{X}$ and the distance between $\boldsymbol{X}$ and $\bar{\boldsymbol{X}}$ under Assumption 2. These inequalities will serve as crucial components in the subsequent analysis of the spectral properties of the weight matrices.

Lemma 6 *Suppose that Assumption 2 holds. Then, we have*

$$\sqrt{1-\theta} \le \sigma_{\min}(\boldsymbol{X}) \le \sigma_{\max}(\boldsymbol{X}) \le \sqrt{1+\theta}, \tag{25}$$

$$\|\boldsymbol{X} - \bar{\boldsymbol{X}}\| \le \sqrt{1+4\theta}, \quad N - K - 4\theta \le \|\boldsymbol{X} - \bar{\boldsymbol{X}}\|_F^2 \le N - K + 4\theta. \tag{26}$$

Proof Using $\|\boldsymbol{A}\| \le \|\boldsymbol{A}\|_F$ for all $\boldsymbol{A} \in \mathbb{R}^{N\times N}$, we have

$$\|\boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{I}_N\|^2 \le \|\boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{I}_N\|_F^2 \le \sum_{i=1}^N \left(\|\boldsymbol{x}_i\|^2 - 1\right)^2 + \sum_{i=1}^N \sum_{j\neq i}^N \left(\boldsymbol{x}_i^T\boldsymbol{x}_j\right)^2 \le \theta^2, \tag{27}$$

where the last inequality follows from (9). This, together with Weyl's inequality, implies $1-\theta \le \sigma_{\min}(\boldsymbol{X}^T\boldsymbol{X}) \le \sigma_{\max}(\boldsymbol{X}^T\boldsymbol{X}) \le 1+\theta$. Therefore, we obtain (25).

Now, we prove (26). For ease of exposition, let $\boldsymbol{A} := (\boldsymbol{X} - \bar{\boldsymbol{X}})^T(\boldsymbol{X} - \bar{\boldsymbol{X}}) = \boldsymbol{X}^T\boldsymbol{X} - \boldsymbol{X}^T\bar{\boldsymbol{X}} - \bar{\boldsymbol{X}}^T\boldsymbol{X} + \bar{\boldsymbol{X}}^T\bar{\boldsymbol{X}}$. First, we consider the elements of diagonal blocks of $\boldsymbol{A}$. For each $i = (k-1)n + j$ and $j \in [n]$, we compute

$$a_{ii} = \|\boldsymbol{x}_{k,j} - \bar{\boldsymbol{x}}_k\|^2 = \left(1 - \frac{1}{n}\right)^2 \|\boldsymbol{x}_{k,j}\|^2 + \frac{1}{n^2}\sum_{j'\neq j} \|\boldsymbol{x}_{k,j'}\|^2 - \frac{2}{n}\sum_{j\neq j'} \langle \boldsymbol{x}_{k,j}, \boldsymbol{x}_{k,j'}\rangle + \frac{1}{n^2}\sum_{j=1}^n \sum_{j'\neq j} \langle \boldsymbol{x}_{k,j}, \boldsymbol{x}_{k,j'}\rangle.$$

This, together with (9) in Assumption 2, yields

$$1 - \frac{1}{n} - \frac{4\theta}{N} \le a_{ii} \le 1 - \frac{1}{n} + \frac{4\theta}{N}. \tag{28}$$

Using the similar argument, we compute for each $i = (k-1)n + j$, $i' = (k-1)n + j'$, and $j \neq j' \in [n]$,

$$-\frac{1}{n} - \frac{4\theta}{N} \le a_{i,i'} \le -\frac{1}{n} + \frac{4\theta}{N}. \tag{29}$$

Now, we consider the elements of off-diagonal blocks of $\boldsymbol{A}$. Similarly, we compute for each $i = (k-1)n + j$, $i' = (k'-1)n + j'$, $k \neq k'$, and $j, j' \in [n]$,

$$-\frac{4\theta}{N} \le a_{i,i'} \le \frac{4\theta}{N}. \tag{30}$$

This, together with (28) and (29), yields $\boldsymbol{A} = \boldsymbol{I}_N - \frac{1}{n}\boldsymbol{I}_K \otimes \boldsymbol{E}_n + \boldsymbol{\Delta}$, where $|\delta_{ij}| \le 4\theta/N$. Therefore, we have

$$\|\boldsymbol{A}\| \le \left\|\boldsymbol{I}_N - \frac{1}{n} I_K \otimes \boldsymbol{E}_n\right\| + \|\boldsymbol{\Delta}\|_F \le 1 + 4\theta,$$

which implies the first inequality in (26). Moreover, we have

$$\|\boldsymbol{X} - \bar{\boldsymbol{X}}\|_F^2 = \|\boldsymbol{X}\|_F^2 - 2\langle \boldsymbol{X}, \bar{\boldsymbol{X}}\rangle + \|\bar{\boldsymbol{X}}\|_F^2. \tag{31}$$

Then, we bound the above terms in turn. First, noting that $\|\boldsymbol{X}\|_F^2 = \sum_{i=1}^N \|\boldsymbol{x}_i\|^2$ and using (9) in Assumption 2 yield

$$N - \theta \leq \|\boldsymbol{X}\|_F^2 \leq N + \theta. \tag{32}$$

Second, it follows from (23) that

$$\langle \boldsymbol{X}, \bar{\boldsymbol{X}} \rangle = \sum_{k=1}^K \sum_{i=1}^n \langle \boldsymbol{x}_{k,i}, \bar{\boldsymbol{x}}_k \rangle = \sum_{k=1}^K \sum_{i=1}^n \frac{1}{n} \sum_{j=1}^n \langle \boldsymbol{x}_{k,i}, \boldsymbol{x}_{k,j} \rangle = \sum_{k=1}^K \sum_{i=1}^n \frac{1}{n} \left(\|\boldsymbol{x}_{k,i}\|^2 + \sum_{j \neq i}^n \langle \boldsymbol{x}_{k,i}, \boldsymbol{x}_{k,j} \rangle \right).$$

This, together with Assumption 2, yields

$$K - \theta \leq \langle \boldsymbol{X}, \bar{\boldsymbol{X}} \rangle \leq K + \theta. \tag{33}$$

Finally, we compute

$$\|\bar{\boldsymbol{X}}\|_F^2 = n \sum_{k=1}^K \|\bar{\boldsymbol{x}}_k\|^2 = \frac{1}{n} \sum_{k=1}^K \left\| \sum_{i=1}^n \boldsymbol{x}_{k,i} \right\|^2 = \frac{1}{n} \sum_{k=1}^K \left(\sum_{i=1}^n \|\boldsymbol{x}_{k,i}\|^2 + \sum_{i \neq j} \langle \boldsymbol{x}_{k,i}, \boldsymbol{x}_{k,j} \rangle \right)$$

This, together with Assumption 2, yields $K - \theta \leq \|\bar{\boldsymbol{X}}\|_F^2 \leq K + \theta$. Substituting this, (32), and (33) into (31) yields the second inequality in (26). ■

Next, we present a lemma that shows that each singular value of the first $L-1$ weight matrices is exactly the same, and each singular value of the last weight matrix is approximately equal to that of the first $L-1$ weight matrices under the balancedness (11) in Assumption 3.

Lemma 7 *Suppose that the weight matrices $\{\boldsymbol{W}_l\}_{l=1}^L$ satisfy* (11) *and admit the SVD in* (21) *and* (22)*. Then, it holds that*

$$\boldsymbol{\Sigma}_{l+1} = \boldsymbol{\Sigma}_l, \ \forall l \in [L-2], \ \|\boldsymbol{\Sigma}_L^T \boldsymbol{\Sigma}_L - \boldsymbol{\Sigma}_l^2\|_F \leq \delta, \ \forall l \in [L-1]. \tag{34}$$

Proof Since $\boldsymbol{W}_{l+1}^T \boldsymbol{W}_{l+1} = \boldsymbol{W}_l \boldsymbol{W}_l^T$ for all $l \in [L-2]$ and (21), we have

$$\boldsymbol{V}_{l+1} \boldsymbol{\Sigma}_{l+1}^2 \boldsymbol{V}_{l+1}^T = \boldsymbol{U}_l \boldsymbol{\Sigma}_l^2 \boldsymbol{U}_l^T. \tag{35}$$

For a given $l \in [L-2]$, the two sides of the above equation are essentially eigenvalue decomposition of the same matrix. This, together with the fact that $\boldsymbol{\Sigma}_{l+1}$ and $\boldsymbol{\Sigma}_l$ have non-increasing and non-negative diagonal elements by (21), yields $\boldsymbol{\Sigma}_{l+1} = \boldsymbol{\Sigma}_l$. Using $\|\boldsymbol{W}_L^T \boldsymbol{W}_L - \boldsymbol{W}_{L-1} \boldsymbol{W}_{L-1}^T\|_F \leq \delta$, (21), and (22), we have

$$\|\boldsymbol{V}_L \boldsymbol{\Sigma}_L^T \boldsymbol{\Sigma}_L \boldsymbol{V}_L^T - \boldsymbol{U}_{L-1} \boldsymbol{\Sigma}_{L-1}^2 \boldsymbol{U}_{L-1}^T\|_F \leq \delta. \tag{36}$$

This, together with (Arora et al., 2018a, Lemma 4), implies $\|\boldsymbol{\Sigma}_L^T \boldsymbol{\Sigma}_L - \boldsymbol{\Sigma}_{L-1}^2\|_F \leq \delta$. According to this and $\boldsymbol{\Sigma}_{l+1} = \boldsymbol{\Sigma}_l$ for all $l \in [L-2]$, implies (34) for all $l \in [L-1]$. ■

Based on the above two lemmas, we are ready to estimate the top K singular values of each weight matrix $\boldsymbol{W}_l$ for all $l \in [L]$ under Assumptions 2 and 3.

Lemma 8 *Suppose that Assumption 2 holds, and the weight matrices* $\{\boldsymbol{W}_l\}_{l=1}^L$ *satisfy* (10) *and* (11) *with*

$$\delta \leq \frac{(2n)^{1/L}}{30L^2}. \tag{37}$$

Then, the following statements hold:
(i) It holds that

$$\frac{\sqrt{n}}{\sqrt{1+\theta}} \leq \sigma_K(\boldsymbol{W}_{L:1}) \leq \sigma_1(\boldsymbol{W}_{L:1}) \leq \frac{\sqrt{n}}{\sqrt{1-\theta}}, \tag{38}$$

(ii) It holds that

$$\left(\sqrt{\frac{n}{2}}\right)^{1/L} \leq \sigma_K\left(\boldsymbol{W}_l\right) \leq \sigma_1\left(\boldsymbol{W}_l\right) \leq \left(2\sqrt{n}\right)^{1/L}, \ \forall l \in [L], \tag{39}$$

and

$$\left(\left(\frac{1}{1+\theta} - 4\delta\right)n\right)^{1/2L} \leq \sigma_K(\boldsymbol{W}_l) \leq \sigma_1(\boldsymbol{W}_l) \leq \left(\left(\frac{1}{1-\theta} + 4\delta\right)n\right)^{1/2L}, \ \forall l \in [L-1], \tag{40}$$

$$\left(\left(\frac{1}{1+\theta} - 4\delta\right)n\right)^{1/2L} - \sqrt{\delta} \leq \sigma_K(\boldsymbol{W}_L) \leq \sigma_1(\boldsymbol{W}_L) \leq \left(\left(\frac{1}{1-\theta} + 4\delta\right)n\right)^{1/2L} + \sqrt{\delta}. \tag{41}$$

Proof According to (25) in Lemma 6, we obtain that $\boldsymbol{X}$ is of full column rank, and thus $\boldsymbol{X}^+ := (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T$ is of full row rank with $1/\sqrt{1+\theta} \leq \sigma_{\min}(\boldsymbol{X}^+) \leq \sigma_{\max}(\boldsymbol{X}^+) \leq 1/\sqrt{1-\theta}$. This, together with $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \boldsymbol{1}_n^T$, (10), and Lemma 15, yields

$$\frac{\sqrt{n}}{\sqrt{1+\theta}} \leq \sigma_K(\boldsymbol{W}_{L:1}) \leq \sigma_1(\boldsymbol{W}_{L:1}) \leq \frac{\sqrt{n}}{\sqrt{1-\theta}}, \tag{42}$$

which, together with $\theta \in [0, 1/4]$, implies

$$\frac{2\sqrt{n}}{\sqrt{5}} \leq \sigma_K(\boldsymbol{W}_{L:1}) \leq \sigma_1(\boldsymbol{W}_{L:1}) \leq \frac{2\sqrt{n}}{\sqrt{3}}. \tag{43}$$

Using this, (11) with (37), and invoking (Arora et al., 2018a, Lemma 6), we obtain $\sigma_1(\boldsymbol{W}_l) \leq (2\sqrt{n})^{1/L}$ for all $l \in [L]$. Using $\boldsymbol{W}_{l+1}^T\boldsymbol{W}_{l+1} = \boldsymbol{W}_l\boldsymbol{W}_l^T$ for all $l \in [L-2]$ by (11), we compute

$$\begin{aligned}
\|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\|_F &= \|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \boldsymbol{W}_{L-1:1}^T\boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\boldsymbol{W}_{L-1:1}\|_F \\
&= \|\boldsymbol{W}_{L-1:1}^T\left(\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\right)\boldsymbol{W}_{L-1:1}\|_F \\
&\leq \|\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\|_F \prod_{l=1}^{L-1}\|\boldsymbol{W}_l\|^2 \\
&\leq \delta\left(4n\right)^{(L-1)/L} \leq \frac{2n}{15L^2},
\end{aligned} \tag{44}$$

where the second inequality uses (11), (37), and (39) for all $l \in [L-1]$, and the last inequality follows from (37). Using this, (42), and Weyl's inequality gives

$$\begin{aligned}\sigma_K\left(\left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\right) &\geq \sigma_K\left(\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\right) - \|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\| \\ &\geq \frac{n}{1+\theta} - \|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\|_F \geq \left(\frac{1}{1+\theta} - \frac{2}{15L^2}\right)n,\end{aligned}$$

which implies $\sigma_K(\boldsymbol{W}_1) \geq \left(\left(\frac{1}{1+\theta} - \frac{2}{15L^2}\right)n\right)^{1/2L}$. This, together with Lemma 7, implies

$$\sigma_K(\boldsymbol{W}_l) \geq \left(\left(\frac{1}{1+\theta} - \frac{2}{15L^2}\right)n\right)^{1/2L}, \ \forall l \in [L-1]. \tag{45}$$

Using Weyl's inequality again and (11), we have

$$\begin{aligned}\sigma_K\left(\boldsymbol{W}_L^T\boldsymbol{W}_L\right) &\geq \sigma_K\left(\boldsymbol{W}_{L-1}^T\boldsymbol{W}_{L-1}\right) - \|\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\| \\ &\geq \left(\frac{1}{1+\theta} - \frac{2}{15L^2}\right)^{1/L} n^{1/L} - \delta \geq \left(\frac{n}{2}\right)^{1/L},\end{aligned}$$

where the last inequality follows from $\delta \leq (2n)^{1/L}/(30L^2)$ and

$$\left(\frac{1}{1+\theta} - \frac{2}{15L^2}\right)^{1/L} \geq \left(\frac{3}{4}\right)^{1/L} \geq \frac{1}{15L^2} + \left(\frac{1}{2}\right)^{1/L}, \ \forall L \geq 1.$$

Here the first inequality uses $L \geq 2$, and the second inequality is due to $f(x) = (3/4)^x - (1/2)^x - x^2/15$ is increasing for $x \in (0,1]$ and $f(0) = 0$. Therefore, we have $\sigma_K(\boldsymbol{W}_L) \geq \left(\frac{n}{2}\right)^{1/2L}$. This, together with (45) and $\sigma_1(\boldsymbol{W}_l) \leq (2\sqrt{n})^{1/L}$ for all $l \in [L]$, yields (39).

