One Rank at a Time: Cascading Error Dynamics in Sequential Learning

Anonymous authors
Paper under double-blind review

Abstract

Sequential learning –where complex tasks are broken down into simpler, hierarchical components– has emerged as a paradigm in AI. This paper views sequential learning through the lens of low-rank linear regression, focusing specifically on how errors propagate when learning rank-1 subspaces sequentially. We present an analysis framework that decomposes the learning process into a series of rank-1 estimation problems, where each subsequent estimation depends on the accuracy of previous steps. Our aim is explanatory rather than comparative: we analyze error propagation and derive compute allocation guidance without claiming superiority over joint or one-shot training. Our contribution is a characterization of the error propagation in this sequential process, establishing bounds on how errors –e.g., due to limited computational budgets and finite precision– affect the overall model accuracy. We prove that these errors compound in predictable ways, with implications for both algorithmic design and stability guarantees.

1 Introduction

Sequential learning (Barto & Mahadevan, 2003; Pateria et al., 2021; Sahni et al., 2017; Ostapenko et al., 2024; Sordoni et al., 2024; Page-Caccia et al., 2024) is a concept found in cognitive science that posits that learning could be structured as a series of stages or levels (Shapere, 1964; Richardson, 2019; Okano et al., 2000; Carey & Markman, 1999; Ericsson et al., 1993; Council et al., 2000). Sequential learning is relevant when the notion of "skills" is orthogonal or correlated to each other, or even layered hierarchically (Farajtabar et al., 2020; Chaudhry et al., 2020). For instance, in a multitask learning environment (Caruana, 1997; Ruder, 2017; Andrychowicz et al., 2016; Crawshaw, 2020), basic skills might serve for common tasks, while specialized skills might be required for specific tasks.

Fully understanding sequential learning is an open question, even for simple models: researchers study not just how AI systems learn, but why they fail to learn when they do, and under what conditions they can learn better (Zhang et al., 2017; Bjorck et al., 2018; Baldi & Sadowski, 2013; Sun, 2020; Liu et al., 2020; Zhou et al., 2022; Wang et al., 2021; Yang et al., 2020). Just to provide a non-exhaustive list of recent efforts: Zhao et al. (2024a) presents a sequential learning strategy on videos and text for sentiment analysis, where learning features sequentially –from simpler to more complex– led to better performance. McAlister et al. (2024) combines deep learning with symbolic AI to tackle sequential learning tasks. Bian et al. (2023) deals with recommendation systems, such as those used by Netflix or Amazon and propose a method for these systems to learn from sequences of user interactions with multiple types of data (e.g., text, images, videos) for better recommendations.

Such efforts in understanding the empirical performance of sequential strategy necessitate a theoretical foundation of the training scheme. A line of prior works tackle the problem of sequential learning from a rank-incremental perspective. In particular, (Jin et al., 2023) proves the behavior of rank-incremental recovery in the setting of matrix sensing. A later work (Kwon et al., 2024) studies a similar behavior for the task of deep linear networks, and shows that the sequential learning of the principal components of the model can provably lead to an efficient compression algorithm. In this paper, we focus on the theoretical aspects underpinning the explicitly enforced sequential learning algorithms.

Our setting. We focus on low-rank subspaces as feature representations (Sanyal et al., 2018). Low-rank models are mostly compelling due to their interpretable solutions that capture influential factors in the data (Jolliffe, 1995; Koren et al., 2009; Vidal & Favaro, 2014). Assuming the simple low-rank linear regression (Candes & Recht, 2012; Recht et al., 2010; Rohde & Tsybakov, 2011; Ha & Foygel Barber, 2015) and given input $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, the goal is to approximate the relationship between a dependent variable $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathbb{R}^{m\times n}$ and independent unknown variables $\mathbf{B} \in \mathbb{R}^{m\times r}$ and $\mathbf{A} \in \mathbb{R}^{r\times d}$ that result in a lower rank $r \ll \min(m, d)$ matrix $\mathbf{W} = \mathbf{BA} \in \mathbb{R}^{m\times d}$ such that:

$$\mathbf{Y} \approx \mathbf{BAX} = \mathbf{WX}.$$

Here, $(\mathbf{x}_i, \mathbf{y}_i)$ represents a data sample of a dataset $\mathcal{D} := \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$. Conceptually, the matrix $\mathbf{W}$ projects the original d features onto an m-dimensional space, given that first $\mathbf{x}_i$ is passed through an r-dimensional "funnel", thus reducing the dimensionality and complexity of the model.

In view of this, we utilize linear low-rank regression as a framework to study theoretically sequential learning processes, and how errors propagate through sequential rank-1 subspace learning. Algorithmically, the approach we consider relates to deflation in PCA (Hotelling, 1933; Mackey, 2008; Zhang, 2006; Sriperumbudur et al., 2007; Saad, 1988; Danisman et al., 2014). That is, find a rank-1 estimate based on the deflated target $\mathbf{Y}_k$, as in:

$$\mathbf{a}_k, \mathbf{b}_k = \underset{\mathbf{a}\in\mathbb{R}^d, \mathbf{b}\in\mathbb{R}^m}{\operatorname{argmin}} \tfrac{1}{2} \left\|\mathbf{Y}_k - \mathbf{b}\mathbf{a}^\top \mathbf{X}\right\|_F^2. \tag{1}$$

Our approach starts off with $\mathbf{Y}_1 = \mathbf{Y}$ and we obtain $\mathbf{a}_1, \mathbf{b}_1$. The matrix $\mathbf{Y}_1$ is further processed to lie in a "subspace" where the contributions of $(\mathbf{a}_1, \mathbf{b}_1)$ are removed: $\mathbf{Y}_2 := \mathbf{Y}_1 - \mathbf{b}_1\mathbf{a}_1^\top \mathbf{X}$. This process is repeated by applying sequentially rank-1 updates on the deflated matrix, which leads to an approximation of the second pair $(\mathbf{a}_2, \mathbf{b}_2)$, and so on. Overall:

$$\mathbf{Y}_1 = \mathbf{Y}; \quad (\mathbf{a}_k, \mathbf{b}_k) = \texttt{rank-1}(\mathbf{Y}_k, \mathbf{X}, t); \quad \mathbf{Y}_{k+1} := \mathbf{Y}_k - \mathbf{b}_k\mathbf{a}_k^\top \mathbf{X}, \tag{2}$$

where $\texttt{rank-1}(\mathbf{Y}_k, \mathbf{X}, t)$ returns an approximation of a rank-1 estimate in equation 1 that minimizes the mean-squared error $\frac{1}{2}\left\|\mathbf{Y}_k - \mathbf{b}\mathbf{a}^\top \mathbf{X}\right\|_F^2$, using t iterations. We then estimate the subsequent subspaces by running the same rank-1 algorithm repetitively.

Motivation. While subspace tracking have a long history (see (Yang, 1995; Vaswani et al., 2018; Balzano et al., 2018; Peng et al., 2023) and references to these), low-rank subspaces have recently gained attention due to emerging applications in AI. Parameter-efficient fine-tuning (PEFT) methods, such as LoRA (Hu et al., 2021), have demonstrated that representing weight updates as low-rank matrices can effectively adapt large language models, while maintaining performance. We note here that many practical pipelines use sequential rank-1 updates because the procedure: *i*) aligns with deflation-style estimators long used in PCA and subspace tracking; *ii*) mirrors how low-rank adapters are deployed incrementally across tasks or capacity budgets, often without knowing the optimal rank a priori; and *iii*) fits streaming or compute-constrained settings where one adds rank one component at a time and can stop early. In these regimes, the key question is often how errors introduced at each rank-1 step propagate and when the process remains stable.

Contributions. This work presents a mathematical formulation of linear low-rank regression that emphasizes its decomposition into rank-1 problems. We focus on the scenario where the sub-routine, $\texttt{rank-1}(\mathbf{Y}_k, \mathbf{X}, t)$, incurs numerical errors: even solving equation 2 for a single pair, our estimate is only an approximation to the true pair. This view offers a pathway to examine how errors from each rank-1 estimation affect subsequent estimations. The following contributions are made:

- *Error Propagation Study.* We provide an examination of how errors propagate through the deflation process in linear low-rank regression, highlighting implications in the stability and accuracy.
- *Generalization Ability.* We analyze in the setting of noiseless and noisy labels how sequentially discovering rank-1 components can learn a model that enjoys provable generalization guarantees.
- *Empirical Validation.* We validate our theory on both linear low-rank matrix regression problems and simple PEFT settings. An interesting finding is the benefit of front-loading resource allocation for sequential procedures that are already used in PEFT and streaming low-rank updates.

2 Background

We use $\|\boldsymbol{a}\|_2$ to denote the ℓ_2-norm of vector $\mathbf{a}$; $\|\mathbf{A}\|_2$ denotes the spectral norm, $\|\mathbf{A}\|_F$ the Frobenius norm of matrix $\mathbf{A}$; $\mathrm{sv}_{\mathrm{L}}(\mathbf{A})$ and $\mathrm{sv}_{\mathrm{R}}(\mathbf{A})$ denote the normalized top left and right singular vectors of $\mathbf{A}$.

