Emergent Low-Rank Training Dynamics in MLPs with Smooth Activations

Alec S. Xu
University of Michigan
alecx@umich.edu

Can Yaras
Princeton University
cy4900@princeton.edu

Matthew Asato
University of Michigan
asamatt@umich.edu

Qing Qu
University of Michigan
qingqu@umich.edu

Laura Balzano
University of Michigan
girasole@umich.edu

Abstract

Recent empirical studies have shown the training dynamics of deep neural networks mostly occur within low-dimensional subspaces. While this has inspired new research in low-rank training, compression, and adaptation, theoretical justification for these dynamics in nonlinear networks remains limited. To address this gap, this paper analyzes the learning dynamics of multi-layer perceptrons (MLPs) under gradient descent (GD). We demonstrate that the weight dynamics concentrate within invariant low-dimensional subspaces throughout training. Theoretically, we precisely characterize these invariant subspaces for two-layer networks with smooth nonlinear activations, providing insight into their emergence. Experimentally, we validate that this phenomenon extends well beyond our theoretical setting. Leveraging these insights, we empirically show there exists a low-rank MLP parameterization that, when initialized in the appropriate subspaces, nearly matches the performance of their fully-parameterized counterparts on classification tasks. Code for our experiments can be found at https://github.com/alecxu00/mlp-low-dim-gd.

1 Introduction

Recently, low-rank approaches have achieved great success in training and fine-tuning large neural networks. For example, low-rank adaptation (LoRA) [1] is a popular fine-tuning technique for large language models (LLMs) by adding a low-rank adapter to frozen pre-trained weights. Other works have proposed projecting the parameters [2, 3] or optimizer states [4–7] onto low-dimensional subspaces, and then updating the parameters or optimizer states there; see [8] for a survey.

Empirical observations indicate that large model training and fine-tuning naturally occur within low-dimensional subspaces [9–11], which potentially explains the success of the aforementioned low-rank training approaches. However, a precise characterization of *which* subspaces the optimization occurs within remains unclear. To address this question, [12–14] proved that when deep *linear* networks are trained via gradient descent (GD), the weights are updated within fixed subspaces that depend on the weights and the first-layer gradient at initialization. However, practical neural networks are highly nonlinear, and no prior work has theoretically investigated how introducing nonlinearities impacts this phenomenon. To this end, we extend these works by focusing on multi-layer perceptrons (MLPs). We find that for the MLP with small output dimension and smooth nonlinear activation functions, the training dynamics are highly concentrated within unchanging low-dimensional subspaces. Our contributions are as follows:

- **Theoretical analysis of two-layer networks (Section 3).** We provide the first theoretical analysis of this phenomenon in two-layer networks trained via GD. Specifically, we show that there exists a

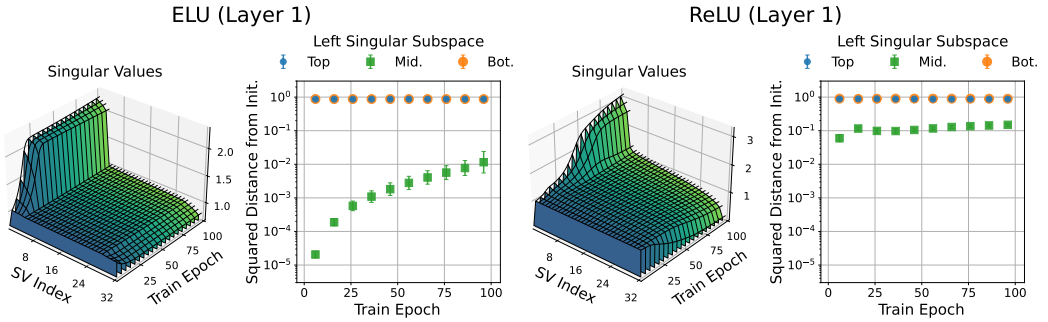


Figure 1: The middle left singular subspace of the first-layer weight matrix in the ELU network evolves noticeably slower than that in the ReLU network, and the corresponding singular values remain closer to their initialization. The right singular subspaces exhibit similar dynamics in both cases (see Appendix B.1).

fixed subspace such that, in each GD step, the change of the first-layer weights in this subspace is bounded. This subspace depends on the first layer's weights and gradient at initialization.

- **Low-rank training beyond theory (Section 4).** Empirically, we show that the phenomenon of low-rank training dynamics extends well beyond our theoretical settings. Specifically, in deep MLPs, we provide precise characterizations of layer-specific, low-dimensional subspaces in which each layer's weight matrix is mostly trained; these layer-specific subspaces also depend on the network at initialization. We empirically demonstrate that GD, SGD with momentum, Adam, and Muon all train deep MLP weight matrices in these identified subspaces.
- **Subspace-initialized low-rank MLPs (Section 5).** Based on the previous insights, we empirically show there exists a low-rank MLP parameterization that, *if initialized using the identified subspaces from Sections 3 and 4*, achieves near-equivalent performance compared to their fully parameterized counterparts under the same training recipe. We conduct experiments on Fashion MNIST and CIFAR-10 using deep MLPs demonstrating this near-equivalence and the importance of the initialization.

Related work. There have been several works on low-dimensional learning in neural networks [2, 10, 11, 13–16], as well as low-rank gradients in nonlinear networks [4, 17–19]. Another relevant line of work is the implicit bias towards low-rank weights in nonlinear networks [20–23]. See Appendix A for more detailed discussions.

2 Problem Setup

Here, we set up and motivate our problem of interest.

Notation. We use unbolded letters x, X for scalars, bold lower case letters $\boldsymbol{x}$ for vectors, and bold capital letters $\boldsymbol{X}$ for matrices. For some $N \in \mathbb{N}$, $[N]$ denotes the set $\{1, 2, \ldots, N\}$. For scalars a, b, we say $a \lesssim \mathcal{O}(b)$ if there exists a constant C s.t. $a \leq C \cdot b$, $a \gtrsim \Omega(b)$ if $a \geq C \cdot b$, and $a = \Theta(b)$ if $a = C \cdot b$. We use $\sigma_i(\boldsymbol{X})$, $\|\boldsymbol{X}\|_F$, $\|\boldsymbol{X}\|_1$, and $\|\boldsymbol{X}\|_{\max}$ to respectively denote the i^{th} singular value, Frobenius norm, matrix-1 norm, and maximum magnitude element. Finally, $\mathcal{R}(\boldsymbol{X})$ denotes the range (or column space) of $\boldsymbol{X}$, and $\mathcal{R}^{\perp}(\boldsymbol{X})$ its orthogonal complement.

Data. We consider data with inputs $\boldsymbol{X} \in \mathbb{R}^{d \times N}$ and labels $\boldsymbol{Y} \in \mathbb{R}^{K \times N}$, where d is the data dimension, N is the number of data points, and K is the label dimension. We consider the case where K is much smaller compared to d and N. Finally, we define $p := d - 2K$.

Network architecture. We define an L-layer neural network $f_{\boldsymbol{\Theta}} : \mathbb{R}^{d \times N} \to \mathbb{R}^{K \times N}$ as follows:

$$f_{\boldsymbol{\Theta}}(\boldsymbol{X}) = \boldsymbol{W}_L \phi\left(\boldsymbol{W}_{L-1} \phi\left(\ldots \phi\left(\boldsymbol{W}_1 \boldsymbol{X}\right) \ldots\right)\right), \tag{1}$$

where $\phi(\cdot)$ is the element-wise nonlinear activation function, $\boldsymbol{W}_l$ denotes the l^{th} layer's weight matrix, and $\boldsymbol{\Theta} = \{\boldsymbol{W}_1, \ldots, \boldsymbol{W}_L\}$ denote the model parameters. Here, $\boldsymbol{W}_1 \in \mathbb{R}^{m_1 \times d}$, $\boldsymbol{W}_l \in \mathbb{R}^{m_l \times m_{l-1}}$ for $l \in \{2, \ldots, L-1\}$, and $\boldsymbol{W}_L \in \mathbb{R}^{K \times m_{L-1}}$. For simplicity, we assume $m_1 = \cdots = m_{L-1} := m$.

Training. We train $f_{\boldsymbol{\Theta}}$ defined in (1) via the loss $\min_{\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{\Theta}) := \ell\left(f_{\boldsymbol{\Theta}}(\boldsymbol{X}), \boldsymbol{Y}\right)$ using gradient descent $\boldsymbol{\Theta}(t+1) = \boldsymbol{\Theta}(t) - \eta \nabla \mathcal{L}(\boldsymbol{\Theta}(t))$, where t denotes the iteration index.

2.1 Case Study: Smooth Activations Encourage Lower-Rank Training Dynamics

To motivate our problem of interest, we present a case study showcasing when low-rank training dynamics emerge in MLPs. We trained six different MLPs using full-batch GD on synthetic classification data with $d = 32$ dimensions and $K = 4$ classes. Each MLP contained a different activation function: three had smooth activations (ELU, GELU, SiLU), while the other three had nonsmooth ones (ReLU, $\mathrm{Leaky\text{-}ReLU}$, $\mathrm{Randomized-ReLU}$). During training, we tracked each of the first three layers' singular values, as well as the changes in their top-K, bottom-K, and middle $d-2K$ singular subspaces. We show results for the first layer of the ELU and ReLU networks in Figure 1, while deferring other results and experimental details to Appendix B.1.

In the ELU network's first layer, the middle $d-2K$ singular subspace evolves very slowly throughout training, especially compared to that of the ReLU network. These singular subspace changes are also reflected in their corresponding singular values. This is not unique to ELU and ReLU: the changes in singular values and subspaces are noticeably smaller in MLPs with smooth activations compared to nonsmooth (see Figures 7 to 9 in Appendix B.1). For MLPs with smooth activations, the training dynamics appear much more concentrated within a lower-dimensional subspace, implying *low-rank training dynamics emerge more prominently for MLPs with smooth activations.* Therefore, we focus on such activations.

3 Analysis on Two-Layer Nonlinear Networks

In this section, we provide our main theoretical result on low-rank training in two-layer networks.

3.1 Definitions and Assumptions

In this section, we provide a preliminary definition, and then state and discuss our assumptions. We first define quantities that measure the alignment between two subspaces.

Definition 3.1 (Principal angles between subspaces)**.** Let $\boldsymbol{U}_1, \boldsymbol{U}_2 \in \mathbb{R}^{d\times r}$ be orthonormal bases of two r-dimensional subspaces of $\mathbb{R}^d$. Then, for all $i \in [r]$, the i^{th} principal angle ψ_i between $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ is defined as such:

$$\psi_i = \arccos\left(\sigma_i\left(\boldsymbol{U}_1^\top \boldsymbol{U}_2\right)\right) \in [0, \pi/2]\,.$$

Also define $\Psi := [\psi_1 \quad \dots \quad \psi_r] \in \mathbb{R}^r$. We measure the alignment between $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ through the following metric:

$$\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2 := \sqrt{\sum_{i=1}^{r} \sin^2(\psi_i)} \in \left[0, \sqrt{r}\right].$$

Smaller angles ψ_i indicate $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ are well aligned, and also leads to smaller $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2$. Thus, smaller $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2$ indicates greater alignment between $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$.

We now provide our assumptions. First, we state our assumptions on the input data $\boldsymbol{X}$ and labels $\boldsymbol{Y}$.

Assumption 3.2 (Input data)**.** The data $\boldsymbol{X}$ is whitened, and the label dimension K satisfies $K < d/2$.

Before proceeding, we briefly discuss Assumption 3.2.

- **Whitened input data.** Many previous theoretical studies of neural networks assume whitened input data (e.g., [12, 24–26]), an assumption that provides valuable insights into practical phenomena across more generic settings. In Section 4.2, we empirically observe this phenomenon approximately holds for unwhitened $\boldsymbol{X}$.
- **Small output dimension $K \ll d$.** We consider a setting where the $K \ll d$, which is commonly studied. For instance, [27] studied neural collapse [28] in this exact setting. Classification is another example of where this setting is common, as K represents the number of classes. In our analysis, we show K governs the rank of the low-rank training dynamics.

Next, we state our assumptions on the network architecture and training.

Assumption 3.3 (Network architecture and training)**.** The network (1) contains $L = 2$ layers, i.e., $f_{\boldsymbol{\Theta}}(\boldsymbol{X}) = \boldsymbol{W}_2\phi(\boldsymbol{W}_1\boldsymbol{X})$, with $\boldsymbol{\Theta} = \{\boldsymbol{W}_2, \boldsymbol{W}_1\}$, $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$, and $\boldsymbol{W}_2 \in \mathbb{R}^{K\times m}$. Furthermore,

- the width m satisfies $m \geq d$,
- the activation function ϕ satisfies $\phi(0) = 0$, $|\phi'(x)| \leq \beta$, and $|\phi''(x)| \leq \mu$ for all $x \in \mathbb{R}$.
- the first-layer weight matrix $\boldsymbol{W}_1$ is trained using GD with step size η on the squared error loss

$$\min_{\boldsymbol{W}_1} \mathcal{L}(\boldsymbol{W}_1) = \frac{1}{2} \left\| \boldsymbol{W}_2\phi\left(\boldsymbol{W}_1\boldsymbol{X}\right) - \boldsymbol{Y} \right\|_F^2, \tag{2}$$

  while $\boldsymbol{W}_2$ is fixed during training and is full row rank.

From now on, we will use $\boldsymbol{G}_1(t)$ to denote $\nabla_{\boldsymbol{W}_1}\mathcal{L}\left(\boldsymbol{W}_1(t)\right)$. We now provide some brief remarks on Assumption 3.3.

- **Squared-error loss.** Several works on neural network analyses considered squared-error loss [29–32], even in classification [33–35]. In Section 4.2, we empirically observe similar low-rank training dynamics in classification even with the cross-entropy loss.
- **Fixed second layer.** In our problem, we aim to show the weight matrices in nonlinear networks mostly get updated in low-dimensional subspaces. Recall $K \ll d \leq m$. Since $\boldsymbol{W}_2 \in \mathbb{R}^{K\times m}$ and $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$, we focus our analysis on the GD dynamics of $\boldsymbol{W}_1$. To make the setting analytically tractable, we keep $\boldsymbol{W}_2$ fixed during training. In Section 4 and Appendix C.1, we empirically observe that low-rank training dynamics hold even when $\boldsymbol{W}_2$ (more generally $\boldsymbol{W}_L$ with $L > 2$) is trained, indicating that the phenomenon is not strictly tied to this assumption. Furthermore, fixing the second layer is standard in two-layer network analyses [20, 21, 29, 30, 36].

While we recognize that training $\boldsymbol{W}_2$ jointly with $\boldsymbol{W}_1$ aligns more closely with practical scenarios, the highly coupled and non-linear optimization dynamics of joint training present significant analytical challenges. Relaxing this assumption remains an important direction for future research. Nevertheless, our current framework isolates the core mechanisms of the network, and provides a necessary foundational step toward understanding the fully joint dynamics.

Finally, for Theorem 3.5, we make the following technical assumption.

Assumption 3.4. For all $t \geq 0$, $\|\boldsymbol{W}_2\phi(\boldsymbol{W}_1(t)\boldsymbol{X}) - \boldsymbol{Y}\|_{\max} \leq M$ for some finite constant M.

This assumption states that the maximum magnitude element in the residual is bounded by some finite constant, which we make for technical convenience. This assumption is quite mild: defining $\boldsymbol{\Delta}_2(t) := \boldsymbol{W}_2\phi\left(\boldsymbol{W}_1(t)\boldsymbol{X}\right) - \boldsymbol{Y}$, if we did have $\|\boldsymbol{\Delta}_2(t)\|_{\max} \to \infty$, then we would also have $\mathcal{L}\left(\boldsymbol{W}_1(t)\right) = \frac{1}{2}\|\boldsymbol{\Delta}_2(t)\|_F^2 \to \infty$.

3.2 Main Results

We now present our first theoretical result.

Theorem 3.5 (Hidden-Layer Low-Rank Training)**.** *Suppose Assumptions 3.2 through 3.4 hold. Let $\boldsymbol{L}_{1,1}(t)$ and $\boldsymbol{R}_{1,1}(t)$ denote top-K left and right singular subspaces of $\boldsymbol{G}_1(t)$, and we define the alignment of the singular subspace at iteration t with respect to its initialization as*

$$A(t) := \max\left\{ \left\|\sin\Psi\left(\boldsymbol{L}_{1,1}(t), \boldsymbol{L}_{1,1}(0)\right)\right\|_2, \left\|\sin\Psi\left(\boldsymbol{R}_{1,1}(t), \boldsymbol{R}_{1,1}(0)\right)\right\|_2 \right\}.$$

Suppose $\boldsymbol{W}_1(0)$ satisfies $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) = \epsilon^2\boldsymbol{I}_d$ with $\epsilon \lesssim \mathcal{O}\left(\sigma_K\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\boldsymbol{X}^\top\right)\right)$, and let the step size satisfy $\eta \lesssim \min\left\{1, \frac{1}{\Omega\left(\|\boldsymbol{W}_2\|_1 + \sigma_1^2(\boldsymbol{W}_2)\right)}\right\}$. Then, there exist orthogonal matrices $\boldsymbol{U} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ that only depend on $\boldsymbol{W}_1(0)$ and $\boldsymbol{G}_1(0)$ such that, for all $t \geq 0$, $\boldsymbol{W}_1(t)$ admits the following decomposition:

$$\boldsymbol{W}_1(t) = \boldsymbol{U}\widetilde{\boldsymbol{W}}_1(t)\boldsymbol{V}^\top = \boldsymbol{U}\begin{bmatrix} \widetilde{\boldsymbol{W}}_{1,1}(t) & \widetilde{\boldsymbol{W}}_{1,2}(t) \\ \widetilde{\boldsymbol{W}}_{1,3}(t) & \widetilde{\boldsymbol{W}}_{1,4}(t) \end{bmatrix}\boldsymbol{V}^\top,$$

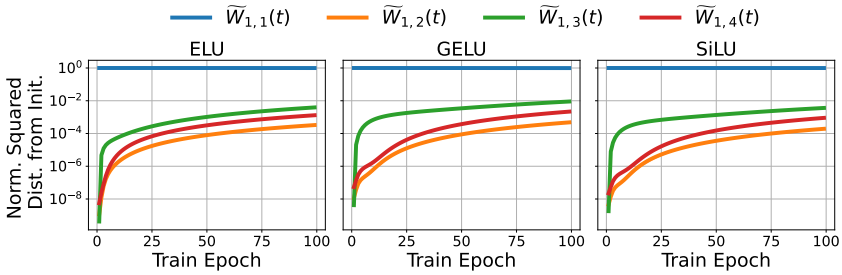


Figure 2: Under our theoretical setting, nearly all of the change in $\widetilde{\boldsymbol{W}}_1(t)$ is from $\widetilde{\boldsymbol{W}}_{1,1}(t)$.

where $\widetilde{\boldsymbol{W}}_{1,1}(t) \in \mathbb{R}^{(m-p)\times 2K}$, $\widetilde{\boldsymbol{W}}_{1,2}(t) \in \mathbb{R}^{(m-p)\times p}$, $\widetilde{\boldsymbol{W}}_{1,3}(t) \in \mathbb{R}^{p\times 2K}$, *and* $\widetilde{\boldsymbol{W}}_{1,4}(t) \in \mathbb{R}^{p\times p}$, *with*

$$\widetilde{\boldsymbol{W}}_{1,2}(0) = \mathbf{0},\ \widetilde{\boldsymbol{W}}_{1,3}(0) = \mathbf{0},\ \frac{\|\widetilde{\boldsymbol{W}}_{1,4}(0)\|_F}{\sqrt{p}} = \epsilon,$$

and, for $i \in \{2,3,4\}$,

$$\frac{\|\widetilde{\boldsymbol{W}}_{1,i}(t+1) - \widetilde{\boldsymbol{W}}_{1,i}(t)\|_F}{\sqrt{p}} \lesssim \eta \cdot \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right) \cdot \frac{A^2(t)}{p} + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)}.$$

Discussion on Theorem 3.5. For ease of exposition, suppose $m = d$. Let $\boldsymbol{U} := [\boldsymbol{U}_1 \quad \boldsymbol{U}_2]$ and $\boldsymbol{V} = [\boldsymbol{V}_1 \quad \boldsymbol{V}_2]$, where $\boldsymbol{U}_1 \in \mathbb{R}^{d\times 2K}$, $\boldsymbol{U}_2 \in \mathbb{R}^{d\times p}$, $\boldsymbol{V}_1 \in \mathbb{R}^{d\times 2K}$, and $\boldsymbol{V}_2 \in \mathbb{R}^{d\times p}$ all have orthonormal columns, and recall $p = d - 2K$. See Appendix B.3 on how to compute $\boldsymbol{U}$ and $\boldsymbol{V}$ in practice. From Theorem 3.5, we have

$$\boldsymbol{W}_1(t) = \underbrace{\boldsymbol{U}_1\widetilde{\boldsymbol{W}}_{1,1}(t)\boldsymbol{V}_1^\top}_{\text{rank } 2K} + \underbrace{\boldsymbol{U}_1\widetilde{\boldsymbol{W}}_{1,2}(t)\boldsymbol{V}_2^\top + \boldsymbol{U}_2\widetilde{\boldsymbol{W}}_{1,3}(t)\boldsymbol{V}_1^\top + \boldsymbol{U}_2\widetilde{\boldsymbol{W}}_{1,4}(t)\boldsymbol{V}_2^\top}_{\text{“perturbation terms”}}.$$

Theorem 3.5 states the perturbation term update magnitudes are bounded in each iteration, implying the remainder of the GD update occurs in $\widetilde{\boldsymbol{W}}_{1,1}(t)$. The bound depends on the gradient singular subspace alignment at iteration t vs. initialization.

Proof sketch. We now summarize our proof strategy for Theorem 3.5. Let $\boldsymbol{R}_{1,2}(t)$ and $\boldsymbol{L}_{1,1}(t)$ respectively denote the bottom $d-K$ and top-K right and left singular subspaces of $\boldsymbol{G}_1(t)$. We identify the following p-dimensional subspace where $\boldsymbol{G}_1(t)$ has a bounded component, which is analogous to the subspace identified in [12, 13]:

$$\mathcal{S}_{small} = \mathcal{R}\left(\boldsymbol{R}_{1,2}(0)\right) \cap \mathcal{R}^\perp\left(\boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right). \tag{3}$$

Let $\boldsymbol{V}_2$ be an orthonormal basis for $\mathcal{S}_{small}$ and $\boldsymbol{U}_2 = \boldsymbol{W}_1(0)\boldsymbol{V}_2/\epsilon$. We bound $\|\boldsymbol{G}_1(t)\boldsymbol{V}_2\|_F$ and $\|\boldsymbol{G}_1^\top(t)\boldsymbol{U}_2\|_F$ based on the alignment between the singular subspaces of $\boldsymbol{G}_1(t)$ and $\boldsymbol{G}_1(0)$.

Empirical validation. We trained the first layer of two layer networks with various activations under the setting of Theorem 3.5, deferring specific details to Appendix B.2. Figure 2 shows the squared distance of each $\widetilde{\boldsymbol{W}}_{1,i}(t)$ from $\widetilde{\boldsymbol{W}}_{1,i}(0)$ normalized by the squared distance between $\widetilde{\boldsymbol{W}}_1(t)$ and $\widetilde{\boldsymbol{W}}_1(0)$, i.e., $\|\widetilde{\boldsymbol{W}}_{1,i}(t) - \widetilde{\boldsymbol{W}}_{1,i}(0)\|_F^2/\|\widetilde{\boldsymbol{W}}_1(t) - \widetilde{\boldsymbol{W}}_1(0)\|_F^2$, averaged over 10 trials. Clearly, for all activations, $\widetilde{\boldsymbol{W}}_{1,1}(t)$ accounts for almost all of the change in $\widetilde{\boldsymbol{W}}_1(t)$, and thus $\boldsymbol{W}_1(t)$. This support the main message in Theorem 3.5: each GD update mostly occurs in a low-dimensional subspace.

Quantifying the bound. In our next result, we provide some quantitative intuition on the bound in Theorem 3.5. To do so, we rely on empirical observations within our theoretical setting; see Figures 11 and 12 in Appendix C.2. We formalize these observations in the following assumption, which we recognize is quite strict in broader cases.

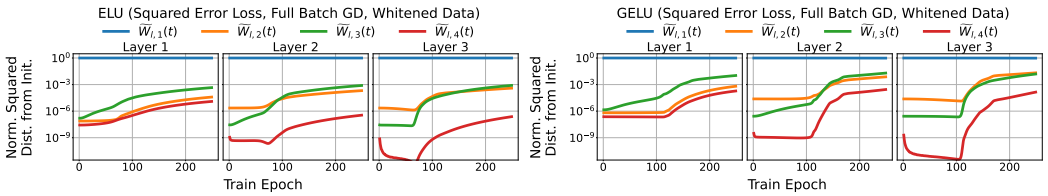


Figure 3: For every $l \in [L-1]$ in deep MLPs, almost all of the change in $\widetilde{\boldsymbol{W}}_l(t)$ is from $\widetilde{\boldsymbol{W}}_{l,1}(t)$.