Next, we are devoted to proving (40) and (41). It follows from (44) that

$$\|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\|_F \leq 4n\delta. \tag{46}$$

This, together with Weyl's inequality and (42), yields

$$\begin{aligned}\sigma_1\left(\left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\right) &\leq \sigma_1\left(\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\right) + \|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\| \leq \left(\frac{1}{1-\theta} + 4\delta\right)n, \\ \sigma_K\left(\left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\right) &\geq \sigma_K\left(\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\right) - \|\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \left(\boldsymbol{W}_1^T\boldsymbol{W}_1\right)^L\| \geq \left(\frac{1}{1+\theta} - 4\delta\right)n.\end{aligned}$$

This, together with Lemma 7, directly implies (40). Using Weyl's inequality and (11), we have

$$\begin{aligned}\sigma_1\left(\boldsymbol{W}_L^T\boldsymbol{W}_L\right) &\leq \sigma_1\left(\boldsymbol{W}_{L-1}^T\boldsymbol{W}_{L-1}\right) + \|\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\| \leq \left(\frac{1}{1-\theta} + 4\delta\right)^{1/L} n^{1/L} + \delta, \\ \sigma_K\left(\boldsymbol{W}_L^T\boldsymbol{W}_L\right) &\geq \sigma_K\left(\boldsymbol{W}_{L-1}^T\boldsymbol{W}_{L-1}\right) - \|\boldsymbol{W}_L^T\boldsymbol{W}_L - \boldsymbol{W}_{L-1}\boldsymbol{W}_{L-1}^T\| \geq \left(\frac{1}{1-\theta} + 4\delta\right)^{1/L} n^{1/L} - \delta,\end{aligned}$$

which directly implies (41). ■

Using this lemma and $\varepsilon \in (0,1)$, we conclude that the leading K singular values of $\boldsymbol{W}_l$ for all $l \in [L-1]$ are well separated from the remaining $d-K$ singular values according to (12). Consequently, we can handle them with the associated singular vectors separately. Since $\boldsymbol{\Sigma}_{l+1} = \boldsymbol{\Sigma}_l$ for all $l \in [L-2]$ according to Lemma 7, suppose that $\boldsymbol{\Sigma}_{l,1}$ and $\boldsymbol{\Sigma}_{l,2}$ have distinct $p \geq 1$ and $q \geq 1$ singular values for all $l \in [L-1]$, respectively. Let $\sigma_1 > \cdots > \sigma_p > \sigma_{p+1} > \cdots > \sigma_{p+q}$ be the distinct singular values with the corresponding multiplicities $h_1, h_2, \ldots, h_{p+q} \in \mathbb{N}$. In particular, we have $\sum_{i=1}^{p} h_i = K$, $\sum_{i=p+1}^{p+q} h_i = d-K$, and $\sigma_{p+1} \leq \varepsilon$ due to (12). Then, we can write

$$\boldsymbol{\Sigma}_l = \tilde{\boldsymbol{\Sigma}} = \mathrm{BlkDiag}(\tilde{\boldsymbol{\Sigma}}_1, \tilde{\boldsymbol{\Sigma}}_2) = \mathrm{BlkDiag}\left(\sigma_1 \boldsymbol{I}_{h_1}, \ldots, \sigma_p \boldsymbol{I}_{h_p}, \ldots, \sigma_{p+q} \boldsymbol{I}_{h_{p+q}}\right). \tag{47}$$

Using the same partition, we can also write

$$\boldsymbol{U}_l = \left[\boldsymbol{U}_l^{(1)}, \ldots, \boldsymbol{U}_l^{(p)}, \ldots, \boldsymbol{U}_l^{(p+q)}\right], \ \boldsymbol{V}_l = \left[\boldsymbol{V}_l^{(1)}, \ldots, \boldsymbol{V}_l^{(p)}, \ldots, \boldsymbol{V}_l^{(p+q)}\right], \tag{48}$$

where $\boldsymbol{U}_l^{(i)}, \boldsymbol{V}_l^{(i)} \in \mathcal{O}^{d \times h_i}$ for all $i \in [p+q]$. Using the above definitions, we can characterize the relationship between singular vector matrices $\boldsymbol{U}_l$ and $\boldsymbol{V}_{l+1}$.

Lemma 9 *Suppose that Assumptions 2 and 3 hold. Then, the following statements hold: (i) For all $l \in [L-2]$, there exists orthogonal matrix $\boldsymbol{Q}_{l,i} \in \mathcal{O}^{h_i}$ such that*

$$\boldsymbol{U}_l^{(i)} = \boldsymbol{V}_{l+1}^{(i)} \boldsymbol{Q}_{l,i}, \ \forall i \in [p+q]. \tag{49}$$

(ii) It holds that

$$\|\boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,2}\|_F \leq \frac{2\sqrt{\delta}\sqrt[4]{K}}{n^{1/2L}}, \ \sigma_{\min}\left(\boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,1}\right) \geq 1 - \frac{2\sqrt{\delta}\sqrt[4]{K}}{n^{1/2L}}. \tag{50}$$

Proof (i) Using the first equation in (34), (35), (47), and (48), we obtain

$$\sum_{i=1}^{p+q} \sigma_i^2 \boldsymbol{V}_{l+1}^{(i)} \boldsymbol{V}_{l+1}^{(i)^T} = \sum_{i=1}^{p+q} \sigma_i^2 \boldsymbol{U}_l^{(i)} \boldsymbol{U}_l^{(i)^T}$$

Multiplying $\boldsymbol{U}_l^{(1)}$ on the both sides of the above equality, we obtain

$$\sum_{i=1}^{p+q} \sigma_i^2 \boldsymbol{U}_l^{(1)^T} \boldsymbol{V}_{l+1}^{(i)} \boldsymbol{V}_{l+1}^{(i)^T} \boldsymbol{U}_l^{(1)} = \sigma_1^2 \boldsymbol{I},$$

where the equality follows from the fact that $\boldsymbol{U}_l \in \mathcal{O}^d$ takes the form of (48). Therefore, for any $\boldsymbol{x} \in \mathbb{R}^{h_1}$ with $\|\boldsymbol{x}\| = 1$, we have

$$\begin{aligned}
\sigma_1^2 &= \sum_{i=1}^{p+q} \sigma_i^2 \left\|\boldsymbol{V}_{l+1}^{(i)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2 \leq \sigma_1^2 \left\|\boldsymbol{V}_{l+1}^{(1)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2 + \sigma_2^2 \sum_{i=2}^{p+q} \left\|\boldsymbol{V}_{l+1}^{(i)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2 \\
&= \sigma_1^2 \left\|\boldsymbol{V}_{l+1}^{(1)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2 + \sigma_2^2 \left(1 - \left\|\boldsymbol{V}_{l+1}^{(1)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2\right) = \sigma_2^2 + \left(\sigma_1^2 - \sigma_2^2\right) \left\|\boldsymbol{V}_{l+1}^{(1)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x}\right\|^2,
\end{aligned}$$

where the inequality follows from $\sigma_1 > \sigma_2 > \cdots > \sigma_{p+q}$, and the second equality uses $\boldsymbol{V}_{l+1} \in \mathcal{O}^d$. This, together with $\sigma_1 > \sigma_2 > 0$, implies $\left\| \boldsymbol{V}_{l+1}^{(1)^T} \boldsymbol{U}_l^{(1)} \boldsymbol{x} \right\|^2 = 1$. Using this and $\|\boldsymbol{x}\| = 1$, we have $\boldsymbol{U}_l^{(1)} = \boldsymbol{V}_{l+1}^{(1)} \boldsymbol{Q}_{l,1}$ for some $\boldsymbol{Q}_{l,1} \in \mathcal{O}^{h_1}$. By repeatedly applying the same argument to $\boldsymbol{U}_l^{(i)}$ for all $i \geq 2$, we prove (49).

(ii) According to (34) and the block structures of $\boldsymbol{\Sigma}_L, \boldsymbol{\Sigma}_{L-1}$, we obtain

$$\|\boldsymbol{\Sigma}_{L-1,2}^2\|_F \leq \delta. \tag{51}$$

In addition, it follows from (21), (22), and (36) that

$$\|\boldsymbol{V}_{L,1}\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{V}_{L,1}^T - \boldsymbol{U}_{L-1,1}\boldsymbol{\Sigma}_{L-1,1}^2\boldsymbol{U}_{L-1,1}^T - \boldsymbol{U}_{L-1,2}\boldsymbol{\Sigma}_{L-1,2}^2\boldsymbol{U}_{L-1,2}^T\|_F \leq \delta.$$

Using this, $\|\boldsymbol{U}^T\boldsymbol{A}\boldsymbol{U}\|_F \leq \|\boldsymbol{A}\|_F$ for any $\boldsymbol{U} \in \mathcal{O}^{d\times(d-K)}$, we further obtain

$$\|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2} - \boldsymbol{\Sigma}_{L-1,2}^2\|_F \leq \delta.$$

This, together with (51), implies

$$\|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F \leq 2\delta.$$

Using Lemma 14 with the fact that $\boldsymbol{\Sigma}_{L,1}\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}$ is of rank K, we further have

$$\|\boldsymbol{\Sigma}_{L,1}\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F^2 \leq 2\delta\sqrt{K}. \tag{52}$$

Noting that $\|\boldsymbol{\Sigma}_{L,1}\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F \geq \sigma_{\min}(\boldsymbol{\Sigma}_{L,1})\|\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F$, we have

$$\|\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F \leq \frac{\|\boldsymbol{\Sigma}_{L,1}\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\|_F}{\sigma_{\min}(\boldsymbol{\Sigma}_{L,1})} \leq \frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \leq \frac{2\sqrt{\delta}\sqrt[4]{K}}{n^{1/2L}}, \tag{53}$$

where the second inequality follows from (52) and Lemma 8. Now, we compute

$$\begin{aligned}
\sigma_{\min}^2(\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1}) &= \min_{\|\boldsymbol{x}\|=1} \|\boldsymbol{U}_{L-1,1}^T\boldsymbol{V}_{L,1}\boldsymbol{x}\|^2 = \min_{\|\boldsymbol{x}\|=1} \boldsymbol{x}^T\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1}\boldsymbol{U}_{L-1,1}^T\boldsymbol{V}_{L,1}\boldsymbol{x} \\
&= \min_{\|\boldsymbol{x}\|=1} \boldsymbol{x}^T\boldsymbol{V}_{L,1}^T\left(\boldsymbol{I} - \boldsymbol{U}_{L-1,2}\boldsymbol{U}_{L-1,2}^T\right)\boldsymbol{V}_{L,1}\boldsymbol{x} = 1 - \max_{\|\boldsymbol{x}\|=1} \|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\boldsymbol{x}\|^2 \\
&= 1 - \sigma_{\max}^2(\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}) \geq 1 - \|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\|_F^2,
\end{aligned}$$

where the third equality follows from $\boldsymbol{U}_{L-1} \in \mathcal{O}^d$. This, together with (53), implies

$$\sigma_{\min}(\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1}) \geq \sqrt{1 - \|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\|_F^2} \geq 1 - \|\boldsymbol{U}_{L-1,2}^T\boldsymbol{V}_{L,1}\|_F \geq 1 - \frac{2\sqrt{\delta}\sqrt[4]{K}}{n^{1/2L}}.$$

■

A.2 Analysis of Progressive Within-Class Compression

In this subsection, our goal is to prove progressive compression of within-class features across layers. Towards this goal, we study the behavior of $\mathrm{Tr}(\boldsymbol{\Sigma}_W)$ and $\mathrm{Tr}(\boldsymbol{\Sigma}_B)$ across layers, respectively. For ease of exposition, let

$$\boldsymbol{\Delta}_W := [\boldsymbol{\delta}_{1,1}, \ldots, \boldsymbol{\delta}_{1,n}, \ldots, \boldsymbol{\delta}_{K,1}, \ldots, \boldsymbol{\delta}_{K,n}] \in \mathbb{R}^{d\times N}, \text{ where } \boldsymbol{\delta}_{k,i} = \boldsymbol{x}_{k,i} - \bar{\boldsymbol{x}}_k, \ \forall k, i, \tag{54}$$

$$\boldsymbol{\Delta}_B := \left[\bar{\boldsymbol{\delta}}_1, \ldots, \bar{\boldsymbol{\delta}}_K\right] \in \mathbb{R}^{d\times K}, \text{ where } \bar{\boldsymbol{\delta}}_k = \bar{\boldsymbol{x}}_k - \bar{\boldsymbol{x}}, \ \forall k \in [K], \tag{55}$$

where $\bar{\boldsymbol{x}}_k$ and $\bar{\boldsymbol{x}}$ are defined in (23). According to Definition 1 and $n_k = n$ for all $k \in [K]$, one can verify

$$\mathrm{Tr}(\boldsymbol{\Sigma}_W^l) = \frac{1}{N}\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_W\|_F^2, \quad \mathrm{Tr}(\boldsymbol{\Sigma}_B^l) = \frac{1}{K}\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2. \tag{56}$$

Recall from (47) that $\tilde{\boldsymbol{\Sigma}} = \mathrm{BlkDiag}(\tilde{\boldsymbol{\Sigma}}_1, \tilde{\boldsymbol{\Sigma}}_2)$, where

$$\tilde{\boldsymbol{\Sigma}}_1 = \mathrm{BlkDiag}\left(\sigma_1\boldsymbol{I}_{h_1}, \ldots, \sigma_p\boldsymbol{I}_{h_p}\right), \ \tilde{\boldsymbol{\Sigma}}_2 = \mathrm{BlkDiag}\left(\sigma_{p+1}\boldsymbol{I}_{h_{p+1}}, \ldots, \sigma_{p+q}\boldsymbol{I}_{h_{p+q}}\right) \tag{57}$$

According to (21), (47), and (49), there exist block diagonal matrices $\boldsymbol{O}_{l,1} = \mathrm{Blkdiag}(\boldsymbol{Q}_{l,1}, \ldots, \boldsymbol{Q}_{l,p}) \in \mathbb{R}^{K\times K}$ and $\boldsymbol{O}_{l,2} = \mathrm{Blkdiag}(\boldsymbol{Q}_{l,p+1}, \ldots, \boldsymbol{Q}_{l,p+q}) \in \mathbb{R}^{(d-K)\times(d-K)}$ with $\boldsymbol{Q}_{l,i} \in \mathcal{O}^{h_i}$ for all $i \in [p+q]$ such that for each $l \in [L-1]$,

$$\boldsymbol{W}_{l:1} = \boldsymbol{U}_{l,1}\boldsymbol{O}_{l,1}\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{l,2}\boldsymbol{O}_{l,2}\tilde{\boldsymbol{\Sigma}}_2^l\boldsymbol{V}_{1,2}^T = \boldsymbol{U}_{l,1}\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{O}_{l,1}\boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{l,2}\tilde{\boldsymbol{\Sigma}}_2^l\boldsymbol{O}_{l,2}\boldsymbol{V}_{1,2}^T, \tag{58}$$

where $\boldsymbol{U}_{l,1}$, $\boldsymbol{U}_{l,2}$, $\boldsymbol{V}_{l,1}$, and $\boldsymbol{V}_{l,2}$ for ell $l \in [L-1]$ are defined in (21).