Problem setup. Let $\mathbf{X} \in \mathbb{R}^{d\times n}$ be the input matrix of n data points, each with d features. For simplicity, we will assume that $\mathbf{X}$ is sampled entrywise from the normal distribution with zero mean and unit variance, followed by a row-wise normalizing process, unless otherwise stated. Let $\mathbf{Y} \in \mathbb{R}^{m\times n}$ be the output matrix based on the noiseless generative model, as in: $\mathbf{Y} = \mathbf{W}^\star\mathbf{X}$, that simulates the process of "inserting" data samples (columns of $\mathbf{X}$) through a low-rank linear channel $\mathbf{W}^\star \in \mathbb{R}^{m\times d}$ to obtain the corresponding column in $\mathbf{Y}$. The goal is, then, to estimate the best low rank parameter $\mathbf{W}$ given data $(\mathbf{Y}, \mathbf{X})$ as a low-rank linear regression problem:

$$\min_{\mathbf{W}\in\mathbb{R}^{m\times d}} \quad f(\mathbf{W}) := \tfrac{1}{2}\|\mathbf{Y} - \mathbf{W}\mathbf{X}\|_F^2 \quad \text{s.t.} \quad \mathrm{rank}(\mathbf{W}) \le r. \tag{3}$$

Solutions. This problem has a long history with various approaches, including convex (Recht et al., 2010; Lee & Bresler, 2009; Liu & Vandenberghe, 2009), non-convex projected-gradient descent (Jain et al., 2010; Lee & Bresler, 2010; Kyrillidis & Cevher, 2014; 2011; Khanna & Kyrillidis, 2017; Xu et al., 2018), as well as matrix factorization ones (Burer & Monteiro, 2003; Jain & Dhillon, 2013; Chen & Wainwright, 2015; Zhao et al., 2015; Zheng & Lafferty, 2015; Tu et al., 2016; Kyrillidis et al., 2018b; Park et al., 2016b; Sun & Luo, 2016; Bhojanapalli et al., 2016a;b; Park et al., 2016a; Ge et al., 2017; Hsieh et al., 2017; Kyrillidis et al., 2018a; Kim et al., 2023). In the latter, the problem turns into:

$$\min_{\mathbf{A}\in\mathbb{R}^{r\times d},\ \mathbf{B}\in\mathbb{R}^{m\times r}} f(\mathbf{A}, \mathbf{B}) := \tfrac{1}{2}\|\mathbf{Y} - \mathbf{B}\mathbf{A}\mathbf{X}\|_F^2, \tag{4}$$

which is related to modernized task adaptation in neural network training, like LoRA (Hu et al., 2021; Ostapenko et al., 2024). *Key difference in our analysis is that we study the sequential nature of learning*; in contrast, in the above scenarios, one often utilizes factorized gradient descent, a low-rank solver that *updates all r rank-1 components simultaneously*, as follows:

$$\mathbf{A}_{t+1} = \mathbf{A}_t - \eta_{\mathbf{A}}\nabla_{\mathbf{A}} f(\mathbf{A}_t, \mathbf{B}_t), \quad \mathbf{B}_{t+1} = \mathbf{B}_t - \eta_{\mathbf{B}}\nabla_{\mathbf{B}} f(\mathbf{A}_t, \mathbf{B}_t), \quad \text{with } \eta_{\mathbf{A}},\ \eta_{\mathbf{B}} \text{ learning rates.}$$

We acknowledge solving (3)-(4) directly with these methods when r is known is more efficient and could be preferable in terms of accuracy, yet does not fall into the sequential scenario we focus on, where r could increase over time or is considered unknown.

Learning rank-1 subspaces sequentially. Our aim is to study routines like the ones described in (1) and (2). To do so, we will need to understand the behavior of both the *exact sequential low-rank recovery*, as well as the *inexact sequential low-rank recovery*. We describe some simple algorithms to motivate our work.

Algorithm 1 Exact Sequential Low-Rank

Require: Input data $\mathbf{X} \in \mathbb{R}^{d\times n}$, output data $\mathbf{Y} \in \mathbb{R}^{m\times n}$, target rank r
Ensure: Rank-1 components $\{(\mathbf{a}_k^\star, \mathbf{b}_k^\star)\}_{k=1}^r$
1: $\mathbf{Y}_1^\star \leftarrow \mathbf{Y}$
2: **for** $k = 1$ **to** r **do**
3: $\quad (\mathbf{a}_k^\star, \mathbf{b}_k^\star) \leftarrow \underset{\mathbf{a}\in\mathbb{R}^d, \mathbf{b}\in\mathbb{R}^m}{\mathrm{argmin}} \ \frac{1}{2}\left\|\mathbf{Y}_k^\star - \mathbf{b}\mathbf{a}^\top\mathbf{X}\right\|_F^2$
4: $\quad \mathbf{Y}_{k+1}^\star \leftarrow \mathbf{Y}_k^\star - \mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}$
5: **end for**
6: **return** $\{(\mathbf{a}_k^\star, \mathbf{b}_k^\star)\}_{k=1}^r$

Algorithm 2 Inexact Sequential Low-Rank

Require: Input data $\mathbf{X} \in \mathbb{R}^{d\times n}$, output data $\mathbf{Y} \in \mathbb{R}^{m\times n}$, target rank r, sub-routine steps T
Ensure: Approx. rank-1 components $\{(\mathbf{a}_k, \mathbf{b}_k)\}_{k=1}^r$
1: $\mathbf{Y}_1 \leftarrow \mathbf{Y}$
2: **for** $k = 1$ **to** r **do**
3: $\quad (\mathbf{a}_k, \mathbf{b}_k) \leftarrow \texttt{rank-1}(\mathbf{Y}_k, \mathbf{X}, t)$
4: $\quad \mathbf{Y}_{k+1} \leftarrow \mathbf{Y}_k - \mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}$
5: **end for**
6: **return** $\{(\mathbf{a}_k, \mathbf{b}_k)\}_{k=1}^r$

Algorithm 1 aims to find exact low-rank subspaces by iteratively computing pairs of vectors $(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$. It starts with the original output data $\mathbf{Y}$ as $\mathbf{Y}_1^\star$ (Line 1). In each iteration k, an optimization problem is solved to find the **best** pair $(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$ that minimizes the Frobenius norm of the difference between the current matrix $\mathbf{Y}_k^\star$ and the rank-1 estimate $\mathbf{b}\mathbf{a}^\top\mathbf{X}$ (see Line 3)[1]. By applying the singular value decomposition (SVD) to $\mathbf{Y}$,

[1]Note that, while $\sum_{k=1}^r \mathbf{b}_k^\star\mathbf{a}_k^\star = \mathbf{W}^\star$, $\left(\mathbf{b}_k^\star, \mathbf{a}_k^\star\right)$ not necessarily align with the kth left- and right- singular vectors of $\mathbf{W}^\star$.

Table 1: Rank-1 objects used at step k.

Symbol	Meaning
$(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$	Exact (ideal) rank-1 pair
$(\overline{\mathbf{a}}_k, \overline{\mathbf{b}}_k)$	Subroutine-optimal pair (minimizer of step-k objective)
$(\mathbf{a}_k, \mathbf{b}_k)$	Inexact pair returned by the finite-iteration routine
$\mathbf{\Psi}_k := \mathbf{b}_k\mathbf{a}_k^\top - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top$	Numerical error of the rank-1 step
$\mathbf{Y}_k$	Residual output after removing first $(k-1)$ rank-1 components

we decompose it as: $\mathbf{Y} = \sum_{k=1}^{p} \sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^{\star\top}$, where $\sigma_k^\star$ are the singular values, and $\mathbf{u}_k^\star$ and $\mathbf{v}_k^\star$ are the left and right singular vectors, respectively. Notice that we denote $p = \text{rank}(\mathbf{Y})$, but when executing Algorithm 1 and Algorithm 2 we may choose a target rank $r \neq p$.

Lemma 1. *Under our defined settings and deflation method, for each k, we have that $\mathbf{Y}_k^\star = \sum_{k'=k}^{p} \sigma_{k'}^\star \mathbf{u}_{k'}^\star \mathbf{v}_{k'}^{\star\top}$ and $\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} = \sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^{\star\top}$.*

The proof of Lemma 1 is provided in Appendix B.1. Namely, when $\mathbf{X}$ has full rank with $n \geq d$, $\mathbf{b}_k^\star$ and $\mathbf{a}_k^\star$ can be uniquely identified up to scalar multiplication.

After determining the pair $(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$, the matrix $\mathbf{Y}_{k+1}^\star$ is updated by subtracting the rank-1 component $\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X}$ from $\mathbf{Y}_k^\star$ (see Line 4). This iterative process continues for r iterations, generating r pairs of vectors, which collectively represent the exact low-rank subspaces.