Assumption 3.6. There exist constants $G_1, G_2, G_3 > 0$ such that $\boldsymbol{G}_1(t)$ satisfies the following:

$$\frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F} \leq G_1 \cdot (1-\Theta(\eta))^{\Theta(t)}, \quad \frac{\sigma_i(\boldsymbol{G}_1(t))}{\sigma_i(\boldsymbol{G}_1(0))} \leq G_2 \cdot \frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F} \text{ for all } i \leq K, \text{ and}$$
$$\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - \sigma_{K+1}\left(\boldsymbol{G}_1(t)\right) \geq G_3 \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right).$$

We now state the result below.

Corollary 3.7. *Suppose the setting of Theorem 3.5 and Assumption 3.6 hold. Then, for some constants C_1 and C_2 that depend on G_1, G_2, G_3, β, μ, ϵ, $\phi'(0)$, $\sigma_1(\boldsymbol{W}_2)$, and $\sigma_K(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top)$, we have*

$$\sqrt{\sigma_1^2(\boldsymbol{G}_1(t)) \cdot \frac{A^2(t)}{p} + \sigma_{K+1}^2(\boldsymbol{G}_1(t))} \leq \begin{cases} C_1 \cdot \epsilon & t = 0 \\ C_2 \cdot (1-\Theta(\eta))^{\Theta(t)} \cdot \left(1 - (1-\Theta(\eta))^{\Theta(t)}\right) & t > 0 \end{cases}.$$

Discussion on Corollary 3.7. In the early training stages, the $1-(1-\Theta(\eta))^{\Theta(t)}$ term indicates the singular subspaces are well-aligned with those at initialization. As training progresses, these singular subspaces become less aligned. However, from the $(1-\Theta(\eta))^{\Theta(t)}$ term, eventually the GD update itself decreases in magnitude as GD converges to a stationary point. We provide a proof of Corollary 3.7 in Appendix E.5.

Why activation smoothness matters. In Corollary 3.7, a key step in our proof is showing the bottom $d-K$ singular values of $\boldsymbol{G}_1(0)$ are at most $\Theta(\epsilon)$, which leads to the $t=0$ result. In Appendix D, we provide a result showing this does not hold for a nonsmooth ϕ. This implies when ϕ is nonsmooth, even a *single* GD update could increase the perturbation terms by a "large" amount, which we precisely observe in Figure 1.

Technical challenges compared to linear networks. We now discuss the main technical challenges of considering a general (smooth) nonlinearity ϕ compared to a linear network. Recall $\boldsymbol{\Delta}_2(t) := \boldsymbol{W}_2\phi(\boldsymbol{W}_1(t)\boldsymbol{X}) - \boldsymbol{Y}$. For general nonlinear ϕ, we have $\boldsymbol{G}_1(t) = \left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t) \odot \phi'(\boldsymbol{W}_1(t)\boldsymbol{X})\right)\boldsymbol{X}^\top$, where $\odot$ denotes the Hadamard product and ϕ' is the derivative of ϕ, which is applied element-wise. For linear networks, $\phi'(x) = 1$ for all $x \in \mathbb{R}$, meaning $\boldsymbol{G}_1(t) = \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\boldsymbol{X}^\top$. Prior work [13, 14] heavily relied on this linear form to show a specific high-dimensional subspace remains strictly orthogonal to $\boldsymbol{G}_1(t)$ for all iterations t. When ϕ is nonlinear, the Hadamard product with $\phi'(\boldsymbol{W}_1(t)\boldsymbol{X})$ breaks this exact orthogonality, preventing direct application of the proofs in [13, 14]. To overcome these challenges, our theoretical framework establishes that low-rank training more generally depends on how the singular subspaces of $\boldsymbol{G}_1(t)$ evolve throughout training, and the smoothness of ϕ allows us to bound these subspace shifts (see Lemmas E.6 to E.8 in Appendix E).

4 Empirical Observations Beyond Theory

In this section, we show low-rank training dynamics persist beyond our theoretical setting. We defer specific experimental details to Appendix B.4.

4.1 Deeper Networks

Here, we show that low-rank training GD updates emerge in deeper MLPs, where each layer l has a corresponding "small-update subspace." Since $\boldsymbol{W}_L \in \mathbb{R}^{K \times m}$, where $K \ll d \leq m$, and we are

interested in identifying if the weight matrices are mostly updated in low-dimensional subspaces, we focus on the first $L-1$ layers, where $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$ and $\boldsymbol{W}_l \in \mathbb{R}^{m\times m}$ for $l = 2, \dots, L-1$. However, we still trained $\boldsymbol{W}_L$ in all experiments.

Network architecture and training. We considered $L = 4$ layer networks with activations $\phi = \text{ELU}$ and GELU. We trained the networks on squared-error loss using full-batch GD, and initialized the first 3 layers to be ϵ-scaled (semi-)orthogonal matrices with $\epsilon = 0.1$. We trained on synthetic data as described in Appendix B.2.

Left and right factors for deeper layers. For the hidden layers in deep MLPs, we discuss how to find decompositions $\boldsymbol{W}_l(t) = \boldsymbol{U}_l \widetilde{\boldsymbol{W}}_l(t) \boldsymbol{V}_l^\top$ that are analogous to the $\boldsymbol{W}_1(t)$ decomposition in Theorem 3.5. Define $\boldsymbol{V}_{1,2} \in \mathbb{R}^{d\times p}$ as an orthonormal basis for $\mathcal{S}_{small}$ from Equation (3), and $\boldsymbol{U}_{1,2} := \boldsymbol{W}_1(0)\boldsymbol{V}_{1,2}/\epsilon \in \mathbb{R}^{m\times p}$. Again see Appendix B.3 for more details on computing $\boldsymbol{U}_{1,2}$ and $\boldsymbol{V}_{1,2}$. Inspired by [12, 13], we define corresponding $\boldsymbol{V}_{l,2}$ and $\boldsymbol{U}_{l,2}$ in $\boldsymbol{W}_l$ for $l = 2, \dots, L-1$ as follows:

$$\boldsymbol{V}_{l,2} = \boldsymbol{U}_{l-1,2} \quad \text{and} \quad \boldsymbol{U}_{l,2} = \boldsymbol{W}_l(0)\boldsymbol{V}_{l,2}/\epsilon, \tag{4}$$

assuming $\boldsymbol{W}_l(0)$ is initialized as an ϵ-scaled semi-orthogonal matrix. Equivalently,

$$\boldsymbol{V}_{l,2} = \boldsymbol{W}_{l-1}(0)\cdots\boldsymbol{W}_1(0)\boldsymbol{V}_{1,2}/\epsilon^{l-1} \quad \text{and} \quad \boldsymbol{U}_{l,2} = \boldsymbol{W}_l(0)\cdots\boldsymbol{W}_1(0)\boldsymbol{V}_{1,2}/\epsilon^{l}. \tag{5}$$

Define $\boldsymbol{V}_l := [\boldsymbol{V}_{l,1} \quad \boldsymbol{V}_{l,2}] \in \mathbb{R}^{m\times m}$ and $\boldsymbol{U}_l := [\boldsymbol{U}_{l,1} \quad \boldsymbol{U}_{l,2}] \in \mathbb{R}^{m\times m}$, where $\boldsymbol{V}_{l,1} \in \mathbb{R}^{m\times(m-p)}$ and $\boldsymbol{U}_{l,1} \in \mathbb{R}^{m\times(m-p)}$ are respectively orthogonal to $\boldsymbol{V}_{l,2}$ and $\boldsymbol{U}_{l,2}$. Finally, let

$$\widetilde{\boldsymbol{W}}_l(t) = \begin{bmatrix} \boldsymbol{U}_{l,1}^\top \boldsymbol{W}_l(t)\boldsymbol{V}_{l,1} & \boldsymbol{U}_{l,1}^\top \boldsymbol{W}_l(t)\boldsymbol{V}_{l,2} \\ \boldsymbol{U}_{l,2}^\top \boldsymbol{W}_l(t)\boldsymbol{V}_{l,1} & \boldsymbol{U}_{l,2}^\top \boldsymbol{W}_l(t)\boldsymbol{V}_{l,2}, \end{bmatrix} := \begin{bmatrix} \widetilde{\boldsymbol{W}}_{l,1}(t) & \widetilde{\boldsymbol{W}}_{l,2}(t) \\ \widetilde{\boldsymbol{W}}_{l,3}(t) & \widetilde{\boldsymbol{W}}_{l,4}(t) \end{bmatrix} \in \mathbb{R}^{m\times m}, \tag{6}$$

which implies $\boldsymbol{W}_l(t) = \boldsymbol{U}_l \widetilde{\boldsymbol{W}}_l(t) \boldsymbol{V}_l^\top$. Note $\widetilde{\boldsymbol{W}}_{l,1}(t) \in \mathbb{R}^{(m-p)\times(m-p)}$, $\widetilde{\boldsymbol{W}}_{l,2}(t) \in \mathbb{R}^{(m-p)\times p}$, $\widetilde{\boldsymbol{W}}_{l,3}(t) \in \mathbb{R}^{p\times(m-p)}$, and $\widetilde{\boldsymbol{W}}_{l,4}(t) \in \mathbb{R}^{p\times p}$.

Results. In Figure 3, we show $\|\widetilde{\boldsymbol{W}}_{l,i}(t) - \widetilde{\boldsymbol{W}}_{l,i}(0)\|_F^2 / \|\widetilde{\boldsymbol{W}}_l(t) - \widetilde{\boldsymbol{W}}_l(0)\|_F^2$ averaged over 10 trials. For all $l \in [L-1]$, $\widetilde{\boldsymbol{W}}_{l,1}(t)$ accounts for most of the change in $\widetilde{\boldsymbol{W}}_l(t)$, and thus $\boldsymbol{W}_l(t)$. This implies **the GD updates of the deeper layers are also concentrated within low-dimensional subspaces.** Furthermore, here we trained $\boldsymbol{W}_L$, which is $\boldsymbol{W}_2$ in Section 3, and still observed low-rank training dynamics in the hidden layers. This implies freezing $\boldsymbol{W}_2$ in Assumption 3.3 may not be strictly necessary for low-rank training dynamics to arise.

4.2 Loss Function, Optimizer, and Unwhitened Data

So far, we have been training MLPs using full batch GD on squared-error loss with whitened data. Here, we investigate changing the loss function and optimizer, as well as not whitening the data. We again trained $L = 4$ layer MLPs with activations $\phi = \text{ELU}$ and GELU. We generated $\boldsymbol{X}$ as described in Appendix B.2, but did not whiten $\boldsymbol{X}$. We also trained on cross-entropy loss using 1) minibatch SGD with momentum, 2) Adam, and 3) Muon [37].

Figure 4 again shows the normalized squared distance of each $\widetilde{\boldsymbol{W}}_{l,i}(t)$ from their initializations. Once again, $\widetilde{\boldsymbol{W}}_{l,1}(t)$ accounts for a majority of the change in $\widetilde{\boldsymbol{W}}_l(t)$, and thus $\boldsymbol{W}_l(t)$, meaning low-rank training approximately holds in more realistic training settings. Interestingly, this holds even for the Muon optimizer, which orthogonalizes the update matrices for $\boldsymbol{W}_l(t)$, i.e., the updates are no longer low-rank. This implies low-rank updates may not be strictly necessary for low-rank training to emerge, despite being an assumption for Corollary 3.7.

5 Low-Rank Parameterization in MLPs

From Sections 3 and 4, for $l \in [L-1]$, we find $\boldsymbol{W}_l$ is mostly updated within a subspace that depends on $\boldsymbol{G}_1(0)$ and $\boldsymbol{W}_l(0)$. From these insights, we find there exists a low-rank MLP that, when initialized

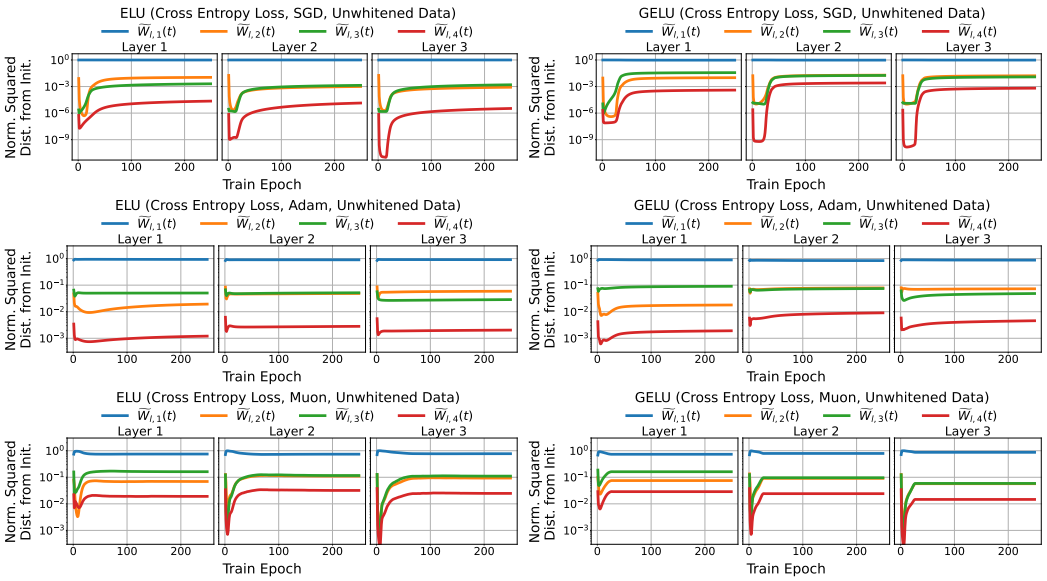


Figure 4: Training with SGD plus momentum (top), Adam (middle), or Muon (bottom) on unwhitened input data using cross-entropy loss maintains the previously observed low-rank training dynamics.

properly, achieves near-identical performance to their fully parameterized counterpart. For some width parameter r, we refer to the following parameterization as a *low-rank MLP:*

$$\boldsymbol{W}_L\phi\Big(\widetilde{\boldsymbol{U}}\widetilde{\boldsymbol{W}}_{L-1}\phi(\widetilde{\boldsymbol{W}}_{L-2}\dots\widetilde{\boldsymbol{W}}_2\phi(\widetilde{\boldsymbol{W}}_1\widetilde{\boldsymbol{V}}^\top\boldsymbol{X})\dots)\Big), \tag{7}$$

where $\boldsymbol{W}_L \in \mathbb{R}^{K\times m}$, $\widetilde{\boldsymbol{U}} \in \mathbb{R}^{m\times r}$, $\widetilde{\boldsymbol{V}} \in \mathbb{R}^{d\times r}$ and $\widetilde{\boldsymbol{W}}_l \in \mathbb{R}^{r\times r}$ for $l \in [L-1]$.

Initializing and training $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$. For ease of exposition, suppose $m = d$, so $m-p = 2K$. From Section 4, we observe each $\boldsymbol{W}_l$ is updated in a $2K$-dimensional subspace defined from (5). Thus, setting $r = 2K$, and initializing $\boldsymbol{W}_l(0)$ as ϵ-scaled orthogonal matrices for all $l \in [L-1]$, we find $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ should be initialized as follows. Recall $\mathcal{S}^{(1)}_{small}$ in (3), and denote its orthogonal complement as $\mathcal{S}^{(1)}_{big}$. We initialize $\widetilde{\boldsymbol{V}} \in \mathbb{R}^{d\times 2K}$ as an orthonormal basis of $\mathcal{S}^{(1)}_{big}$, and then initialize $\widetilde{\boldsymbol{U}}$ according to (5), i.e., $\widetilde{\boldsymbol{U}} = \boldsymbol{W}_{L-1}(0)\cdots\boldsymbol{W}_1(0)\widetilde{\boldsymbol{V}}/\epsilon^{L-1} \in \mathbb{R}^{d\times 2K}$. We call this the $\mathcal{S}_{\mathbf{big}}$ **initialization**. After initialization, we train $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ using the same learning rate as $\widetilde{\boldsymbol{W}}_l$.

5.1 Experiments

In this section, we provide empirical evidence showing that, when $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ are properly initialized, **the low-rank MLP achieves near-equivalent performance** (test loss and/or accuracy) **compared to the fully parameterized MLP**, assuming the hyperparameters remain the same when training the two types of networks. We again ran all experiments in `PyTorch` using an NVIDIA A40 GPU.

We emphasize that the goal is not to find optimal hyperparameters to maximize performance. Rather, we aim to show that, for a given MLP that is trained using some (possibly sub-optimal) training recipe, there exists a low-rank MLP that, if initialized properly, achieves similar performance throughout training.

5.1.1 Fashion MNIST using MLPs

We trained fully parameterized and low-rank MLPs on Fashion MNIST [38]. We pre-processed all images by re-scaling the pixel values to lie in $[0, 1]$, and then normalized them to have zero mean and unit variance.

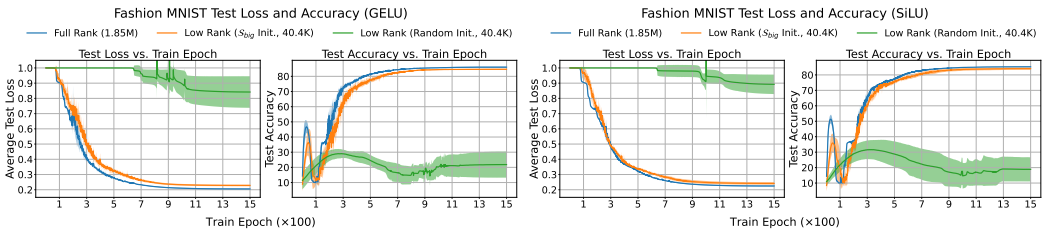


Figure 5: The properly-initialized low-rank MLP (orange) achieves nearly-identical test loss and accuracy trajectories as the fully parameterized MLP (blue). Meanwhile, the low-rank MLP with random subspace initialization (green) gets stuck at a noticeably worse local minimum.

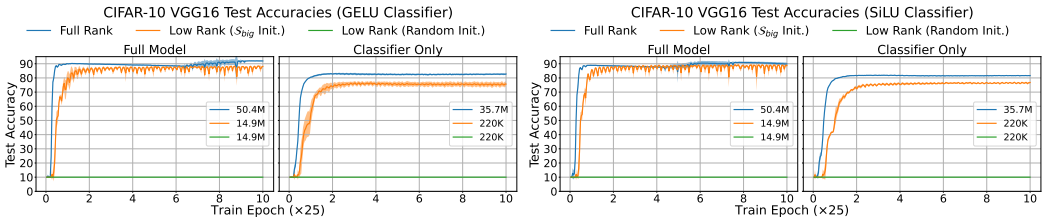


Figure 6: The properly-initialized low-rank MLP classifier head (orange) achieves similar test accuracies to the fully parameterized version (blue) in VGG-16 on CIFAR-10, while the random-subspace initialized low-rank MLP (green) is stuck in the random guessing stage.

Network architecture and training. We used $L = 4$ layer networks with GELU and SiLU activations, setting $m = d = 784$ and $r = 2K$. We initialized each $\boldsymbol{W}_l$ and $\widetilde{\boldsymbol{W}}_l$ as ϵ-scaled semi-orthogonal matrices, with $\epsilon = 0.1$. We initialized $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ in two different ways: 1) the $\mathcal{S}_{big}$ initialization scheme, and 2) as random semi-orthogonal matrices. We trained all networks on all $N = 5 \times 10^4$ training images for $T = 1500$ epochs using full-batch GD on (total) squared-error loss, which aligns with the training algorithm in our theoretical setting. Since this is a classification task, we used one-hot encoded labels. Finally, we set $\eta = 10^{-5}$ and used a cosine annealing scheduler.

Results. Figure 5 shows the test losses and accuracies for the fully parameterized and low-rank MLPs, averaged over 5 trials. The legends include each models' number of trainable parameters in parentheses. Note that we trained the networks using vanilla full-batch GD, so all networks (likely) converged to local minima. Nevertheless, keeping the training algorithm and hyperparameters fixed between the two parameterizations, the low-rank MLP with $\mathcal{S}_{big}$ initialization achieves nearly identical performance to the fully parameterized MLP. Meanwhile, the low-rank MLP with random subspace initialization gets stuck in a much worse local minimum. We provide an ablation study on the initialization of $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ in Appendix G.1.

5.1.2 CIFAR-10 using VGG-16

Here, we train a modified VGG-16 on CIFAR-10 [39]. We pre-processed the data as described in Section 5.1.1, but after re-sizing the images to $224 \times 224 \times 3$ via bilinear interpolation.

Network architecture and training. We used VGG-16 models [40] modified as such: we reduced the final average pooling layer size from 7×7 to 3×3, and changed the classifier head activation from ReLU to GELU and SiLU. Before training, we initialized the convolutional layers using ImageNet pre-trained weights, and initialized $\boldsymbol{W}_l$ and $\widetilde{\boldsymbol{W}}_l$ in the classifier head in the same manner as in Section 5.1.1, again with $r = 2K$. To initialize $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$, we used 1) an estimate of $\mathcal{S}_{big}^{(1)}$ via a mini-batch esimate of $\boldsymbol{G}_1(0)$ using 128 randomly selected images, and 2) semi-orthogonal matrices sampled uniformly at random. For all model types, we conducted two training paradigms: 1) full model training, where we fined-tuned the convolutional layers from the ImageNet pre-trained weights, in addition to training the classifier head from initialization, and 2) classifier-only training, where we kept the convolutional layers frozen at the ImageNet pre-trained weights, and only trained the classifier head from initialization. We trained all models on cross-entropy loss using SGD with

momentum for 250 epochs. We set $\eta = 5 \times 10^{-3}$ with a cosine annealing scheduler, the batch size to 128, the momentum to 0.9, and weight decay to 5×10^{-4}.

Results. Figure 6 shows the test accuracies of VGG-16 with the fully-parameterized and low-rank classifier heads, again averaged over 5 trials. The legends again include each models' number of trainable parameters in parentheses. Under full model training, VGG-16 with the low-rank MLP and $\mathcal{S}_{big}$ initialization consistently achieved very similar test accuracy to the corresponding model with the fully parameterized MLP. Meanwhile, under the same training setting, the VGG-16 with random-subspace initialized MLP did not escape the random guessing stage.

When we only trained the classifier head, there was a noticeable gap in test accuracy (about $5\% - 10\%$) between the low-rank MLP with $\mathcal{S}_{big}$ initialization and the fully parameterized version. In Appendix G.2, we observe setting the width $r = 4K$ noticeably closes this gap. Additionally, the model with the low-rank classifier head took noticeably more epochs to (nearly) match the performance of the fully parameterized version. We believe this is because we initialized $\widetilde{\boldsymbol{V}}$ and $\widetilde{\boldsymbol{U}}$ using an estimate of $\mathcal{S}_{big}^{(1)}$ via a minibatch estimate of $\boldsymbol{G}_1(0)$ in the classifier head, rather than using the full-batch $\boldsymbol{G}_1(0)$ to compute the true $\mathcal{S}_{big}^{(1)}$.

6 Conclusion

In this work, we investigated when low-rank training dynamics emerge in nonlinear networks, using MLP architectures as a case study. We showed that in MLPs with smooth activation functions, the training dynamics are highly concentrated within invariant low-dimensional subspaces. We provided theoretical insight into this phenomenon on two-layer networks trained with GD, and our experiments show the phenomenon holds well beyond our theoretical setting. From these insights, we showed there exists a low-rank MLP parameterization that, if initialized in the proper subspaces, nearly matches the performance of their high-rank counterparts in classification tasks.

6.1 Limitations and Future Work

We discuss the limitations of this work, and possible directions for future work.

Limitations. We specifically focused on MLP architectures, so it is unclear if or how our results extend to other architectures. Furthermore, our theoretical results rely on several assumptions, such as a fixed second layer in a two-layer network, full-batch gradient descent, whitened input data, orthogonal initialization, and a small label dimension K relative to the input dimension d. Corollary 3.7 also relies on specific technical assumptions regarding gradient descent that are not proved explicitly in our setting. Finally, we evaluated the low-rank parameterization in Section 5 using small-scale models; it is unclear how the parameterization and initialization scheme would scale across different and larger architectures, such as transformers [41].