To bound $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$ and $\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)$ across layers, we present a lemma that establishes a relationship between the magnitudes of $\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_W$ and $\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_W$, as well as a relationship between the magnitudes of $\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B$ and $\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B$ based on the above setup.

Lemma 10 *Suppose that Assumptions 2 and 3 hold with*

$$\delta \le \frac{n^{1/L}}{64\sqrt{K}}, \ n \ge 16. \tag{59}$$

Then, the following statements hold:
(i) It holds that

$$\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_W\|_F \le \frac{2\sqrt{2\delta}\sqrt[4]{K}\varepsilon^{L-1}}{\sqrt{n}}\left\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_W\right\|_F. \tag{60}$$

(ii) It holds that

$$\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F \le 2\theta\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F. \tag{61}$$

Proof (i) It follows from (10) that $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. This, together with $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \boldsymbol{1}_n$ and (23), implies $\boldsymbol{W}_{L:1}\boldsymbol{x}_{k,i} = \boldsymbol{e}_k$ and $\boldsymbol{W}_{L:1}\bar{\boldsymbol{x}}_k = \boldsymbol{e}_k$ for all $i \in [n]$ and $k \in [K]$. It follows from this and (54) that

$$\boldsymbol{W}_{L:1}\boldsymbol{\Delta}_W = \boldsymbol{0}. \tag{62}$$

Using this and (58), there exist block diagonal matrices $\boldsymbol{O}_1 = \mathrm{BlkDiag}(\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_p) \in \mathbb{R}^{K\times K}$ and $\boldsymbol{O}_2 = \mathrm{BlkDiag}(\boldsymbol{Q}_{p+1}, \ldots, \boldsymbol{Q}_{p+q}) \in \mathbb{R}^{(d-K)\times(d-K)}$ with $\boldsymbol{Q}_i \in \mathcal{O}^{h_i}$ for all $i \in [p+q]$ such that

$$\boldsymbol{U}_L \boldsymbol{\Sigma}_{L,1} \boldsymbol{V}_{L,1}^T \left(\boldsymbol{U}_{L-1,1} \tilde{\boldsymbol{\Sigma}}_1^{L-1} \boldsymbol{O}_1 \boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{L-1,2} \tilde{\boldsymbol{\Sigma}}_2^{L-1} \boldsymbol{O}_2 \boldsymbol{V}_{1,2}^T \right) \boldsymbol{\Delta}_W = \boldsymbol{0}.$$

This, together with $\boldsymbol{U}_L \in \mathcal{O}^K$ and (39) in Lemma 8, yields

$$\boldsymbol{V}_{L,1}^T \left(\boldsymbol{U}_{L-1,1} \tilde{\boldsymbol{\Sigma}}_1^{L-1} \boldsymbol{O}_1 \boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{L-1,2} \tilde{\boldsymbol{\Sigma}}_2^{L-1} \boldsymbol{O}_2 \boldsymbol{V}_{1,2}^T \right) \boldsymbol{\Delta}_W = \boldsymbol{0}. \tag{63}$$

Therefore, we have

$$\begin{aligned} \left\| \boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,1} \tilde{\boldsymbol{\Sigma}}_1^{L-1} \boldsymbol{O}_1 \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W \right\|_F &= \left\| \boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,2} \tilde{\boldsymbol{\Sigma}}_2^{L-1} \boldsymbol{O}_2 \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W \right\|_F \\ &\le \|\boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,2}\|_F \|\tilde{\boldsymbol{\Sigma}}_2^{L-1}\| \left\| \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W \right\|_F \\ &\le \frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \varepsilon^{L-1} \left\| \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W \right\|_F, \end{aligned} \tag{64}$$

where the last inequality follows from (50) in Lemma 9 and $\|\tilde{\boldsymbol{\Sigma}}_2\| \le \varepsilon$ due to (12) and (57). On the other hand, we compute

$$\begin{aligned} \left\| \boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,1} \tilde{\boldsymbol{\Sigma}}_1^{L-1} \boldsymbol{O}_1 \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W \right\|_F &\ge \sigma_{\min}\left(\boldsymbol{V}_{L,1}^T \boldsymbol{U}_{L-1,1} \right) \sigma_{\min}\left(\tilde{\boldsymbol{\Sigma}}_1^{L-1} \right) \left\| \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W \right\|_F \\ &\ge \left(1 - \frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \right) \left(\frac{n}{2} \right)^{\frac{L-1}{2L}} \left\| \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W \right\|_F, \end{aligned}$$

where the first inequality uses Lemma 15, and the last inequality follows from (39) in Lemma 8 and (50) in Lemma 9. This, together with (64), yields

$$\left\| \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W \right\|_F \le \frac{\frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \varepsilon^{L-1}}{\left(1 - \frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \right) \left(\frac{n}{2} \right)^{\frac{L-1}{2L}}} \left\| \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W \right\|_F \le \frac{2\sqrt{2\delta}\sqrt[4]{K}\varepsilon^{L-1}}{\sqrt{n}} \left\| \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W \right\|_F,$$

where the second inequality follows from $(n/2)^{1/2L} - \sqrt{2\delta}\sqrt[4]{K} \ge n^{1/2L}/\sqrt{2}$ due to (59) and $n \ge 1$.

(ii) For ease of exposition, let

$$\boldsymbol{P}_1 := \boldsymbol{U}_{L-1,1}^T \boldsymbol{V}_{L,1} \in \mathbb{R}^{K\times K}, \ \boldsymbol{P}_2 := \boldsymbol{U}_{L-1,2}^T \boldsymbol{V}_{L,1} \in \mathbb{R}^{(d-K)\times K}. \tag{65}$$

According to (58), there exist block diagonal matrices $\boldsymbol{O}_1 = \mathrm{BlkDiag}(\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_p) \in \mathbb{R}^{K\times K}$ and $\boldsymbol{O}_2 = \mathrm{BlkDiag}(\boldsymbol{Q}_{p+1}, \ldots, \boldsymbol{Q}_{p+q}) \in \mathbb{R}^{(d-K)\times(d-K)}$ with $\boldsymbol{Q}_i \in \mathcal{O}^{h_i}$ for all $i \in [p+q]$ such that

$$\boldsymbol{W}_{L-1:1} = \boldsymbol{U}_{L-1,1} \tilde{\boldsymbol{\Sigma}}_1^{L-1} \boldsymbol{O}_1 \boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{L-1,2} \tilde{\boldsymbol{\Sigma}}_2^{L-1} \boldsymbol{O}_2 \boldsymbol{V}_{1,2}^T.$$

This, together with (22) and (65), yields

$$\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} = \boldsymbol{W}_{L-1:1}^T\boldsymbol{W}_L^T\boldsymbol{W}_L\boldsymbol{W}_{L-1:1} = \underbrace{\boldsymbol{V}_{1,1}\boldsymbol{O}_1^T\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{P}_1\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{P}_1^T\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{O}_1\boldsymbol{V}_{1,1}^T}_{=:\boldsymbol{\Delta}_1} + \underbrace{\boldsymbol{V}_{1,1}\boldsymbol{O}_1^T\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{P}_1\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{P}_2^T\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{O}_2\boldsymbol{V}_{1,2}^T}_{=:\boldsymbol{\Delta}_2} + \underbrace{\boldsymbol{V}_{1,2}\boldsymbol{O}_2^T\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{P}_2\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{P}_1^T\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{O}_1\boldsymbol{V}_{1,1}^T}_{=:\boldsymbol{\Delta}_3} + \underbrace{\boldsymbol{V}_{1,2}\boldsymbol{O}_2^T\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{P}_2\boldsymbol{\Sigma}_{L,1}^2\boldsymbol{P}_2^T\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{O}_2\boldsymbol{V}_{1,2}^T}_{=:\boldsymbol{\Delta}_4}. \tag{66}$$

Then, we bound the above terms in turn. First, we compute

$$\begin{aligned}\|\boldsymbol{\Delta}_2\|_F = \|\boldsymbol{\Delta}_3\|_F &\le \|\tilde{\boldsymbol{\Sigma}}_1^{L-1}\|\|\boldsymbol{P}_1\|\|\boldsymbol{\Sigma}_{L,1}^2\|\|\tilde{\boldsymbol{\Sigma}}_2^{L-1}\|\|\boldsymbol{P}_2\|_F \\ &\le \left(\sqrt{2n}\right)^{\frac{L+1}{L}}\varepsilon^{L-1}\frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \le 2^{1+\frac{1}{L}}\varepsilon^{L-1}\sqrt{n\delta}\sqrt[4]{K},\end{aligned} \tag{67}$$

where second inequality follows from (12), (39), (50), and (65). Using the same argument, we compute

$$\|\boldsymbol{\Delta}_4\|_F \le \|\boldsymbol{\Sigma}_{L,1}^2\|\|\tilde{\boldsymbol{\Sigma}}_2^{L-1}\|^2\|\boldsymbol{P}_2\|_F^2 \le (2n)^{\frac{1}{L}}\varepsilon^{2(L-1)}\frac{2\delta\sqrt{K}}{(n/2)^{1/L}} \le 2^{1+\frac{2}{L}}\varepsilon^{2(L-1)}\delta\sqrt{K}. \tag{68}$$

Before we proceed further, let

$$\boldsymbol{A}_1 = \boldsymbol{O}_1^T\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{P}_1\boldsymbol{\Sigma}_{L,1}. \tag{69}$$

It follows from (50) in Lemma 9 and (65) that

$$\sigma_{\min}(\boldsymbol{P}_1) \ge 1 - \frac{\sqrt{2\delta}\sqrt[4]{K}}{(n/2)^{1/2L}} \ge \frac{3}{4}, \tag{70}$$

where the second inequality uses (59). Using $\boldsymbol{O}_1 \in \mathcal{O}^K$, we have

$$\sigma_{\min}(\boldsymbol{A}_1) = \sigma_{\min}\left(\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{P}_1\boldsymbol{\Sigma}_{L,1}\right) \ge \sigma_{\min}\left(\tilde{\boldsymbol{\Sigma}}_1^{L-1}\right)\sigma_{\min}(\boldsymbol{P}_1)\,\sigma_{\min}(\boldsymbol{\Sigma}_{L,1}) \ge \frac{3}{4}\sqrt{\frac{n}{2}},$$

where the first inequality uses Lemma 15 and the second inequality follows from (39) in Lemma 8. This implies that $\boldsymbol{A}_1\boldsymbol{A}_1^T \in \mathbb{R}^{K\times K}$ is of full rank, and thus $\boldsymbol{\Delta}_1 = \boldsymbol{V}_{1,1}\boldsymbol{A}_1\boldsymbol{A}_1^T\boldsymbol{V}_{1,1}^T$ is of rank K. Then, let

$$\boldsymbol{\Delta}_1 = \boldsymbol{U}_{\Delta_1}\boldsymbol{\Sigma}_{\Delta_1}\boldsymbol{U}_{\Delta_1}^T \tag{71}$$

be a compact eigenvalue decomposition of $\boldsymbol{\Delta}_1$, where $\boldsymbol{U}_{\Delta_1} \in \mathcal{O}^{d\times K}$ and $\boldsymbol{\Sigma}_{\Delta_1} \in \mathbb{R}^{K\times K}$. Using this and Lemma 16 with $\boldsymbol{\Delta}_1 = \boldsymbol{V}_{1,1}\boldsymbol{A}_1\boldsymbol{A}_1^T\boldsymbol{V}_{1,1}^T$ by (66), we obtain

$$\boldsymbol{V}_{1,1}\boldsymbol{V}_{1,1}^T = \boldsymbol{U}_{\Delta_1,1}\boldsymbol{U}_{\Delta_1,1}^T. \tag{72}$$

Noting that $\boldsymbol{W}_{L:1}$ is of rank K, let

$$\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} = \boldsymbol{U}_1\boldsymbol{\Sigma}_1\boldsymbol{U}_1^T, \tag{73}$$

be a compact eigenvalue decomposition of $\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}$, where $\boldsymbol{U}_1 \in \mathbb{R}^{d\times K}$ and $\boldsymbol{\Sigma}_1 \in \mathbb{R}^{K\times K}$. According to (9) and (25), we have $\sigma_K(\boldsymbol{W}_{L:1}) \geq \sqrt{n/(1-\theta)}$. This, together with (66), (71), and Davis-Kahan Theorem (see, e.g., (Stewart and Sun, 1990, Theorem V.3.6)), yields

$$\begin{aligned}\|\boldsymbol{U}_{\Delta_1,1}\boldsymbol{U}_{\Delta_1,1}^T - \boldsymbol{U}_1\boldsymbol{U}_1^T\|_F &\leq \frac{\|\boldsymbol{\Delta}_2+\boldsymbol{\Delta}_3+\boldsymbol{\Delta}_4\|_F}{n(1-\theta)} \leq \frac{4\varepsilon^{L-1}\sqrt{2n\delta}\sqrt[4]{K} + 4\varepsilon^{2(L-1)}\delta\sqrt{K}}{n(1-\theta)} \\ &\leq \frac{8\varepsilon^{L-1}\sqrt{\delta}\sqrt[4]{K}}{\sqrt{n}},\end{aligned}$$

where the second inequality uses (67) and (68), and the last inequality follows from $4\varepsilon^{L-1}\sqrt{\delta}\sqrt[4]{K} \leq 4\sqrt{\delta}\sqrt[4]{K} \leq (6-4\sqrt{2})\sqrt{n}$ due to $\varepsilon \leq 1$ and (59). Using this and (72) yields

$$\|\boldsymbol{V}_{1,1}\boldsymbol{V}_{1,1}^T - \boldsymbol{U}_1\boldsymbol{U}_1^T\|_F \leq \frac{8\varepsilon^{L-1}\sqrt{\delta}\sqrt[4]{K}}{\sqrt{n}}. \tag{74}$$