Algorithm 2 differs from Algorithm 1 in Lines 3 and 4. In Algorithm 2, Line 3 executes sub-routine for t iterations, denoted by `rank-1`$(\mathbf{Y}_k, \mathbf{X}, t)$, to approximate the solution of equation 1 and return estimates $(\mathbf{a}_k, \mathbf{b}_k)$. The t parameter represents the number of iterations for this approximate computation. An example of the `rank-1` subroutine can be the gradient descent algorithm, which executes:

$$\begin{aligned} \mathbf{a}^{(t+1)} &= \mathbf{a}^{(t)} - \eta_{\mathbf{a}} \mathbf{X} \left(\mathbf{b}^{(t)} \mathbf{a}^{(t)\top} \mathbf{X} - \mathbf{Y_k} \right)^\top \mathbf{b}^{(t)}, \\ \mathbf{b}^{(t+1)} &= \mathbf{b}^{(t)} - \eta_{\mathbf{b}} \left(\mathbf{b}^{(t)} \mathbf{a}^{(t)\top} \mathbf{X} - \mathbf{Y_k} \right) \mathbf{X}^\top \mathbf{a}^{(t)}. \end{aligned} \tag{5}$$

Iterative algorithms such as (5) often produce numerical errors, leading to $\mathbf{b}_k\mathbf{a}_k^\top \neq \mathbf{b}_k^\star \mathbf{a}_k^{\star\top}$. Subsequently, this affects the quality of $\mathbf{Y}_{k+1}$ (Line 4), since the "deflation" step $\mathbf{Y}_k - \mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}$ is based on the approximate matrix $\mathbf{Y}_k$ and the current estimates $\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}$, rather than the exact versions $\mathbf{Y}_k^\star$ and $\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}$ used in Algorithm 1. To study the influence of the numerical errors produced by (5), we define:

Definition 1 (Numerical Error)**.** *Let $(\overline{\mathbf{a}}_k, \overline{\mathbf{b}}_k)$ represents the exact rank-1 solution that approximates the processed label matrix $\mathbf{Y}_k$ using data $\mathbf{X}$:*

$$\overline{\mathbf{a}}_k, \overline{\mathbf{b}}_k = \underset{\mathbf{a}\in\mathbb{R}^d, \mathbf{b}\in\mathbb{R}^m}{\text{argmin}} \frac{1}{2} \left\| \mathbf{Y}_k - \mathbf{b}\mathbf{a}^\top\mathbf{X} \right\|_F^2. \tag{6}$$

and recall that $\mathbf{a}_k, \mathbf{b}_k$ are outputs of `rank-1`$(\mathbf{Y}_k, \mathbf{X}, T)$. We define the numerical errors incurred at iteration k from the `rank-1` sub-routine as:

$$\mathbf{\Psi}_k := \mathbf{b}_k\mathbf{a}_k^\top - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top; \quad \|\mathbf{\Psi}_k\|_F \geq 0. \tag{7}$$

Notice that the definition of $\mathbf{\Psi}_k$ is based on $\mathbf{Y}_k$, not $\mathbf{Y}_k^\star$; recall that $\mathbf{Y}_k$ is constructed recursively using $\mathbf{b}_k\mathbf{a}_k^\top$. When $\mathbf{b}_k\mathbf{a}_k^\top$ is solved inexactly, we cannot guarantee that $\mathbf{Y}_k = \mathbf{Y}_k^\star$. Consequently, it is almost always the case that $(\overline{\mathbf{a}}_k, \overline{\mathbf{b}}_k) \neq (\mathbf{a}_k^\star, \mathbf{b}_k^\star)$, implying $\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top} - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\|_F > 0$. However, since the `rank-1` subroutine only has access to $\mathbf{Y}_k$, its output $(\mathbf{a}_k, \mathbf{b}_k)$ converges to $(\overline{\mathbf{a}}_k, \overline{\mathbf{b}}_k)$ instead of $(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$ as the number of iterations in the `rank-1` subroutine increases.

Related works. Sequential low-rank subspace identification has roots in Principal Component Analysis (PCA) (Jolliffe, 1995; Golub & Van Loan, 2013) and has been explored via hierarchical game-theoretic

models. In these models, rank-1 "players" learn distinct skills by optimizing a deflated objective (Gemp et al., 2021a;b; Hotelling, 1933; Mackey, 2008).

Incremental learning of Eigenspaces. (Arora et al., 2019) has shown that deep matrix factorization models implicitly discover low-rank components sequentially, an observation later extended to symmetric (Jin et al., 2023) and asymmetric (Soltanolkotabi et al., 2023) matrix sensing. This implicit recovery can be used for model compression (Kwon et al., 2024), but typically requires a deep model or specific initialization. In contrast, our work considers the simple model of low-rank linear regression by explicitly enforcing the sequential learning. While the work of (Wang et al., 2023b) is also similar, their design is more specific to matrix completion.

Low-Rank Adapter (LoRA). Our work connects to parameter-efficient fine-tuning (PEFT) methods like LoRA (Hu et al., 2021), particularly in continual learning where adapters are learned sequentially for new tasks, sometimes with orthogonality constraints to prevent catastrophic forgetting (Wistuba et al., 2023; Wang et al., 2023a). While theory shows LoRA can succeed with a sufficiently high rank (Zhang et al., 2023), the error propagation dynamics at lower ranks remain unexplored. Other recent work has focused on merging multiple LoRA adapters (Zhao et al., 2024b; Dimitriadis et al., 2024; Wu et al., 2024; Ostapenko et al.; Huh et al., 2024).

3 Error propagation during training

Recall that Lemma 1 guarantees that the rank-1 components given by the exact Algorithm 1 recovers the top-r singular vector/values of $\mathbf{Y}^\star$. In this section, we study the recovery error under the inexact Algorithm 2. To effectively compare its output with Algorithm 1, we express these outputs in terms of the singular values and singular vectors of the deflated matrices $\mathbf{Y}_k^\star$ and $\mathbf{Y}_k$. We apply similar reasoning as in Lemma 1 for the term $\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}$ based on equation 6. To do so, we define $\sigma_{i_k}, \mathbf{u}_{i_k}$, and $\mathbf{v}_{i_k}$ as the i-th top singular value and vectors pairs of the matrix $\mathbf{Y}_k$. Note that $\sigma_{i_k} \neq \sigma_i^\star$ and $(\mathbf{u}_{i_k}, \mathbf{v}_{i_k}) \neq (\mathbf{u}_i^\star, \mathbf{v}_i^\star)$, for all i. Then, for each k, the SVD on $\mathbf{Y}_k$ gives us: $\mathbf{Y}_k = \sum_{i=1}^{p-k+1} \sigma_{i_k}\mathbf{u}_{i_k}\mathbf{v}_{i_k}^\top$. Since $\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}$ is also rank-1, Eckart-Young-Mirsky theorem implies that it is the optimal rank-1 approximation of $\mathbf{Y}_k$ based on equation 6. Thus $\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X} = \sigma_{1_k}\mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top$, where σ_{1_k}, $\mathbf{u}_{1_k}$, and $\mathbf{v}_{1_k}$ correspond to the top singular value and singular vectors of $\mathbf{Y}_k$.

Recall that $\mathbf{u}_k^\star$ and $\mathbf{v}_k^\star$ are the top left and right singular vectors, respectively, of $\mathbf{Y}_k^\star$. Since singular vectors are unique only up to a sign, both $\mathrm{nsv1_L}(\mathbf{Y}_k)$ and $-\mathrm{nsv1_L}(\mathbf{Y}_k)$ are valid left singular vectors, and similarly, both $\mathrm{nsv1_R}(\mathbf{Y}_k)$ and $-\mathrm{nsv1_R}(\mathbf{Y}_k)$ are valid right singular vectors of $\mathbf{Y}_k$. We will choose $\mathbf{u}_{1_k}$ and $\mathbf{v}_{1_k}$ to be the ones such that $0 \leq \mathbf{v}_k^{\star\top}\mathbf{v}_{1_k}$ and $0 \leq \mathbf{u}_k^{\star\top}\mathbf{u}_{1_k}$:

$$\mathbf{u}_{1_k} := \mathrm{sv_L}(\mathbf{Y}_k) \cdot \operatorname*{argmin}_{s\in\{\pm 1\}} \|s \cdot \mathtt{sv_L}(\mathbf{Y}_k) - \mathbf{u}_k^\star\|_2;\ \mathbf{v}_{1_k} := \mathrm{sv_R}(\mathbf{Y}_k) \cdot \operatorname*{argmin}_{s\in\{\pm 1\}} \| \cdot \mathtt{sv_R}(\mathbf{Y}_k) - \mathbf{v}_k^\star\|_2.$$

We provide a characterization of the error propagation in the deflation methods in Algorithm 2 that is agnostic to the detail of the sub-routine `rank-1`, i.e., when one only has knowledge about $\|\mathbf{\Psi}_k\|_F$. See Appendix A for the proof.