Future work. We believe there are several promising avenues for future work. First, extending the study to consider other architectures, such as self-attention, is certainly an important direction. To sketch an idea, following the setting in [42, 43], consider training a simplified softmax attention model using squared-error loss:

$$\mathcal{L}(\boldsymbol{\Theta}) = \frac{1}{2}\sum_{i=1}^{N} \|\boldsymbol{y}_i - f_{\boldsymbol{\Theta}}(\boldsymbol{X}_i)\|_2^2 \quad \text{where} \quad f_{\boldsymbol{\Theta}}(\boldsymbol{X}_i) = \boldsymbol{W}_V \boldsymbol{X}_i^\top \operatorname{softmax}\left(\boldsymbol{X}_i \boldsymbol{W}_{QK} \boldsymbol{x}_{i,1}\right),$$

where $\boldsymbol{X}_i = \begin{bmatrix}\boldsymbol{x}_{i,1} & \boldsymbol{x}_{i,2} & \dots & \boldsymbol{x}_{i,T}\end{bmatrix}^\top \in \mathbb{R}^{T\times d}$ is a sequence of token embeddings, $\boldsymbol{y}_i \in \mathbb{R}^K$ is the corresponding label, $\boldsymbol{W}_{QK} \in \mathbb{R}^{d\times d}$, and $\boldsymbol{W}_V \in \mathbb{R}^{K\times d}$. Here, $f_{\boldsymbol{\Theta}}(\boldsymbol{X}_i)$ resembles an MLP with a softmax activation function. However, softmax is not applied element-wise, there is a quadratic term $\boldsymbol{X}_i \boldsymbol{W}_{QK} \boldsymbol{x}_{i,1}$, and there is an extra $\boldsymbol{X}_i^\top$ term after the activation. We believe generalizing the findings of this paper to this setting would be interesting. Furthermore, empirically evaluating the low-rank MLP parameterization and initialization scheme in Section 5 at larger scales, such as in transformer [41] and/or MLP-Mixer [44] models, is another important direction for future work.

Acknowledgements

QQ and AX acknowledge support from NSF CAREER CCF-2143904, NSF IIS-2312842, NSF IIS-2402950, NSF CCSS-2532643, and a Google Research Scholar Award. LB and CY acknowledge support from NSF CCF-2331590, NSF CAREER CCF-1845076, and an Intel early career award.

References

[1] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[2] Tao Li, Lei Tan, Zhehao Huang, Qinghua Tao, Yipeng Liu, and Xiaolin Huang. Low dimensional trajectory hypothesis is true: Dnns can be trained in tiny subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3411–3420, 2022.

[3] Zhong Zhang, Bang Liu, and Junming Shao. Fine-tuning happens in tiny subspaces: Exploring intrinsic task-specific subspaces of pre-trained language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1701–1713, 2023.

[4] Jiawei Zhao, Zhenyu Zhang, Beidi Chen, Zhangyang Wang, Anima Anandkumar, and Yuandong Tian. Galore: Memory-efficient llm training by gradient low-rank projection. In *International Conference on Machine Learning*, pages 61121–61143. PMLR, 2024.

[5] Thomas Robert, Mher Safaryan, Ionut-Vlad Modoranu, and Dan Alistarh. Ldadam: Adaptive optimization from low-dimensional gradient statistics. In *The Thirteenth International Conference on Learning Representations*, 2025.

[6] Hanqing Zhu, Zhenyu Zhang, Wenyan Cong, Xi Liu, Sem Park, Vikas Chandra, Bo Long, David Z Pan, Zhangyang Wang, and Jinwon Lee. Apollo: Sgd-like memory, adamw-level performance. *Proceedings of Machine Learning and Systems*, 7, 2025.

[7] Sahar Rajabi, Nayeema Nonta, and Sirisha Rambhatla. Subtrack++: Gradient subspace tracking for scalable llm training. In *Advances in Neural Information Processing Systems*, 2025.

[8] Laura Balzano, Tianjiao Ding, Benjamin D Haeffele, Soo Min Kwon, Qing Qu, Peng Wang, Zhangyang Wang, and Can Yaras. An overview of low-rank structures in the training and adaptation of large models. *arXiv preprint arXiv:2503.19859*, 2025.

[9] Guy Gur-Ari, Daniel A Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

[10] Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the intrinsic dimension of objective landscapes. In *International Conference on Learning Representations*, 2018.

[11] Brett W Larsen, Stanislav Fort, Nic Becker, and Surya Ganguli. How many degrees of freedom do we need to train deep networks: a loss landscape perspective. In *International Conference on Learning Representations*, 2022.

[12] Can Yaras, Peng Wang, Wei Hu, Zhihui Zhu, Laura Balzano, and Qing Qu. The law of parsimony in gradient descent for learning deep linear networks. *arXiv preprint arXiv:2306.01154*, 2023.

[13] Can Yaras, Peng Wang, Laura Balzano, and Qing Qu. Compressible dynamics in deep overparameterized low-rank learning & adaptation. In *International Conference on Machine Learning*, 2024.

[14] Soo Min Kwon, Zekai Zhang, Dogyoon Song, Laura Balzano, and Qing Qu. Efficient low-dimensional compression of overparameterized models. In *International Conference on Artificial Intelligence and Statistics*, 2024.

[15] Steffen Schotthöfer, Emanuele Zangrando, Jonas Kusch, Gianluca Ceruti, and Francesco Tudisco. Low-rank lottery tickets: finding efficient low-rank neural networks via matrix differential equations. In *Advances in Neural Information Processing Systems*, 2022.

[16] Hong-Yi Wang, Di Luo, Tomaso Poggio, Isaac L Chuang, and Liu Ziyin. A universal compression theory for lottery ticket hypothesis and neural scaling laws. In *The Fourteenth International Conference on Learning Representations*, 2026.

[17] Jimmy Ba, Murat A Erdogdu, Taiji Suzuki, Zhichao Wang, Denny Wu, and Greg Yang. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. In *Advances in Neural Information Processing Systems*, 2022.

[18] Ajay Jumar Jaiswal, Yifan Wang, Lu Yin, Shiwei Liu, Runjin Chen, Jiawei Zhao, Ananth Grama, Yuandong Tian, and Zhangyang Wang. From low rank gradient subspace stabilization to low-rank weights: Observations, theories, and applications. In *International Conference on Machine Learning*, 2025.

[19] Rishi Sonthalia, Michael Murray, and Guido Montúfar. Low rank gradients and where to find them. *arXiv preprint arXiv:2510.01303*, 2025.

[20] Spencer Frei, Gal Vardi, Peter Bartlett, Nathan Srebro, and Wei Hu. Implicit bias in leaky relu networks trained on high-dimensional data. In *International Conference on Learning Representations*, 2023.

[21] Yiwen Kou, Zixiang Chen, and Quanquan Gu. Implicit bias of gradient descent for two-layer relu and leaky relu networks on nearly-orthogonal data. In *Advances in Neural Information Processing Systems*, 2023.

[22] Nadav Timor, Gal Vardi, and Ohad Shamir. Implicit regularization towards rank minimization in relu networks. In *International Conference on Algorithmic Learning Theory*, pages 1429–1459. PMLR, 2023.

[23] Hancheng Min, Enrique Mallada, and Rene Vidal. Early neuron alignment in two-layer relu networks with small initialization. In *The Twelfth International Conference on Learning Representations*, 2024.

[24] Sanjeev Arora, Nadav Cohen, Noah Golowich, and Wei Hu. A convergence analysis of gradient descent for deep linear neural networks. In *International Conference on Learning Representations*, 2019.

[25] Lukas Braun, Clémentine Dominé, James Fitzgerald, and Andrew Saxe. Exact learning dynamics of deep linear networks with prior knowledge. In *Advances in Neural Information Processing Systems*, 2022.

[26] Clémentine Carla Juliette Dominé, Nicolas Anguita, Alexandra Maria Proca, Lukas Braun, Daniel Kunin, Pedro AM Mediano, and Andrew M Saxe. From lazy to rich: Exact learning dynamics in deep linear networks. In *The Thirteenth International Conference on Learning Representations*, 2025.

[27] George Andriopoulos, Zixuan Dong, Li Guo, Zifan Zhao, and Keith Ross. The prevalence of neural collapse in neural multivariate regression. In *Advances in Neural Information Processing Systems*, 2024.

[28] Vardan Papyan, XY Han, and David 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.

[29] Samet Oymak and Mahdi Soltanolkotabi. Overparameterized nonlinear learning: Gradient descent takes the shortest path? In *International Conference on Machine Learning*, pages 4951–4960. PMLR, 2019.

[30] Samet Oymak and Mahdi Soltanolkotabi. Toward moderate overparameterization: Global convergence guarantees for training shallow neural networks. *IEEE Journal on Selected Areas in Information Theory*, 1(1):84–105, 2020.

[31] Yajie Bao, Amarda Shehu, and Mingrui Liu. Global convergence analysis of local sgd for two-layer neural network without overparameterization. In *Advances in Neural Information Processing Systems*, 2023.

[32] Dmitry Chistikov, Matthias Englert, and Ranko Lazic. Learning a neuron by a shallow relu network: Dynamics and implicit bias for correlated inputs. In *Advances in Neural Information Processing Systems*, 2023.

[33] Like Hui and Mikhail Belkin. Evaluation of neural architectures trained with square loss vs. cross-entropy in classification tasks. In *The International Conference on Learning Representations*, 2021.

[34] Jinxin Zhou, Chong You, Xiao Li, Kangning Liu, Sheng Liu, Qing Qu, and Zhihui Zhu. Are all losses created equal: A neural collapse perspective. In *Advances in Neural Information Processing Systems*, 2022.

[35] Peng Wang, Xiao Li, Can Yaras, Zhihui Zhu, Laura Balzano, Wei Hu, and Qing Qu. Understanding deep representation learning via layerwise feature compression and discrimination. *arXiv preprint arXiv:2311.02960*, 2023.

[36] Etienne Boursier, Loucas Pillaud-Vivien, and Nicolas Flammarion. Gradient flow dynamics of shallow relu networks for square loss and orthogonal inputs. In *Advances in Neural Information Processing Systems*, 2022.

[37] Keller Jordan, Yuchen Jin, Vlado Boza, Jiacheng You, Franz Cesista, Laker Newhouse, and Jeremy Bernstein. Muon: An optimizer for hidden layers in neural networks, 2024.

[38] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

[39] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[41] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, 2017.

[42] Davoud Ataee Tarzanagh, Yingcong Li, Xuechen Zhang, and Samet Oymak. Max-margin token selection in attention mechanism. In *Advances in Neural Information Processing Systems*, 2023.

[43] Bhavya Vasudeva, Puneesh Deora, and Christos Thrampoulidis. Implicit bias and fast convergence rates for self-attention. *Transactions on Machine Learning Research*, 2025.

[44] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. In *Advances in Neural Information Processing Systems*, 2021.

[45] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2019.

[46] Minyoung Huh, Hossein Mobahi, Richard Zhang, Brian Cheung, Pulkit Agrawal, and Phillip Isola. The low-rank simplicity bias in deep networks. *Transactions on Machine Learning Research*, 2023.

[47] Suriya Gunasekar, Blake E Woodworth, Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. Implicit regularization in matrix factorization. In *Advances in neural information processing systems*, 2017.

[48] Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021.

[49] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. In *Advances in Neural Information Processing Systems*, 2019.

[50] Gal Vardi. On the implicit bias in deep-learning algorithms. *Communications of the ACM*, 66(6):86–93, 2023.

[51] Roger A Horn and Charles R Johnson. *Matrix Analysis*. Cambridge university press, 2012.

[52] Alec S. Xu, Can Yaras, Peng Wang, and Qing Qu. Linearly separable features in shallow nonlinear networks: Width scales polynomially with intrinsic data dimension. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics*, 2026.

[53] Dimitri P Bertsekas. *Nonlinear Programming*. Athena Scientific, 1995.

[54] Michael Spivak. *Calculus*. Cambridge University Press, 2006.

[55] Hermann Weyl. Inequalities between the two kinds of eigenvalues of a linear transformation. *Proceedings of the national academy of sciences*, 35(7):408–411, 1949.

[56] Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12(1):99–111, 1972.

[57] Joel A Tropp et al. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015.

A Related Work

In this section, we provide detailed discussions on related works.

Low-dimensional learning in neural networks. A recent line of empirical work has shown the GD and SGD dynamics of deep neural networks (DNNs) occur within a small subset of the full parameter space. Specifically, [2, 10, 11] observed DNNs can be trained in low-dimensional subspaces of the parameter space. Our work is similar to these works, since we observe and prove a similar phenomenon in MLP architectures. The main difference is [10, 11] showed low-dimensional training is *possible* in *random* subspaces, and [2] determined the subspace to train in via the full-parameter training trajectory of an initial training phase. Meanwhile, in our work, we show the weights are *naturally* trained within low-dimensional subspaces that depend on the weights and first-layer gradient at initialization of the fully parameterized MLP. Identifying these subspaces only requires a single forward and backward pass, and so no training is needed to find the optimization subspaces in our work.

Similarly, [45] proposed the lottery ticket hypothesis: dense, randomly initialized neural networks contain sparse sub-networks that, if trained separately, achieve similar performance to the full network in similar training time. Our work is similar in that we identify low-dimensional spaces where the training largely occurs within. The main difference is that in [45], although the sparse winning lottery tickets emerge at random initialization, identifying these lottery tickets requires an initial training and pruning phase in the full parameter space. In our work, the low-dimensional subspaces in MLPs can be completely identified at initialization of the original network using only a single forward and backward pass.

Another similar work is [15], which proposes a low-rank training algorithm that explicitly sets low-rank constraints on the weight matrices. In particular, they truncate the randomly-initialized weights using SVDs, and then update the resulting factors directly by integrating differential equations. To derive error bounds on their algorithm, [15] assumes the initialized weights have already been truncated to a low-rank approximation, and analyze the training dynamics of the already-truncated weights. In our work, we show that even when the weights in MLPs are trained in their full parameter space, the updates are largely concentrated within low-dimensional subspaces. In other words, even without explicit low-rank constraints on the weights, gradient descent undergoes implicit subspace dynamics.

Furthermore, [16] developed theoretical results showing two-layer networks with m neurons can instead be trained with a $\mathcal{O}\left(\operatorname{poly}\log(m)\right)$-sized subset of the original neurons to achieve nearly identical training dynamics. This subset can be identified at initialization of the original, fully parameterized model via a moment-matching algorithm. Our work is similar in that we also identify compressible dynamics in nonlinear networks, but instead within low-dimensional subspaces of the original parameter space. Similar to [16], these subspaces can be identified at initialization of the original network.

Additionally, [46] empirically studied low-rank simplicity biases in neural networks. They primarily studied the rank of the *data representations* before and after network training, observing that deeper networks typically result in lower-rank penultimate layer embeddings. While [46] briefly studied low-rank structure in the weight matrices, they focused on how the weights' *singular values* evolve throughout training. In our work, we analyzed the training dynamics of the weight matrices in MLP models, showing that these weights mostly get trained in invariant subspaces.

Finally, there is a rich line of literature on the emergence of low-rank structure in matrix factorization [47, 48] and deep linear networks [12–14, 24, 49]; see [50] for a survey. As mentioned in Section 1, among these works, [12–14] are most closely related to ours. They theoretically proved that when deep linear networks are trained via GD, each weight matrix is updated in an unchanging low-dimensional subspace that is determined at initialization. This subspace is also dependent on the weights and the first-layer gradient at initialization. [12] proved this in a setting with whitened input data, while [13, 14] focused on deep matrix factorization (i.e., no input data). This work is largely inspired by the findings of [12], as we extend the investigation to nonlinear MLP architectures.

Low rank gradients in neural networks. Another line of work studied the emergence of low-rank gradients in deep neural networks. In particular, [9] empirically observed the gradients become

Table 1: Initialization and per-iteration FLOP comparison. T is the number of dense training iterations before LTH pruning.

Model	Trainable params	Init. FLOPs	Per-iter. FLOPs
Dense	$1.2M$	N/A	1.59×10^9
Low-rank MLP (this paper, width $= 20$)	$40K$	1.99×10^{10}	5.35×10^7
LTH (sparsity $= 0.968$)	$40K$	$T \times 1.59 \times 10^9$	5.13×10^7
DLRT (rank $= 10$)	$39.4K$	2.12×10^{10}	9.92×10^7

well-aligned with the corresponding Hessian's top eigenspace, which remains approximately constant throughout long training periods. Furthermore, [4, 17–19] theoretically analyzed the emergence of low-rank gradients in nonlinear networks. For instance, [17] showed under a student-teacher model with Gaussian input data and scalar outputs, the gradient of the first layer in a two-layer nonlinear network is approximately rank-1. Likewise, under the same network assumptions but with more relaxed data assumptions, [19] showed the gradient is approximately rank two. In a related direction, [4, 18] studied the low-rank property of the gradients of *reversible* neural networks. Specifically, [4] provided upper bounds on the gradient stable ranks in reversible networks, while [18] showed the gradients asymptotically align with the top eigenspace of their corresponding Hessians, in line with the observations of [9]. Our work is related to the emergence of low-rank gradients in the following way. In Theorem 3.5, we show that the upper bound on the change in the perturbation terms of $\widetilde{\boldsymbol{W}}_1(t)$ depends on the gradient's $(K+1)^{th}$ singular value. The low-rank GD dynamics in $\boldsymbol{W}_1(t)$ relies on the gradient being approximately rank-K.

Implicit bias in two-layer networks. Finally, several other works studied the implicit bias of gradient flow (GF) and GD towards low-rank weights in two-layer nonlinear networks. For example, [20, 21] showed when two-layer ReLU and Leaky-ReLU networks are trained using GF or GD, the first-layer weights have a bounded stable rank at convergence. [23] showed a similar result in ReLU networks for orthogonally separable data. Our analysis on two-layer networks differs from these works, since we show *each GD update* of the first-layer weights mostly occurs in a low-dimensional subspace. Meanwhile, these works focus on the low-rankness of the first-layer weights *at convergence.*

A.1 FLOP Comparison Between Low-Dimensional Training Approaches

In this section, we provide a FLOP comparison between the low-rank MLP in Section 5, Lottery Ticket Hypothesis (LTH) [45], and Dynamic Low-Rank Training (DLRT) [15]. Specifically, we compute the number of FLOPs to initialize each method, as well as each optimization step. For the original dense model, we use a $L = 3$ layer MLP with $m = d = 784$, which contains $1.2M$ parameters. We also set the batch size to 256. We set the low-rank MLP width to $r = 20$, the LTH sparsity level to 0.968, and keep the DLRT rank constant at 10. The low-rank MLP and LTH models contain $40K$ parameters, while DLRT contains $39.4K$ parameters. To measure the per-iteration FLOPs for low-rank MLP and LTH, we used the number of FLOPs in a single SGD step.

Table 1 shows the resulting initialization and per-step FLOPs. The low-rank MLP in Section 5 requires fewer FLOPs to initialize than DLRT, and requires fewer than LTH when LTH requires at least 13 full-model iterations. The low-rank MLP also requires fewer per-iteration FLOPs than DLRT; this is because DLRT requires one-step integrators and QR decompositions in each training step, while the low-rank MLP can be trained with standard SGD steps. The theoretical best per-iteration FLOPs in LTH is lower than that of the low-rank MLP. However, it is difficult to achieve this theoretical best LTH FLOPs in practice: LTH introduces unstructured sparsity in the weights, which modern GPUs cannot fully exploit to maximize practical FLOP reduction. Meanwhile, low-rank matrix products in the low-rank MLP remain GPU friendly, so achieving practical FLOP reduction is more feasible.

B Additional Simulation Details and Results

In this section, we provide additional experimental details and/or results for Section 2.1 (Figure 1), Section 3.2 (Figure 2), and Section 4 (Figures 3 and 4).

B.1 Additional Details and Results for Figure 1

In this section, we provide experimental details and additional results for the experiments introduced in Section 2.1.

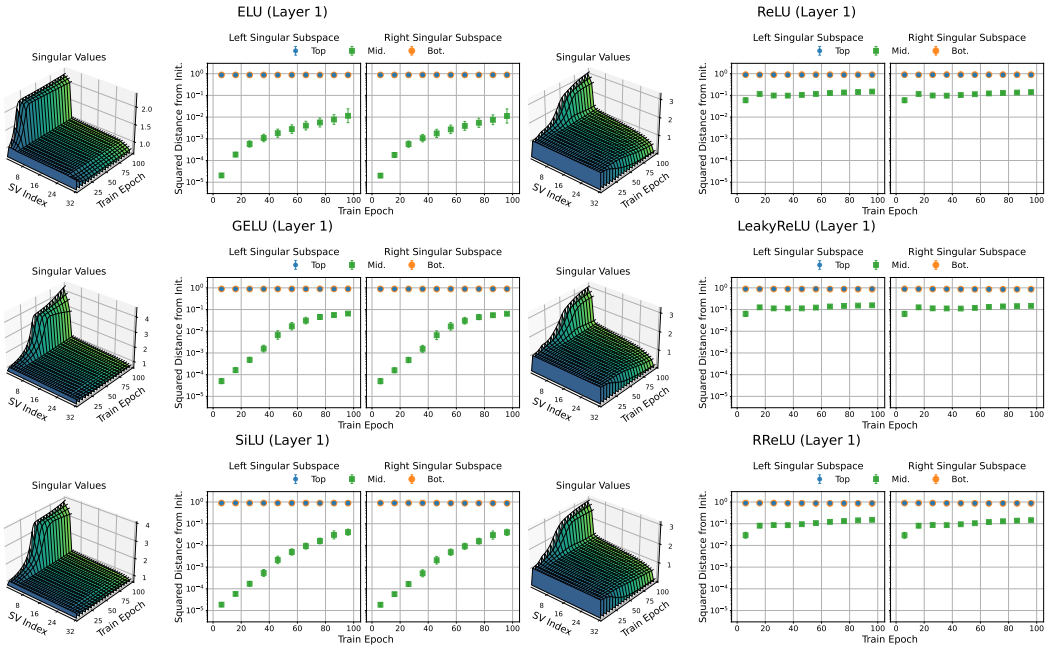


Figure 7: For networks with smooth activation functions, the middle singular subspace of the first layer weights evolves noticeably slower than in networks with non-smooth activation functions.

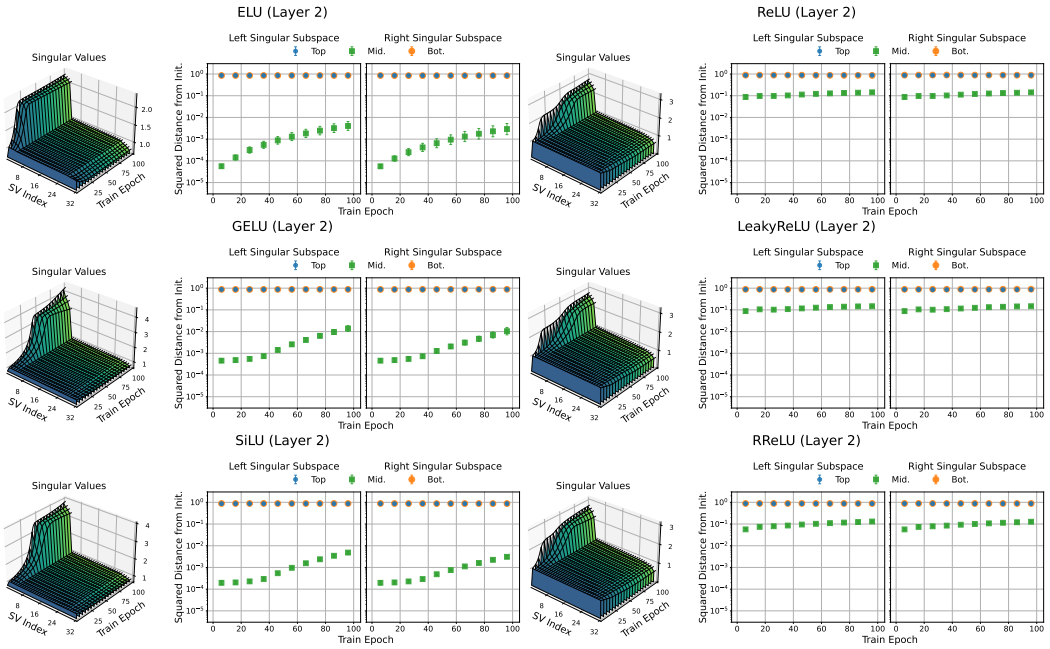


Figure 8: The change in the middle singular subspaces of the second layer weights in networks with smooth vs. nonsmooth activation functions mimic those in the first layer weights (Figure 7).