It follows from (10) that $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. This, together with (1) and (55), implies $\boldsymbol{W}_{L:1}\boldsymbol{\Delta}_B = \boldsymbol{I}_K - \boldsymbol{E}_K/K$. According to this and (10), we compute

$$\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\boldsymbol{\Delta}_B = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{Y}^T\left(\boldsymbol{I}_K - \frac{\boldsymbol{E}_K}{K}\right) = \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\left((\boldsymbol{I}_K - \boldsymbol{E}_K/K)\otimes\boldsymbol{1}_n\right) \tag{75}$$

Noting that $\boldsymbol{X} \in \mathbb{R}^{d\times N}$ is of full column rank due to (25) in Lemma 6, let $\boldsymbol{X} = \boldsymbol{U}_X\boldsymbol{\Sigma}_X\boldsymbol{V}_X^T$ be an SVD of $\boldsymbol{X}$, where $\boldsymbol{U}_X \in \mathcal{O}^{d\times N}$, $\boldsymbol{\Sigma}_X \in \mathbb{R}^{N\times N}$, and $\boldsymbol{V}_X \in \mathcal{O}^{N\times N}$. Then, we obtain $\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1} = \boldsymbol{U}_X\boldsymbol{\Sigma}_X^{-1}\boldsymbol{V}_X^T$. Using this, (75), and $\boldsymbol{\Delta}_B = \boldsymbol{X}\left((\boldsymbol{I}_K - \boldsymbol{E}_K/K)\otimes\boldsymbol{1}_n\right)/n$, we compute

$$\begin{aligned}\left\|\frac{1}{n}\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\boldsymbol{\Delta}_B - \boldsymbol{\Delta}_B\right\|_F &= \frac{1}{n}\left\|\boldsymbol{U}_X\left(\boldsymbol{\Sigma}_X^{-1} - \boldsymbol{\Sigma}_X\right)\boldsymbol{V}_X^T\left((\boldsymbol{I}_K - \boldsymbol{E}_K/K)\otimes\boldsymbol{1}_n\right)\right\|_F \\ &= \frac{1}{n}\left\|\left(\boldsymbol{\Sigma}_X^{-1} - \boldsymbol{\Sigma}_X\right)\boldsymbol{\Sigma}_X^{-1}\boldsymbol{\Sigma}_X\boldsymbol{V}_X^T\left((\boldsymbol{I}_K - \boldsymbol{E}_K/K)\otimes\boldsymbol{1}_n\right)\right\|_F \\ &\leq \frac{1}{n}\left\|\boldsymbol{\Sigma}_X^{-2} - \boldsymbol{I}\right\|\|\boldsymbol{\Delta}_B\|_F \leq \frac{2\theta}{n}\|\boldsymbol{\Delta}_B\|_F,\end{aligned} \tag{76}$$

where the last inequality follows from $\left\|\boldsymbol{\Sigma}_X^{-2} - \boldsymbol{I}\right\| \leq 2\theta$ due to (25) in Lemma 6 and $\theta \leq 1/4$. According to (73), we have

$$\left\|\frac{1}{n}\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \boldsymbol{U}_1\boldsymbol{U}_1^T\right\| = \left\|\frac{\boldsymbol{\Sigma}_1}{n} - \boldsymbol{I}\right\| \leq \frac{\theta}{1-\theta}, \tag{77}$$

where the inequality uses (38) in Lemma 8. Since $\boldsymbol{V}_1 = [\boldsymbol{V}_{1,1}\ \boldsymbol{V}_{1,2}] \in \mathcal{O}^d$, we compute

$$\begin{aligned}\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F = \|(\boldsymbol{I} - \boldsymbol{V}_{1,1}\boldsymbol{V}_{1,1}^T)\boldsymbol{\Delta}_B\|_F &\leq \left\|\left(\boldsymbol{I} - \frac{1}{n}\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1}\right)\boldsymbol{\Delta}_B\right\|_F + \\ &\left\|\left(\frac{1}{n}\boldsymbol{W}_{L:1}^T\boldsymbol{W}_{L:1} - \boldsymbol{U}_1\boldsymbol{U}_1^T\right)\boldsymbol{\Delta}_B\right\|_F + \left\|\left(\boldsymbol{V}_{1,1}\boldsymbol{V}_{1,1}^T - \boldsymbol{U}_1\boldsymbol{U}_1^T\right)\boldsymbol{\Delta}_B\right\|_F \\ &\leq \left(\frac{2\theta}{n} + \frac{\theta}{1-\theta} + \frac{8\varepsilon^{L-1}\sqrt{\delta}\sqrt[4]{K}}{\sqrt{n}}\right)\|\boldsymbol{\Delta}_B\|_F \leq \frac{3}{2}\theta\|\boldsymbol{\Delta}_B\|_F,\end{aligned}$$

where the second inequality follows from (74), (76), and (77), the last inequality uses $\theta \leq 1/4$ and $n \geq 16$. This, together with $\|\boldsymbol{\Delta}_B\|_F^2 = \|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2$, yields

$$\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2 \leq \frac{9\theta^2/4}{1-9\theta^2/4}\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2,$$

which, together with $\theta \leq 1/4$, implies (61). ■

Before we proceed, we make some remarks on the above lemma. According to (56), (58), and (60), it is evident that the leading K singular values with the associated singular vectors of each weight matrix $\boldsymbol{W}_l$ for all $l \in [L-1]$ contribute minimally to the value of $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$. Conversely, the remaining singular values with the associated singular vectors play a dominant role in determining the value of $\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$. In contrast, the primary influence on the value of $\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)$ comes from the leading K singular values along with their corresponding singular vectors, while the impact of the remaining ones is relatively less significant.

With the above preparations, we are ready to prove progressive within-class compression (14) and (15) as presented in Theorem 4. We first provide a formal version of (14) and (15) and their proof. The main idea of the proof is to compute the upper and lower bounds of $\mathrm{Tr}(\boldsymbol{\Sigma}_W^{l+1})/\mathrm{Tr}(\boldsymbol{\Sigma}_W^l)$ and $\mathrm{Tr}(\boldsymbol{\Sigma}_B^{l+1})/\mathrm{Tr}(\boldsymbol{\Sigma}_B^l)$, respectively.

Theorem 11 *Suppose that Assumptions 2 and 3 hold with $\delta, \varepsilon, \rho$ satisfying* (13). *Then, we have*

$$\frac{c_1}{c_2\left((2n)^{1/L}+\varepsilon^2\right)}\varepsilon^2 \leq \frac{C_1}{C_0} \leq \frac{2c_2}{c_1(n/2)^{1/L}}\varepsilon^2, \tag{78}$$

$$\frac{c_1}{c_2\left((2n)^{1/L}+n^{-1/L}\right)}\varepsilon^2 \leq \frac{C_{l+1}}{C_l} \leq \frac{c_2\left(1+n^{-1/L}\right)}{c_1(n/2)^{1/L}}\varepsilon^2,\ \forall l \in [L-2], \tag{79}$$

where

$$c_1 = \frac{(n-3)K-1}{(n-1)K+1},\ c_2 = \frac{\left(1+n^{-1/L}\right)}{\left(1-\varepsilon^{2(L-1)}n^{-1}\right)\left(1-n^{-1/2}\right)}. \tag{80}$$

Proof For ease of exposition, let $\boldsymbol{v}_i \in \mathbb{R}^d$ denote the i-th column of $\boldsymbol{V}_1 \in \mathcal{O}^d$. Therefore, we write

$$\boldsymbol{V}_1 = [\boldsymbol{V}_{1,1}, \boldsymbol{V}_{1,2}],\ \boldsymbol{V}_{1,1} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_K],\ \boldsymbol{V}_{1,2} = [\boldsymbol{v}_{K+1}, \ldots, \boldsymbol{v}_d]. \tag{81}$$

According to (58), there exist block diagonal matrices $\boldsymbol{O}_{l,1} = \mathrm{BlkDiag}(\boldsymbol{Q}_{l,1}, \ldots, \boldsymbol{Q}_{l,p}) \in \mathbb{R}^{K\times K}$ and $\boldsymbol{O}_{l,2} = \mathrm{BlkDiag}(\boldsymbol{Q}_{l,p+1}, \ldots, \boldsymbol{Q}_{l,p+q}) \in \mathbb{R}^{(d-K)\times(d-K)}$ with $\boldsymbol{Q}_{l,i} \in \mathcal{O}^{h_i}$ for all $i \in [p+q]$ such that

$$\boldsymbol{W}_{l:1} = \boldsymbol{U}_{l,1}\boldsymbol{O}_{l,1}\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{l,2}\boldsymbol{O}_{l,2}\tilde{\boldsymbol{\Sigma}}_2^l\boldsymbol{V}_{1,2}^T,\ \forall l \in [L-1]. \tag{82}$$

This implies for all $l \in [L-1]$,

$$\begin{aligned}\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_W\|_F^2 &= \|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \\ &\le \|\tilde{\boldsymbol{\Sigma}}_1^l\|^2 \|\boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l\|^2 \|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \\ &\le (2n)^{l/L} \frac{8\delta\sqrt{K}\varepsilon^{2(L-1)}}{n} \left\|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\right\|_F^2 + \varepsilon^{2l} \|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \\ &\le \left(\frac{16\delta\sqrt{K}}{n^{1/L}}\varepsilon^{2(L-1)} + \varepsilon^{2l}\right)(N-K+4\theta) \\ &\le \left(1 + \frac{1}{n^{1/L}}\right)\varepsilon^{2l}(N-K+1), \end{aligned} \tag{83}$$

where the second inequality uses (12), (39) in Lemma 8, and (60) in Lemma 10, the third inequality follows from $l \in [L-1]$ and $\|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \le \|\boldsymbol{\Delta}_W\|_F^2 \le N-K+4\theta$ due to $\|\boldsymbol{V}_{1,2}\| \le 1$, (26), and (54), and the last inequality uses $\theta \le 1/4$ and $\delta \le 1/(16\sqrt{K})$ due to (13). According to $\boldsymbol{V}_1 \in \mathcal{O}^d$ and (60), we have

$$\begin{aligned}\|\boldsymbol{\Delta}_W\|_F^2 = \|\boldsymbol{V}_1^T \boldsymbol{\Delta}_W\|_F^2 &= \|\boldsymbol{V}_{1,1}^T \boldsymbol{\Delta}_W\|_F^2 + \|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \\ &\le \left(1 + \frac{8\delta\sqrt{K}\varepsilon^{2(L-1)}}{n}\right)\|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \\ &\le \left(1 + \frac{\varepsilon^{2(L-1)}}{n}\right)\|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2, \end{aligned}$$

where the first inequality follows from (13) and (60), and the last inequality uses $\delta \le 1/(16\sqrt{K})$ due to (13). This, together with (26) and (54), implies

$$\|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 \ge \frac{\|\boldsymbol{\Delta}_W\|_F^2}{1+\varepsilon^{2(L-1)}/n} \ge \frac{N-K-1}{1+\varepsilon^{2(L-1)}/n}, \tag{84}$$

where the last inequality follows from $\theta \le 1/4$. Moreover, we compute

$$\begin{aligned}\sum_{i=K+1}^{d-K} \|\boldsymbol{\Delta}_W^T \boldsymbol{v}_i\|^2 = \|\boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 - \sum_{i=d-K+1}^{d} \|\boldsymbol{\Delta}_W^T \boldsymbol{v}_i\|^2 &\ge \frac{N-K-4\theta}{1+\varepsilon^{2(L-1)}/n} - K(1+4\theta) \\ &\ge \left(1 - \frac{\varepsilon^{2(L-1)}}{n}\right)(N-3K-1), \end{aligned} \tag{85}$$

where the equality follows from (81), the first inequality uses (84) and $\|\boldsymbol{\Delta}_W^T \boldsymbol{v}_i\| \le \|\boldsymbol{\Delta}_W\| \le \sqrt{1+4\theta}$ by (26), and the last inequality is due to $\theta \le 1/4$. According to (82), we obtain

$$\begin{aligned}\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_W\|_F^2 \ge \|\tilde{\boldsymbol{\Sigma}}_2^l \boldsymbol{V}_{1,2}^T \boldsymbol{\Delta}_W\|_F^2 = \sum_{i=K+1}^{d} \sigma_i^{2l} \|\boldsymbol{\Delta}_W^T \boldsymbol{v}_i\|^2 &\ge (\varepsilon-\rho)^{2l} \sum_{i=K+1}^{d-K} \|\boldsymbol{\Delta}_W^T \boldsymbol{v}_i\|^2 \\ &\ge \left(1 - \frac{\varepsilon^{2(L-1)}}{n}\right)(\varepsilon-\rho)^{2l}(N-3K-1), \end{aligned} \tag{86}$$

where the equality follows from (81), the second inequality uses (12), and the last inequality is due to (85). This, together with $\theta \le 1/4$, (26), (54), (80), and (83), yields

$$\left(1-\frac{\varepsilon^{2(L-1)}}{n}\right)\left(1-\frac{\rho}{\varepsilon}\right)^2 c_1\varepsilon^2 \le \frac{\|\boldsymbol{W}_1\boldsymbol{\Delta}_W\|_F^2}{\|\boldsymbol{\Delta}_W\|_F^2} \le \left(1+\frac{1}{n^{1/L}}\right)\frac{\varepsilon^2}{c_1}. \tag{87}$$

Using (83) and (86) again, we have for all $l \in [L-2]$,

$$\frac{1-\varepsilon^{2(L-1)}n^{-1}}{1+n^{-1/L}}\left(1-\frac{\rho}{\varepsilon}\right)^{2(l+1)} c_1\varepsilon^2 \le \frac{\|\boldsymbol{W}_{l+1:1}\boldsymbol{\Delta}_W\|_F^2}{\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_W\|_F^2} \le \frac{1+n^{-1/L}}{1-\varepsilon^{2(L-1)}n^{-1}}\frac{\varepsilon^2}{c_1\left(1-\rho/\varepsilon\right)^{2l}}. \tag{88}$$