Theorem 1. *Let $\{(\mathbf{a}_k, \mathbf{b}_k)\}_{k=1}^r$ be the output of Algorithm 2. Let $\mathbf{\Psi}_k$ be given as in Definition 1 with $\|\mathbf{\Psi}_k\|_F > 0$. Let $\sigma_1^\star, \ldots, \sigma_{r^\star}^\star$ denote the singular values of $\mathbf{Y}$. Define the minimum singular value gap as $\mathcal{T}_k^\star := \min\left\{\min_{j>k} |\sigma_k^\star - \sigma_j^\star|, \sigma_k^\star\right\}$. Define an error bound $E(k)$ as:*

$$E(k) := \sigma_{\max}(\mathbf{X}) \sum_{k'=0}^{k-1} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k-1} \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right).$$

If $E(k) < \frac{1}{2}\min_{j>k} |\sigma_k^\star - \sigma_j^\star|$, then the output of Algorithm 2 satisfies:

$$\left\|\mathbf{Y} - \sum_{k=1}^r \mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \left(\sum_{k=r+1}^p \sigma_k^\star\right) + \sigma_{\max}(\mathbf{X}) \sum_{k=1}^r \sum_{k'=0}^k \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^k \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right) \tag{8}$$

Theorem 1 characterizes how errors from approximately solving the rank-1 subroutine propagate through the deflation procedure in sequential low-rank approximations. The theorem asserts that, as long as each

error $\mathbf{\Psi}_k$ is sufficiently small, the compounded effect of errors across the sequence remains bounded, thereby preserving the accuracy of the final low-rank approximation.

Remark 2. *The error bound in Theorem 1 reflects sensitivity to the eigenspectrum of the underlying data matrix. Notice that the upper bound in Theorem 1 involves a summation of summations over components that depend on the error of the sub-routine $\mathbf{\Psi}_{k'}$. In particular, both the number of summands and the multiplicative factor of $\prod_{j=k'+1}^{k}(2+{}^{6\sigma_j^\star}/\mathcal{T}_k^\star)$ in each summand grows as k increases. Notice that a slower decay of singular values —corresponding to a smaller eigengap—error propagation is amplified, making approximation steps more susceptible to the accumulation of individual errors $\mathbf{\Psi}_k$, and vice versa. This dependency on the singular spectrum necessitates more precision in each step for data matrices with dense singular values to avoid error escalation.*

Remark 3. *Our proof relies on an recursive bound on the errors of $\left\|\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}-\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}\right\|_F$, which requires a strict ordering of the pairs $(\mathbf{a}_k,\mathbf{b}_k)$ based on the singular values of $\mathbf{Y}$. In the case where $\mathcal{T}_k^\star=0$, there must exists two singular values $\sigma_k^\star$ and $\sigma_{k+1}^\star$ of $\mathbf{Y}$ that are equal. In this case, the error $\left\|\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}-\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}\right\|_F$ is not well-defined as the pair $(\mathbf{a}_k,\mathbf{b}_k)$ can converge to both $(\mathbf{a}_k^\star,\mathbf{b}_k^\star)$ and $(\mathbf{a}_{k+1}^\star,\mathbf{b}_{k+1}^\star)$. Thus, we require $\mathcal{T}_k^\star\geq 0$ in our proof. The analysis of the case $\mathcal{T}_k^\star=0$ is possible if, instead of tracking individual components errors, we track the error of subspaces corresponding to singular vectors of the same values. We leave this extension as future work.*

Remark 4. *We do not claim our upper bound in Theorem 1 to be tight. Rather, Theorem 1 serves as an upper bound that explicitly relate the loss $\left\|\mathbf{Y}-\sum_{k=1}^{k}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F$ to the numerical errors $\mathbf{\Psi}_k$ in each step. It is possible that, under additional assumptions of the rank-1 algorithms and the structural information of the data, we are able to obtain a tighter bound. However, this direction is beyond the scope of our work, as this work is considered the first such attempt, to our knowledge.*

4 Generalization of sequential rank-1 update

Thus far, we have been focusing on constructing components $\{(\mathbf{a}_k,\mathbf{b}_k)\}_{k=1}^r$ such that $\sum_{k=1}^r\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}$ estimates $\mathbf{Y}$. Given that $\mathbf{X}$ and $\mathbf{Y}$ are considered as training data, the previous section characterized the training error of Algorithm 2. Here, we analyze the generalization ability of Algorithm 2, assuming that the data is generated based on some optimal parameter $\mathbf{W}^\star$ with $\operatorname{rank}(\mathbf{W}^\star)=r^\star$:

$$\mathbf{Y}=\mathbf{Y}^\star+\boldsymbol{\mathcal{E}};\quad \mathbf{Y}^\star=\mathbf{W}^\star\mathbf{X}, \tag{9}$$

where $\boldsymbol{\mathcal{E}}\in\mathbb{R}^{m\times n}$ denotes the label noise generated from a certain distribution, and $\mathbf{Y}^\star$ denotes the noiseless label. In noiseless case where $\boldsymbol{\mathcal{E}}=\mathbf{0}$, we have that $p=\operatorname{rank}(\mathbf{Y})=r^\star$ and, Algorithm 1 can recover $\{(\mathbf{a}_k^\star,\mathbf{b}_k^\star)\}_{k=1}^r$ such that $\mathbf{W}^\star=\sum_{k=1}^r\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}$ when $r=r^\star$. However, it may not be the case that $\mathbf{b}_k$ and $\mathbf{a}_k$ aligns with the kth left and right singular vector of $\mathbf{W}^\star$ under the influence of $\mathbf{X}$. In other words, each pair $(\mathbf{a}_k^\star,\mathbf{b}_k^\star)$ contains component of $\mathbf{W}^\star$ that *extracts a certain information from the input data* $\mathbf{X}$. When $\boldsymbol{\mathcal{E}}\neq\mathbf{0}$, it is possible that $p=\operatorname{rank}(\mathbf{Y})>r^\star$.

Generalization under noiseless labels. As a warm up, we consider the case where the noise $\boldsymbol{\mathcal{E}}=\mathbf{0}$. Intuitively, when $\mathbf{X}$ is full rank, a zero training loss would imply a perfect recovery of the optimal parameter $\mathbf{W}^\star$. We state a more general result below covering the case of non-zero training loss with component-wise generalization error.

Theorem 5. *Let $\{(\mathbf{a}_k,\mathbf{b}_k)\}_{k=1}^r$ be the output of Algorithm 2. Let $\mathbf{\Psi}_k$ be given as in Definition 1 with $\|\mathbf{\Psi}_k\|_F>0$. Let $\sigma_1^\star,\ldots,\sigma_{r^\star}^\star$ denote the singular values of $\mathbf{Y}$. Define the minimum singular value gap as $\mathcal{T}_k^\star:=\min\left\{\min_{j>k}|\sigma_k^\star-\sigma_j^\star|,\sigma_k^\star\right\}$. Also, define an error bound $E(k)$ as:*

$$E(k):=\sigma_{\max}(\mathbf{X})\sum_{k'=0}^{k-1}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k-1}\left(2+\frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right).$$

If $E(k)<\frac{1}{2}\min_{j>k}|\sigma_k^\star-\sigma_j^\star|$, and $\sigma_{\min}(\mathbf{X})>0$, then the output of Algorithm 2 satisfies:

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}-\mathbf{b}_k\mathbf{a}_k^\top\right\|_F\leq\kappa(\mathbf{X})\sum_{k'=0}^{k}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k}\left(2+\frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right);\ \ \forall k\in[r]. \tag{10}$$

Moreover, the aggregation of the components $(\mathbf{a}_k, \mathbf{b}_k)$*'s approximates* $\mathbf{W}^\star$ *as*

$$\left\| \mathbf{W}^\star - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \right\|_F \leq \sum_{k=r+1}^{r^\star} \frac{\sigma_k^\star}{\sigma_{\min}(\mathbf{X})} + \kappa(\mathbf{X}) \sum_{k=1}^{r} \sum_{k'=1}^{k} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_j^\star} \right). \tag{11}$$

Here, $\kappa(\mathbf{X}) = \frac{\sigma_{\max}(\mathbf{X})}{\sigma_{\min}(\mathbf{X})}$ *denotes the condition number of* $\mathbf{X}$*.*

The proof of Theorem 5 is provided in Appendix C. In particular, Theorem 5 states two results. First, (10) measures how the errors of individual components are influenced even by the numerical errors that appear when solving previous components. This bound illustrates key factors contributing to the error at each iteration. As we discussed previously, $(\mathbf{a}_k^\star, \mathbf{b}_k^\star)$ can be considered as the components of $\mathbf{W}^\star$ extracted based on the importance defined by the input data $\mathbf{X}$. From this perspective, (10) shows how well these data-dependent components are approximated by Algorithm 2. Moreover, (11) measures how well the inexact method approximates $\mathbf{W}^\star$, including errors due to inexact computations and limitations of representing $\mathbf{W}^\star$ with rank r. This bound shed light on how the components $(\mathbf{a}_k, \mathbf{b}_k)$'s collaboratively contribute to the overall generalization ability.