Data generation. To generate the data, we set $d = 32$, $K = 4$, and $N = 2000$, so $n = N/K = 500$. We generated $\boldsymbol{X}$ from a Gaussian mixture distribution as follows: first, we sampled K means

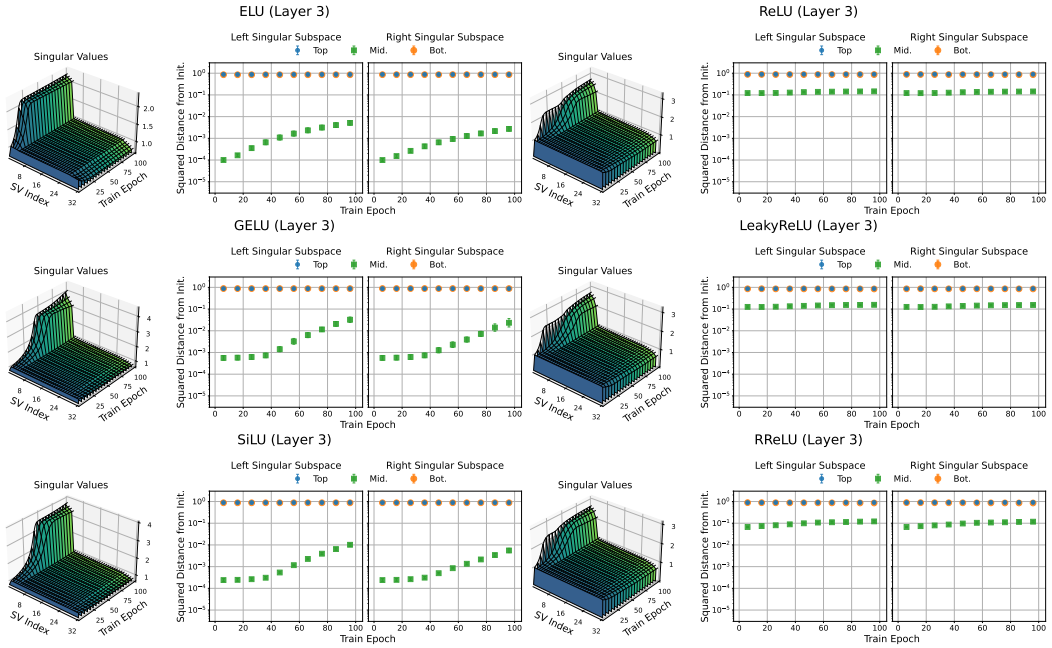


Figure 9: The change in the middle singular subspaces of the third layer weights in networks with smooth vs. nonsmooth activation functions mimic those in the first and second layer weights (Figures 7 and 8).

$\boldsymbol{\mu}_k \sim \mathcal{N}(\mathbf{0}_d, \boldsymbol{I}_d)$. Next, for each $k \in [K]$, we generated $n = 500$ samples in the k^{th} class via $\boldsymbol{x}_{k,i} \sim \mathcal{N}(\boldsymbol{\mu}_k, \sigma^2 \boldsymbol{I}_d)$ with $\sigma^2 = 3$. Then, we set $\boldsymbol{X} = [\boldsymbol{x}_{1,1} \quad \dots \quad \boldsymbol{x}_{1,n} \quad \dots \quad \boldsymbol{x}_{K,1} \quad \dots \quad \boldsymbol{x}_{K,n}]$. Finally, we pre-processed $\boldsymbol{X}$ by whitening, e.g., $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$. We generated the label matrix $\boldsymbol{Y}$ via $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \mathbf{1}_n$.

Network architectures and training. We considered six different $L = 4$ layer MLPs: three with smooth activation functions ELU, GELU, SiLU, and three with nonsmooth activations ReLU, Leaky-ReLU, and a randomized Leaky-ReLU, called RReLU. For ELU, we set the `PyTorch` parameter $\alpha = 1$, while for Leaky-ReLU, we set the `PyTorch` slope parameter $\alpha = 0.01$. In all networks, we set $m = d = 32$, and initialized all the weight matrices as ϵ-scaled orthogonal matrices, with $\epsilon = 1$.

Let $\boldsymbol{W}_l(t) = [\boldsymbol{A}_{l,1}(t) \quad \boldsymbol{A}_{l,2}(t) \quad \boldsymbol{A}_{l,3}(t)] \begin{bmatrix} \boldsymbol{S}_{l,1}(t) & & \\ & \boldsymbol{S}_{l,2}(t) & \\ & & \boldsymbol{S}_{l,3}(t) \end{bmatrix} [\boldsymbol{B}_{l,1}(t) \quad \boldsymbol{B}_{l,2}(t) \quad \boldsymbol{B}_{l,3}(t)]^\top$ be an SVD of $\boldsymbol{W}_l(t)$, where $\boldsymbol{A}_{l,1}(t)\boldsymbol{S}_{l,1}(t)\boldsymbol{B}_{l,1}^\top(t)$, $\boldsymbol{A}_{l,2}(t)\boldsymbol{S}_{l,2}(t)\boldsymbol{B}_{l,2}^\top(t)$, and $\boldsymbol{A}_{l,3}(t)\boldsymbol{S}_{l,3}(t)\boldsymbol{B}_{l,3}^\top(t)$ respectively denote the top-K, middle $d - 2K$, and bottom K SVD components. We tracked the change in top and bottom-K left singular subspaces via

$$\|\sin\Psi(\boldsymbol{A}_{l,1}(t), \boldsymbol{A}_{l,1}(0))\|_2^2 \quad \text{and} \quad \|\sin\Psi(\boldsymbol{A}_{l,3}(t), \boldsymbol{A}_{l,3}(0))\|_2^2,$$

and similarly for the top and bottom-K right singular subspaces, where $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2$ is defined in Definition 3.1 for $\boldsymbol{U}_1, \boldsymbol{U}_2$ with orthonormal columns. For the middle $d - 2K$ singular subspaces, we computed

$$\|\sin\Psi(\boldsymbol{A}_{l,2}(t), \boldsymbol{U}_{l,2})\|_F^2 \quad \text{and} \quad \|\sin\Psi(\boldsymbol{B}_{l,2}(t), \boldsymbol{V}_{l,2})\|_F^2,$$

where $\boldsymbol{U}_{l,2}$ and $\boldsymbol{V}_{l,2}$ are defined in Equation (5).

Additional results. Figures 7 to 9 shows the same results as in Figure 1, but with all nonlinear activations and layers. The behavior in the MLPs with smooth activations is similar to that of the ELU network in Figure 1, while the MLPs with nonsmooth activations behave similarly to that of the ReLU network in Figure 1. This supports our conjecture that smooth activations encourage lower-rank training dynamics in MLPs.

B.2 Experimental Details for Figure 2

Here, we provide additional details on the experimental setup to generate Figure 2. All experiments were run in `PyTorch` using an NVIDIA A40 GPU.

Data generation. We set $d = 64$, $N = 1000$, and $K = 4$. Under these parameters, we generated the data as described in Appendix B.1.

Network architecture and training. We trained $\boldsymbol{W}_1$ only in a two-layer neural network $f_{\boldsymbol{W}_1}(\boldsymbol{X}) = \boldsymbol{W}_2\phi(\boldsymbol{W}_1\boldsymbol{X})$ with $m = 72$ on squared error loss (2) using full-batch GD with $\eta = 10^{-2}$. We considered $\phi = \text{ELU}, \text{GELU}$, and SiLU, which are all smooth. We initialized $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$ to be an ϵ-scaled semi-orthogonal matrix sampled uniformly at random, with $\epsilon = 10^{-2}$, and then set $\boldsymbol{W}_2$ to have frozen iid uniform entries between -1 and 1.

B.3 Computing Left and Right Weight Factors

In this section, we provide additional details on how to compute the $\boldsymbol{U} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ factors from Theorem 3.5. Recall $p := d - 2K$, as well as $\mathcal{S}_{small}$ from Equation (3):

$$\mathcal{S}_{small} := \mathcal{R}\left(\boldsymbol{R}_{1,2}(0)\right) \cap \mathcal{R}^{\perp}\left(\boldsymbol{W}_1^{\top}(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right).$$

Assuming $\boldsymbol{W}_1(0)$ is initialized as an ϵ-scaled semi-orthogonal matrix, $\boldsymbol{U}$ and $\boldsymbol{V}$ factors in Theorem 3.5 can be computed as follows:

1. Compute an orthonormal basis of $\mathcal{S}_{small}$, denoted as $\boldsymbol{V}_2 \in \mathbb{R}^{d\times p}$, via Algorithm 1.
2. Take $\boldsymbol{Q}$ factor of QR $(\boldsymbol{V}_2)$ to obtain $\boldsymbol{V}_1$, and define $\boldsymbol{V} := [\boldsymbol{V}_1 \quad \boldsymbol{V}_2]$
3. Set $\boldsymbol{U} = \boldsymbol{W}_1(0)\boldsymbol{V}/\epsilon$, or equivalently $[\boldsymbol{U}_1 \quad \boldsymbol{U}_2] = [\boldsymbol{W}_1(0)\boldsymbol{V}_1/\epsilon \quad \boldsymbol{W}_1(0)\boldsymbol{V}_2/\epsilon]$.

Overall, this procedure requires 1 SVD, and at most 3 QR decompositions. We emphasize these computations **only need to be done once:** after a single forward-backward pass of the original model.

Algorithm 1 Computing orthonormal basis of $\mathcal{S}_{small}$

1. $\boldsymbol{L}_1(0), \boldsymbol{R}_1(0) \leftarrow \text{SVD}\left(\boldsymbol{G}_1(0)\right)$
2. By definition of SVD, take $\boldsymbol{R}_{1,2}(0)$ as orthonormal basis for $\mathcal{R}\left(\boldsymbol{R}_{1,2}(0)\right)$.
3. Do $\boldsymbol{W}_1^{\top}(0)\boldsymbol{L}_{1,2}(0)/\epsilon$ if $\boldsymbol{W}_1(0)$ is square, otherwise take last $d - K$ columns of $\boldsymbol{Q}$ factor of QR $\left(\boldsymbol{W}_1^{\top}(0)\boldsymbol{L}_{1,1}(0)\right)$, to obtain orthonormal basis for $\mathcal{R}^{\perp}\left(\boldsymbol{W}_1^{\top}(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right)$. Denote this basis as $\widetilde{\boldsymbol{L}}_{1,1}(0)$.
4. Take $\boldsymbol{Q}$ factor of QR $\left(\left[\boldsymbol{R}_{1,2}(0) \quad \widetilde{\boldsymbol{L}}_{1,1}(0)\right]\right)$ to obtain orthonormal basis for $\mathcal{S}_{small}$.

B.4 Experimental Details for Section 4

In this section, we provide additional experimental details for Section 4. We ran all experiments in `PyTorch` using an NVIDIA A40 GPU.

Data generation. In Sections 4.1 and 4.2 (Figures 3 and 4 respectively), we adhered to the exact same data generation process as in Appendix B.1, but we skipped the whitening pre-processing step on $\boldsymbol{X}$ in Section 4.2 (Figure 4).

Network architecture and training. We considered $L = 4$ layer networks of width $m = 72$ with activations $\phi = \text{ELU}$ and GELU. We initialized the first 3 layers to be ϵ-scaled (semi-)orthogonal matrices with $\epsilon = 0.1$, and the last layer $\boldsymbol{W}_L$ with iid uniform entries between -1 and 1.

In Section 4.1 (Figure 3), we trained the network on squared-error loss using full-batch GD for 250 epochs. For the ELU network, we set $\eta = 10^{-3}$, while for the GELU network, we set $\eta = 5 \times 10^{-3}$. Meanwhile, in Section 4.2 (Figure 4), we trained the networks using 1) SGD with momentum and

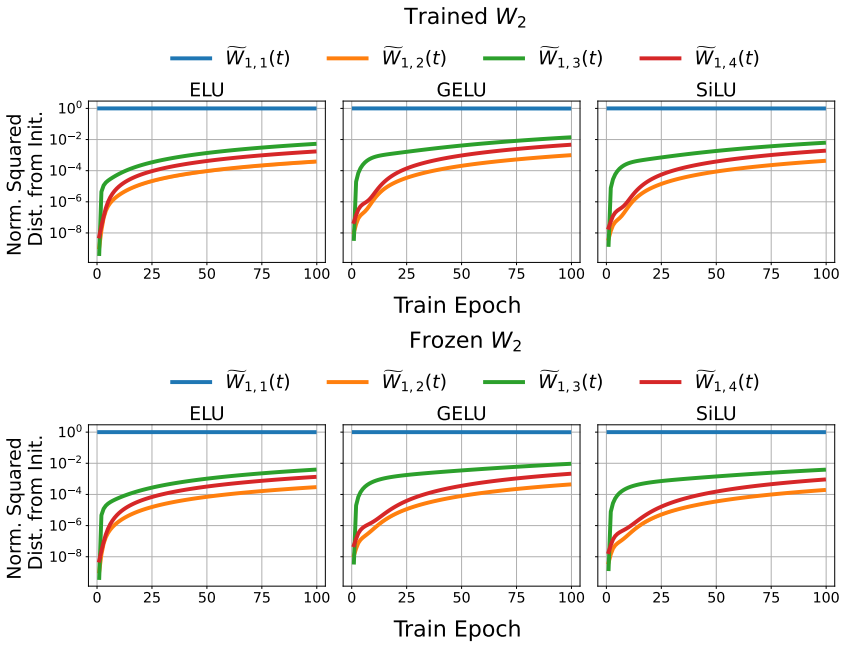


Figure 10: **Top:** even when $\boldsymbol{W}_2$ is trained, nearly all of the change in $\widetilde{\boldsymbol{W}}_1(t)$ is from $\widetilde{\boldsymbol{W}}_{1,1}(t)$. **Bottom:** repeat of Figure 2 as reference.

2) Adam. For SGD, we set the momentum to 0.9, while for Adam, we used the default `PyTorch` parameters. For both optimizers, we set the batch size to 32, $\eta = 10^{-4}$ for the ELU network, and $\eta = 5 \times 10^{-4}$ for GELU.

C Empirical Justifications for Assumptions

In this section, we provide some empirical justification for fixed $\boldsymbol{W}_2$ in Assumption 3.3, as well as the gradient norm and singular values in Assumption 3.6. In all experiments, we adopt the exact same data generation process, network architecture, and training method as in Figure 2.

C.1 Fixed vs. Trained Second Layer

Here, we show training $\boldsymbol{W}_2$ within our theoretical setting still leads to low-rank training dynamics in $\boldsymbol{W}_1$, implying a frozen $\boldsymbol{W}_2$ is not strictly necessary for low-rank training to arise. Specifically, Figure 10 (top) shows training $\boldsymbol{W}_2$ leads to nearly identical results as Figure 2 (also included in Figure 10 (bottom) as reference), where $\boldsymbol{W}_2$ is frozen. In other words, even when $\boldsymbol{W}_2$ is trained, a large majority of the change in $\boldsymbol{W}_1(t)$ is due to $\widetilde{\boldsymbol{W}}_{1,1}(t)$. This implies the overall low-rank training phenomenon is not strictly because $\boldsymbol{W}_2$ is fixed in our theory. Rather, fixing $\boldsymbol{W}_2$ leads to an analytically tractable setting to provide some theoretical characterization of low-rank training dynamics in nonlinear networks.

C.2 Gradient Norm and Singular Value Decay

In this section, we provide empirical justifications for Assumption 3.6. We trained all models for 250 epochs instead of 100.

Gradient norm decay. In Figure 11, we show $\|\boldsymbol{G}_1(t)\|_F$ decays monotonically with respect to $\|\boldsymbol{G}_1(0)\|_F$, and that the top-$K$ singular values of $\boldsymbol{G}_1(t)$ decay at a similar rate. Specifically, we plot $\frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F}$, $\max_{i\in[K]} \frac{\sigma_i(\boldsymbol{G}_1(t))}{\sigma_i(\boldsymbol{G}_1(0))}$, and $G_2 \cdot \frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F}$, where $G_2 = 2$. The gradient norm clearly monotonically decays throughout training, despite the objective being non-convex. This supports the

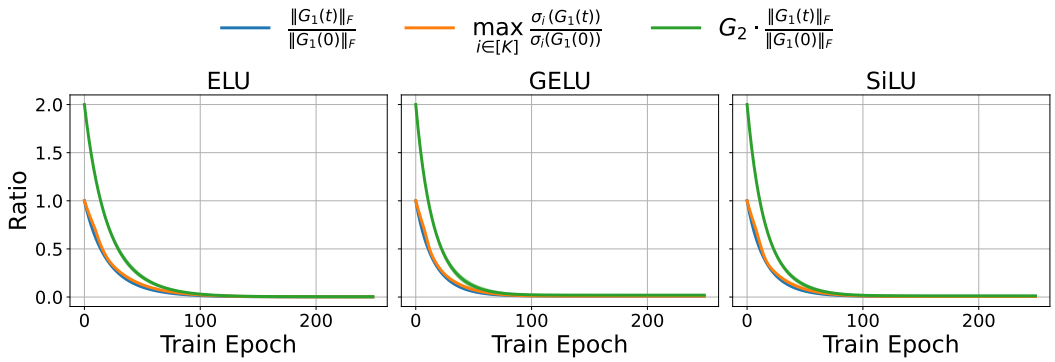


Figure 11: The gradient norm and top-K singular values appear to monotonically decay from their initial values, despite the objective being non-convex.

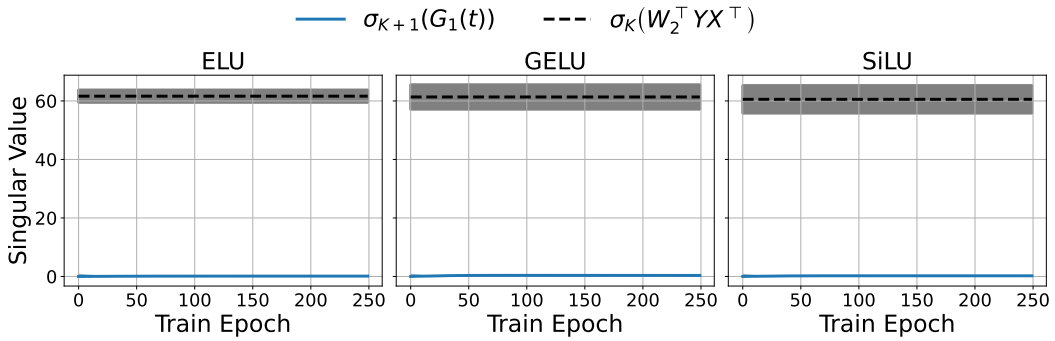


Figure 12: Throughout training, $\sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)$ remains significantly smaller than $\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)$.

assumption that $\|\boldsymbol{G}_1(t)\|_F \leq G_1 \cdot (1-\Theta(\eta))^{\Theta(t)} \cdot \|\boldsymbol{G}_1(0)\|_F$. Additionally, all of the top-K singular values of $\boldsymbol{G}_1(t)$ appear to decay at a similar rate, as Figure 11 shows $\frac{\sigma_i(\boldsymbol{G}_1(t))}{\sigma_i(\boldsymbol{G}_1(0))} \leq G_2 \cdot \frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F}$ for $i \in [K]$, which supports that assumption as well.

Gradient singular values. Finally, we show the tail singular values of $\boldsymbol{G}_1(t)$ remain much smaller than $\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)$. Figure 12 plots $\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)$ vs. $\sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)$ for different ϵ, all averaged over 10 trials. Clearly, $\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)$ is much larger than $\sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)$, justifying our assumption that $\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - \sigma_{K+1}\left(\boldsymbol{G}_1(t)\right) \geq G_3 \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)$. For all 3 activations, $\sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)$ remains very small for all GD iterations.

D Why Activation Smoothness Matters: Some Intuition

From Figure 1, we observed that compared to MLPs with smooth activation functions, the training dynamics in MLPs with non-smooth activations are noticeably less prominent. In this section, we provide some brief intuition for why this is the case. In particular, recall a step in our proof sketch of Theorem 3.5: when we assume ϕ is smooth, then $\sigma_i\left(\boldsymbol{G}_1(0)\right) = \Theta(\epsilon)$ for all $i \geq K+1$. Our next result shows that in a simplified setting, this step does not hold for a nonsmooth ϕ. Specifically, we show (most of) these tail singular values of $\boldsymbol{G}_1(0)$ are *not* on the same order as ϵ. Although the theoretical setting is highly simplified, we empirically observe similar conclusions hold in broader settings; see Section D.1. The proof of Proposition D.1 is provided in Appendix F.

Proposition D.1. *Suppose* $d = N$, $\boldsymbol{W}_1(0)_{ij} \overset{iid}{\sim} \mathcal{N}(0, \frac{\epsilon^2}{m})$, *and* $\phi = \mathrm{ReLU}$. *Then, for any* $\delta \in (0,1)$, *with probability at least* $1-\delta$ *w.r.t. the randomness in* $\boldsymbol{W}_1(0)$,

$$\sigma_{d-K}\left(\boldsymbol{G}_1(0)\right) \geq \sqrt{\frac{\lambda_{\min}\left(\boldsymbol{V}^\top \boldsymbol{D}\boldsymbol{V}\right)}{4} - \left(\frac{R'}{6}\cdot \log\left(\frac{2(d-K)}{\delta}\right) + \sqrt{2\cdot\log\left(\frac{2(d-K)}{\delta}\right)\cdot\frac{R'\cdot\lambda_{\max}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right)}{16}}\right)},$$

where $\boldsymbol{V} \in \mathbb{R}^{d\times(d-K)}$ *is an orthonormal basis for* $\mathcal{N}\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\right)$, $\boldsymbol{D} := \mathrm{diag}\left(\left\|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{:,1}\right\|_2^2, \ldots, \left\|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{:,N}\right\|_2^2\right)$, *and* $R' := \max_{i\in[m]} \|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{i,:}\|_2^2$.

Intuition behind Proposition D.1. Below, we provide some intuition behind Proposition D.1. First, note for small ϵ, we have $\boldsymbol{\Delta}_2(0) \approx -\boldsymbol{Y}$, and so $\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0) \approx -\boldsymbol{W}_2^\top\boldsymbol{Y} \in \mathbb{R}^{m\times N}$. If we consider a classification task with balanced classes and one-hot encoded labels, we have $\boldsymbol{Y} = \boldsymbol{I}_K \otimes \boldsymbol{1}_n^\top$, where $n = N/K$. Thus,

$$\boldsymbol{W}_2^\top\boldsymbol{Y} = \begin{bmatrix} (\boldsymbol{W}_2)_{1,1}\boldsymbol{1}_n^\top & \ldots & (\boldsymbol{W}_2)_{K,1}\boldsymbol{1}_n^\top \\ & \ddots & \\ (\boldsymbol{W}_2)_{1,m}\boldsymbol{1}_n^\top & \ldots & (\boldsymbol{W}_2)_{K,m}\boldsymbol{1}_n^\top \end{bmatrix},$$

i.e., the columns of $\boldsymbol{W}_2^\top\boldsymbol{Y}$ are (repeated) rows of $\boldsymbol{W}_2$ which are of size m, and each row of $\boldsymbol{W}_2^\top\boldsymbol{Y}$ contains $n = N/K = d/K$ copies of the corresponding column entries of $\boldsymbol{W}_2$, which are each of size K. Next, if $\boldsymbol{W}_2$ contains, e.g., iid sub-Gaussian entries with zero mean and unit variance, then $\left\|-\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\right)_{i,:}\right\|_2^2 \approx d$ and $\left\|-\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\right)_{:,j}\right\|_2^2 \approx m$, and so $\boldsymbol{D} \approx m\boldsymbol{I}_N \implies \boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V} \approx m\boldsymbol{I}_{d-K} \implies \lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right) \approx \lambda_{\max}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right) \approx m$, and $R' \approx d$. For a fixed failure probability $\delta \in (0,1)$, we have

$$\sigma_{d-K}\left(\boldsymbol{G}_1(0)\right) \gtrsim \sqrt{\Theta(m) - \Theta(d\log d) - \Theta\left(\sqrt{md\log d}\right)}.$$

Then, if $m \gtrsim \Omega\left(d\log d\right)$, this lower bound is non-vacuous and not of the same order as the initialization scale ϵ. Therefore, even after a single GD step, the component of $\boldsymbol{W}_1(t)$ outside of $\mathcal{S}_{small}$ in Equation (3) can undergo a "large" change, which is reflected in Figure 1 and Figures 7 to 9.

New assumptions on $\boldsymbol{X}$ and $\boldsymbol{W}_1(0)$. In Proposition D.1, we assume $d = N$, i.e., $\boldsymbol{X}$ is an exactly orthogonal matrix, and $\left(\boldsymbol{W}_1(0)\right)_{ij} \sim \mathcal{N}\left(0, \frac{\epsilon^2}{m}\right)$. This is mostly for ease of analysis, specifically to introduce iid random variables in the mask $\phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)$. Although we recognize $d = N$ is quite restrictive, we empirically show the broader conclusions hold in more general settings, i.e., $d < N$. Next, since independent Gaussian vectors are approximately orthogonal in high dimensions, $\boldsymbol{W}_1(0)$ in Proposition D.1 satisfies $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) \approx \epsilon^2\boldsymbol{I}_d$, which is the assumption we make on $\boldsymbol{W}_1(0)$ in Theorem 3.5.

D.1 Verifying Proposition D.1

Here, we provide empirical evidence supporting Proposition D.1. In particular, we show that when $\phi = \mathrm{ReLU}$, the tail singular values of $\boldsymbol{G}_1(0)$ are independent of the (small) initialization scale ϵ. Here, all experiments were done in `numpy` on a Macbook Air with an M3 chip. We adhered to the same data generation process and network initialization schemes as in Figure 2, but with $\epsilon \in \{10^{-3}, 10^{-2}, 10^{-1}\}$. We note this setting violates the assumptions on $\boldsymbol{X}$ and $\boldsymbol{W}_1(0)$ in Proposition D.1. We computed the singular values of $\boldsymbol{G}_1(0)$ for both $\phi = \mathrm{GELU}$ and $\phi = \mathrm{ReLU}$ over 10 trials.