Using the Bernoulli's inequality and $\rho \le \varepsilon/(2L\sqrt{n})$ by (13), we have $(1-\rho/\varepsilon)^{2l} \ge 1 - 2l\rho/\varepsilon \ge 1 - 2L\rho/\varepsilon \ge 1 - n^{-1/2}$ due to for all $l \in [L-1]$. This, together with (80), (87), (88), implies for all $l \in [L-1]$,

$$\frac{c_1}{c_2}\varepsilon^2 \le \frac{\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_W\|_F^2}{\|\boldsymbol{W}_{l-1:1}\boldsymbol{\Delta}_W\|_F^2} \le \frac{c_2}{c_1}\varepsilon^2. \tag{89}$$

Next, we are devoted to bounding $\|\boldsymbol{W}_{l-1:1}\boldsymbol{\Delta}_B\|_F^2/\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2$. Using (82), we have for all $l = 0, 1, \ldots, L-1$,

$$\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2 = \|\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2.$$

Therefore, we compute

$$\frac{\|\boldsymbol{\Delta}_B\|_F^2}{\|\boldsymbol{W}_1\boldsymbol{\Delta}_B\|_F^2} = \frac{\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2}{\|\tilde{\boldsymbol{\Sigma}}_1\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2} \le \frac{1+4\theta^2}{(n/2)^{1/L}} \le \frac{2}{(n/2)^{1/L}},$$

where the first inequality follows from (39) and (61), and the last inequality uses $\theta \le 1/4$. Moreover, we compute

$$\frac{\|\boldsymbol{\Delta}_B\|_F^2}{\|\boldsymbol{W}_1\boldsymbol{\Delta}_B\|_F^2} \ge \frac{\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2}{(2n)^{1/L}\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \varepsilon^2\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2} \ge \frac{1}{(2n)^{1/L}+\varepsilon^2},$$

where the first inequality uses (39) and (12), and the last inequality follows from (61) and $\theta \le 1/4$. These, together with (8) and (87), yield (78). Next, we compute for all $l \in [L-2]$,

$$\begin{aligned}\frac{\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2}{\|\boldsymbol{W}_{l+1:1}\boldsymbol{\Delta}_B\|_F^2} &= \frac{\|\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2}{\|\tilde{\boldsymbol{\Sigma}}_1^{l+1}\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^{l+1}\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2} \le \frac{\|\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \varepsilon^{2l}\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2}{(n/2)^{1/L}\|\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2} \\ &\le \frac{1+\frac{\varepsilon^{2l}}{4(n/2)^{l/L}}}{(n/2)^{1/L}} \le \frac{1+n^{-1/L}}{(n/2)^{1/L}},\end{aligned}$$

where the first inequality uses (12) and (39), and the second inequality follows from

$$\|\tilde{\boldsymbol{\Sigma}}_1^l\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F \ge (n/2)^{l/2L}\,\|\boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F \ge 2\,(n/2)^{l/2L}\,\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F \tag{90}$$

due to (39) in Lemma 8 and (61) with $\theta \le 1/4$, and the last inequality is due to $\varepsilon < 1$. On the other hand, we compute for all $l \in [L-1]$,

$$\begin{aligned}\frac{\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2}{\|\boldsymbol{W}_{l+1:1}\boldsymbol{\Delta}_B\|_F^2} &= \frac{\|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l \boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2}{\|\tilde{\boldsymbol{\Sigma}}_1^{l+1} \boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + \|\tilde{\boldsymbol{\Sigma}}_2^{l+1} \boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2} \\ &\ge \frac{\|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2}{(2n)^{1/L}\|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T\boldsymbol{\Delta}_B\|_F^2 + (\varepsilon-\rho)^{2(l+1)}\|\boldsymbol{V}_{1,2}^T\boldsymbol{\Delta}_B\|_F^2} \\ &\ge \frac{1}{(2n)^{1/L} + \frac{(\varepsilon-\rho)^{2(l+1)}}{4(n/2)^{l/L}}} \ge \frac{1}{(2n)^{1/L} + n^{-1/L}},\end{aligned}$$

where the first inequality uses (12) and (39), the second inequality follows from (90), and the last inequality uses $0 \le \rho \le \varepsilon \le 1$. Therefore, we have for all $l \in [L-2]$,

$$\frac{1}{(2n)^{1/L} + n^{-1/L}} \le \frac{\|\boldsymbol{W}_{l:1}\boldsymbol{\Delta}_B\|_F^2}{\|\boldsymbol{W}_{l+1:1}\boldsymbol{\Delta}_B\|_F^2} \le \frac{1+n^{-1/L}}{(n/2)^{1/L}}.$$

This, together with (8) and (89), yield (79).

■

A.3 Analysis of Progressive Between-Class Discrimination

Now, let us turn to study the progressive discrimination of between-class features across layers. Recall that the mean of features in the k-th class at the l-th layer is

$$\boldsymbol{\mu}_k^l = \frac{1}{n}\sum_{i=1}^n \boldsymbol{z}_{k,i}^l = \boldsymbol{W}_{l:1}\bar{\boldsymbol{x}}_k,\ \forall k \in [K],\ l \in [L-1], \tag{91}$$

where $\boldsymbol{z}_{k,i}^l$ and $\bar{\boldsymbol{x}}_k$ are defined in (4) and (23), respectively. Moreover, recall from (21) that $\boldsymbol{V}_{1,1}$ is the right singular matrix of $\boldsymbol{W}_1$. Using the results in Sections A.1 and A.2, we are ready to derive the explicit form of $\boldsymbol{V}_{1,1}\bar{\boldsymbol{x}}_k$ and bound the norm of $\boldsymbol{\mu}_k^l$.

Lemma 12 *Suppose that Assumptions 2 and 3 hold with δ and ε satisfying* (13)*. Then, the following statements hold:*
(i) For each $k \in [K]$, there exist a block diagonal matrix $\boldsymbol{O} = \mathrm{BlkDiag}(\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_p) \in \mathbb{R}^{K\times K}$, where $\boldsymbol{Q}_i \in \mathcal{O}^{h_i}$ for all $i \in [p]$, and a matrix $\boldsymbol{P} \in \mathcal{O}^K$ such that

$$\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_k = \boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k + \boldsymbol{\omega}_k, \tag{92}$$

where $\|\boldsymbol{\omega}_k\| \le 4\sqrt{\delta}\sqrt[4]{K}/n^{\frac{1}{2}+\frac{1}{2L}}$.
(ii) It holds for all $k \in [K]$ and $l \in [L-1]$ that

$$\|\boldsymbol{\mu}_k^l\| \ge \frac{1}{2\sqrt{2n}}\left(\frac{n}{2}\right)^{\frac{l}{2L}}. \tag{93}$$

Proof According to (9), we have $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. This, together with $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \mathbf{1}_n^T$ and (23), implies

$$\boldsymbol{W}_{L:1}\bar{\boldsymbol{x}}_k = \boldsymbol{e}_k,\ \forall k \in [K]. \tag{94}$$

This, together with (22) and (58), yields that there exist block diagonal matrices $\boldsymbol{O}_1 = \text{BlkDiag}\ (\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_p) \in \mathbb{R}^{K\times K}$ and $\boldsymbol{O}_2 = \text{BlkDiag}(\boldsymbol{Q}_{p+1}, \ldots, \boldsymbol{Q}_{p+q}) \in \mathbb{R}^{(d-K)\times(d-K)}$ with $\boldsymbol{Q}_i \in \mathcal{O}^{h_i}$ for all $i \in [p+q]$ such that for all $k \in [K]$,

$$\boldsymbol{U}_L\boldsymbol{\Sigma}_{L,1}\boldsymbol{V}_{L,1}^T\left(\boldsymbol{U}_{L-1,1}\boldsymbol{O}_1\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{L-1,2}\boldsymbol{O}_2\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{V}_{1,2}^T\right)\bar{\boldsymbol{x}}_k = \boldsymbol{e}_k. \tag{95}$$

Since $\boldsymbol{U}_L \in \mathcal{O}^K$ and $\boldsymbol{\Sigma}_{L,1} \in \mathbb{R}^{K\times K}$ is invertible due to (39), we have

$$\boldsymbol{V}_{L,1}^T\left(\boldsymbol{U}_{L-1,1}\boldsymbol{O}_1\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{V}_{1,1}^T + \boldsymbol{U}_{L-1,2}\boldsymbol{O}_2\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{V}_{1,2}^T\right)\bar{\boldsymbol{x}}_k = \boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k \tag{96}$$

Let $\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1} = \boldsymbol{P}\boldsymbol{\Lambda}\boldsymbol{Q}^T$ be an SVD of $\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1}$, where $\boldsymbol{P}, \boldsymbol{Q} \in \mathcal{O}^K$ and $\boldsymbol{\Lambda} \in \mathbb{R}^{K\times K}$ is diagonal. This allows us to compute

$$\left\|\boldsymbol{P}\boldsymbol{Q}^T - \boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1}\right\| = \left\|\boldsymbol{P}\left(\boldsymbol{I} - \boldsymbol{\Lambda}\right)\boldsymbol{Q}^T\right\| = \|\boldsymbol{I} - \boldsymbol{\Lambda}\| \le \frac{\sqrt{2\delta}\sqrt[4]{K}}{n^{1/2L}}, \tag{97}$$

where the inequality follows from (50) in Theorem 9. Now, we rewrite (96) as

$$\boldsymbol{P}\boldsymbol{Q}^T\boldsymbol{O}_1\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_k = \boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k + \boldsymbol{\xi}_k, \tag{98}$$

where $\boldsymbol{\xi}_k := (\boldsymbol{P}\boldsymbol{Q}^T - \boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1})\boldsymbol{O}_1\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_k - \boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\boldsymbol{O}_2\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{V}_{1,2}^T\bar{\boldsymbol{x}}_k$. This, together with (39), yields

$$\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_k = \tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{O}_1^T\boldsymbol{Q}\boldsymbol{P}^T\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k + \boldsymbol{\omega}_k, \tag{99}$$

where $\boldsymbol{\omega}_k := \tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{O}_1^T\boldsymbol{Q}\boldsymbol{P}^T\boldsymbol{\xi}_k$. Then, we are devoted to bounding the norm of $\boldsymbol{\omega}_k$. We compute

$$\begin{aligned}\|\boldsymbol{\xi}_k\| &\le \|(\boldsymbol{P}\boldsymbol{Q}^T - \boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,1})\boldsymbol{O}_1\tilde{\boldsymbol{\Sigma}}_1^{L-1}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_k\| + \|\boldsymbol{V}_{L,1}^T\boldsymbol{U}_{L-1,2}\boldsymbol{O}_2\tilde{\boldsymbol{\Sigma}}_2^{L-1}\boldsymbol{V}_{1,2}^T\bar{\boldsymbol{x}}_k\| \\ &\le \frac{\sqrt{2\delta}\sqrt[4]{K}}{n^{1/2L}}\left(\sqrt{2n}\right)^{\frac{L-1}{L}}\|\bar{\boldsymbol{x}}_k\| + \frac{\sqrt{2\delta}\sqrt[4]{K}\varepsilon^{L-1}}{n^{1/2L}}\|\bar{\boldsymbol{x}}_k\| \le \frac{5\sqrt{\delta}\sqrt[4]{K}}{4n^{1/2L}}\left(\sqrt{2n}\right)^{\frac{L-1}{L}}\left(\frac{1}{\sqrt{n}} + \sqrt{\frac{\theta}{nK}}\right),\end{aligned}$$

where the second inequality follows from (12), (39), (50), and (97), and the last inequality uses $\varepsilon \le n^{1/2L}/4$ by (13) and $\|\bar{\boldsymbol{x}}_k\|^2 = \|\sum_{i=1}^n \boldsymbol{x}_{k,i}\|^2/n^2 \le 1/n + \theta/N$ due to (9) and $N = nK$. This, together with (39), implies

$$\begin{aligned}\|\boldsymbol{\omega}_k\| &= \|\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{O}_1^T\boldsymbol{Q}\boldsymbol{P}^T\boldsymbol{\xi}_k\| \le \|\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\|\|\boldsymbol{\xi}_k\| \\ &\le \frac{5\sqrt{\delta}\sqrt[4]{K}}{4n^{1/2L}}\left(\frac{1}{\sqrt{n}} + \sqrt{\frac{\theta}{nK}}\right)(2n)^{\frac{L-1}{2L}}\left(\frac{2}{n}\right)^{\frac{L-1}{2L}} \le \frac{4\sqrt{\delta}\sqrt[4]{K}}{n^{\frac{1}{2}+\frac{1}{2L}}},\end{aligned} \tag{100}$$

where the last inequality follows from $\theta/K \le 1/8$. This, together with (99) and interchangeness between $\tilde{\boldsymbol{\Sigma}}_1$ and $\boldsymbol{O}_1$, completes the proof of (i).

Next, we prove (ii). According to (58) and (91), we have

$$\|\boldsymbol{\mu}_k^l\|^2 = \|\boldsymbol{W}_{l:1}\bar{\boldsymbol{x}}_k\|^2 = \|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T \bar{\boldsymbol{x}}_k\|^2 + \|\tilde{\boldsymbol{\Sigma}}_2^l \boldsymbol{V}_{1,2}^T \bar{\boldsymbol{x}}_k\|^2. \tag{101}$$

According to (92), we compute

$$\begin{aligned}\|\tilde{\boldsymbol{\Sigma}}_1^l \boldsymbol{V}_{1,1}^T \bar{\boldsymbol{x}}_k\| &= \left\|\tilde{\boldsymbol{\Sigma}}_1^l \left(\boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)} \boldsymbol{Q}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T \boldsymbol{e}_k + \boldsymbol{\omega}_k\right)\right\| \\ &\geq \sigma_{\min}\left(\tilde{\boldsymbol{\Sigma}}_1^l\right)\left(\left\|\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{Q}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k\right\| - \|\boldsymbol{\omega}_k\|\right) \\ &\geq \left(\frac{n}{2}\right)^{\frac{l}{2L}}\left(\frac{1}{\sqrt{2n}} - \frac{4\sqrt{\delta}\sqrt[4]{K}}{n^{\frac{1}{2}+\frac{1}{2L}}}\right) \geq \frac{1}{2\sqrt{2n}}\left(\frac{n}{2}\right)^{\frac{l}{2L}},\end{aligned}$$

where the first inequality uses the triangular inequality and the fact that $\boldsymbol{O}$ and $\tilde{\boldsymbol{\Sigma}}_1$ can be interchanged, the second inequality follows from (39) and (100), and the last inequality follows from (13). This, together with (99), implies (93). ■

With the above preparations, we are ready to show (16) in Theorem 4. We now provide a formal version of (16) and its proof.