Generalization under noisy labels. In the previous section, we studied the generalization ability of Algorithm 2 under a noiseless scenario with $\mathbf{Y} = \mathbf{W}^\star \mathbf{X}$, where the algorithmic choice of choosing $r = \operatorname{rank}(\mathbf{Y})$ can be shown to be optimal. However, this argument may not hold when the labels are generated with a non-zero additive noise. In this section, we consider the noise matrix $\boldsymbol{\mathcal{E}}$ to consist of I.I.D. entries $\boldsymbol{\mathcal{E}}_{ij} \sim \mathcal{N}(0, \varepsilon^2)$, where ε controls the magnitude of the noise. In this case, with a high probability, we have $p = \operatorname{rank}(\mathbf{Y}) = m$. Let $\{(\mathbf{a}_k, \mathbf{b}_k)\}_{k=1}^r$ be the recovery result according to Algorithm 2. Then we are interested in an upper bound on $\|\mathbf{W}^\star - \sum_{k=1}^r \mathbf{b}_k \mathbf{a}_k\|_F$. In particular, we have the following guarantee on the generalization error.

Theorem 6. *Consider the scenario of finding the top-r rank-1 subspaces that minimize the loss in equation 1. Let* $\mathcal{T}_k^\star := \min\{\min_{j>k} |\sigma_k^\star - \sigma_{j_k}|, \sigma_k^\star\}$*, and* $\mathcal{T}_{\min}^\star := \min_{k \in [1,r]} \mathcal{T}_k^\star$*, if the noise scale satisfies* $\varepsilon \leq O\left(\frac{\mathcal{T}_{\min}^\star}{\sqrt{n} + \sqrt{\log 1/\gamma}} \right)$ *then with probability at least* $1 - \gamma$*, the output of Algorithm 2 satisfies:*

$$\begin{aligned} \left\| \mathbf{W}^\star - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \right\|_F \leq \kappa(\mathbf{X}) & \left(\sum_{k=r+1}^{r^\star} \sigma_r(\mathbf{W}^\star) + \sum_{k=1}^{r} \sum_{k'=0}^{k} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star} \right) \right) \\ & + O\left(\frac{\varepsilon \sqrt{n \log 1/\gamma}}{\sigma_{\min}(\mathbf{X})} \left(r + \sqrt{\frac{\min\{r^\star, r\}}{\mathcal{T}_{\min}^\star}} \right) \right) \end{aligned} \tag{12}$$

The proof of Theorem 6 is given in Appendix C.3. In particular, Theorem 6 characterizes how Algorithm 2 recovers component that can generalize even under label noise. The first term in the upper bound of (12) comes from the numerical errors in the inexact solving of each `rank-1` subroutine. Then second term demonstrates the influence of the additive noise $\boldsymbol{\mathcal{E}}$ on the generalization ability. To start, a larger noise scale ϵ implies a worse generalization error. Moreover, it should be noticed that a good choice of r can greatly impact the generalization error as well: choosing $r < r^\star$ can result in a larger error in the first term due to the incomplete estimation of the components in $\mathbf{W}^\star$. On the other hand, since the second term scales with r, choosing a larger r can result in a larger error caused by the noise. This is the scenario where the noise is overfitted by increasing the complexity of the model. From this perspective, Theorem 6 characterizes the bias-variance trade-off in the sequential rank-1 recovery algorithm. Lastly, the requirement of the noise scale is to make sure that after adding the noise, the ordering of the rank-1 components is not changed.

5 Experiments

We present experimental results to corroborate our theoretical findings, as well as to illustrate possible extensions to the non-linear case of LoRA training. These experiments are not meant to show that sequential deflation dominates one-shot training; rather, they validate our error-propagation theory and show how compute scheduling across components materially changes outcomes, as the theory predicts.

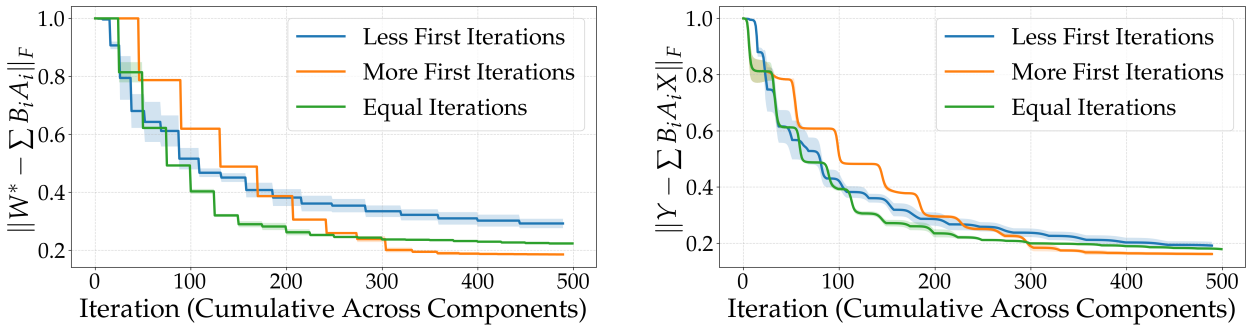


Figure 1: Impact of iteration allocation strategy under a fixed iteration budget. *Left*: $\mathbf{W}^\star$ reconstruction error. *Right*: Training objective error.

5.1 Synthetic validation of theoretical setting.

Our experiments aim to demonstrate how the distribution of computational resources across rank-1 components affects the overall approximation quality, particularly focusing on how errors in early components propagate to later stages of the sequential learning process. Following the general set-up in (9), we consider three different iteration allocation strategies:

1. **Equal:** Same number of optimization iterations to each rank-1 component.
2. **More First:** More iterations allocated to the earlier components and fewer to later ones.
3. **Less First:** Fewer iterations allocated to the earlier components and more to later ones.

Our analysis dictates that errors in early components propagate to later components, suggesting that allocating more iterations to earlier components leads to better overall performance. Figure 1 shows the reconstruction and training errors, respectively, for the three allocation strategies under a fixed computational budget.

The results confirm our theoretical predictions. Allocating more iterations to earlier components leads to better final reconstruction and training errors compared to allocating fewer iterations initially. The equal iteration strategy performs better than the "less first iterations" approach, but worse than the "more first iterations" strategy. This validates our theoretical finding that errors in early components propagate and compound through the sequential process. This cascading effect means that if early components are poorly approximated, their errors get magnified in subsequent components. Details and further results in the synthetic setting are deferred to Appendix E.

5.2 Experimental analysis using LoRA on simple vision tasks.

We evaluate our sequential rank-1 approach in LoRA adaptation on three standard image classification datasets: MNIST, CIFAR10, and CIFAR100. We design the experiments such that each dataset present a different level of challenge to assess how our sequential LoRA adaptation performs under varying initial conditions. *The purpose here is* ***not*** *to attain top-notch performance in these scenarios neither to claim these as "real scenarios"; rather, to assess how sequential learning behaves on well- to –intentionally– badly-pretrained scenarios. This is also expected, given that the baseline model is a feedforward neural network.*

Problem setting. We employ a simple feedforward network as our base architecture across all experiments. For each dataset, we first train the baseline model on a subset of classes (in particular, the first half of available classes). We then apply our sequential rank-1 LoRA adaptation approach to handle the remaining classes for 3 sequential rank-1 trainings, i.e., $r = 3$.

Our architecture consists of three fully-connected layers that map flattened input images to class logits. As usual, for MNIST, inputs are 784-dimensional (28×28 grayscale images), while for CIFAR10 and CIFAR100, inputs are 3072-dimensional (32×32×3 RGB images). Hidden layers have 512 units with ReLU activations, and the output layer dimension matches the number of classes in each dataset. We analyze three distinct scenarios:

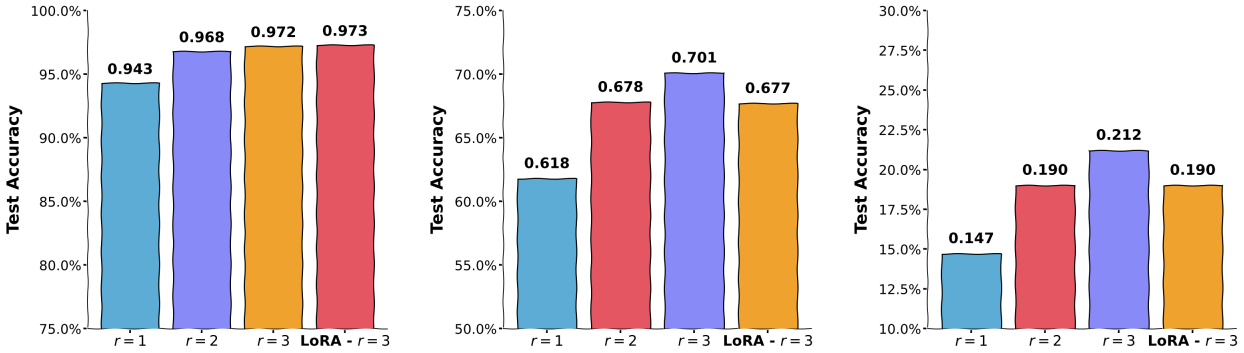


Figure 2: Test accuracy of sequential rank-1 LoRA on new classes: MNIST (left), CIFAR10 (center), CIFAR100 (right).