Results. All d singular values of $\boldsymbol{G}_1(0)$ are shown in Figure 13. Clearly, when $\phi = \mathrm{GELU}$, the tail singular values of $\boldsymbol{G}_1(0)$ scale linearly with ϵ. Meanwhile, when $\phi = \mathrm{ReLU}$, the bottom $d-K$ singular values appear to be on a similar order as the top-K and *independent* of ϵ. This supports our broader conclusions from Proposition D.1, despite the specific theoretical assumptions in Proposition D.1 being violated here.

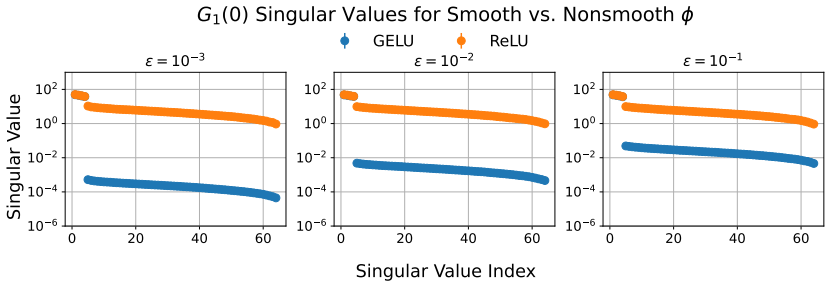


Figure 13: The bottom $d - K$ singular values of $\boldsymbol{G}_1(0)$ scale with ϵ when ϕ is smooth (GELU), but do not change with ϵ when ϕ is nonsmooth (ReLU). The top-K singular values of $\boldsymbol{G}_1(0)$ overlap for both smooth and nonsmooth ϕ.

E Proofs for Section 3

In this section, we first re-state our assumptions in Appendix E.1, and derive the analytical form of the loss gradient with respect to (w.r.t.) $\boldsymbol{W}_1$ in Appendix E.2. We then provide supporting lemmas for Theorem 3.5 with their proofs in Appendix E.3, and the proof of Theorem 3.5 in Section E.4. Finally, we provide the proof of Corollary 3.7 in Appendix E.5.

E.1 Notation and Assumptions

Here, we re-state our notation and assumptions for Theorem 3.5 for convenience, and introduce some new notation.

Notation. We use unbolded letters x, X for scalars, bold lower case letters $\boldsymbol{x}$ for vectors, and bold capital letters $\boldsymbol{X}$ for matrices. For some $N \in \mathbb{N}$, $[N]$ denotes the set $\{1, 2, \ldots, N\}$. For scalars a, b, we say $a \lesssim \mathcal{O}(b)$ if there exists a constant C s.t. $a \leq C \cdot b$, $a \gtrsim \Omega(b)$ if $a \geq C \cdot b$, and $a = \Theta(b)$ if $a = C \cdot b$. We use $\sigma_i(\boldsymbol{X})$, $\|\boldsymbol{X}\|_F$, $\|\boldsymbol{X}\|_1$, $\|\boldsymbol{X}\|_\infty$, and $\|\boldsymbol{X}\|_{\max}$ to respectively denote the i^{th} singular value, Frobenius norm, matrix-1 norm, matrix-∞ norm, and maximum magnitude element, and use $\boldsymbol{X}_{i,:}$ and $\boldsymbol{X}_{:,j}$ to respectively denote the i^{th} row and j^{th} column of $\boldsymbol{X}$, where $\boldsymbol{X}_{i,:}$ is written as a column vector. Finally, $\mathcal{R}(\boldsymbol{X})$ denotes the range (or column space) of $\boldsymbol{X}$, and $\mathcal{R}^\perp(\boldsymbol{X})$ its orthogonal complement.

We now re-state our assumptions here for convenience.

Assumption 3.2 (Input data)**.** The data $\boldsymbol{X}$ is whitened, and the label dimension K satisfies $K < d/2$.

Assumption 3.3 (Network architecture and training)**.** The network (1) contains $L = 2$ layers, i.e., $f_{\boldsymbol{\Theta}}(\boldsymbol{X}) = \boldsymbol{W}_2\phi(\boldsymbol{W}_1\boldsymbol{X})$, with $\boldsymbol{\Theta} = \{\boldsymbol{W}_2, \boldsymbol{W}_1\}$, $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$, and $\boldsymbol{W}_2 \in \mathbb{R}^{K\times m}$. Furthermore,

- the width m satisfies $m \geq d$,
- the activation function ϕ satisfies $\phi(0) = 0$, $|\phi'(x)| \leq \beta$, and $|\phi''(x)| \leq \mu$ for all $x \in \mathbb{R}$.
- the first-layer weight matrix $\boldsymbol{W}_1$ is trained using GD with step size η on the squared error loss

$$\min_{\boldsymbol{W}_1} \mathcal{L}(\boldsymbol{W}_1) = \frac{1}{2}\left\|\boldsymbol{W}_2\phi\left(\boldsymbol{W}_1\boldsymbol{X}\right) - \boldsymbol{Y}\right\|_F^2, \tag{2}$$

  while $\boldsymbol{W}_2$ is fixed during training and is full row rank.

Assumption 3.4. For all $t \geq 0$, $\|\boldsymbol{W}_2\phi(\boldsymbol{W}_1(t)\boldsymbol{X}) - \boldsymbol{Y}\|_{\max} \leq M$ for some finite constant M.

Assumption 3.6. There exist constants $G_1, G_2, G_3 > 0$ such that $\boldsymbol{G}_1(t)$ satisfies the following:

$$\frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F} \leq G_1 \cdot (1 - \Theta(\eta))^{\Theta(t)}, \quad \frac{\sigma_i\left(\boldsymbol{G}_1(t)\right)}{\sigma_i\left(\boldsymbol{G}_1(0)\right)} \leq G_2 \cdot \frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F} \text{ for all } i \leq K, \text{ and}$$
$$\sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - \sigma_{K+1}\left(\boldsymbol{G}_1(t)\right) \geq G_3 \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right).$$

We emphasize Theorem 3.5 only relies on Assumptions 3.2 through 3.4, while Corollary 3.7 additionally relies on Assumption 3.6.

E.2 Analytical Gradient Form

Here, we derive the analytical form of the loss gradient w.r.t. $\boldsymbol{W}_1$, which we denote as $\boldsymbol{G}_1$. Recall

$$\boldsymbol{H}_0 = \boldsymbol{X} \in \mathbb{R}^{d\times N}, \quad \boldsymbol{Z}_1 = \boldsymbol{W}_1\boldsymbol{H}_0 \in \mathbb{R}^{m\times N}, \quad \boldsymbol{H}_1 = \phi\left(\boldsymbol{Z}_1\right) \in \mathbb{R}^{m\times N}, \quad \text{and} \quad \boldsymbol{Z}_2 = \boldsymbol{W}_2\boldsymbol{H}_1 \in \mathbb{R}^{K\times N}.$$

Notice $\boldsymbol{Z}_2 = f_{\boldsymbol{W}_1}(\boldsymbol{X})$. Define

$$\boldsymbol{\Delta}_2 = \nabla_{\boldsymbol{Z}_2}\mathcal{L}\left(\boldsymbol{W}_1\right) = \boldsymbol{Z}_2 - \boldsymbol{Y} \in \mathbb{R}^{K\times N} \text{ and}$$
$$\boldsymbol{\Delta}_1 = \nabla_{\boldsymbol{Z}_1}\mathcal{L}\left(\boldsymbol{W}_1\right) = \left(\nabla_{\boldsymbol{Z}_1}\boldsymbol{H}_1\right)^\top\left(\nabla_{\boldsymbol{H}_1}\boldsymbol{Z}_2\right)^\top\nabla_{\boldsymbol{Z}_2}\mathcal{L}\left(\boldsymbol{W}_1\right) = \boldsymbol{W}_2^\top\boldsymbol{\Delta}_2 \odot \phi'(\boldsymbol{Z}_1) \in \mathbb{R}^{m\times N},$$

where $\phi'(\cdot)$ is the derivative of activation function $\phi(\cdot)$. The gradient of the loss w.r.t. $\boldsymbol{W}_1$ via backpropagation is

$$\boldsymbol{G}_1 \coloneqq \nabla_{\boldsymbol{W}_1}\mathcal{L}\left(\boldsymbol{W}_1\right) = \boldsymbol{\Delta}_1\boldsymbol{H}_0^\top = \boldsymbol{\Delta}_1\boldsymbol{X}^\top. \tag{8}$$

For all $t > 0$ and $l \in \{1, 2\}$, let $\boldsymbol{Z}_l(t)$, $\boldsymbol{H}_1(t)$, $\boldsymbol{\Delta}_l(t)$, and $\boldsymbol{G}_1(t)$ denote the values of their corresponding matrices at iteration t. Then,

$$\boldsymbol{\Delta}_2(t) = \boldsymbol{Z}_2(t) - \boldsymbol{Y}, \quad \boldsymbol{\Delta}_1(t) = \boldsymbol{W}_2^\top(t)\boldsymbol{\Delta}_2(t) \odot \phi'(\boldsymbol{Z}_1(t)), \quad \text{and} \quad \boldsymbol{G}_1(t) = \boldsymbol{\Delta}_1(t)\boldsymbol{X}^\top.$$

Substituting the expression for $\boldsymbol{\Delta}_1(t)$ into $\boldsymbol{G}_1(t)$ yields

$$\boldsymbol{G}_1(t) = \boldsymbol{\Delta}_1(t)\boldsymbol{H}_0^\top = \Big(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(t) \odot \phi'(\boldsymbol{Z}_1(t))\Big)\boldsymbol{X}^\top = \Big(\boldsymbol{W}_2^\top\Big(\boldsymbol{Z}_2(t) - \boldsymbol{Y}\Big) \odot \phi'(\boldsymbol{W}_1(t)\boldsymbol{X})\Big)\boldsymbol{X}^\top. \tag{9}$$

E.3 Supporting Results

In this section, we provide supporting results that are useful in proving Theorem 3.5. For notational brevity, we use $\ell(t) \coloneqq \mathcal{L}\left(\boldsymbol{W}_1(t)\right) = \ell(\boldsymbol{Z}_2(t), \boldsymbol{Y})$ to denote the value of the loss at GD iteration t.

E.3.1 Auxiliary Results

In this section, we provide auxiliary results in linear algebra. First, for a matrix $\boldsymbol{A}$, we define $|\boldsymbol{A}|$ as applying $|\cdot|$ element-wise to $\boldsymbol{A}$, i.e., $|\boldsymbol{A}|_{ij} = |\boldsymbol{A}_{ij}|$. Our first result relates the spectral norm of $\boldsymbol{A}$ with $|\boldsymbol{A}|$.

Lemma E.1. *For any $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, we have $\sigma_1\left(\boldsymbol{A}\right) \le \sigma_1\left(|\boldsymbol{A}|\right)$.*

Proof. For some square matrix $\boldsymbol{B} \in \mathbb{R}^{n\times n}$, let $\rho\left(\boldsymbol{B}\right)$ denote its spectral radius. Without loss of generality, suppose $m \ge n$. By definition, we have $\sigma_1^2\left(\boldsymbol{A}\right) = \rho\left(\boldsymbol{A}^\top\boldsymbol{A}\right)$, so we focus on upper bounding $\rho\left(\boldsymbol{A}^\top\boldsymbol{A}\right)$ in terms of $\rho\left(\left|\boldsymbol{A}^\top\boldsymbol{A}\right|\right)$. First, we trivially have $\boldsymbol{A}^\top\boldsymbol{A} \le |\boldsymbol{A}^\top\boldsymbol{A}|$, where $\le$ is done element-wise. From [51, Exercise 8.1.9],

$$|\boldsymbol{A}^\top\boldsymbol{A}| \le |\boldsymbol{A}|^\top|\boldsymbol{A}|.$$

Then, applying [51, Theorem 8.1.18] twice yields

$$\rho\left(\boldsymbol{A}^\top\boldsymbol{A}\right) \le \rho\left(\left|\boldsymbol{A}^\top\boldsymbol{A}\right|\right) \le \rho\left(|\boldsymbol{A}|^\top|\boldsymbol{A}|\right),$$

which implies

$$\sigma_1^2\left(\boldsymbol{A}\right) = \rho\left(\boldsymbol{A}^\top\boldsymbol{A}\right) \le \rho\left(|\boldsymbol{A}|^\top|\boldsymbol{A}|\right) = \sigma_1^2\left(|\boldsymbol{A}|\right),$$

which completes the proof. □

Our next result applies upper bounds on the matrix ∞ and 1 norms on products of particular matrix types.

Lemma E.2. *Suppose $\boldsymbol{W} \in \mathbb{R}^{m\times d}$ satisfies $\boldsymbol{W}^\top\boldsymbol{W} = \epsilon^2\boldsymbol{I}_d$, and $\boldsymbol{X} \in \mathbb{R}^{d\times N}$ satisfies $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$. Then, we have*

$$\|\boldsymbol{W}\boldsymbol{X} \odot \boldsymbol{W}\boldsymbol{X}\|_\infty \le \epsilon^2 \quad \textit{and} \quad \|\boldsymbol{W}\boldsymbol{X} \odot \boldsymbol{W}\boldsymbol{X}\|_1 \le \epsilon^2.$$

Proof. Define $\boldsymbol{Z} := \boldsymbol{W}\boldsymbol{X}$. Notice $(\boldsymbol{Z} \odot \boldsymbol{Z})_{ij} = \boldsymbol{Z}_{ij}^2$, and so $\|\boldsymbol{Z} \odot \boldsymbol{Z}\|_\infty$ and $\|\boldsymbol{Z} \odot \boldsymbol{Z}\|_1$ are the maximum row and column squared Euclidean norms of $\boldsymbol{Z}$. Next, notice the i^{th} column of $\boldsymbol{Z}$ is $\boldsymbol{Z}_{i,:} := \boldsymbol{W}_{i,:}^\top \boldsymbol{X}$. Therefore,

$$\|\boldsymbol{Z}_{i,:}\|_2^2 = \|\boldsymbol{W}_{i,:}^\top \boldsymbol{X}\|_2^2 = \boldsymbol{W}_{i,:}^\top \boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{W}_{i,:} = \boldsymbol{W}_{i,:}^\top \boldsymbol{W}_{i,:} = \|\boldsymbol{W}_{i,:}\|_2^2.$$

Since $\boldsymbol{W}$ has ϵ-scaled orthonormal columns, we have $\|\boldsymbol{W}_{i,:}\|_2^2 \leq \epsilon^2$ for all $i \in [m]$. Therefore,

$$\|\boldsymbol{Z} \odot \boldsymbol{Z}\|_\infty = \max_{i\in[m]} \|\boldsymbol{Z}_{i,:}\|_2^2 = \max_{i\in[m]} \|\boldsymbol{W}_{i,:}^\top \boldsymbol{X}\|_2^2 = \max_{i\in[m]} \|\boldsymbol{W}_{i,:}\|_2^2 \leq \epsilon^2.$$

Similarly, the j^{th} column of $\boldsymbol{Z}$ is $\boldsymbol{Z}_{:,j} = \boldsymbol{W}\boldsymbol{X}_{:,j}$. Therefore,

$$\|\boldsymbol{Z}_{:,j}\|_2^2 = \|\boldsymbol{W}\boldsymbol{X}_{:,j}\|_2^2 = \boldsymbol{X}_{:,j}^\top \boldsymbol{W}^\top \boldsymbol{W}\boldsymbol{X}_{:,j} = \epsilon^2 \boldsymbol{X}_{:,j}^\top \boldsymbol{X}_{:,j} = \epsilon^2 \|\boldsymbol{X}_{:,j}\|_2^2.$$

Since $\boldsymbol{X}$ is whitened, i.e., $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$, we have $\|\boldsymbol{X}_{:,j}\|_2^2 \leq 1$ for all $j \in [N]$. Therefore,

$$\|\boldsymbol{Z} \odot \boldsymbol{Z}\|_1 = \max_{j\in[N]} \|\boldsymbol{Z}_{:,j}\|_2^2 = \max_{j\in[N]} \|\boldsymbol{W}\boldsymbol{X}_{:,j}\|_2^2 = \max_{j\in[N]} \epsilon^2 \|\boldsymbol{X}_{:,j}\|_2^2 \leq \epsilon^2.$$

This completes the proof. □

Next, for two matrices $\boldsymbol{U}_1 \in \mathbb{R}^{d\times r}, \boldsymbol{U}_2 \in \mathbb{R}^{d\times r}$ with orthonormal columns, recall the definition of $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2$ in Definition 3.1. Also define

$$\operatorname{dist}(\boldsymbol{U}_1, \boldsymbol{U}_2) = \left\|\boldsymbol{U}_1\boldsymbol{U}_1^\top - \boldsymbol{U}_2\boldsymbol{U}_2^\top\right\|_F. \tag{10}$$

The next result relates $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2$ and $\operatorname{dist}(\boldsymbol{U}_1, \boldsymbol{U}_2)$.

Lemma E.3. *Let $\boldsymbol{U}_1, \boldsymbol{U}_2 \in \mathbb{R}^{d\times K}$ have orthonormal columns, and $\boldsymbol{U}_{1,\perp} \in \mathbb{R}^{d\times(d-K)}$ have orthonormal columns that satisfy $\boldsymbol{U}_1^\top \boldsymbol{U}_{1,\perp} = \boldsymbol{0}_{K\times(d-K)}$. Then,*

$$\operatorname{dist}(\boldsymbol{U}_1, \boldsymbol{U}_2) = \sqrt{2} \cdot \|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2 = \sqrt{2} \cdot \|\boldsymbol{U}_{1,\perp}^\top \boldsymbol{U}_2\|_F.$$

Proof. Let $\psi_1, \ldots, \psi_K$ denote the K principal angles between $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$, which are exactly $\psi_i = \arccos\left(\sigma_i\left(\boldsymbol{U}_1^\top \boldsymbol{U}_2\right)\right)$. Then, from [52, Lemma 2], we have

$$\operatorname{dist}(\boldsymbol{U}_1, \boldsymbol{U}_2) = \sqrt{2} \cdot \sqrt{\sum_{i=1}^K \sin^2(\psi_i)} = \sqrt{2} \cdot \|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2.$$

It suffices to show that $\|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2 = \|\boldsymbol{U}_{1,\perp}^\top \boldsymbol{U}_2\|_F$. We can write

$$\boldsymbol{U}_2 = \boldsymbol{U}_1\boldsymbol{U}_1^\top \boldsymbol{U}_2 + \boldsymbol{U}_{1,\perp}\boldsymbol{U}_{1,\perp}^\top \boldsymbol{U}_2.$$

Therefore,

$$\begin{aligned}
\|\boldsymbol{U}_{1,\perp}^\top \boldsymbol{U}_2\|_F^2 &= \|\boldsymbol{U}_2\|_F^2 - \|\boldsymbol{U}_1^\top \boldsymbol{U}_2\|_F^2 = K - \sum_{i=1}^K \sigma_i^2(\boldsymbol{U}_1^\top \boldsymbol{U}_2) \\
&= K - \sum_{i=1}^K \cos^2(\psi_i) = \sum_{i=1}^K \left(1 - \cos^2(\psi_i)\right) = \sum_{i=1}^K \sin^2(\psi_i) = \|\sin\Psi(\boldsymbol{U}_1, \boldsymbol{U}_2)\|_2^2.
\end{aligned}$$

This completes the proof. □

Our next result upper bounds the alignment between two subspaces after being projected onto a third subspace.

Lemma E.4. *Let $\boldsymbol{Q} \in \mathbb{R}^{m\times d}$ and $\boldsymbol{U}_1, \boldsymbol{U}_2 \in \mathbb{R}^{m\times K}$ all have orthonormal columns. Define $\boldsymbol{U}_{1,\perp} \in \mathbb{R}^{m\times(m-K)}$ to have orthonormal columns that satisfy $\boldsymbol{U}_1^\top \boldsymbol{U}_{1,\perp} = \boldsymbol{0}_{K\times(m-K)}$, and $\widetilde{\boldsymbol{U}}_{1,\perp} \in \mathbb{R}^{d\times(d-K)}$ to be an orthonormal basis for the subspace $\mathcal{R}^\perp\left(\boldsymbol{Q}^\top \boldsymbol{U}_1\right)$. Then, we have*

$$\left\|\boldsymbol{U}_2^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}\right\|_F \leq \|\boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\|_F.$$

Proof. Since $\boldsymbol{U}_1 \in \mathbb{R}^{m\times K}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{m\times(m-K)}$ each have orthonormal columns, and $\boldsymbol{U}_1^\top \boldsymbol{U}_{1,\perp} = \boldsymbol{0}_{m\times(m-K)}$, we have

$$\boldsymbol{I}_m = \boldsymbol{U}_1\boldsymbol{U}_1^\top + \boldsymbol{U}_{1,\perp}\boldsymbol{U}_{1,\perp}^\top.$$

Therefore,

$$\begin{aligned}\boldsymbol{U}_2^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp} &= \boldsymbol{U}_2^\top \left(\boldsymbol{U}_1\boldsymbol{U}_1^\top + \boldsymbol{U}_{1,\perp}\boldsymbol{U}_{1,\perp}^\top\right)\boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}\\ &= \underbrace{\boldsymbol{U}_2^\top \boldsymbol{U}_1\boldsymbol{U}_1^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}}_{(a)} + \boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\boldsymbol{U}_{1,\perp}^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}.\end{aligned}$$

Since $\widetilde{\boldsymbol{U}}_{1,\perp} \in \mathcal{R}^\perp\left(\boldsymbol{Q}^\top\boldsymbol{U}_1\right)$, we have $\boldsymbol{U}_1^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp} = \left(\boldsymbol{Q}^\top\boldsymbol{U}_1\right)^\top \widetilde{\boldsymbol{U}}_{1,\perp} = \boldsymbol{0}_{K\times(d-K)}$, and so $(a) = \boldsymbol{0}_{K\times(d-K)}$. Thus,

$$\begin{aligned}&\left\|\boldsymbol{U}_2^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}\right\|_F = \left\|\boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\boldsymbol{U}_{1,\perp}^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}\right\|_F\\ &\le \left\|\boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\right\|_F \cdot \sigma_1\left(\boldsymbol{U}_{1,\perp}^\top \boldsymbol{Q}\widetilde{\boldsymbol{U}}_{1,\perp}\right) \le \|\boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\|_F \cdot \sigma_1\left(\boldsymbol{U}_{1,\perp}\right)\cdot\sigma_1\left(\boldsymbol{Q}\right)\cdot\sigma_1\left(\widetilde{\boldsymbol{U}}_{1,\perp}\right) = \|\boldsymbol{U}_2^\top \boldsymbol{U}_{1,\perp}\|_F.\end{aligned}$$

This completes the proof. □

E.3.2 Gradient Bounds

In this section, we provide bounds related to the gradient $\boldsymbol{G}_1(t)$. First, we upper bound $\|\boldsymbol{G}_1(t)\|_F$.

Lemma E.5. *For all $t \ge 0$, we have*

$$\|\boldsymbol{G}_1(t)\|_F \le \beta\cdot\sigma_1\left(\boldsymbol{W}_2\right)\cdot\sqrt{2\ell(t)}.$$

Proof. Recall from Equation (9), we have

$$\boldsymbol{G}_1(t) = \Big(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\odot\phi'\left(\boldsymbol{Z}_1(t)\right)\Big)\boldsymbol{X}^\top,$$

and so

$$\begin{aligned}\|\boldsymbol{G}_1(t)\|_F &= \left\|\Big(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\odot\phi'\left(\boldsymbol{Z}_1(t)\right)\Big)\boldsymbol{X}^\top\right\|_F \le \left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\odot\phi'\left(\boldsymbol{Z}_1(t)\right)\right\|_F\cdot\sigma_1(\boldsymbol{X})\\ &\overset{(i)}{=} \left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\odot\phi'\left(\boldsymbol{Z}_1(t)\right)\right\|_F \overset{(ii)}{\le} \beta\cdot\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(t)\|_F = \beta\cdot\sigma_1(\boldsymbol{W}_2)\cdot\|\boldsymbol{\Delta}_2(t)\|_F,\end{aligned}$$

where (i) is because $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$, and (ii) is because ϕ is β-Lipschitz. Finally, note

$$\ell(t) = \frac{1}{2}\|\boldsymbol{W}_2\phi\left(\boldsymbol{W}_2(t)\boldsymbol{X}\right) - \boldsymbol{Y}\|_F^2 = \frac{1}{2}\|\boldsymbol{\Delta}_2(t)\|_F^2 \implies \|\boldsymbol{\Delta}_2(t)\|_F = \sqrt{2\ell(t)}.$$

Putting everything together yields

$$\|\boldsymbol{G}_1(t)\|_F \le \beta\cdot\sigma_1(\boldsymbol{W}_2)\cdot\sqrt{2\ell(t)},$$

which completes the proof. □

Next, we show the gradient $\boldsymbol{G}_1(t)$ is locally Lipschitz.