Theorem 13 *Suppose that Assumptions 2 and 3 hold with $\delta, \varepsilon, \rho$ satisfying* (13). *Then, we have*

$$D_l \geq 1 - 32\left(\theta + 4\delta\right)\left(2 - \frac{l+1}{L}\right) - n^{-\Omega(1)},\ \forall l \in [L-1]. \tag{102}$$

Proof According to (58) and (91), we compute for all $k \neq k'$,

$$\langle \boldsymbol{\mu}_k^l, \boldsymbol{\mu}_{k'}^l\rangle = \bar{\boldsymbol{x}}_k^T \boldsymbol{W}_{l:1}^T \boldsymbol{W}_{l:1}\bar{\boldsymbol{x}}_{k'} = \bar{\boldsymbol{x}}_k^T \boldsymbol{V}_{1,1}\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_{k'} + \bar{\boldsymbol{x}}_k^T\boldsymbol{V}_{1,2}\tilde{\boldsymbol{\Sigma}}_2^{2l}\boldsymbol{V}_{1,2}^T\bar{\boldsymbol{x}}_{k'}. \tag{103}$$

First, substituting (92) in Lemma 12 into the above first term yields

$$\begin{aligned}\bar{\boldsymbol{x}}_k^T\boldsymbol{V}_{1,1}\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_{k'} \leq &\left|\boldsymbol{e}_k^T\boldsymbol{U}_L\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{P}^T\tilde{\boldsymbol{\Sigma}}_1^{-2(L-1)+2l}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| + \left|\boldsymbol{\omega}_{k'}^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{P}\right. \\ &\left.\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k\right| + \left|\boldsymbol{\omega}_k^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| + \left|\boldsymbol{\omega}_k^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{\omega}_{k'}\right|,\end{aligned} \tag{104}$$

where $\boldsymbol{P}, \boldsymbol{O}$ are defined in (i) of Lemma 12. Now, we bound the above terms in turn. According to (40), we compute for all $l \in [L-1]$,

$$\begin{aligned}&\left\|\tilde{\boldsymbol{\Sigma}}_1^{-2(L-1)+2l} - n^{\frac{-(L-1)+l}{L}}\boldsymbol{I}\right\| \\ &\leq n^{\frac{-(L-1)+l}{L}}\max\left\{1 - \left(\frac{1}{1-\theta}+3\delta\right)^{\frac{-(L-1)+l}{L}}, \left(\frac{1}{1+\theta}-3\delta\right)^{\frac{-(L-1)+l}{L}} - 1\right\} \\ &\leq n^{\frac{-(L-1)+l}{L}}\frac{2\left(\theta+4\delta\right)}{L}(L-l-1),\end{aligned} \tag{105}$$

where the last inequality follows from Bernoulli's inequality. Using the similar argument, it follows from (41) that

$$\|\boldsymbol{\Sigma}_{L,1}^{-2} - n^{-\frac{1}{L}}\boldsymbol{I}\| \leq 2n^{-\frac{1}{L}}(\theta + 4\delta). \tag{106}$$

Then, we compute

$$\begin{aligned}
&\left|\boldsymbol{e}_k^T\boldsymbol{U}_L\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{P}^T\tilde{\boldsymbol{\Sigma}}_1^{-2(L-1)+2l}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| \\
\leq &\left|\boldsymbol{e}_k^T\boldsymbol{U}_L\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{P}^T\left(\tilde{\boldsymbol{\Sigma}}_1^{-2(L-1)+2l} - n^{\frac{-(L-1)+l}{L}}\boldsymbol{I}\right)\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| + \\
&\left|n^{\frac{-(L-1)+l}{L}}\boldsymbol{e}_k^T\boldsymbol{U}_L\left(\boldsymbol{\Sigma}_{L,1}^{-2} - n^{-\frac{1}{L}}\boldsymbol{I}\right)\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| \\
\leq &\|\boldsymbol{\Sigma}_{L,1}\|^{-2}\left\|\tilde{\boldsymbol{\Sigma}}_1^{-2(L-1)+2l} - n^{\frac{-(L-1)+l}{L}}\boldsymbol{I}\right\| + n^{\frac{-(L-1)+l}{L}}\|\boldsymbol{\Sigma}_{L,1}^{-2} - n^{-\frac{1}{L}}\boldsymbol{I}\| \\
\leq &\, 2^{1+1/L}\,(\theta + 4\delta)\,n^{\frac{-L+l}{L}}\frac{L-l-1}{L} + 2(\theta+4\delta)n^{\frac{-L+l}{L}},
\end{aligned}$$

where the second inequality follows from (105) and (106). Next, we compute

$$\begin{aligned}
\left|\boldsymbol{\omega}_{k'}^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_k\right| &\leq \|\boldsymbol{\omega}_{k'}\|\|\tilde{\boldsymbol{\Sigma}}_1\|^{-L+2l+1}\|\boldsymbol{\Sigma}_{L,1}\|^{-1} \\
&\leq 8\sqrt{\delta}\sqrt[4]{K}n^{\frac{-L+l}{L}}n^{-\frac{1}{2L}},
\end{aligned}$$

where the second inequality follows from (i) in Lemma 12 and (39). Using the same argument, we have

$$\left|\boldsymbol{\omega}_k^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{O}\tilde{\boldsymbol{\Sigma}}_1^{-(L-1)}\boldsymbol{P}\boldsymbol{\Sigma}_{L,1}^{-1}\boldsymbol{U}_L^T\boldsymbol{e}_{k'}\right| \leq 8\sqrt{\delta}\sqrt[4]{K}n^{\frac{-L+l}{L}}n^{-\frac{1}{2L}}.$$

and

$$\left|\boldsymbol{\omega}_k^T\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{\omega}_{k'}\right| \leq 16\delta\sqrt{K}n^{\frac{-L+l}{L}}n^{-\frac{1}{L}} \leq \sqrt{\delta}\sqrt[4]{K}n^{\frac{-L+l}{L}}n^{-\frac{1}{2L}}.$$

where the last inequality follows from (13). These, together with (104), yield

$$\begin{aligned}
\bar{\boldsymbol{x}}_k^T\boldsymbol{V}_{1,1}\tilde{\boldsymbol{\Sigma}}_1^{2l}\boldsymbol{V}_{1,1}^T\bar{\boldsymbol{x}}_{k'} \leq 2\sqrt{2}\,(\theta+4\delta)\,n^{\frac{-L+l}{L}}\frac{L-l-1}{L} + 2(\theta+4\delta)n^{\frac{-L+l}{L}} + \\
9\sqrt{\delta}\sqrt[4]{K}n^{\frac{-L+l}{L}}n^{-\frac{1}{2L}}.
\end{aligned} \tag{107}$$

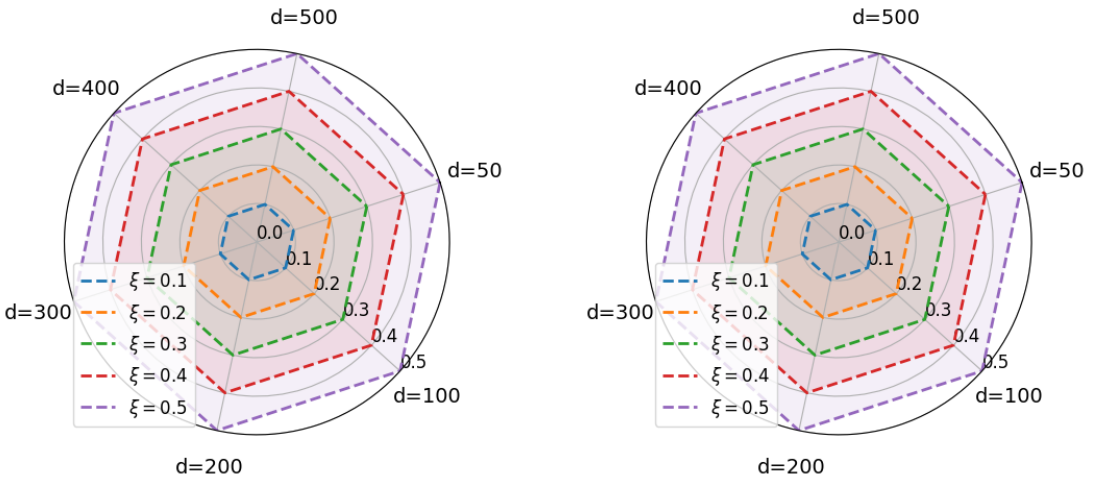
Moreover, we compute

$$\bar{\boldsymbol{x}}_k^T\boldsymbol{V}_{1,2}\tilde{\boldsymbol{\Sigma}}_2^{2l}\boldsymbol{V}_{1,2}^T\bar{\boldsymbol{x}}_{k'} \leq \|\bar{\boldsymbol{x}}_k\|\|\bar{\boldsymbol{x}}_{k'}\|\|\tilde{\boldsymbol{\Sigma}}_2^{2l}\| \leq \left(\frac{1}{n} + \frac{\theta}{N}\right)\varepsilon^{2l}.$$

where the second inequality follows from $\|\bar{\boldsymbol{x}}_k\|^2 \leq \|\sum_{i=1}^n \boldsymbol{x}_{k,i}/n\|^2 \leq 1/n + \theta/N$ due to Assumption 2. This, together with (93) in Lemma 12, (103), and (107), yields for all k, l,

$$\begin{aligned}
\frac{\langle\boldsymbol{\mu}_k^l, \boldsymbol{\mu}_{k'}^l\rangle}{\|\boldsymbol{\mu}_k^l\|\|\boldsymbol{\mu}_{k'}^l\|} \leq &\, 32\,(\theta+4\delta)\left(2 - \frac{l+1}{L}\right) \\
&+ 144\sqrt{\delta}\sqrt[4]{K}n^{-\frac{1}{2L}} + 32n^{-\frac{l}{L}}\left(1 + \frac{\theta}{K}\right)\varepsilon^{2l}.
\end{aligned}$$

This, together with (8), yields (102). ■



(a) $\min_{l\in[L-1],K+1\leq i\leq d-K}\ \sigma_i(\boldsymbol{W}_l)$ (b) $\max_{l\in[L-1],K+1\leq i\leq d-K}\ \sigma_i(\boldsymbol{W}_l)$

Figure 14: **Approximate low-rankness hold universally across different training and network configurations.** The experimental setup is deferred to Section 5.2.1. For each figure, the vertex of each polygon denotes the singular value of the weights for each d and ξ. As we observe, for each fixed ξ, the polygon is a hexagon, implying that the singular values are approximately equal and independent of the network width d. Second, the singular values grow when the initialization scale ξ increases and also confirms Theorem 5 that $\varepsilon = \xi$. Moreover, we conclude that $\sigma_{\min} \approx \sigma_{\max}$ when comparing (a) and (b). Therefore, we can see that both the minimum and maximum singular values remain unchanged, demonstrating that the GD operates only within a 2K-dimensional invariant subspace.

A.4 Proof of Theorem 4

Based on Theorem 11 and Theorem 13, we can directly obtain Theorem 4.

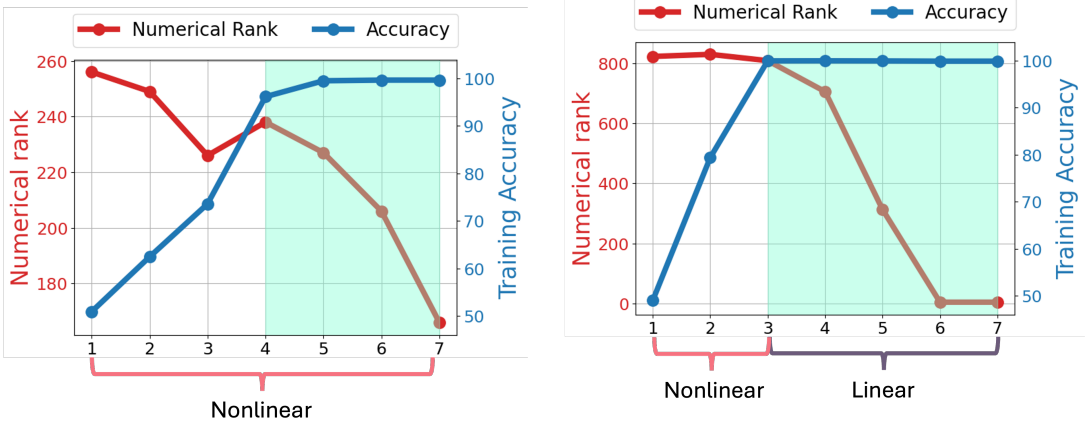
A.5 Proof of Theorem 5

Now, we study the convergence behavior of gradient flow with the initialization (18) for solving Problem (3).

Proof Since GD converges to a global optimal solution, we have $\boldsymbol{W}_{L:1}\boldsymbol{X} = \boldsymbol{Y}$. This, together with the fact that $\boldsymbol{X}$ is square and orthogonal, yields (10). According to Du et al. (2018); Arora et al. (2018b), the iterates of gradient flow for solving Problem (3) satisfy (19). This, together with the initialization (18), yields for all $t \geq 0$,

$$\|\boldsymbol{W}_{l+1}(t)^T\boldsymbol{W}_{l+1}(t) - \boldsymbol{W}_l(t)\boldsymbol{W}_l(t)^T\|_F = 0, \forall l \in [L-2],$$
$$\|\boldsymbol{W}_L(t)^T\boldsymbol{W}_L(t) - \boldsymbol{W}_{L-1}(t)\boldsymbol{W}_{L-1}(t)^T\|_F = \xi^2\sqrt{d-K}.$$

This implies that (11) holds with $\delta = \xi^2\sqrt{d-K}$. Moreover, using (Yaras et al., 2023, Theorem 1), we obtain that (12) holds with $\varepsilon = \xi$ and $\rho = 0$. Then, the proof is completed. ■



(a) A 7-layer MLP network with ReLU

(b) A 7-layer hybrid network

Figure 15: **Numerical rank and training accuracy across layers on the SST-5 language dataset.** We train two networks with different architectures on the SST-5 dataset: (a) A 7-layer multilayer perceptron (MLP) network with ReLU activation, (b) A hybrid network formed by 3-layer MLP with ReLU activation followed by a 4-layer linear network. For each figure, we plot the numerical rank of the features of each layer and training accuracy by applying linear probing to the output of each layer against the number of layers, respectively. The green shade indicates that the features at these layers are *approximately* linearly separable, as evidenced by the near-perfect accuracy achieved by a linear classifier.