1. **MNIST (Strong Baseline)**: The baseline network achieves high accuracy ($\sim$98%) on classes 0-4, providing a strong foundation for adaptation on the remaining 5-9 classes.
2. **CIFAR10 (Moderate Baseline)**: The baseline network reaches moderate accuracy ($\sim$40%) on classes 0-4, representing a partially optimized model (reminder that the model is not a CNN-based model but just a FF connected network).
3. **CIFAR100 (Weak Baseline)**: The baseline network attains lower accuracy ($\sim$20%) on classes 0-49, exemplifying a relatively poor initial representation, where LoRA models adapt over the remaining 50-99 classes.

Mathematical formulation. In standard LoRA, we parameterize the weight change during fine-tuning as a low-rank decomposition: $\Delta\mathbf{W} = \mathbf{B}\mathbf{A}^\top \in \mathbb{R}^{m\times n}$, where $\mathbf{A} \in \mathbb{R}^{n\times r}$ and $\mathbf{B} \in \mathbb{R}^{m\times r}$ with $r \ll \min(m, n)$. In these experiments and w.l.o.g., $r = 3$. In our approach, instead of optimizing all r components simultaneously, we optimize one rank-1 component at a time, using the residual error from previous components to guide each subsequent step.

In particular, let $\mathbf{W}_0 \in \mathbb{R}^{m\times n}$ be a pre-trained weight matrix (in our case, we have $\mathbf{W}_0$ for every layer of the pretrained fully-connected network). Let $\mathcal{L}$ be the task-specific loss function, f is the network function, and $(\mathbf{x}, \mathbf{y})$ represents the task data. Then,

$$\Delta\mathbf{W} = \underset{\Delta\hat{\mathbf{W}}\in\mathbb{R}^{m\times n},\mathrm{rank}(\Delta\hat{\mathbf{W}})=r}{\arg\min} \mathcal{L}(f(\mathbf{x}; \mathbf{W}_0 + \Delta\hat{\mathbf{W}}), \mathbf{y}). \tag{13}$$

We define our sequential rank-1 adaptation procedure as follows: For $k = 1, 2, \ldots, r$, find the rank-1 update that minimizes the task loss given the previously learned components:

$$\mathbf{a}_k, \mathbf{b}_k = \underset{\mathbf{a}\in\mathbb{R}^n,\mathbf{b}\in\mathbb{R}^m}{\arg\min} \mathcal{L}\Big(f\Big(\mathbf{x}; \mathbf{W}_0 + \sum_{j=1}^{k-1} \mathbf{b}_j\mathbf{a}_j^\top + \mathbf{b}\mathbf{a}^\top\Big), \mathbf{y}\Big) \tag{14}$$

The final adapted model uses the weight matrix on the new data domain: $\mathbf{W} = \mathbf{W}_0 + \sum_{k=1}^{r} \mathbf{b}_k\mathbf{a}_k^\top$. We approximate the optimal rank-1 updates using (stochastic) gradient descent on (14). All experiments were conducted using Google Colab Pro+ with NVIDIA A100 GPU (40GB memory).

Adaptation performance across datasets. Figure 2 presents the test accuracy of sequential rank-1 LoRA components when adapting to new classes across the three datasets. The accuracy is measured solely on the new classes (classes "5-9" for MNIST and CIFAR10, classes "50-99" for CIFAR100), highlighting the adaptation capabilities rather than overall performance. The standard deviation for all the LoRA experiments is maximum $\sim$1.5% around the mean over 5 different runs, and the key message of this section remains consistent over runs.

Our results demonstrate that sequential LoRA adaptation effectively transfers knowledge across all three scenarios, though with varying degrees of success depending on the quality of the baseline model. In all cases, we observe that sequential rank-1 LoRA training works at least comparable to standard LoRA, where all $r = 3$ components are trained simultaneously.

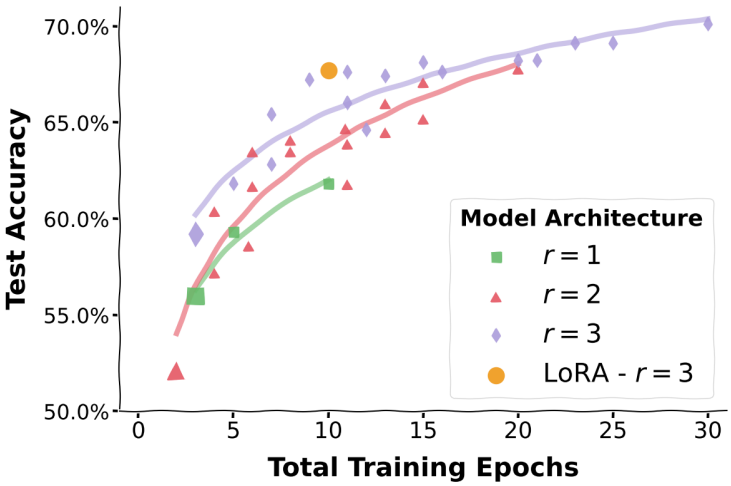


Figure 3: Marker sizes indicate relative efficiency of each configuration.

Definitely, this performance comes with a cost. Figure 3 displays the relationship between compute efficiency (measured by test accuracy per training epoch) and total training epochs for different model architectures. Here, Rank-1 architectures correspond to just using $r = 1$ for different number of epochs; Rank-2 architectures correspond to $r = 2$, where the components are trained for different *combinations* of total epochs (e.g., some models have been trained with 1→ 1 epochs, while others have been trained with 10 → 10 epochs; more about this in the next paragraph), and so on.

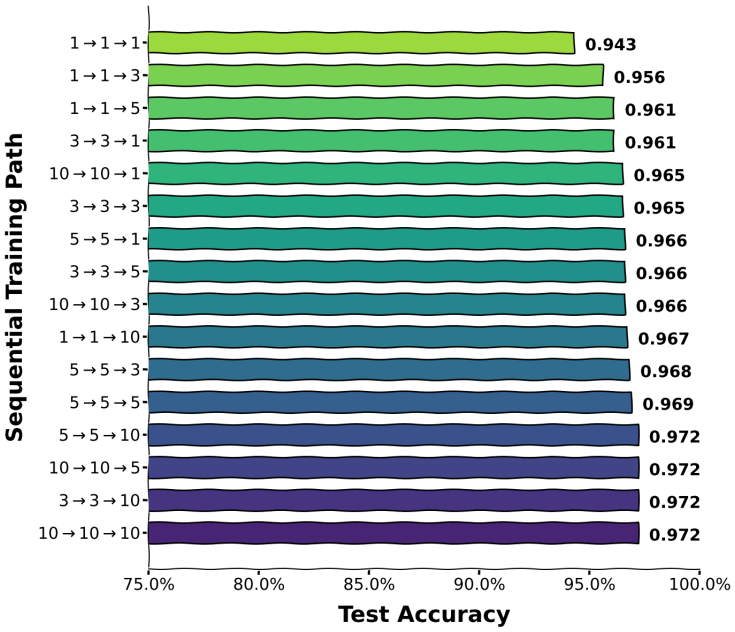


Figure 4: $\alpha \to \beta \to \gamma$ denotes sequential training with α, β, γ epochs for each component. Not all combinations are shown.

Across all datasets, we observe that sequential rank-1 approaches (noted as "Rank-1", "Rank-2", and "Rank-3") achieve comparable compute efficiency (with a slight loss of accuracy) compared to standard LoRA. Sequential rank-1 models require more total training for comparable accuracy, thus creating a tradeoff, but still maintain favorable parameter-to-performance ratios. *Yet our approach introduces an interesting property: sequential rank-1 does not require to know apriori the rank of the adaptation; one could check online whether accuracy is sufficient and stop further training. Such a property lacks in standard LoRA: either the user needs to know a good value for r, or one needs to consider different r values from scratch before making the final decision.*

Sequential training paths. Figure 4 illustrates the effectiveness of different sequential training paths for the case of MNIST, where each path represents a sequence of component training durations. For example, path "1→3→5" indicates a rank-3 LoRA where the 1st component is trained for 1 epoch, the 2nd component 3 epochs, and the 3rd component 5 epochs.