Lemma E.6. *Suppose $\boldsymbol{W}_1(0) \in \mathbb{R}^{m\times d}$ is an arbitrary ϵ-scaled semi-orthogonal matrix, i.e., $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) = \epsilon^2\boldsymbol{I}_d$. Define $\boldsymbol{\Delta}_2\left(\boldsymbol{W}_1\right) := \boldsymbol{W}_2\phi(\boldsymbol{W}_1\boldsymbol{X}) - \boldsymbol{Y}$, and $\mathcal{M} := \left\{\boldsymbol{W}_1 \in \mathbb{R}^{m\times d} : \|\boldsymbol{\Delta}_2(\boldsymbol{W}_1)\|_{\max} \le M\right\}$. Also define $\boldsymbol{G}_1\left(\boldsymbol{W}_1\right)$ to be $\boldsymbol{G}_1$ at some $\boldsymbol{W}_1 \in \mathbb{R}^{m\times d}$. Then, for all $\boldsymbol{W}_1, \widehat{\boldsymbol{W}}_1 \in \mathcal{M}$, we have*

$$\|\boldsymbol{G}_1\left(\boldsymbol{W}_1\right) - \boldsymbol{G}_1(\widehat{\boldsymbol{W}}_1)\|_F \le \left(\mu\cdot M\cdot\|\boldsymbol{W}_2\|_1 + \beta^2\cdot\sigma_1^2(\boldsymbol{W}_2)\right)\cdot\|\boldsymbol{W}_1 - \widehat{\boldsymbol{W}}_1\|_F.$$

Proof. Suppose $\boldsymbol{W}_1, \widehat{\boldsymbol{W}}_1 \in \mathcal{M}$. Let $\boldsymbol{G}_1(\boldsymbol{W}_1), \boldsymbol{G}_1(\widehat{\boldsymbol{W}}_1)$ denote $\boldsymbol{G}_1$ at $\boldsymbol{W}_1$ and $\widehat{\boldsymbol{W}}_1$, and let $\boldsymbol{\Delta}_2 := \boldsymbol{\Delta}_2(\boldsymbol{W}_1)$ and $\hat{\boldsymbol{\Delta}}_2 := \boldsymbol{\Delta}_2(\widehat{\boldsymbol{W}}_1)$. Then, we have

$$
\begin{aligned}
&\left\|\boldsymbol{G}_1(\boldsymbol{W}_1) - \boldsymbol{G}_1(\widehat{\boldsymbol{W}}_1)\right\|_F \\
&= \left\|\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'\left(\boldsymbol{W}_1 \boldsymbol{X}\right) - \boldsymbol{W}_2^\top \hat{\boldsymbol{\Delta}}_2 \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right) \boldsymbol{X}^\top\right\|_F \\
&\leq \left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'\left(\boldsymbol{W}_1 \boldsymbol{X}\right) - \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right\|_F \\
&= \left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'(\boldsymbol{W}_1 \boldsymbol{X}) - \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X}) + \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X}) - \boldsymbol{W}_2^\top \hat{\boldsymbol{\Delta}}_2 \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right\|_F \\
&\leq \left(\left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \left(\phi'(\boldsymbol{W}_1 \boldsymbol{X}) - \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right)\right\|_F + \left\|\boldsymbol{W}_2^\top \left(\boldsymbol{\Delta}_2 - \hat{\boldsymbol{\Delta}}_2\right) \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right\|_F\right) \\
&= \left(\left\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2 \odot \left(\phi'(\boldsymbol{W}_1 \boldsymbol{X}) - \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right)\right\|_F + \left\|\boldsymbol{W}_2^\top \boldsymbol{W}_2 \left(\phi(\boldsymbol{W}_1 \boldsymbol{X}) - \phi(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right) \odot \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right\|_F\right) \\
&\leq \left(\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2\|_{\max} \cdot \left\|\phi'(\boldsymbol{W}_1 \boldsymbol{X}) - \phi'(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\right\|_F + \beta \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \|\phi(\boldsymbol{W}_1 \boldsymbol{X}) - \phi(\widehat{\boldsymbol{W}}_1 \boldsymbol{X})\|_F\right) \\
&\overset{(i)}{\leq} \left(\mu \cdot \|\boldsymbol{W}_2\|_1 \cdot \|\boldsymbol{\Delta}_2\|_{\max} \cdot \|\boldsymbol{W}_1 \boldsymbol{X} - \widehat{\boldsymbol{W}}_1 \boldsymbol{X}\|_F + \beta^2 \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \|\boldsymbol{W}_1 \boldsymbol{X} - \widehat{\boldsymbol{W}}_1 \boldsymbol{X}\|_F\right) \\
&\overset{(ii)}{\leq} \left(\mu \cdot M \cdot \|\boldsymbol{W}_2\|_1 + \beta^2 \cdot \sigma_1^2(\boldsymbol{W}_2)\right) \cdot \|\boldsymbol{W}_1 - \widehat{\boldsymbol{W}}_1\|_F,
\end{aligned}
$$

where (i) is because ϕ and β-Lipschitz and μ-smooth, and (ii) is because $\boldsymbol{W}_1, \widehat{\boldsymbol{W}}_1 \in \mathcal{M}$, which implies $\|\boldsymbol{\Delta}_2\|_{\max} \leq M$ by definition. This completes the proof. □

From now on, we define $\gamma_L := \mu \cdot M \cdot \|\boldsymbol{W}_2\|_1 + \beta^2 \cdot \sigma_1^2(\boldsymbol{W}_2)$. Note that by the Descent Lemma [53], if $\eta \leq \frac{1}{\gamma_L}$, then the loss does not increase. Thus, we assume $\eta \leq \frac{1}{\gamma_L}$.

E.3.3 Gradient Singular Values and Subspaces

In this section, we provide results characterizing the singular values and subspaces of the gradient. Before proceeding, we provide some additional definitions. Let $\boldsymbol{G}_1(t) = \boldsymbol{L}_1(t) \boldsymbol{\Sigma}_1(t) \boldsymbol{R}_1^\top(t)$ denote an SVD of $\boldsymbol{G}_1(t)$, with $\boldsymbol{L}_{1,1}(t) \in \mathbb{R}^{m \times K}$, $\boldsymbol{\Sigma}_{1,1}(t) \in \mathbb{R}^{K \times K}$, and $\boldsymbol{R}_{1,1}(t) \in \mathbb{R}^{d \times K}$ denoting the top-K components, and $\boldsymbol{L}_{1,2}(t) \in \mathbb{R}^{m \times (d-K)}$, $\boldsymbol{\Sigma}_{1,2}(t) \in \mathbb{R}^{(d-K) \times (d-K)}$, and $\boldsymbol{R}_{1,2}(t) \in \mathbb{R}^{d \times (d-K)}$ denoting the bottom $d - K$ components.

Our first result shows under small initialization scale ϵ, $\boldsymbol{G}_1(0)$ is approximately rank-K.

Lemma E.7. *Suppose Assumptions 3.2 and 3.3 hold. Define*

$$
r(\epsilon) = \epsilon \cdot \phi'(0) \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \left(\phi'(0) + \frac{\mu}{2} \cdot \epsilon\right) + \mu \cdot \|\boldsymbol{W}_1(0) \boldsymbol{X}\|_{\max} \cdot \sigma_1(\boldsymbol{W}_2) \cdot \left(\beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{d} \cdot \epsilon + \|\boldsymbol{Y}\|_F\right).
$$

If $\boldsymbol{W}_1^\top(0) \boldsymbol{W}_1(0) = \epsilon^2 \boldsymbol{I}_d$, where ϵ satisfies $r(\epsilon) < \frac{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top\right)}{2}$, then, for all $i \in [d]$, we have

$$
\sigma_i\left(\boldsymbol{G}_1(0)\right) \in \begin{cases} \left[\phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top) - r(\epsilon), \phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top) + r(\epsilon)\right] & i = 1, \ldots, K \\ [0, r(\epsilon)] & i = K+1, \ldots, d \end{cases}
$$

Proof. Recall from (9) that

$$
\boldsymbol{G}_1(0) = \left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \phi'(\boldsymbol{W}_1(0) \boldsymbol{X})\right) \boldsymbol{X}^\top.
$$

Now, define a term $\boldsymbol{P}(0) := \left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \phi'(\boldsymbol{W}_1(0) \boldsymbol{X})\right) - \phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)$, and $\boldsymbol{\Gamma}(0) := \boldsymbol{\Delta}_2(0) \boldsymbol{X}^\top$. This means

$$
\boldsymbol{G}_1(0) = \phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \boldsymbol{X}^\top + \boldsymbol{P}(0) \boldsymbol{X}^\top = \phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0) + \boldsymbol{P}(0) \boldsymbol{X}^\top. \tag{11}
$$

Here, $\phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)$ is the linear component of $\boldsymbol{G}_1(0)$, while $\boldsymbol{P}(0) \boldsymbol{X}^\top$ is a perturbation of $\phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)$.

Singular values of $\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)$. We first analyze the singular values of $\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)$, which is (at most) rank-K. Note

$$\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0) = \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \boldsymbol{X}^\top = \left(\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi(\boldsymbol{W}_1(0)\boldsymbol{X}) - \boldsymbol{W}_2^\top \boldsymbol{Y}\right) \boldsymbol{X}^\top = (\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi(\boldsymbol{W}_1(0)\boldsymbol{X}) \boldsymbol{X}^\top - \boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top).$$

Also note

$$\begin{aligned}
&\sigma_1(\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right) \boldsymbol{X}^\top) \leq \sigma_1\left(\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)\right) \leq \sigma_1^2(\boldsymbol{W}_2) \cdot \sigma_1(\phi(\boldsymbol{W}_1(0)\boldsymbol{X})) \\
&\leq \sigma_1^2(\boldsymbol{W}_2) \cdot \Big(\phi'(0) \cdot \sigma_1(\boldsymbol{Z}_1(0)) + \sigma_1(\boldsymbol{R}(\boldsymbol{Z}_1(0))\Big),
\end{aligned}$$

where the last inequality is from taking the Taylor expansion of $\phi(\boldsymbol{W}_1(0)\boldsymbol{X})$ element-wise around $\boldsymbol{0}_m \boldsymbol{0}_N^\top$, with $\boldsymbol{R}(\boldsymbol{Z}_1(0))$ denoting the remainder, and then applying the triangle inequality. Since $\boldsymbol{W}_1(0)$ is an ϵ-scaled semi-orthogonal matrix, and $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$, we have $\sigma_1(\boldsymbol{Z}_1(0)) \leq \sigma_1\left(\boldsymbol{W}_1(0)\right) = \epsilon$. Next, since $\phi(\cdot)$ is μ-smooth, by Taylor's Remainder Theorem [54], we have

$$|\boldsymbol{R}(\boldsymbol{Z}_1(0))_{ij}| \leq \frac{\mu}{2} \left|(\boldsymbol{Z}_1(0))_{ij}\right|^2 = \frac{\mu}{2} \left(\boldsymbol{Z}_1(0)\right)_{ij}^2,$$

or equivalently,

$$|\boldsymbol{R}(\boldsymbol{Z}_1(0))| \leq \frac{\mu}{2} \left(\boldsymbol{Z}_1(0) \odot \boldsymbol{Z}_1(0)\right),$$

where $|\cdot|$ and $\leq$ are element-wise. We then have

$$\begin{aligned}
&\sigma_1\Big(\boldsymbol{R}(\boldsymbol{Z}_1(0))\Big) \overset{(i)}{\leq} \sigma_1\Big(|\boldsymbol{R}(\boldsymbol{Z}_1(0))|\Big) \overset{(ii)}{\leq} \frac{\mu}{2} \cdot \sigma_1\Big(\boldsymbol{Z}_1(0) \odot \boldsymbol{Z}_1(0)\Big) \\
&\leq \frac{\mu}{2} \cdot \sqrt{\|\boldsymbol{Z}_1(0) \odot \boldsymbol{Z}_1(0)\|_\infty \|\boldsymbol{Z}_1(0) \odot \boldsymbol{Z}_1(0)\|_1} \\
&\overset{(iii)}{\leq} \frac{\mu}{2} \cdot \sqrt{\epsilon^2 \cdot \epsilon^2} = \frac{\mu}{2} \cdot \epsilon^2,
\end{aligned}$$

where (i) is from Lemma E.1, (ii) is from [51, Theorem 8.1.18], and (iii) is from Lemma E.2. In summary,

$$\begin{aligned}
&\sigma_1\left(\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi(\boldsymbol{W}_1(0)\boldsymbol{X})\right) \leq \sigma_1^2(\boldsymbol{W}_2) \cdot \left(\phi'(0) \cdot \sigma_1(\boldsymbol{Z}_1(0)) + \sigma_1\left(\boldsymbol{R}(\boldsymbol{Z}_1(0))\right)\right) \\
&\leq \sigma_1^2(\boldsymbol{W}_2) \cdot \Big(\phi'(0) \cdot \epsilon + \frac{\mu}{2} \cdot \epsilon^2\Big) = \epsilon \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \Big(\phi'(0) + \frac{\mu}{2} \cdot \epsilon\Big).
\end{aligned}$$

Thus, by Weyl's inequality [55]:

$$\left|\sigma_i(\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)) - \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top)\right| \leq \sigma_1(\boldsymbol{W}_2^\top \boldsymbol{W}_2 \phi(\boldsymbol{W}_1(0)\boldsymbol{X})) \leq \epsilon \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \Big(\phi'(0) + \frac{\mu}{2} \cdot \epsilon\Big). \tag{12}$$

Let $r_1(\epsilon) := \epsilon \cdot \phi'(0) \cdot \sigma_1^2\left(\boldsymbol{W}_2\right) \cdot \left(\phi'(0) + \frac{\mu}{2} \cdot \epsilon\right)$. By rearranging (12), for all $i = 1, \ldots, K$, we have

$$\phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top) - r_1(\epsilon) \leq \phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)) \leq \phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top) + r_1(\epsilon) \tag{13}$$

For $i = K+1, \ldots, d$, we have $\sigma_i\left(\phi'(0) \cdot \boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)\right) = 0$ since $\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)$ is at most rank-K.

Singular values of $\boldsymbol{G}_1(0)$. We now analyze the singular values of $\boldsymbol{G}_1(0)$. Recall $\phi'(x)$ is μ-Lipschitz over $\mathbb{R}$, so $|\phi'(x) - \phi'(0)| \leq \mu \cdot |x|$ for all $x \in \mathbb{R}$. Therefore,

$$\begin{aligned}
&|\phi'(\boldsymbol{W}_1(0)\boldsymbol{X})_{ij} - \phi'(0)| \leq \mu \left|(\boldsymbol{W}_1(0)\boldsymbol{X})_{ij}\right| \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \\
&\implies \phi'(0) - \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \leq \phi'(\boldsymbol{W}_1(0)\boldsymbol{X})_{ij} \leq \phi'(0) + \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \\
&\implies -\mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \leq \phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)_{ij} - \phi'(0) \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max}.
\end{aligned}$$

Therefore,

$$-\mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\right)_{ij} \leq \underbrace{\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\right)_{ij} \cdot \Big(\phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)_{ij} - \phi'(0)\Big)}_{:= (\boldsymbol{P}(0))_{ij}} \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\right)_{ij}.$$

As a result,

$$\left|\left(\boldsymbol{P}(0)\right)_{ij}\right| \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \left|\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\right)_{ij}\right| \implies \|\boldsymbol{P}(0)\|_F \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\|_F.$$

Note $\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\|_F \leq \sigma_1(\boldsymbol{W}_2) \cdot \|\boldsymbol{\Delta}_2(0)\|_F$, and so

$$\|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\|_F \leq \sigma_1(\boldsymbol{W}_2) \cdot \|\boldsymbol{W}_2\phi(\boldsymbol{W}_1(0)\boldsymbol{X}) - \boldsymbol{Y}\|_F \leq \sigma_1(\boldsymbol{W}_2) \cdot \left(\beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{d} \cdot \epsilon + \|\boldsymbol{Y}\|_F\right)$$

Again by Weyl's inequality, Equation (11), and Equation (13),

$$\begin{aligned}&\left|\sigma_i(\boldsymbol{G}_1(0)) - \phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0))\right| \leq \sigma_1(\boldsymbol{P}(0)\boldsymbol{X}^\top) \leq \|\boldsymbol{P}(0)\boldsymbol{X}^\top\|_F \\ &\leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \|\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)\|_F \leq \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \sigma_1(\boldsymbol{W}_2) \cdot \left(\beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{d} \cdot \epsilon + \|\boldsymbol{Y}\|_F\right).\end{aligned}$$

Let $r_2(\epsilon) := \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \sigma_1(\boldsymbol{W}_2) \cdot \left(\beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{d} \cdot \epsilon + \|\boldsymbol{Y}\|_F\right)$. Therefore, for all $i \in [d]$, we have

$$\begin{aligned}&\phi'(0) \cdot \sigma_i\left(\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)\right) - r_2(\epsilon) \leq \sigma_i\left(\boldsymbol{G}_1(0)\right) \leq \phi'(0) \cdot \sigma_i\left(\boldsymbol{W}_2^\top \boldsymbol{\Gamma}(0)\right) + r_2(\epsilon) \\ &\implies \phi'(0) \cdot \sigma_i\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - r_1(\epsilon) - r_2(\epsilon) \leq \sigma_i\left(\boldsymbol{G}_1(0)\right) \leq \phi'(0) \cdot \sigma_i\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) + r_1(\epsilon) + r_2(\epsilon).\end{aligned}$$

Define $r(\epsilon) = r_1(\epsilon) + r_2(\epsilon)$. Since $\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top$ is rank-K, for all $i \in [d]$, we have

$$\sigma_i\left(\boldsymbol{G}_1(0)\right) \in \begin{cases} \left[\phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top) - r(\epsilon), \phi'(0) \cdot \sigma_i(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top) + r(\epsilon)\right] & i = 1, \ldots, K \\ [0, r(\epsilon)] & i = K+1, \ldots, d \end{cases}$$

Finally, note that we require $\sigma_K(\boldsymbol{G}_1(0)) - \sigma_{K+1}(\boldsymbol{G}_1(0)) > 0$, which implies

$$\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - 2 \cdot r(\epsilon) > 0 \implies r(\epsilon) < \frac{\phi'(0) \cdot \sigma_K(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top)}{2},$$

which completes the proof. □

Our next result characterizes how the bottom $d - K$ left and singular subspaces change between $\boldsymbol{G}_1(t)$ and $\boldsymbol{G}_1(0)$.

Lemma E.8. *Suppose Assumptions 3.2 through 3.6 hold. Define*

$$\gamma_L := \mu \cdot M \cdot \|\boldsymbol{W}_2\|_1 + \beta^2 \cdot \sigma_1^2\left(\boldsymbol{W}_2\right), \quad \textit{and}$$
$$r(\epsilon) = \epsilon \cdot \phi'(0) \cdot \sigma_1^2(\boldsymbol{W}_2) \cdot \left(\phi'(0) + \frac{\mu}{2} \cdot \epsilon\right) + \mu \cdot \|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max} \cdot \sigma_1(\boldsymbol{W}_2) \cdot \left(\beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{d} \cdot \epsilon + \|\boldsymbol{Y}\|_F\right).$$

If $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) = \epsilon^2 \boldsymbol{I}_d$ where ϵ satisfies $r(\epsilon) < \frac{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right)}{2}$, and $\eta \leq \frac{1}{\gamma_L}$, then for all $t \geq 1$, we have

$$\left\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{L}_{1,2}(0)\right\|_F \leq \frac{\gamma_L \cdot \beta \cdot \sigma_1\left(\boldsymbol{W}_2\right) \cdot \sqrt{2\ell(0)} \cdot \Theta(1) \cdot \left(1 - (1 - \Theta(\eta))^{\Theta(t)}\right)}{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - r(\epsilon) - \sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)}, \quad \textit{and}$$

$$\|\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{R}_{1,2}(0)\|_F \leq \frac{\gamma_L \cdot \beta \cdot \sigma_1\left(\boldsymbol{W}_2\right) \cdot \sqrt{2\ell(0)} \cdot \Theta(1) \cdot \left(1 - (1 - \Theta(\eta))^{\Theta(t)}\right)}{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top\right) - r(\epsilon) - \sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)},$$

which provides upper bounds on the change in the top-K left and right singular subspaces of $\boldsymbol{G}_1(t)$ from $\boldsymbol{G}_1(0)$.

Proof. Before proceeding, we note that at $t = 0$, we have $\operatorname{dist}\left(\boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(t)/\epsilon, \boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right) = \operatorname{dist}\left(\boldsymbol{R}_{1,2}(t), \boldsymbol{R}_{1,2}(0)\right) = 0$ exactly, so we consider $t \geq 1$. This is an direct application of Wedin's Sin Theorem [56]. We first upper bound

$\|\boldsymbol{G}_1(t) - \boldsymbol{G}_1(0)\|_F$:

$$\|\boldsymbol{G}_1(t) - \boldsymbol{G}_1(0)\|_F \overset{(i)}{\leq} \gamma_L \cdot \|\boldsymbol{W}_1(t) - \boldsymbol{W}_1(0)\|_F = \gamma_L \cdot \eta \cdot \left\| \sum_{\tau=0}^{t-1} \boldsymbol{G}_1(\tau) \right\|_F$$
$$\leq \gamma_L \cdot \eta \cdot \sum_{\tau=0}^{t-1} \|\boldsymbol{G}_1(\tau)\|_F \overset{(ii)}{\leq} \gamma_L \cdot \eta \cdot \|\boldsymbol{G}_1(0)\|_F \cdot \sum_{\tau=0}^{t-1} (1 - \Theta(\eta))^{\Theta(\tau)}$$
$$\overset{(iii)}{\leq} \gamma_L \cdot \beta \cdot \eta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{2\ell(0)} \cdot \sum_{\tau=0}^{t-1} (1 - \Theta(\eta))^{\Theta(\tau)}$$
$$\leq \gamma_L \cdot \beta \cdot \eta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{2\ell(0)} \cdot \frac{1 - (1 - \Theta(\eta))^{\Theta(t)}}{\Theta(\eta)}$$
$$= \gamma_L \cdot \beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{2\ell(0)} \cdot \Theta(1) \cdot \left(1 - (1 - \Theta(\eta))^{\Theta(t)}\right),$$

where (i) is from Lemma E.6, (ii) is from Assumption 3.6, and (iii) is from Lemma E.5. Then, directly from Wedin's Sin Theorem [56] and Lemma E.3,

$$\|\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{R}_{1,2}(0)\|_F \leq \frac{\|\boldsymbol{G}_1(t) - \boldsymbol{G}_1(0)\|_F}{\sigma_K(\boldsymbol{G}_1(0)) - \sigma_{K+1}(\boldsymbol{G}_1(t))} \leq \frac{\gamma_L \cdot \beta \cdot \sigma_1(\boldsymbol{W}_2) \cdot \sqrt{2\ell(0)} \cdot \Theta(1) \cdot \left(1 - (1 - \Theta(\eta))^{\Theta(t)}\right)}{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top\right) - r(\epsilon) - \sigma_{K+1}(\boldsymbol{G}_1(t))}, \tag{14}$$

which also applies to $\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{L}_{1,2}(0)\|_F$. Note from Assumption 3.6 and our condition on $r(\epsilon)$, the denominator is strictly positive. This completes the proof.

□

E.4 Proof of Theorem 3.5

In this section, we provide a proof of Theorem 3.5. First, we provide a result that identifies a p-dimensional subspace where $\boldsymbol{G}_1(t)$ has a bounded component for all $t \geq 0$ — Theorem 3.5 follows straightforwardly after.