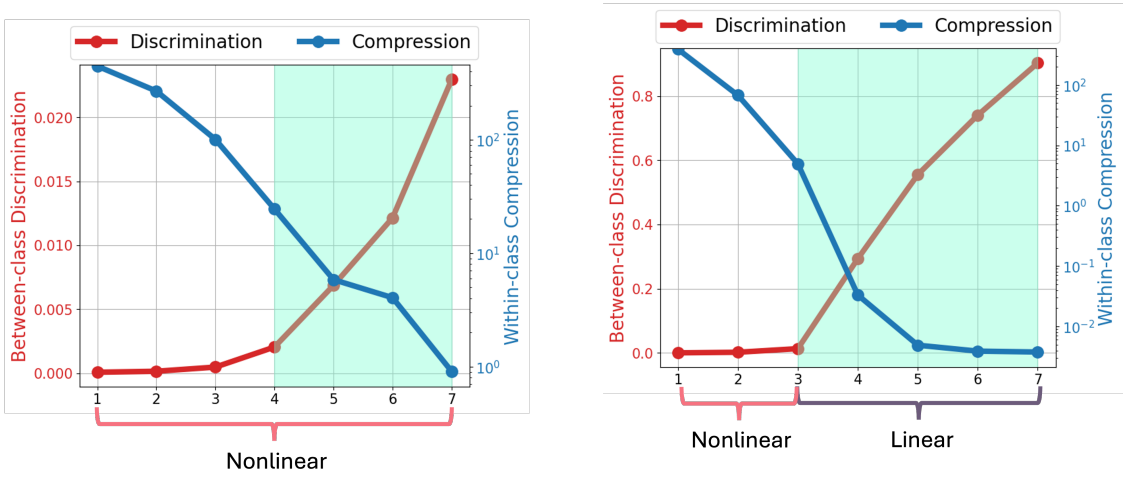
Appendix B. Extra Experiments

B.1 Assumption Discussion

Here, we provide more experimental results for validating the low-rankness property in Theorem 3 of trained weights in DLN. To validate the low-rankness assumption, we train 4-layer DLNs using the orthogonal initialization with width $d \in \{50, 100, 200, 300, 400, 500\}$ and initialization scale $\xi \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. The training setting is the same with Section 5.2. For the weights $\{\boldsymbol{W}_l\}_{l=1}^{L-1}$ of each network that have been trained until convergence, we plot

$$\text{minimum singular value} \quad \sigma_{\min} := \min_{K+1 \le i \le d-K,\, l \in [L-1]} \sigma_i(\boldsymbol{W}_l),$$
$$\text{maximum singular value} \quad \sigma_{\max}; := \max_{K+1 \le i \le d-K,\, l \in [L-1]} \sigma_i(\boldsymbol{W}_l).$$

The results are shown in Figure 14. Here, for each $l \in [L-1]$, we assume that $\sigma_1(\boldsymbol{W}_l) \ge \sigma_2(\boldsymbol{W}_l) \ge \cdots \ge \sigma_d(\boldsymbol{W}_l)$. For each figure, the vertex of each polygon denotes the singular value of the weights for each d and ξ. As we observe, for each fixed ξ, the polygon is a hexagon, implying that the singular values are approximately equal and independent of the network width d. Second, the singular values grow when the initialization scale ξ increases and also confirms Theorem 5 that $\varepsilon = \xi$. Moreover, we conclude that $\sigma_{\min} \approx \sigma_{\max}$ when comparing Figure 14a and Figure 14b. Therefore, we can see that both the minimum and



(a) A 7-layer MLP network with ReLU

(b) A 7-layer hybrid network

Figure 16: **Within-class compression and between-class discrimination across layers on the SST-5 text dataset.** We use the same setup as in Figure 15 and plot the metrics of within-class compression and between-class discrimination of the two networks across layers after training, respectively.

maximum singular values remain unchanged, demonstrating that the GD operates only within a 2K-dimensional invariant subspace.

B.2 Exploration on Other Datasets

In the main body of the paper, we primarily present results on the CIFAR and Fashion-MNIST datasets. In this subsection, we extend our experiments to a broader range of datasets.

Experiments on Language Dataset. To further illustrate the analogous roles of deep linear and nonlinear layers across different data modalities, we reproduce the experiments from Figure 1 using the SST-5 text dataset (Socher et al., 2013), keeping all experimental settings unchanged except for the dataset. As shown in Figure 15, we observe a similar trend: early nonlinear layers enhance linear separability, while deeper layers—regardless of their nonlinearity—progressively compress the features. We also use this experimental setup to verify Theorem 4, with results presented in Figure 16. As observed, within-class compression and between-class discrimination respectively exhibit progressively collapsing and increasing trends, further confirming that our theoretical insights generalize beyond the image domain.

Experiments on ImageNet. Due to resource constraints, we use the PyTorch pre-trained VGG11 network (Simonyan and Zisserman, 2015) on ImageNet (Deng et al., 2009) to extract features and evaluate within-class compression and between-class discrimination. The results are presented in Figure 17. We observe that the compression metric continues to exhibit an approximately geometric decay, and the discrimination metric shows increasing pattern, thereby validating our theory on the large-scale benchmark dataset.

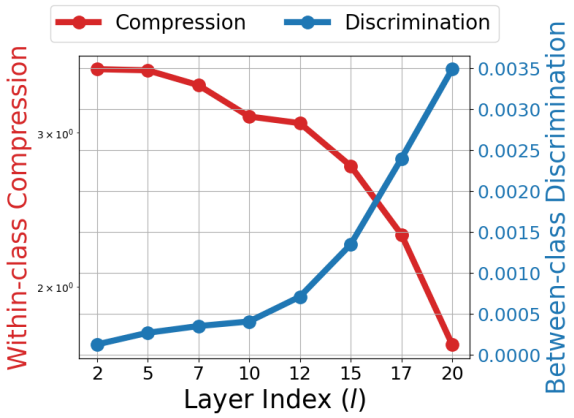


Figure 17: **Within-class compression and between-class discrimination of features from a VGG11 network pre-trained on ImageNet.** We use the PyTorch-provided pre-trained VGG11 model and extract features from layers 2, 5, 7, 10, 12, 15, 17, 20.

Appendix C. Auxiliary Results

Lemma 14 *Given a matrix $\boldsymbol{A} \in \mathbb{R}^{m\times n}$ of rank $r \in \mathbb{N}_+$, we have*

$$\|\boldsymbol{A}^T\boldsymbol{A}\|_F \le \|\boldsymbol{A}\|_F^2 \le \sqrt{r}\|\boldsymbol{A}^T\boldsymbol{A}\|_F. \tag{108}$$

Proof Let $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ be a singular value decomposition of $\boldsymbol{A}$, where

$$\boldsymbol{\Sigma} = \begin{bmatrix} \tilde{\boldsymbol{\Sigma}} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}, \ \tilde{\boldsymbol{\Sigma}} = \mathrm{diag}(\sigma_1, \ldots, \sigma_r),$$

$\boldsymbol{U} \in \mathcal{O}^m$, and $\boldsymbol{V} \in \mathcal{O}^n$. Then we compute

$$\|\boldsymbol{A}^T\boldsymbol{A}\|_F^2 = \|\boldsymbol{\Sigma}^T\boldsymbol{\Sigma}\|_F^2 = \sum_{i=1}^r \sigma_i^4, \ \|\boldsymbol{A}\|_F^2 = \|\boldsymbol{\Sigma}\|_F^2 = \sum_{i=1}^r \sigma_i^2,$$

which, together with the AM-QM inequality, directly implies (108). ■

Lemma 15 *For arbitrary matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$, if $\boldsymbol{A}$ has full column rank and $\boldsymbol{B} \neq \boldsymbol{0}$ or $\boldsymbol{B}$ has full row rank and $\boldsymbol{A} \neq \boldsymbol{0}$, it holds that*

$$\sigma_{\min}(\boldsymbol{A}\boldsymbol{B}) \ge \sigma_{\min}(\boldsymbol{A})\,\sigma_{\min}(\boldsymbol{B}). \tag{109}$$

Lemma 16 *Suppose that $\boldsymbol{Q} \in \mathcal{O}^{d\times K}$ is an orthonormal matrix and $\boldsymbol{A} \in \mathbb{R}^{K\times K}$ is a symmetric matrix of full rank, where $d > K$. Let $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \boldsymbol{Q}\boldsymbol{A}\boldsymbol{Q}^T$ be a compact singular value decomposition of $\boldsymbol{Q}\boldsymbol{A}\boldsymbol{Q}^T$, where $\boldsymbol{U} \in \mathcal{O}^{d\times K}$ and $\boldsymbol{\Lambda} \in \mathbb{R}^{K\times K}$ is a diagonal matrix. Then, it holds that*

$$\boldsymbol{Q}\boldsymbol{Q}^T = \boldsymbol{U}\boldsymbol{U}^T. \tag{110}$$

Proof It follows from $\boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T = \boldsymbol{Q}\boldsymbol{A}\boldsymbol{Q}^T$, $\boldsymbol{Q} \in \mathcal{O}^{d\times K}$, and $\boldsymbol{U} \in \mathcal{O}^{d\times K}$ that

$$\boldsymbol{Q}^T\boldsymbol{U}\boldsymbol{\Lambda} = \boldsymbol{A}\boldsymbol{Q}^T\boldsymbol{U}. \tag{111}$$

Now, left-multiplying both sides by $\boldsymbol{Q}$ yields

$$\boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{U}\boldsymbol{\Lambda} = \boldsymbol{Q}\boldsymbol{A}\boldsymbol{Q}^T\boldsymbol{U} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{U}\boldsymbol{\Lambda}.$$

Since $\boldsymbol{A}$ is a symmetric matrix of full rank, $\boldsymbol{\Lambda}$ is a diagonal matrix with non-zero entries. We can multiply both sides by $\boldsymbol{\Lambda}^{-1}$ and obtain $\boldsymbol{Q}\boldsymbol{Q}^T\boldsymbol{U} = \boldsymbol{U}$, which implies (110) by $\boldsymbol{U} \in \mathcal{O}^{d\times K}$. ■

Lemma 17 *For every real number $1 \le \theta \le 1$ and $x \ge -1$, it holds that*

$$(1+x)^\theta \le 1 + \theta x.$$

References

E. M. Achour, F. Malgouyres, and S. Gerchinovitz. The loss landscape of deep linear neural networks: a second-order analysis. *Journal of Machine Learning Research*, 25(242):1–76, 2024.

G. Alain and Y. Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations*, 2017.

Z. Allen-Zhu and Y. Li. Backward feature correction: How deep learning performs deep (hierarchical) learning. In *Conference on Learning Theory*, pages 4598–4598. PMLR, 2023.

A. Ansuini, A. Laio, J. H. Macke, and D. Zoccolan. Intrinsic dimension of data representations in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

S. Arora, R. Ge, and A. Moitra. New algorithms for learning incoherent and overcomplete dictionaries. In *Conference on Learning Theory*, pages 779–806. PMLR, 2014.

S. Arora, N. Cohen, N. Golowich, and W. Hu. A convergence analysis of gradient descent for deep linear neural networks. In *International Conference on Learning Representations*, 2018a.

S. Arora, N. Cohen, and E. Hazan. On the optimization of deep networks: Implicit acceleration by overparameterization. In *International Conference on Machine Learning*, pages 244–253. PMLR, 2018b.

S. Arora, N. Cohen, W. Hu, and Y. Luo. Implicit regularization in deep matrix factorization. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

B. Bah, H. Rauhut, U. Terstiege, and M. Westdickenberg. Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers. *Information and Inference: A Journal of the IMA*, 11(1):307–353, 2022.

P. L. Bartlett, P. M. Long, G. Lugosi, and A. Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

M. Belkin, D. Hsu, S. Ma, and S. Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

I. Ben-Shaul and S. Dekel. Nearest class-center simplification through intermediate layers. In *Topological, Algebraic and Geometric Learning Workshops 2022*, pages 37–47. PMLR, 2022.

A. Bietti, J. Bruna, C. Sanford, and M. J. Song. Learning single-index models with shallow neural networks. In *Advances in Neural Information Processing Systems*, volume 35, pages 9768–9783, 2022.

E. Boursier, L. Pillaud-Vivien, and N. Flammarion. Gradient flow dynamics of shallow relu networks for square loss and orthogonal inputs. In *Advances in Neural Information Processing Systems*, volume 35, pages 20105–20118, 2022.

Y. Cao, D. Zou, Y. Li, and Q. Gu. The implicit bias of batch normalization in linear models and two-layer linear convolutional neural networks. In *Conference on Learning Theory*, volume 195, pages 5699–5753. PMLR, 2023.

K. H. R. Chan, Y. Yu, C. You, H. Qi, J. Wright, and Y. Ma. Redunet: A white-box deep network from the principle of maximizing rate reduction. *Journal of Machine Learning Research*, 23(114):1–103, 2022.

N. S. Chatterji and P. M. Long. Deep linear networks can benignly overfit when shallow ones do. *Journal of Machine Learning Research*, 24(117):1–39, 2023.

N. S. Chatterji, P. M. Long, and P. L. Bartlett. The interplay between implicit bias and benign overfitting in two-layer linear networks. *Journal of Machine Learning Research*, 23(1):12062–12109, 2022.

M. Chen, D. Y. Fu, A. Narayan, M. Zhang, Z. Song, K. Fatahalian, and C. Ré. Perfectly balanced: Improving transfer and robustness of supervised contrastive learning. In *International Conference on Machine Learning*, pages 3090–3122. PMLR, 2022a.

P. Chen, R. Jiang, and P. Wang. A complete loss landscape analysis of regularized deep matrix factorization. *arXiv preprint arXiv:2506.20344*, 2025.

T. Chen, S. Kornblith, M. Norouzi, and G. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020.

X. Chen and K. He. Exploring simple siamese representation learning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15745–15753, 2021.

Y. Chen, A. Yuille, and Z. Zhou. Which layer is learning faster? A systematic exploration of layer-wise convergence rate for deep neural networks. In *International Conference on Learning Representations*, 2022b.

A. Damian, J. Lee, and M. Soltanolkotabi. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pages 5413–5452. PMLR, 2022.

H. Dang, T. T. Huu, S. Osher, H. T. Tran, N. Ho, and T. M. Nguyen. Neural collapse in deep linear networks: From balanced to imbalanced data. In *International Conference on Machine Learning*, pages 6873–6947. PMLR, 2023.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009.

D. L. Donoho and M. Elad. Optimally sparse representation in general (nonorthogonal) dictionaries via l1 minimization. *Proceedings of the National Academy of Sciences*, 100 (5):2197–2202, 2003.