In all cases (experiments for all cases convey a similar message), it is evident that, oftentimes, good first component implies (almost all the times) a better combined final model: front-loaded training schedules

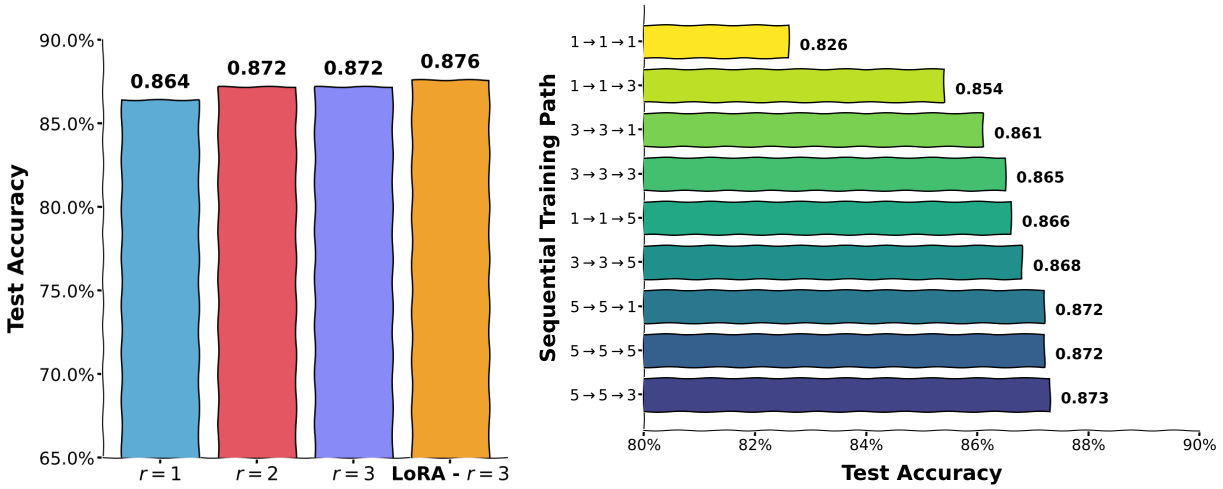


Figure 5: (Left) Final test accuracy after each sequential rank-1 component is added (achieving 0.864, 0.872, and 0.872 for Rank-1, Rank-2, and Rank-3 respectively), compared to a standard, jointly-trained rank-3 LoRA (0.876). (Right) Effectiveness of different sequential training paths for the rank-3 LoRA adaptation on SST-2. The path $\alpha \to \beta \to \gamma$ denotes a strategy where the first component is trained for α epochs, the second for β epochs, and the third for γ epochs. Final test accuracy is shown on the x-axis.

often perform better, indicating that the first component could capture most of the necessary adaptation, with diminishing returns for extensive training of later components.

5.3 Experimental analysis using LoRA on simple language tasks.

Problem setting. We employ a `distilbert-base-uncased` model (Sanh et al., 2019), a smaller and faster variant of BERT, and fine-tune it on the GLUE SST-2 dataset for binary sentiment classification (Yang et al., 2024). Following the same methodology as in the vision tasks, we freeze all base model parameters and inject LoRA adapters into the query (`q_lin`) and value (`v_lin`) projections of each attention layer. We again compare the standard, joint-training LoRA (with $r = 3$) against our sequential rank-1 adaptation, which applies the non-linear optimization from equation (14) for a total target rank of $r = 3$.

Adaptation performance. Figure 5(Left) shows the test accuracy as each sequential component is added. The first rank-1 component alone achieves accuracy of 0.864, capturing the majority of the required adaptation. Adding a second component (Rank-2) provides some benefit, increasing accuracy to 0.872. However, adding a third component (Rank-3) yields no further improvement, indicating that the "useful" rank for this task and model is likely 2. The standard, jointly-trained LoRA (rank 3) achieves the highest accuracy at 0.876, slightly outperforming but on par with sequential strategies.

Sequential training paths. This experiment probes the implications of our theoretical hypothesis in this non-linear setting. As shown in Figure 5 (Right), a good first component implies (almost all the times) a better combined final model. Front-loaded training schedules perform better, indicating that the first component captures most of the necessary adaptation. Conversely, a severely under-trained first component (e.g., "1→1→1") yields the worst performance (0.826) by a large margin, aligning with our hypothesis that large initial errors propagate and degrade the final solution.

However, the non-linear dynamics here introduce further complexities. Notably, a back-loaded strategy like "1→1→5" (0.866) slightly outperforms a balanced one ("3→3→3" at 0.865). A potential explanation is that the early rank-1 and rank-2 components become accurate very quickly, so spending extra epochs on them (e.g., 3 vs. 1) barely shrinks the error and yields diminishing returns. If the third component is working on a harder residual, allocating the computational budget there (the "5" in "1→1→5") may improve the final accuracy more than over-tuning the first stages. This suggests a more nuanced allocation principle than

simple front-loading: front-load compute, but only until early stages stop giving meaningful improvement per extra epoch, then reallocate the compute to later stages.

6 Limitations

Our theoretical analysis and experimental results on sequential rank-1 learning have certain limitations. First, the theoretical analysis is constrained to linear low-rank regression, leaving open challenges in extending to non-linear transformations and complex architectures. Moreover, while sequential rank-1 updates offer flexible rank determination, they may demand more training iterations than simultaneous rank optimization. Furthermore, our LoRA adaptation experiments are restricted to feedforward networks and basic classification tasks. Lastly, it remains an open question to characterize the optimal allocation of training epochs across components.

7 Conclusion

By examining error propagation in hierarchical learning frameworks, we demonstrate how the accuracy of sequential components is interconnected, with each subsequent step dependent on the precision of preceding estimations. Experimental results on low-rank linear matrix regression and LoRA adaptation (main text and Appendix) validate our hypotheses. From a practical point of view, this work suggests that more computational resources (larger T_k) should be allocated to earlier components to reduce their approximation errors, as these have the largest impact on the final result. To this end, our work opens up the following future directions:
—*Adaptive Procedures*: We can develop adaptive procedures that adjust the number of gradient steps T_k based on the estimated approximation error; this connects with learning schedules literature.
—*Component Reoptimization*: Periodically refine earlier components after extracting new ones.
—*Orthogonality Constraints*: Enforce orthogonality between components to reduce interference.
—*Hybrid Approaches*: Combine sequential rank-1 updates with occasional full-rank steps.

References

Marcin Andrychowicz, Misha Denil, Sergio Gomez, Matthew W Hoffman, David Pfau, Tom Schaul, Brendan Shillingford, and Nando De Freitas. Learning to learn by gradient descent by gradient descent. *Advances in neural information processing systems*, 29, 2016.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization, 2019. URL `https://arxiv.org/abs/1905.13655`.

Pierre Baldi and Peter J Sadowski. Understanding dropout. *Advances in neural information processing systems*, 26, 2013.

Laura Balzano, Yuejie Chi, and Yue M Lu. Streaming PCA and subspace tracking: The missing data case. *Proceedings of the IEEE*, 106(8):1293–1310, 2018.

Andrew G Barto and Sridhar Mahadevan. Recent advances in hierarchical reinforcement learning. *Discrete event dynamic systems*, 13:341–379, 2003.

Srinadh Bhojanapalli, Anastasios Kyrillidis, and Sujay Sanghavi. Dropping convexity for faster semi-definite optimization. In *Conference on Learning Theory*, pp. 530–582, 2016a.

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

Shuqing Bian, Xingyu Pan, Wayne Xin Zhao, Jinpeng Wang, Chuyuan Wang, and Ji-Rong Wen. Multi-modal mixture of experts represetation learning for sequential recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 110–119, 2023.

Nils Bjorck, Carla P Gomes, Bart Selman, and Kilian Q Weinberger. Understanding batch normalization. *Advances in neural information processing systems*, 31, 2018.

Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

Emmanuel Candes and Benjamin Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

Susan Carey and Ellen M Markman. Cognitive development. In *Cognitive science*, pp. 201–254. Elsevier, 1999.

Rich Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997.

Arslan Chaudhry, Naeemullah Khan, Puneet Dokania, and Philip Torr. Continual learning in low-rank orthogonal subspaces. *Advances in Neural Information Processing Systems*, 33:9900–9911, 2020.

Yudong Chen and Martin J Wainwright. Fast low-rank estimation by projected gradient descent: General statistical and algorithmic guarantees. *arXiv preprint arXiv:1509.03025*, 2015.

National Research Council, Division of Behavioral, Board on Behavioral, Sensory Sciences, Committee on Developments in the Science of Learning with additional material from the Committee on Learning Research, and Educational Practice. *How people learn: Brain, mind, experience, and school: Expanded edition*, volume 1. National Academies Press, 2000.

Michael Crawshaw. Multi-task learning with deep neural networks: A survey. *arXiv preprint arXiv:2009.09796*, 2020.

Y Danisman, MF Yilmaz, A Ozkaya, and I Comlekciler. A comparison of eigenvalue methods for principal component analysis. *Appl. and Comput. Math*, 13:316–331, 2014.

Nikolaos Dimitriadis, Pascal Frossard, and Francois Fleuret. Pareto low-rank adapters: Efficient multi-task learning with preferences. *arXiv preprint arXiv:2407.08056*, 2024.

K Anders Ericsson, Ralf T Krampe, and Clemens Tesch-Römer. The role of deliberate practice in the acquisition of expert performance. *Psychological review*, 100(3):363, 1993.

Mehrdad Farajtabar, Navid Azizan, Alex Mott, and Ang Li. Orthogonal gradient descent for continual learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 3762–3773. PMLR, 2020.

Rong Ge, Chi Jin, and Yi Zheng. No spurious local minima in nonconvex low rank problems: A unified geometric analysis. *arXiv preprint arXiv:1704.00708*, 2017.