Lemma E.9. *Suppose Assumptions 3.2 through 3.4 hold. If $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) = \epsilon^2 \boldsymbol{I}_d$ where ϵ satisfies $r(\epsilon) < \frac{\phi'(0) \cdot \sigma_K\left(\boldsymbol{W}_2^\top \boldsymbol{Y} \boldsymbol{X}^\top\right)}{2}$, and $\eta \leq \frac{1}{\gamma_L}$, then there exist semi-orthogonal matrices $\boldsymbol{V} \in \mathbb{R}^{d \times p}$ and $\boldsymbol{U} \in \mathbb{R}^{m \times p}$ such that*

$$\boldsymbol{W}_1(0)\boldsymbol{V} = \epsilon \boldsymbol{U}, \quad \boldsymbol{W}_1^\top(0)\boldsymbol{U} = \epsilon \boldsymbol{V}, \quad \textit{and}$$
$$\max\left\{ \frac{\|\boldsymbol{W}_1(t+1)\boldsymbol{V} - \boldsymbol{W}_1(t)\boldsymbol{V}\|_F}{\sqrt{p}}, \frac{\|\boldsymbol{W}_1^\top(t+1)\boldsymbol{U} - \boldsymbol{W}_1^\top(t)\boldsymbol{U}\|_F}{\sqrt{p}} \right\} \leq \eta \cdot \sqrt{\sigma_1^2(\boldsymbol{G}_1(t)) \cdot \frac{A^2(t)}{p} + \sigma_{K+1}^2(\boldsymbol{G}_1(t))}.$$

Proof. Define

$$\mathcal{S} := \mathcal{R}(\boldsymbol{R}_{1,2}(0)) \cap \mathcal{R}^\perp\left(\boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right),$$

Both $\mathcal{R}(\boldsymbol{R}_{1,2}(0))$ and $\mathcal{R}^\perp\left(\boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right)$ are of dimension $d-K$, so $\dim(\mathcal{S}) = d - 2K := p$. Let $\boldsymbol{V} \in \mathbb{R}^{d \times p}$ be an orthonormal basis for $\mathcal{S}$, and $\boldsymbol{U} := \boldsymbol{W}_1(0)\boldsymbol{V}/\epsilon \in \mathbb{R}^{m \times p}$. From the definitions of $\boldsymbol{U}$ and $\boldsymbol{V}$, we trivially have

$$\boldsymbol{W}_1(0)\boldsymbol{V} - \epsilon \boldsymbol{U} = \boldsymbol{0}_{m \times p} \quad \text{and} \quad \boldsymbol{W}_1^\top(0)\boldsymbol{U} - \epsilon \boldsymbol{V} = \boldsymbol{0}_{d \times p}.$$

Now consider an arbitrary t for $t \geq 0$. From the GD update,

$$\|\boldsymbol{W}_1(t+1)\boldsymbol{V} - \boldsymbol{W}_1(t)\boldsymbol{V}\|_F = \eta \cdot \|\boldsymbol{G}_1(t)\boldsymbol{V}\|_F \quad \text{and}$$
$$\|\boldsymbol{W}_1^\top(t+1)\boldsymbol{U} - \boldsymbol{W}_1^\top(t)\boldsymbol{U}\|_F = \eta \cdot \|\boldsymbol{G}_1^\top(t)\boldsymbol{U}\|_F.$$

We first upper bound $\|\boldsymbol{G}_1(t)\boldsymbol{V}\|_F$, which then also applies for $\|\boldsymbol{G}_1^\top(t)\boldsymbol{U}\|_F$.

Upper bounding $\|\boldsymbol{G}_1(t)\boldsymbol{V}\|_F$ and $\|\boldsymbol{G}_1^\top(t)\boldsymbol{U}\|_F$. Recall $\boldsymbol{G}_1(t) = \boldsymbol{L}_{1,1}(t)\boldsymbol{\Sigma}_{1,1}(t)\boldsymbol{R}_{1,1}^\top(t) + \boldsymbol{L}_{1,2}(t)\boldsymbol{\Sigma}_{1,2}(t)\boldsymbol{R}_{1,2}^\top(t)$. Therefore,

$$\begin{aligned}
&\boldsymbol{G}_1(t)\boldsymbol{V} = \boldsymbol{L}_{1,1}(t)\boldsymbol{\Sigma}_{1,1}(t)\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{V} + \boldsymbol{L}_{1,2}(t)\boldsymbol{\Sigma}_{1,2}(t)\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V} \\
\implies \|\boldsymbol{G}_1(t)\boldsymbol{V}\|_F &= \sqrt{\left\|\boldsymbol{L}_{1,1}(t)\boldsymbol{\Sigma}_{1,1}(t)\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{V}\right\|_F^2 + \left\|\boldsymbol{L}_{1,2}(t)\boldsymbol{\Sigma}_{1,2}(t)\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\right\|_F^2} \\
&= \sqrt{\left\|\boldsymbol{\Sigma}_{1,1}(t)\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{V}\right\|_F^2 + \left\|\boldsymbol{\Sigma}_{1,2}(t)\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\right\|_F^2} \\
&\le \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{V}\right\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\right\|_F^2} \\
&\le \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\boldsymbol{R}_{1,1}^\top(t)\boldsymbol{R}_{1,2}(0)\right\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\right\|_F^2} \\
&\overset{(i)}{=} \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\sin\Psi\left(\boldsymbol{R}_{1,1}(t),\boldsymbol{R}_{1,1}(0)\right)\right\|_2^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\left\|\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\right\|_F^2}.
\end{aligned}$$

where (i) is from Lemma E.3. Next, define $\boldsymbol{W}_1(0) := \epsilon\boldsymbol{Q} \implies \boldsymbol{Q} := \boldsymbol{W}_1(0)/\epsilon$ and $\widetilde{\boldsymbol{L}}_{1,2}(0)$ to be an orthonormal basis for $\mathcal{R}^\perp\left(\boldsymbol{W}_1^\top(0)\boldsymbol{L}_{1,1}(0)/\epsilon\right) = \mathcal{R}^\perp\left(\boldsymbol{Q}^\top\boldsymbol{L}_{1,1}(0)\right)$. Recall by definition, we have $\boldsymbol{V} \in \mathcal{R}\left(\widetilde{\boldsymbol{L}}_{1,2}(0)\right)$. Therefore,

$$\begin{aligned}
&\boldsymbol{G}_1^\top(t)\boldsymbol{U} = \boldsymbol{R}_{1,1}(t)\boldsymbol{\Sigma}_{1,1}^\top(t)\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{U} + \boldsymbol{R}_{1,2}(t)\boldsymbol{\Sigma}_{1,2}^\top(t)\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{U} \\
\implies \|\boldsymbol{G}_1^\top(t)\boldsymbol{U}\|_F &= \sqrt{\|\boldsymbol{R}_{1,1}(t)\boldsymbol{\Sigma}_{1,1}^\top(t)\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{U}\|_F^2 + \|\boldsymbol{R}_{1,2}(t)\boldsymbol{\Sigma}_{1,2}^\top(t)\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{U}\|_F^2} \\
&= \sqrt{\|\boldsymbol{\Sigma}_{1,1}^\top(t)\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{U}\|_F^2 + \|\boldsymbol{\Sigma}_{1,2}^\top(t)\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{U}\|_F^2} \\
&\le \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{U}\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{U}\|_F^2} \\
&= \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{W}_1(0)\boldsymbol{V}/\epsilon\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{W}_1(0)\boldsymbol{V}/\epsilon\|_F^2} \\
&:= \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2} \\
&\le \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{Q}\widetilde{\boldsymbol{L}}_{1,2}(0)\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2} \\
&\overset{(ii)}{\le} \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,1}^\top(t)\boldsymbol{L}_{1,2}(0)\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2} \\
&\overset{(iii)}{=} \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\sin\Psi\left(\boldsymbol{L}_{1,1}(t),\boldsymbol{L}_{1,1}(0)\right)\|_F^2 + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2},
\end{aligned}$$

where (ii) is from Lemma E.4, and (iii) is from Lemma E.3. The definition of $A(t)$, combined with the fact that $\|\boldsymbol{R}_{1,2}^\top(t)\boldsymbol{V}\|_F^2 \le p$ and $\|\boldsymbol{L}_{1,2}^\top(t)\boldsymbol{Q}\boldsymbol{V}\|_F^2 \le p$, gives us

$$\max\left\{\|\boldsymbol{G}_1(t)\boldsymbol{V}\|_F, \|\boldsymbol{G}_1^\top(t)\boldsymbol{U}\|_F\right\} \le \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot A^2(t) + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot p} := \rho(t)\cdot\sqrt{p}.$$

This completes the proof. □

We now provide a proof of Theorem 3.5. We re-state the (formal) Theorem statement here for convenience.

Theorem E.10 (Formal version of Theorem 3.5)**.** *Suppose Assumptions 3.2 through 3.4 hold. Define*

$$\gamma_L := \mu\cdot M\cdot\|\boldsymbol{W}_2\|_1 + \beta^2\cdot\sigma_1^2\left(\boldsymbol{W}_2\right),$$

$$r(\epsilon) := \epsilon\cdot\phi'(0)\cdot\sigma_1^2(\boldsymbol{W}_2)\cdot\left(\phi'(0) + \frac{\mu}{2}\epsilon\right) + \mu\cdot\|\boldsymbol{W}_1(0)\boldsymbol{X}\|_{\max}\cdot\sigma_1(\boldsymbol{W}_2)\cdot\left(\beta\cdot\sigma_1(\boldsymbol{W}_2)\cdot\sqrt{d}\cdot\epsilon + \|\boldsymbol{Y}\|_F\right),$$

$$A(t) := \max\left\{\|\sin\Psi\left(\boldsymbol{L}_{1,1}(t),\boldsymbol{L}_{1,1}(0)\right)\|_2, \|\sin\Psi\left(\boldsymbol{R}_{1,1}(t),\boldsymbol{R}_{1,1}(0)\right)\|_2\right\}.$$

If $\boldsymbol{W}_1^\top(0)\boldsymbol{W}_1(0) = \epsilon^2\boldsymbol{I}_d$ where ϵ satisfies $r(\epsilon) < \frac{\phi'(0)\cdot\sigma_K\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\boldsymbol{X}^\top\right)}{2}$, and $\eta \le \frac{1}{\gamma_L}$, then there exist orthogonal matrices $\boldsymbol{U} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{V} \in \mathbb{R}^{d\times d}$ such that, for all $t \ge 0$, $\boldsymbol{W}_1(t)$ admits the following decomposition:

$$\boldsymbol{W}_1(t) = \boldsymbol{U}\widetilde{\boldsymbol{W}}_1(t)\boldsymbol{V}^\top = \boldsymbol{U}\begin{bmatrix}\widetilde{\boldsymbol{W}}_{1,1}(t) & \widetilde{\boldsymbol{W}}_{1,2}(t)\\ \widetilde{\boldsymbol{W}}_{1,3}(t) & \widetilde{\boldsymbol{W}}_{1,4}(t)\end{bmatrix}\boldsymbol{V}^\top,$$

where $\widetilde{\boldsymbol{W}}_{1,1} \in \mathbb{R}^{(m-p)\times 2K}$, $\widetilde{\boldsymbol{W}}_{1,2} \in \mathbb{R}^{(m-p)\times p}$, $\widetilde{\boldsymbol{W}}_{1,3} \in \mathbb{R}^{p\times 2K}$, *and* $\widetilde{\boldsymbol{W}}_{1,4} \in \mathbb{R}^{p\times p}$, *with*

$$\widetilde{\boldsymbol{W}}_{1,2}(0) = \mathbf{0}_{(m-p)\times p}, \quad \widetilde{\boldsymbol{W}}_{1,3}(0) = \mathbf{0}_{p\times 2K}, \quad \frac{\|\widetilde{\boldsymbol{W}}_{1,4}(0)\|_F}{\sqrt{p}} = \epsilon,$$

and

$$\frac{\|\widetilde{\boldsymbol{W}}_{1,i}(t+1) - \widetilde{\boldsymbol{W}}_{1,i}(t)\|_F}{\sqrt{p}} \leq \eta \cdot \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right) \cdot \frac{A^2(t)}{p} + \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)}$$

for all $i \in \{2,3,4\}$.

Proof. Define

$$\rho(t) := \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right) \cdot A^2(t) + p \cdot \sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)}.$$

From Lemma E.9, there exist $\boldsymbol{V}_2 \in \mathbb{R}^{d\times p}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{m\times p}$ with orthonormal columns such that for any $t \geq 0$:

$$\boldsymbol{W}_1(t+1)\boldsymbol{V}_2 = \boldsymbol{W}_1(t)\boldsymbol{V}_2 + \boldsymbol{T}_1(t),$$

where $\|\boldsymbol{T}_1(t)\|_F \leq \eta \cdot \rho(t)$, and $\boldsymbol{W}_1(0)\boldsymbol{V}_2 = \epsilon \boldsymbol{U}_2$. Similarly,

$$\boldsymbol{W}_1^\top(t+1)\boldsymbol{U}_2 = \boldsymbol{W}_1^\top(t)\boldsymbol{U}_2 + \boldsymbol{T}_2(t),$$

where again $\|\boldsymbol{T}_2(t)\|_F \leq \eta \cdot \rho(t)$ and $\boldsymbol{W}_1^\top(0)\boldsymbol{U} = \epsilon \boldsymbol{V}$. Complete $\boldsymbol{V}_2$ and $\boldsymbol{U}_2$ to be orthogonal matrices, defined as $\boldsymbol{V} := [\boldsymbol{V}_1 \quad \boldsymbol{V}_2] \in \mathbb{R}^{d\times d}$ and $\boldsymbol{U} := [\boldsymbol{U}_1 \quad \boldsymbol{U}_2] \in \mathbb{R}^{m\times m}$. Then, we have

$$\begin{aligned}
&\boldsymbol{U}_1^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_2 - \boldsymbol{U}_1^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2 = \boldsymbol{U}_1^\top \boldsymbol{T}_1(t) \\
&\implies \|\boldsymbol{U}_1^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_2 - \boldsymbol{U}_1^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2\|_F = \|\boldsymbol{U}_1^\top \boldsymbol{T}_1(t)\|_F \leq \|\boldsymbol{T}_1(t)\|_F \leq \eta \cdot \rho(t),
\end{aligned}$$

where

$$\boldsymbol{U}_1^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2 = \epsilon \boldsymbol{U}_1^\top \boldsymbol{U}_2 = \mathbf{0}_{2K\times p} \implies \|\boldsymbol{U}_1^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2\|_F = 0.$$

Similarly, we have

$$\|\boldsymbol{V}_1^\top \boldsymbol{W}_1(t+1)\boldsymbol{U}_2 - \boldsymbol{V}_1^\top \boldsymbol{W}_1(t)\boldsymbol{U}_2\|_F = \|\boldsymbol{V}_1^\top \boldsymbol{T}_2(t)\|_F \leq \|\boldsymbol{T}_2(t)\|_F \leq \eta \cdot \rho(t),$$

where

$$\boldsymbol{V}_1^\top \boldsymbol{W}_1^\top(0)\boldsymbol{U}_2 = \epsilon \boldsymbol{V}_1^\top \boldsymbol{V}_2 = \mathbf{0}_{2K\times p} \implies \|\boldsymbol{V}_1^\top \boldsymbol{W}_1(0)\boldsymbol{U}_2\|_F = 0.$$

Finally, we have

$$\|\boldsymbol{U}_2^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_2 - \boldsymbol{U}_2^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2\|_F = \|\boldsymbol{U}_2^\top \boldsymbol{T}_1(t)\|_F \leq \eta \cdot \rho(t),$$

where

$$\boldsymbol{U}_2^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2 = \epsilon \boldsymbol{U}_2^\top \boldsymbol{U}_2 = \epsilon \boldsymbol{I}_p \implies \|\boldsymbol{U}_2^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2\|_F = \epsilon\sqrt{p}.$$

Putting everything together:

$$\boldsymbol{U}^\top \boldsymbol{W}_1(t)\boldsymbol{V} = \begin{bmatrix} \boldsymbol{U}_1^\top \\ \boldsymbol{U}_2^\top \end{bmatrix} \boldsymbol{W}_1(t) \begin{bmatrix} \boldsymbol{V}_1 & \boldsymbol{V}_2 \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_1^\top \boldsymbol{W}_1(t)\boldsymbol{V}_1 & \boldsymbol{U}_1^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2 \\ \boldsymbol{U}_2^\top \boldsymbol{W}_1(t)\boldsymbol{V}_1 & \boldsymbol{U}_2^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2 \end{bmatrix} := \begin{bmatrix} \widetilde{\boldsymbol{W}}_{1,1}(t) & \widetilde{\boldsymbol{W}}_{1,2}(t) \\ \widetilde{\boldsymbol{W}}_{1,3}(t) & \widetilde{\boldsymbol{W}}_{1,4}(t) \end{bmatrix},$$

where

$$\begin{aligned}
\|\widetilde{\boldsymbol{W}}_{1,2}(0)\|_F &= \|\boldsymbol{U}_1^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2\|_F = 0 \\
\|\widetilde{\boldsymbol{W}}_{1,3}(0)\|_F &= \|\boldsymbol{U}_2^\top \boldsymbol{W}_1(0)\boldsymbol{V}_1\|_F = 0, \quad \text{and} \\
\|\widetilde{\boldsymbol{W}}_{1,4}(0)\|_F &= \|\boldsymbol{U}_2^\top \boldsymbol{W}_1(0)\boldsymbol{V}_2\|_F = \epsilon\sqrt{p},
\end{aligned}$$

as well as

$$\begin{aligned}
\|\widetilde{\boldsymbol{W}}_{1,2}(t+1) - \widetilde{\boldsymbol{W}}_{1,2}(t)\|_F &= \|\boldsymbol{U}_1^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_2 - \boldsymbol{U}_1^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2\|_F \leq \eta \cdot \rho(t), \\
\|\widetilde{\boldsymbol{W}}_{1,3}(t+1) - \widetilde{\boldsymbol{W}}_{1,3}(t)\|_F &= \|\boldsymbol{U}_2^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_1 - \boldsymbol{U}_2^\top \boldsymbol{W}_1(t)\boldsymbol{V}_1\|_F \leq \eta \cdot \rho(t), \quad \text{and} \\
\|\widetilde{\boldsymbol{W}}_{1,4}(t+1) - \widetilde{\boldsymbol{W}}_{1,4}(t)\|_F &= \|\boldsymbol{U}_2^\top \boldsymbol{W}_1(t+1)\boldsymbol{V}_2 - \boldsymbol{U}_2^\top \boldsymbol{W}_1(t)\boldsymbol{V}_2\|_F \leq \eta \cdot \rho(t).
\end{aligned}$$

Dividing everything by $\sqrt{p}$ completes the proof. □

E.5 Proof of Corollary 3.7

Here, we provide a proof of Corollary 3.7. We re-state the result here for convenience.

Corollary 3.7. *Suppose the setting of Theorem 3.5 and Assumption 3.6 hold. Then, for some constants C_1 and C_2 that depend on G_1, G_2, G_3, β, μ, ϵ, $\phi'(0)$, $\sigma_1(\boldsymbol{W}_2)$, and $\sigma_K(\boldsymbol{W}_2^\top \boldsymbol{Y}\boldsymbol{X}^\top)$, we have*

$$\sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\frac{A^2(t)}{p}+\sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)} \le \begin{cases} C_1\cdot\epsilon & t=0 \\ C_2\cdot(1-\Theta(\eta))^{\Theta(t)}\cdot\left(1-(1-\Theta(\eta))^{\Theta(t)}\right) & t>0 \end{cases}.$$

Proof. Define

$$\rho(t) := \sqrt{\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot A^2(t) + p\cdot\sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)}$$

and

$$\delta(t) := \frac{\gamma_L\cdot\beta\cdot\sigma_1\left(\boldsymbol{W}_2\right)\cdot\sqrt{2\ell(0)}\cdot\Theta(1)\cdot\left(1-(1-\Theta(\eta))^{\Theta(t)}\right)}{\phi'(0)\cdot\sigma_K\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\boldsymbol{X}^\top\right)-r(\epsilon)-\sigma_{K+1}\left(\boldsymbol{G}_1(t)\right)}.$$

We now upper bound $\rho(t)$. First, note that at $t=0$, we have $A^2(t)=0$, and $p\cdot\sigma_{K+1}^2\left(\boldsymbol{G}_1(0)\right) = p\cdot r^2(\epsilon)$ via Lemma E.7. Thus, we focus on $t\ge 1$. Recall by definition, $\boldsymbol{V}\in\mathcal{R}\left(\boldsymbol{R}_{1,2}(0)\right)$. From Lemma E.8, we have

$$A(t)\le\delta(t) \implies \sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot A^2(t)\le\sigma_1^2\left(\boldsymbol{G}_1(t)\right)\cdot\delta^2(t).$$

Then, we have

$$\frac{\sigma_1\left(\boldsymbol{G}_1(t)\right)}{\sigma_1\left(\boldsymbol{G}_1(0)\right)}\le G_2\cdot\frac{\|\boldsymbol{G}_1(t)\|_F}{\|\boldsymbol{G}_1(0)\|_F}\le G_2\cdot G_1\cdot(1-\Theta(\eta))^{\Theta(t)}$$

$$\overset{(i)}{\implies}\sigma_1(\boldsymbol{G}_1(t))\le\bar{G}\cdot(1-\Theta(\eta))^{\Theta(t)}\cdot\sigma_1\left(\boldsymbol{G}_1(0)\right)\overset{(ii)}{\le}\bar{G}\cdot(1-\Theta(\eta))^{\Theta(t)}\cdot\delta(t)\cdot\left(\phi'(0)\cdot\sigma_1\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\boldsymbol{X}^\top\right)+r(\epsilon)\right),$$

where $\bar{G}:=G_2\cdot G_1$, (i) is from Assumption 3.6, and (ii) is from Lemma E.7. In summary,

$$\rho(t)\le\begin{cases}\dfrac{r(\epsilon)\cdot\sqrt{p}}{\sqrt{\bar{G}^2\cdot(1-\Theta(\eta))^{\Theta(t)}\cdot\delta^2(t)\cdot\left(\phi'(0)\cdot\sigma_1\left(\boldsymbol{W}_2^\top\boldsymbol{Y}\boldsymbol{X}^\top\right)+r(\epsilon)\right)^2+\sigma_{K+1}^2\left(\boldsymbol{G}_1(t)\right)\cdot p}} & t=0 \\ & t\ge 1\end{cases}.$$

Dividing both sides by $\sqrt{p}$ completes the proof. □

F Proof of Proposition D.1

In this section, we provide a proof of Proposition D.1. We use the same notation as in Appendix E. We re-state Proposition D.1 below for convenience.

Proposition F.1 (Proposition D.1 re-stated)**.** *Suppose $d=N$, $\boldsymbol{W}_1(0)_{ij}\overset{iid}{\sim}\mathcal{N}\left(0,\frac{\epsilon^2}{m}\right)$, and $\phi=$ ReLU. With probability at least $1-\delta$ w.r.t. the randomness in $\boldsymbol{W}_1(0)$,*

$$\sigma_{d-K}\left(\boldsymbol{G}_1(0)\right)\ge\sqrt{\frac{\lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right)}{4}-\left(\frac{R'}{6}\cdot\log\left(\frac{2(d-K)}{\delta}\right)+\sqrt{2\cdot\log\left(\frac{2(d-K)}{\delta}\right)\cdot\frac{R'\cdot\lambda_{\max}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right)}{16}}\right)},$$

where $\boldsymbol{V}$ is an orthonormal basis for $\mathcal{N}\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)$, $\boldsymbol{D}:=\operatorname{diag}\left(\left\|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{:,1}\right\|_2^2,\ldots,\left\|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{:,N}\right\|_2^2\right)$, and $R':=\max_{i\in[m]}\left\|\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\right)_{i,:}\right\|_2^2$.