S. S. Du and W. Hu. Width provably matters in optimization for deep linear neural networks. In *International Conference on Machine Learning*, pages 1655–1664. PMLR, 2019.

S. S. Du, W. Hu, and J. D. Lee. Algorithmic regularization in learning deep homogeneous models: Layers are automatically balanced. In *Advances in Neural Information Processing Systems*, volume 31, pages 6151–6160, 2018.

A. Eftekhari. Training linear neural networks: Non-local convergence and complexity results. In *International Conference on Machine Learning*, pages 2836–2847. PMLR, 2020.

A. Esteva, A. Robicquet, B. Ramsundar, V. Kuleshov, M. DePristo, K. Chou, C. Cui, G. Corrado, S. Thrun, and J. Dean. A guide to deep learning in healthcare. *Nature Medicine*, 25(1):24–29, 2019.

C. Fang, H. He, Q. Long, and W. J. Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43), 2021.

S. Frei, G. Vardi, P. Bartlett, N. Srebro, and W. Hu. Implicit bias in leaky ReLU networks trained on high-dimensional data. In *International Conference on Learning Representations*, 2023.

T. Galanti, L. Galanti, and I. Ben-Shaul. On the implicit bias towards minimal depth of deep neural networks. *arXiv preprint arXiv:2202.09028*, 2022a.

T. Galanti, A. György, and M. Hutter. On the role of neural collapse in transfer learning. In *International Conference on Learning Representations*, 2022b.

B. Geshkovski, C. Letrouit, Y. Polyanskiy, and P. Rigollet. The emergence of clusters in self-attention dynamics. In *Advances in Neural Information Processing Systems*, volume 36, pages 57026–57037, 2023.

G. Gidel, F. Bach, and S. Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, volume 32, pages 3202–3211, 2019.

X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

S. Gunasekar, B. E. Woodworth, S. Bhojanapalli, B. Neyshabur, and N. Srebro. Implicit regularization in matrix factorization. In *Advances in Neural Information Processing Systems*, volume 30, pages 6151–6159, 2017.

S. Guo, J. M. Alvarez, and M. Salzmann. Expandnets: Linear over-parameterization to train compact convolutional networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 1298–1310, 2020.

X. Han, V. Papyan, and D. L. Donoho. Neural collapse under MSE loss: Proximity to and dynamics on the central path. In *International Conference on Learning Representations*, 2021.

M. Hardt and T. Ma. Identity matters in deep learning. In *International Conference on Learning Representations*, 2016.

H. He and W. J. Su. A law of data separation in deep learning. *Proceedings of the National Academy of Sciences*, 120(36):e2221704120, 2023.

K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1026–1034, 2015.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

W. Hong and S. Ling. Beyond unconstrained features: Neural collapse for shallow neural networks with general data. *arXiv preprint arXiv:2409.01832*, 2024.

W. Hu, L. Xiao, and J. Pennington. Provable benefit of orthogonal initialization in optimizing deep linear networks. In *International Conference on Learning Representations*, 2019.

W. Hu, L. Xiao, B. Adlam, and J. Pennington. The surprising simplicity of the early-time learning dynamics of neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 17116–17128, 2020.

J. Huang and H.-T. Yau. Dynamics of deep neural networks and neural tangent hierarchy. In *International Conference on Machine Learning*, pages 4542–4551. PMLR, 2020.

M. Huh, H. Mobahi, R. Zhang, B. Cheung, P. Agrawal, and P. Isola. The low-rank simplicity bias in deep networks. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856.

L. Hui and M. Belkin. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. In *International Conference on Learning Representations*, 2020.

L. Hui, M. Belkin, and P. Nakkiran. Limitations of neural collapse for understanding generalization in deep learning. *arXiv preprint arXiv:2202.08384*, 2022.

S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456. PMLR, 2015.

A. Jacot, F. Gabriel, and C. Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, volume 31, pages 8571–8580, 2018.

A. Jacot, P. Súkeník, Z. Wang, and M. Mondelli. Wide neural networks trained with weight decay provably exhibit neural collapse. In *International Conference on Learning Representations*, 2025.

Z. Ji and M. Telgarsky. Gradient descent aligns the layers of deep linear networks. In *International Conference on Learning Representations*, 2018.

Z. Ji and M. Telgarsky. The implicit bias of gradient descent on nonseparable data. In *Conference on Learning Theory*, pages 1772–1798. PMLR, 2019.

K. Kawaguchi. Deep learning without poor local minima. In *Advances in Neural Information Processing Systems*, volume 29, pages 586–594, 2016.

S. Kornblith, T. Chen, H. Lee, and M. Norouzi. Why do better loss functions lead to less transferable features? In *Advances in Neural Information Processing Systems*, volume 34, pages 28648–28662, 2021.

V. Kothapalli and T. Tirer. Can kernel methods explain how the data affects neural collapse? *Transactions on Machine Learning Research*, 2025. ISSN 2835-8856.

V. Kothapalli, T. Tirer, and J. Bruna. A neural collapse perspective on feature evolution in graph neural networks. In *Advances in Neural Information Processing Systems*, volume 36, pages 14134–14191, 2023.

A. Krizhevsky and G. E. Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25, 2012.

D. Kunin, A. Yamamura, C. Ma, and S. Ganguli. The asymmetric maximum margin bias of quasi-homogeneous neural networks. In *International Conference on Learning Representations*, 2022.

S. M. Kwon, Z. Zhang, D. Song, L. Balzano, and Q. Qu. Efficient low-dimensional compression of overparameterized models. In *International Conference on Artificial Intelligence and Statistics*, volume 238, pages 1009–1017. PMLR, 2024.

A. K. Lampinen and S. Ganguli. An analytic theory of generalization dynamics and transfer learning in deep linear networks. In *International Conference on Learning Representations*, 2018.

T. Laurent and J. Brecht. Deep linear networks with arbitrary loss: All local minima are global. In *International Conference on Machine Learning*, pages 2902–2907. PMLR, 2018.

Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

P. Li, X. Li, Y. Wang, and Q. Qu. Neural collapse in multi-label learning with pick-all-label loss. In *International Conference on Machine Learning*, pages 28060–28094, 2024a.

X. Li, S. Liu, J. Zhou, X. Lu, C. Fernandez-Granda, Z. Zhu, and Q. Qu. Understanding and improving transfer learning of deep models via neural collapse. *Transactions on Machine Learning Research*, 2024b. ISSN 2835-8856.

I. Loshchilov and F. Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017.

H. Lu and K. Kawaguchi. Depth creates no bad local minima. *arXiv preprint arXiv:1702.08580*, 2017.

K. Lyu and J. Li. Gradient descent maximizes the margin of homogeneous neural networks. In *International Conference on Learning Representations*, 2020.

Y. Ma, D. Tsao, and H.-Y. Shum. On the principles of parsimony and self-consistency for the emergence of intelligence. *Frontiers of Information Technology & Electronic Engineering*, 23(9):1298–1323, 2022.

W. Masarczyk, M. Ostaszewski, E. Imani, R. Pascanu, P. Miłoś, and T. Trzcinski. The tunnel effect: Building data representations in deep neural networks. In *Advances in Neural Information Processing Systems*, volume 36, pages 76772–76805, 2023.

L. McInnes, J. Healy, N. Saul, and L. Großberger. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861, 2018.

H. Min, S. Tarmoun, R. Vidal, and E. Mallada. On the explicit role of initialization on the convergence and implicit bias of overparametrized linear networks. In *International Conference on Machine Learning*, pages 7760–7768. PMLR, 2021.

D. G. Mixon, H. Parshall, and J. Pi. Neural collapse with unconstrained features. *arXiv preprint arXiv:2011.11619*, 2020.

M. S. Nacson, J. Lee, S. Gunasekar, P. H. P. Savarese, N. Srebro, and D. Soudry. Convergence of gradient descent on separable data. In *International Conference on Artificial Intelligence and Statistics*, pages 3420–3428. PMLR, 2019.

B. Neyshabur. Implicit regularization in deep learning. *arXiv preprint arXiv:1709.01953*, 2017.

G. M. Nguegnang, H. Rauhut, and U. Terstiege. Convergence of gradient descent for learning linear neural networks. *arXiv preprint arXiv:2108.02040*, 2021.

E. Nichani. *An Empirical and Theoretical Analysis of the Role of Depth in Convolutional Neural Networks*. PhD thesis, Massachusetts Institute of Technology, 2021.

V. Papyan, X. Han, and D. L. Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117 (40):24652–24663, 2020.

J. Pennington, S. Schoenholz, and S. Ganguli. The emergence of spectral universality in deep networks. In *International Conference on Artificial Intelligence and Statistics*, pages 1924–1932. PMLR, 2018.

M. Phuong and C. H. Lampert. The inductive bias of relu networks on orthogonally separable data. In *International Conference on Learning Representations*, 2020.

Q. Qu, Y. Zhai, X. Li, Y. Zhang, and Z. Zhu. Geometric analysis of nonconvex optimization landscapes for overcomplete learning. In *International Conference on Learning Representations*, 2020.

A. Rangamani, M. Lindegaard, T. Galanti, and T. A. Poggio. Feature learning in deep classifiers through intermediate neural collapse. In *International Conference on Machine Learning*, pages 28729–28745. PMLR, 2023.

S. Recanatesi, M. Farrell, M. Advani, T. Moore, G. Lajoie, and E. Shea-Brown. Dimensionality compression and expansion in deep neural networks. *arXiv preprint arXiv:1906.00443*, 2019.

A. M. Saxe, J. L. McClelland, and S. Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116 (23):11537–11546, 2019.

H. Shah, K. Tamuly, A. Raghunathan, P. Jain, and P. Netrapalli. The pitfalls of simplicity bias in neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9573–9585, 2020.

Y. Shin. Effects of depth, width, and initialization: A convergence analysis of layer-wise training for deep linear neural networks. *Analysis and Applications*, 20(01):73–119, 2022.

R. Shwartz Ziv and Y. LeCun. To compress or not to compress—self-supervised learning and information theory: A review. *Entropy*, 26(3):252, 2024.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642. Association for Computational Linguistics, 2013.

D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

G. W. Stewart and J.-g. Sun. Matrix perturbation theory. 1990.

P. Súkeník, M. Mondelli, and C. H. Lampert. Deep neural collapse is provably optimal for the deep unconstrained features model. *Advances in Neural Information Processing Systems*, 36:52991–53024, 2023.

J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere i: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2):853–884, 2016.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

S. Tarmoun, G. Franca, B. D. Haeffele, and R. Vidal. Understanding the dynamics of gradient flow in overparameterized linear models. In *International Conference on Machine Learning*, pages 10153–10161. PMLR, 2021.

T. Tirer and J. Bruna. Extended unconstrained features model for exploring deep neural collapse. In *International Conference on Machine Learning*, pages 21478–21505. PMLR, 2022.

T. Tirer, H. Huang, and J. Niles-Weed. Perturbation analysis of neural collapse. In *International Conference on Machine Learning*, pages 34301–34329. PMLR, 2023.

G. Valle-Perez, C. Q. Camargo, and A. A. Louis. Deep learning generalizes because the parameter-function map is biased towards simple functions. In *International Conference on Learning Representations*, 2018.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

P. Wang, H. Liu, C. Yaras, L. Balzano, and Q. Qu. Linear convergence analysis of neural collapse with unconstrained features. In *OPT 2022: Optimization for Machine Learning*, 2022.

S. Wang, K. Gai, and S. Zhang. Progressive feedforward collapse of resnet training. *arXiv preprint arXiv:2405.00985*, 2024.

H. Xiao, K. Rasul, and R. Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

L. Xiao, Y. Bahri, J. Sohl-Dickstein, S. Schoenholz, and J. Pennington. Dynamical isometry and a mean field theory of cnns: How to train 10,000-layer vanilla convolutional neural networks. In *International Conference on Machine Learning*, pages 5393–5402. PMLR, 2018.

S. Xie, J. Qiu, A. Pasad, L. Du, Q. Qu, and H. Mei. Hidden state variability of pretrained language models can guide computation reduction for transfer learning. In *Conference on Empirical Methods in Natural Language Processing*, 2022.

M. Xu, A. Rangamani, Q. Liao, T. Galanti, and T. Poggio. Dynamics in deep classifiers trained with the square loss: Normalization, low rank, neural collapse, and generalization bounds. *Research*, 6:0024, 2023.

C. Yaras, P. Wang, Z. Zhu, L. Balzano, and Q. Qu. Neural collapse with normalized features: A geometric analysis over the Riemannian manifold. *Advances in Neural Information Processing Systems*, 35:11547–11560, 2022.

C. Yaras, P. Wang, W. Hu, Z. Zhu, L. Balzano, and Q. Qu. The law of parsimony in gradient descent for learning deep linear networks. *arXiv preprint arXiv:2306.01154*, 2023.

C. Yaras, P. Wang, L. Balzano, and Q. Qu. Compressible dynamics in deep overparameterized low-rank learning & adaptation. In *International Conference on Machine Learning*, pages 56946–56965, 2024.

J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Y. Yu, K. H. R. Chan, C. You, C. Song, and Y. Ma. Learning diverse and discriminative representations via the principle of maximal coding rate reduction. In *Advances in Neural Information Processing Systems*, volume 33, pages 9422–9434, 2020.

Y. Yu, S. Buchanan, D. Pai, T. Chu, Z. Wu, S. Tong, H. Bai, Y. Zhai, B. D. Haeffele, and Y. Ma. White-box transformers via sparse rate reduction: Compression is all there is? *Journal of Machine Learning Research*, 25(300):1–128, 2024.

Y. Zhai, Z. Yang, Z. Liao, J. Wright, and Y. Ma. Complete dictionary learning via L4-norm maximization over the orthogonal group. *Journal of Machine Learning Research*, 21(1): 6622–6689, 2020.

C. Zhang, S. Bengio, and Y. Singer. Are all layers created equal? *Journal of Machine Learning Research*, 23(1):2930–2957, 2022.

J. Zhou, X. Li, T. Ding, C. You, Q. Qu, and Z. Zhu. On the optimization landscape of neural collapse under MSE loss: Global optimality with unconstrained features. In *International Conference on Machine Learning*, pages 27179–27202. PMLR, 2022a.

J. Zhou, C. You, X. Li, K. Liu, S. Liu, Q. Qu, and Z. Zhu. Are all losses created equal: A neural collapse perspective. In *Advances in Neural Information Processing Systems*, volume 35, pages 31697–31710, 2022b.

Z. Zhu, T. Ding, J. Zhou, X. Li, C. You, J. Sulam, and Q. Qu. A geometric analysis of neural collapse with unconstrained features. In *Advances in Neural Information Processing Systems*, volume 34, 2021.