Proof. Recall that

$$\boldsymbol{G}_1(0)=\left(\boldsymbol{W}_2^\top\boldsymbol{\Delta}_2(0)\odot\phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)\right)\boldsymbol{X}^\top.$$

Since $d = N$ and $\boldsymbol{X}\boldsymbol{X}^\top = \boldsymbol{I}_d$, $\boldsymbol{X}$ is exactly an orthogonal matrix. Therefore, we have

$$\sigma_{d-K}\left(\boldsymbol{G}_1(0)\right) = \sigma_{d-K}\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)\right),$$

so we aim to lower bound $\sigma_{K+1}\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)\right)$. First, note that $\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)$ is rank-K. Next, define $\boldsymbol{M} := \phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)$. Notice that $\boldsymbol{M}$ contains iid Bernoulli entries with probability parameter $q = 0.5$, as $\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)_{ij} \overset{iid}{\sim} \mathcal{N}\left(0, \frac{\epsilon^2}{m}\right)$, and $\phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)_{ij} = \mathbb{I}\left[\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)_{ij} > 0\right]$, where $\mathbb{I}$ denotes the indicator function. Finally, note $\boldsymbol{M} = q\mathbf{1}_m\mathbf{1}_N^\top + \left(\boldsymbol{M} - q\mathbf{1}_m\mathbf{1}_N^\top\right)$. Combining everything yields

$$\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \boldsymbol{M} = \underbrace{q\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)}_{:=\boldsymbol{S}} + \underbrace{\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \left(\boldsymbol{M} - q\mathbf{1}_m\mathbf{1}_N^\top\right)}_{:=\boldsymbol{N}}.$$

Therefore,

$$\begin{aligned}
&\sigma_{d-K}\left(\boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0) \odot \phi'\left(\boldsymbol{W}_1(0)\boldsymbol{X}\right)\right) = \sigma_{d-K}\left(\boldsymbol{S} + \boldsymbol{N}\right) \\
&\overset{(i)}{=} \max_{\mathcal{T}\subset\mathbb{R}^N : \dim(\mathcal{T}) = d-K} \min_{\boldsymbol{x}\in\mathcal{T}, \|\boldsymbol{x}\|_2 = 1} \left\|\left(\boldsymbol{S} + \boldsymbol{N}\right)\boldsymbol{x}\right\|_2 \\
&\overset{(ii)}{\geq} \min_{\boldsymbol{x}\in\mathcal{N}(\boldsymbol{S}), \|\boldsymbol{x}\|_2 = 1} \left\|\left(\boldsymbol{S} + \boldsymbol{N}\right)\boldsymbol{x}\right\|_2 = \min_{\boldsymbol{x}\in\mathcal{N}(\boldsymbol{S}), \|\boldsymbol{x}\|_2 = 1} \left\|\boldsymbol{N}\boldsymbol{x}\right\|_2.
\end{aligned}$$

where (i) is from the Courant-Fischer Min-Max Theorem for singular values, i.e., [51, Theorem 4.2.2] applied to $\left(\boldsymbol{S} + \boldsymbol{N}\right)^\top \left(\boldsymbol{S} + \boldsymbol{N}\right)$, and (ii) is using the fact that $\dim\left(\mathcal{N}\left(\boldsymbol{S}\right)\right) = d - K$. Let $\boldsymbol{V} \in \mathbb{R}^{N\times(d-K)}$ be an orthonormal basis for $\mathcal{N}\left(\boldsymbol{S}\right)$. Then,

$$\min_{\boldsymbol{x}\in\mathcal{N}(\boldsymbol{S}), \|\boldsymbol{x}\|_2 = 1} \left\|\boldsymbol{N}\boldsymbol{x}\right\|_2 = \min_{\boldsymbol{z}\in\mathbb{R}^{d-K} : \|\boldsymbol{z}\|_2 = 1} \left\|\boldsymbol{N}\boldsymbol{V}\boldsymbol{z}\right\|_2 = \sigma_{\min}\left(\boldsymbol{N}\boldsymbol{V}\right).$$

Now, it suffices to lower bound $\sigma_{\min}\left(\boldsymbol{N}\boldsymbol{V}\right)$. Notice $\boldsymbol{N}\boldsymbol{V} \in \mathbb{R}^{m\times(d-K)}$ with $m \geq N > d - K$. Therefore, $\sigma_{\min}^2\left(\boldsymbol{N}\boldsymbol{V}\right) = \lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right)$, so we aim to lower bound $\lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right)$. Let $\widetilde{\boldsymbol{M}} := \boldsymbol{M} - q\mathbf{1}_m\mathbf{1}_N^\top$ and $\boldsymbol{E} = \boldsymbol{W}_2^\top \boldsymbol{\Delta}_2(0)$, so $\boldsymbol{N} = \boldsymbol{E} \odot \widetilde{\boldsymbol{M}}$. Then,

$$\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V} = \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right] + \left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right),$$

and so by Weyl's inequality [55],

$$\begin{aligned}
&\lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right) = \lambda_{\min}\Big(\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right] + \left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right)\Big) \\
&\geq \lambda_{\min}\left(\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right) + \lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right) \\
&\geq \underbrace{\lambda_{\min}\left(\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right)}_{(a)} - \underbrace{\sigma_1\left(\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right)}_{(b)}
\end{aligned}$$

We analyze (a) and (b) individually. To do so, we first determine $\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]$.

Expectation of $\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}$. First, since $\boldsymbol{V}$ is deterministic w.r.t. $\boldsymbol{N}$, we have

$$\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right] = \boldsymbol{V}^\top\mathbb{E}_{\widetilde{\boldsymbol{M}}}\left[\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)^\top\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)\right]\boldsymbol{V}.$$

Therefore,

$$\begin{aligned}
&\left(\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)^\top\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)\right)_{ij} = \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)^\top_{:,i}\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{:,j} = \sum_{k=1}^m \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{ki}\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{kj} = \sum_{k=1}^m \boldsymbol{E}_{ki}\boldsymbol{E}_{kj}\cdot\widetilde{\boldsymbol{M}}_{ki}\widetilde{\boldsymbol{M}}_{kj} \\
&\implies \mathbb{E}_{\widetilde{\boldsymbol{M}}}\left[\left(\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)^\top\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)\right)_{ij}\right] = \sum_{k=1}^m \boldsymbol{E}_{ki}\boldsymbol{E}_{kj}\cdot\mathbb{E}\left[\widetilde{\boldsymbol{M}}_{ki}\widetilde{\boldsymbol{M}}_{kj}\right] = \begin{cases} 0 & i \neq j \\ q^2\cdot\|\boldsymbol{E}_{:,i}\|_2^2 & i = j \end{cases} \\
&\implies \mathbb{E}_{\widetilde{\boldsymbol{M}}}\left[\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)^\top\left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)\right] = q^2\cdot\mathrm{diag}\left(\|\boldsymbol{E}_{:,1}\|_2^2, \ldots, \|\boldsymbol{E}_{:,N}\|_2^2\right) := q^2\cdot\boldsymbol{D} \\
&\implies \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right] = q^2\cdot\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V},
\end{aligned}$$

where $\boldsymbol{A}_{:,i}$ denotes the i^{th} column in the matrix $\boldsymbol{A}$. Thus, we have

$$(a) = \lambda_{\min}\left(\mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top\boldsymbol{N}^\top\boldsymbol{N}\boldsymbol{V}\right]\right) = q^2\cdot\lambda_{\min}\left(\boldsymbol{V}^\top\boldsymbol{D}\boldsymbol{V}\right),$$

so now we aim to lower bound (b).

Bounding (b). We aim to use the Matrix Bernstein inequality, i.e., [57, Theorem 6.1.1] to bound (b). First, define $\overline{\boldsymbol{N}} := \boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N}\boldsymbol{V}\right]$. Note we can write $\overline{\boldsymbol{N}}$ as a sum of independent, zero-mean, symmetric random matrices:

$$\boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N}\boldsymbol{V} - \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N}\boldsymbol{V}\right] = \sum_{i=1}^{m} \boldsymbol{V}^\top \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:} \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:}^\top \boldsymbol{V} - \mathbb{E}\left[\boldsymbol{V}^\top \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:} \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:}^\top \boldsymbol{V}\right]$$

where $\boldsymbol{A}_{i,:}$ denotes the i^{th} row in $\boldsymbol{A}$, but written as a column vector. The terms in the sum are independent since the rows of $\widetilde{\boldsymbol{M}}$ are independent. Next, define $\boldsymbol{n}_i := \boldsymbol{V}^\top \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:}$ and $\overline{\boldsymbol{N}}_i = \boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]$. Then, we have $\overline{\boldsymbol{N}} = \sum_{i=1}^{m} \left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right) = \sum_{i=1}^{m} \overline{\boldsymbol{N}}_i$. To leverage the Matrix Bernstein inequality, we must bound $\sigma_1\left(\overline{\boldsymbol{N}}_i\right)$ and $\nu\left(\overline{\boldsymbol{N}}\right) := \sigma_1\left(\mathbb{E}\left[\overline{\boldsymbol{N}}^2\right]\right)$.

Bounding $\sigma_1\left(\overline{\boldsymbol{N}}_i\right)$. We first bound $\sigma_1\left(\overline{\boldsymbol{N}}_i\right)$. First, recall $\overline{\boldsymbol{N}}_i = \boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]$, and notice it is the difference of two symmetric matrices, and thus is itself symmetric. Therefore,

$$\sigma_1\left(\overline{\boldsymbol{N}}_i\right) = \max\left\{\lambda_{\max}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right), \left|\lambda_{\min}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right)\right|\right\}.$$

Next, $\boldsymbol{n}_i \boldsymbol{n}_i^\top \succeq 0$ with probability 1, which implies $\mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right] \succeq 0$. Therefore,

$$\lambda_{\max}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right) \leq \lambda_{\max}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top\right) = \|\boldsymbol{n}_i\|_2^2,$$

and

$$\lambda_{\min}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right) \geq \lambda_{\min}\left(-\mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right) = -\lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right)$$

$$\implies \left|\lambda_{\min}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right)\right| \leq \lambda_{\max}\left(\mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right) \overset{(i)}{\leq} \mathbb{E}\left[\lambda_{\max}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top\right)\right] = \mathbb{E}\left[\|\boldsymbol{n}_i\|_2^2\right],$$

where (i) is from Jensen's inequality. Thus, we focus on analyzing $\|\boldsymbol{n}_i\|_2^2$. Note

$$\boldsymbol{n}_i = \boldsymbol{V}^\top \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:} = \boldsymbol{V}^\top \mathrm{diag}\left(\boldsymbol{E}_{i,:}\right) \widetilde{\boldsymbol{M}}_{i,:},$$

where $\mathrm{diag}(\boldsymbol{a})$ is a diagonal matrix whose entries are the elements of the vector $\boldsymbol{a}$. Therefore,

$$\|\boldsymbol{n}_i\|_2^2 = \boldsymbol{n}_i^\top \boldsymbol{n}_i = \widetilde{\boldsymbol{M}}_{i,:}^\top \mathrm{diag}\left(\boldsymbol{E}_{i,:}\right)^\top \boldsymbol{V}\boldsymbol{V}^\top \mathrm{diag}\left(\boldsymbol{E}_{i,:}\right) \widetilde{\boldsymbol{M}}_{i,:} \overset{(ii)}{\leq} \widetilde{\boldsymbol{M}}_{i,:}^\top \mathrm{diag}\left(\boldsymbol{E}_{i,:}\right)^\top \mathrm{diag}\left(\boldsymbol{E}_{i,:}\right) \widetilde{\boldsymbol{M}}_{i,:} = q^2 \|\boldsymbol{E}_{i,:}\|_2^2$$

$$\implies \|\boldsymbol{n}_i\|_2^2 \leq q^2 \cdot \max_{i \in [m]} \|\boldsymbol{E}_{i,:}\|_2^2 := q^2 \cdot R'$$

with probability 1, where (ii) is because $\boldsymbol{V}\boldsymbol{V}^\top \preceq \boldsymbol{I}_N$. Therefore, for all $i \in [m]$,

$$\begin{aligned}\sigma_1\left(\overline{\boldsymbol{N}}_i\right) &= \max\left\{\lambda_{\max}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right), \left|\lambda_{\min}\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]\right)\right|\right\} \\ &\leq \max\left\{\|\boldsymbol{n}_i\|_2^2, \mathbb{E}\left[\|\boldsymbol{n}_i\|_2^2\right]\right\} \leq q^2 \cdot R'\end{aligned}$$

with probability 1.

Bounding $\nu\left(\overline{\boldsymbol{N}}\right)$. Next, we bound $\nu\left(\overline{\boldsymbol{N}}\right) = \sigma_1\left(\mathbb{E}\left[\overline{\boldsymbol{N}}^2\right]\right)$. Since each $\overline{\boldsymbol{N}}_i$ is zero mean, we have

$$\sigma_1\left(\mathbb{E}\left[\overline{\boldsymbol{N}}^2\right]\right) = \sigma_1\left(\sum_{i=1}^{m} \mathbb{E}\left[\overline{\boldsymbol{N}}_i^2\right]\right).$$

For an arbitrary $i \in [m]$:

$$\mathbb{E}\left[\overline{\boldsymbol{N}}_i^2\right] = \mathbb{E}\left[\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i\right]\right)^2\right] = \mathbb{E}\left[\left(\boldsymbol{n}_i \boldsymbol{n}_i^\top\right)^2\right] - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]^2$$

$$= \mathbb{E}\left[\|\boldsymbol{n}_i\|_2^2 \cdot \boldsymbol{n}_i \boldsymbol{n}_i^\top\right] - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]^2 \overset{(ii)}{\preceq} q^2 \cdot R' \cdot \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right] - \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]^2 \preceq q^2 \cdot R' \cdot \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right]$$

where (ii) is from the fact that $\|\boldsymbol{n}_i\|_2^2 \leq q^2 \cdot R'$ with probability 1. Therefore,

$$\sum_{i=1}^{m} \mathbb{E}\left[\overline{\boldsymbol{N}}_i^2\right] \preceq q^2 \cdot R' \cdot \sum_{i=1}^{m} \mathbb{E}\left[\boldsymbol{n}_i \boldsymbol{n}_i^\top\right] \overset{(iii)}{=} q^2 \cdot R' \cdot \mathbb{E}_{\boldsymbol{N}}\left[\boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N}\boldsymbol{V}\right] = q^4 \cdot R' \cdot \boldsymbol{V}^\top \boldsymbol{D}\boldsymbol{V},$$

where (iii) is because $\boldsymbol{n}_i := \boldsymbol{V}^\top \left(\boldsymbol{E} \odot \widetilde{\boldsymbol{M}}\right)_{i,:} = \boldsymbol{V}^\top \boldsymbol{N}_{i,:}$, and so $\sum_{i=1}^{m} \boldsymbol{n}_i \boldsymbol{n}_i^\top = \boldsymbol{V}^\top \boldsymbol{N}^\top \boldsymbol{N} \boldsymbol{V}$. As a result,

$$\sigma_1\left(\mathbb{E}\left[\overline{\boldsymbol{N}}^2\right]\right) = \sigma_1\left(\sum_{i=1}^{m} \mathbb{E}\left[\overline{\boldsymbol{N}}_i^2\right]\right) \leq q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right)$$

Matrix Bernstein inequality. From [57, Theorem 6.1.1], for all $\tau \geq 0$:

$$\begin{aligned}\mathbb{P}\left(\sigma_1\left(\overline{\boldsymbol{N}}\right) \geq \tau\right) &\leq 2(d-K) \cdot \exp\left(\frac{-\tau^2/2}{\nu\left(\overline{\boldsymbol{N}}\right) + \max_i \sigma_1\left(\overline{\boldsymbol{N}}_i\right) \cdot \tau/3}\right) \\ &\leq 2(d-K) \cdot \exp\left(\frac{-\tau^2/2}{q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right) + q^2 \cdot R' \cdot \tau/3}\right) \leq \delta,\end{aligned}$$

which implies

$$\begin{aligned}\tau \geq & \frac{q^2 \cdot R'}{3} \cdot \log\left(\frac{2(d-K)}{\delta}\right) \\ & + \sqrt{\left(\frac{q^2 \cdot R'}{3} \cdot \log\left(\frac{2(d-K)}{\delta}\right)\right)^2 + 2 \cdot \log\left(\frac{2(d-K)}{\delta}\right) \cdot \left(q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right)\right)}.\end{aligned}$$

Using $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$, a sufficient condition is

$$\tau \geq \frac{2q^2 \cdot R'}{3} \cdot \log\left(\frac{2(d-K)}{\delta}\right) + \sqrt{2 \cdot \log\left(\frac{2(d-K)}{\delta}\right) \cdot \left(q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right)\right)}.$$

Therefore, with probability at least $1 - \delta$,

$$\sigma_1\left(\overline{\boldsymbol{N}}\right) \leq \frac{2q^2 \cdot R'}{3} \cdot \log\left(\frac{2(d-K)}{\delta}\right) + \sqrt{2 \cdot \log\left(\frac{2(d-K)}{\delta}\right) \cdot \left(q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right)\right)}.$$

Final result. Putting everything together, for any $\delta \in (0,1)$, with probability at least $1 - \delta$,

$$\begin{aligned}\sigma_{d-K}\left(\boldsymbol{G}_1(0)\right) \geq \Bigg(q^2 \cdot \lambda_{\min}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right) - \Bigg(&\frac{2q^2 \cdot R'}{3} \cdot \log\left(\frac{2(d-K)}{\delta}\right) \\ &+ \sqrt{2 \cdot \log\left(\frac{2(d-K)}{\delta}\right) \cdot \left(q^4 \cdot R' \cdot \lambda_{\max}\left(\boldsymbol{V}^\top \boldsymbol{D} \boldsymbol{V}\right)\right)}\Bigg)\Bigg)^{1/2}.\end{aligned}$$

Substituting $q = 0.5$ completes the proof.

□

G Low-Rank MLP Ablations

In this section, we provide ablation studies on the initialization of the $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ factors, as well as the width parameter r. We ran all experiments in this section using `PyTorch` on an NVIDIA A40 GPU.

G.1 Subspace Initialization

From Figures 5 and 6 in Sections 5.1.1 and 5.1.2 respectively, we observe that if we train the low-rank MLP in (7) after initializing $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ as random semi-orthogonal matrices (green), then (S)GD gets stuck in a much worse local minimum compared to the other MLPs (blue and orange). In this section, we ablate the initialization of $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$.

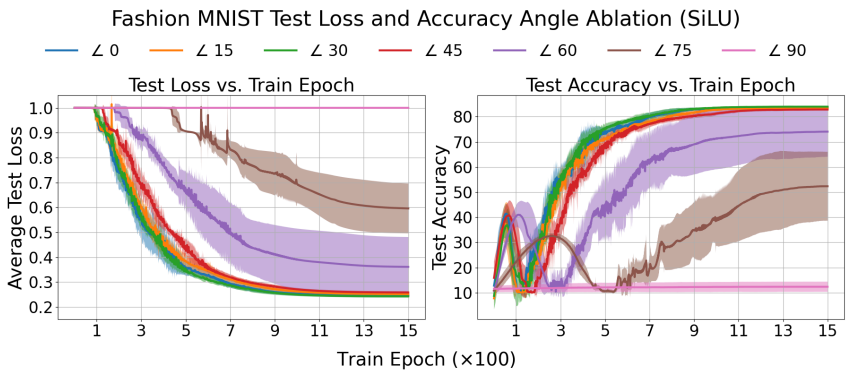


Figure 14: For all low-rank MLPs with $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ initialized close to ($\xi \leq 45$ degrees) $\widetilde{\boldsymbol{U}}_{big}$ and $\widetilde{\boldsymbol{V}}_{big}$, GD reaches similar solutions. Otherwise, GD gets stuck in progressively worse local minima as ξ increases.

Experimental details. We repeated the experiments as described in Section 5.1.1. We used the exact same training algorithm (full batch GD), hyperparameters, and loss function (total squared-error loss). However, we only used $\phi = \text{SiLU}$, and we varied the initialization of $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ as follows. First, we define $\widetilde{\boldsymbol{U}}_{big} \in \mathbb{R}^{d\times r}$ and $\widetilde{\boldsymbol{V}}_{big} \in \mathbb{R}^{d\times r}$ to be the result of the $\mathcal{S}_{big}$ initialization scheme described in Section 5, where we set $r = 2K = 20$. Next, we define $\widetilde{\boldsymbol{U}}^{\perp}_{big}, \widetilde{\boldsymbol{V}}^{\perp}_{big} \in \mathbb{R}^{d\times r}$ to be completely orthogonal to $\widetilde{\boldsymbol{U}}_{big}$ and $\widetilde{\boldsymbol{V}}_{big}$, respectively. Finally, we initialized $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ to be some angle ξ degrees away from $\widetilde{\boldsymbol{U}}_{big}$ and $\widetilde{\boldsymbol{V}}_{big}$ via

$$\widetilde{\boldsymbol{U}} = \cos(\xi) \cdot \widetilde{\boldsymbol{U}}_{big} + \sin(\xi) \cdot \widetilde{\boldsymbol{U}}^{\perp}_{big} \quad \text{and} \quad \widetilde{\boldsymbol{V}} = \cos(\xi) \cdot \widetilde{\boldsymbol{V}}_{big} + \sin(\xi) \cdot \widetilde{\boldsymbol{V}}^{\perp}_{big}. \tag{15}$$

In our experiments, we swept through $\xi \in \{0, 15, 30, 45, 60, 75, 90\}$ degrees.

Results. Figure 14 shows the resulting test loss and accuracy curves during training averaged over 5 trials for each ξ. The test loss and accuracy curves are nearly identical for all $\xi \in \{0, 15, 30, 45\}$ degrees, although at $\xi = 45$, GD required slightly more iterations for the test loss to begin decreasing from its initial value. The performance noticeably deteriorates when $\xi > 45$. For $\xi \in \{60, 75\}$ degrees, the test loss still decreases from their initial values, but GD gets stuck in noticeably worse local minima compared to when $\xi < 45$. Finally, at $\xi = 90$, the initialized $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ are completely orthogonal to $\widetilde{\boldsymbol{U}}_{big}$ and $\widetilde{\boldsymbol{V}}_{big}$. The test loss stays at its initial value for all training epochs, and the low-rank MLP does not escape the random guessing stage. Overall, if $\widetilde{\boldsymbol{U}}$ / $\widetilde{\boldsymbol{V}}$ in the low-rank MLP are initialized to be closer to $\widetilde{\boldsymbol{U}}_{big}$ / $\widetilde{\boldsymbol{V}}_{big}$ than their orthogonal complements, then the low-rank MLP can eventually nearly match the performance of the full-rank counterpart. Otherwise, GD gets stuck in progressively worse local minima as $\xi \to 90$ degrees.

G.2 Width Ablation

From the classifier-only results in Section 5.1.2 (Figure 6), there is a $5 - 10\%$ gap in test accuracy between the low-rank MLP with $\mathcal{S}_{big}$ initialization (orange) and the fully parameterized MLP (blue). In this section, we investigate how much does this gap close by increasing the width r.

Experimental details. We repeated the classifier-only training experiments as described in Section 5.1.2 with both $\phi = \text{GELU}$ and $\phi = \text{SiLU}$, again using the exact same training algorithm (SGD with momentum), hyperparameters, and loss function (cross-entropy loss). However, now we vary the width $r \in \{2K, 4K, 8K, 16K\}$ in the low-rank MLP in (7). We initialized $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ using the $\mathcal{S}_{big}$ initialization scheme as described in Section 5. For $r > 2K$, we initialized the last $r - 2K$ columns of $\widetilde{\boldsymbol{U}}$ and $\widetilde{\boldsymbol{V}}$ to lie in the orthogonal complements of their first $2K$ columns.

Results. Figure 15 shows the test accuracy at every training epoch for each r. Even going from $r = 2K$ to $r = 4K$ noticeably closes the gap between the low-rank and fully-parameterized MLPs,

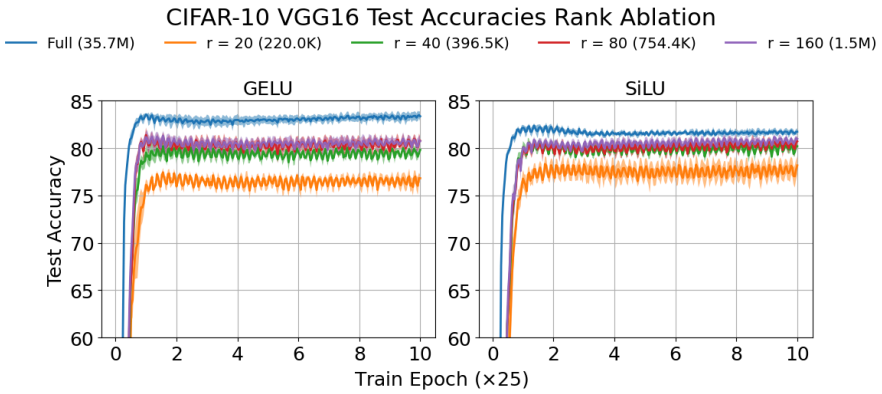


Figure 15: Increasing the width r of the low-rank MLP initially leads to a noticeable increase in test accuracy, but further increasing r leads to diminishing returns in test accuracy gains. Here, we only trained the classifier head in VGG-16, and kept the convolutional layers frozen at their ImageNet pre-trained weights.

especially for the SiLU classifier. However, continuing to double r further leads to diminishing returns in performance gains. Nevertheless, with the appropriate initialization on $\widehat{\boldsymbol{U}}$ and $\widehat{\boldsymbol{V}}$, a width of $r = \Theta(K)$ appears to be sufficient in achieving a test accuracy that is only about $2 - 3\%$ lower than that of the fully parameterized classifier head.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state we show low-rank training dynamics arise in MLP architectures with smooth activation functions, which is the sole focus of the paper.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the limitations of our work in the Conclusion section. We explicitly state our theoretical assumptions and discuss how strong they are. Furthermore, in the Appendix, we provide empirical support for our stronger assumptions.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: We explicitly state our setting of interest and theoretical assumptions in the main text. We provide proofs of all theoretical results in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: In the Appendix, we provide details for all experiments on toy synthetic datasets. In the main text, we provide all details on experiments done on real data, and explicitly state the datasets we used. We also provide a link to our code in the main text.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We provide a link to our code in the main paper. All datasets used in this work are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: In the main text and Appendix, we state the datasets, hyperparameters, and optimizers we used in all experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: In all of our experimental results, we provide 1-σ error bars over several trials, although these error bars may not be visible in some plots.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: In the main text and Appendix, we state the type of GPUs we used in our experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This paper does not include experiments with human subjects, and we do not believe this paper has any societal impacts.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: We do not believe there are any societal impacts in our work.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We believe this paper does not pose any of these risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We provide references for all real datasets we used.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper does not introduce any new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This paper does not include any experiments with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: This paper does not include any experiments with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs are not a part of the core methodology in this work. However, LLMs were used to help write and clean up code, as well as aid writing and editing. The authors take full responsibility for all contents in this paper.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.