Ian Gemp, Brian McWilliams, Claire Vernade, and Thore Graepel. Eigengame: {PCA} as a nash equilibrium. In *International Conference on Learning Representations*, 2021a. URL `https://openreview.net/forum?id=NzTU59SYbNq`.

Ian Gemp, Brian McWilliams, Claire Vernade, and Thore Graepel. Eigengame unloaded: When playing games is better than optimizing. *arXiv preprint arXiv:2102.04152*, 2021b.

Gene H Golub and Charles F Van Loan. *Matrix computations*. JHU press, 2013.

Wooseok Ha and Rina Foygel Barber. Robust PCA with compressed data. *Advances in Neural Information Processing Systems*, 28, 2015.

Harold Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of educational psychology*, 24(6):417, 1933.

Ya-Ping Hsieh, Yu-Chun Kao, Rabeeh Karimi Mahabadi, Yurtsever Alp, Anastasios Kyrillidis, and Volkan Cevher. A non-euclidean gradient descent framework for non-convex matrix factorization. Technical report, Institute of Electrical and Electronics Engineers, 2017.

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*, 2021.

Minyoung Huh, Brian Cheung, Jeremy Bernstein, Phillip Isola, and Pulkit Agrawal. Training neural networks from scratch with parallel low-rank adapters. *arXiv preprint arXiv:2402.16828*, 2024.

Prateek Jain and Inderjit S Dhillon. Provable inductive matrix completion. *arXiv preprint arXiv:1306.0626*, 2013.

Prateek Jain, Raghu Meka, and Inderjit S Dhillon. Guaranteed rank minimization via singular value projection. In *Advances in Neural Information Processing Systems*, pp. 937–945, 2010.

Jikai Jin, Zhiyuan Li, Kaifeng Lyu, Simon S. Du, and Jason D. Lee. Understanding incremental learning of gradient descent: A fine-grained analysis of matrix sensing, 2023. URL `https://arxiv.org/abs/2301.11500`.

I.T. Jolliffe. Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, 22(1):29–35, 1995.

R. Khanna and A. Kyrillidis. IHT dies hard: Provable accelerated iterative hard thresholding. *arXiv preprint arXiv:1712.09379*, 2017.

Junhyung Lyle Kim, George Kollias, Amir Kalev, Ken X Wei, and Anastasios Kyrillidis. Fast quantum state reconstruction via accelerated non-convex programming. In *Photonics*, volume 10, pp. 116. MDPI, 2023.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

Soo Min Kwon, Zekai Zhang, Dogyoon Song, Laura Balzano, and Qing Qu. Efficient compression of overparameterized deep models through low-dimensional learning dynamics, 2024. URL `https://arxiv.org/abs/2311.05061`.

A. Kyrillidis and V. Cevher. Recipes on hard thresholding methods. In *Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP), 2011 4th IEEE International Workshop on*, pp. 353–356. IEEE, 2011.

A. Kyrillidis and V. Cevher. Matrix recipes for hard thresholding methods. *Journal of mathematical imaging and vision*, 48(2):235–265, 2014.

A. Kyrillidis, A. Kalev, D. Park, S. Bhojanapalli, C. Caramanis, and S. Sanghavi. Provable quantum state tomography via non-convex methods. *npj Quantum Information*, 4(36), 2018a.

Anastasios Kyrillidis, Amir Kalev, Dohyung Park, Srinadh Bhojanapalli, Constantine Caramanis, and Sujay Sanghavi. Provable compressed sensing quantum state tomography via non-convex methods. *npj Quantum Information*, 4(1):36, 2018b.

Kiryung Lee and Yoram Bresler. Guaranteed minimum rank approximation from linear observations by nuclear norm minimization with an ellipsoidal constraint. *arXiv preprint arXiv:0903.4742*, 2009.

Kiryung Lee and Yoram Bresler. Admira: Atomic decomposition for minimum rank approximation. *IEEE Transactions on Information Theory*, 56(9):4402–4416, 2010.

Liyuan Liu, Xiaodong Liu, Jianfeng Gao, Weizhu Chen, and Jiawei Han. Understanding the difficulty of training transformers. *arXiv preprint arXiv:2004.08249*, 2020.

Zhang Liu and Lieven Vandenberghe. Interior-point method for nuclear norm approximation with application to system identification. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1235–1256, 2009.

Lester Mackey. Deflation methods for sparse pca. *Advances in neural information processing systems*, 21, 2008.

Hayden McAlister, Anthony Robins, and Lech Szymanski. Sequential learning in the dense associative memory. *arXiv preprint arXiv:2409.15729*, 2024.

Hideyuki Okano, Tomoo Hirano, and Evan Balaban. Learning and memory. *Proceedings of the National Academy of Sciences*, 97(23):12403–12404, 2000.

Oleksiy Ostapenko, Zhan Su, Edoardo Ponti, Laurent Charlin, Nicolas Le Roux, Lucas Caccia, and Alessandro Sordoni. Towards modular LLMs by building and reusing a library of LoRAs. In *Forty-first International Conference on Machine Learning*.

Oleksiy Ostapenko, Zhan Su, Edoardo Maria Ponti, Laurent Charlin, Nicolas Le Roux, Matheus Pereira, Lucas Caccia, and Alessandro Sordoni. Towards modular LLMs by building and reusing a library of LoRAs. *arXiv preprint arXiv:2405.11157*, 2024.

Lucas Page-Caccia, Edoardo Maria Ponti, Zhan Su, Matheus Pereira, Nicolas Le Roux, and Alessandro Sordoni. Multi-head adapter routing for cross-task generalization. *Advances in Neural Information Processing Systems*, 36, 2024.

Dohyung Park, Anastasios Kyrillidis, Constantine Caramanis, and Sujay Sanghavi. Finding low-rank solutions to matrix problems, efficiently and provably. *arXiv preprint arXiv:1606.03168*, 2016a.

Dohyung Park, Anastasios Kyrillidis, Constantine Caramanis, and Sujay Sanghavi. Non-square matrix sensing without spurious local minima via the burer-monteiro approach. *arXiv preprint arXiv:1609.03240*, 2016b.

Shubham Pateria, Budhitama Subagdja, Ah-hwee Tan, and Chai Quek. Hierarchical reinforcement learning: A comprehensive survey. *ACM Computing Surveys (CSUR)*, 54(5):1–35, 2021.

Liangzu Peng, Paris Giampouras, and René Vidal. The ideal continual learner: An agent that never forgets. In *International Conference on Machine Learning*, pp. 27585–27610. PMLR, 2023.

Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

Ken Richardson. *Models of cognitive development*. Psychology Press, 2019.

Angelika Rohde and Alexandre B Tsybakov. Estimation of high-dimensional low-rank matrices1. *The Annals of Statistics*, 39(2):887–930, 2011.

Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

Youcef Saad. Projection and deflation method for partial pole assignment in linear state feedback. *IEEE Transactions on Automatic Control*, 33(3):290–297, 1988.

Himanshu Sahni, Saurabh Kumar, Farhan Tejani, and Charles Isbell. Learning to compose skills. *arXiv preprint arXiv:1711.11289*, 2017.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019.

Amartya Sanyal, Varun Kanade, Philip HS Torr, and Puneet K Dokania. Robustness via deep low-rank representations. *arXiv preprint arXiv:1804.07090*, 2018.

Dudley Shapere. The structure of scientific revolutions. *The Philosophical Review*, 73(3):383–394, 1964.

Mahdi Soltanolkotabi, Dominik Stöger, and Changzhi Xie. Implicit balancing and regularization: Generalization and convergence guarantees for overparameterized asymmetric matrix sensing. In Gergely Neu and Lorenzo Rosasco (eds.), *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 5140–5142. PMLR, 12–15 Jul 2023. URL `https://proceedings.mlr.press/v195/soltanolkotabi23a.html`.

Alessandro Sordoni, Eric Yuan, Marc-Alexandre Côté, Matheus Pereira, Adam Trischler, Ziang Xiao, Arian Hosseini, Friederike Niedtner, and Nicolas Le Roux. Joint prompt optimization of stacked llms using variational inference. *Advances in Neural Information Processing Systems*, 36, 2024.

Bharath K Sriperumbudur, David A Torres, and Gert RG Lanckriet. Sparse eigen methods by DC programming. In *Proceedings of the 24th international conference on Machine learning*, pp. 831–838, 2007.

Ruo-Yu Sun. Optimization for deep learning: An overview. *Journal of the Operations Research Society of China*, 8(2):249–294, 2020.

Ruoyu Sun and Zhi-Quan Luo. Guaranteed matrix completion via non-convex factorization. *IEEE Transactions on Information Theory*, 62(11):6535–6579, 2016.

S. Tu, R. Boczar, M. Simchowitz, M. Soltanolkotabi, and B. Recht. Low-rank solutions of linear matrix equations via Procrustes flow. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pp. 964–973. JMLR. org, 2016.

Namrata Vaswani, Thierry Bouwmans, Sajid Javed, and Praneeth Narayanamurthy. Robust subspace learning: Robust PCA, robust subspace tracking, and robust subspace recovery. *IEEE signal processing magazine*, 35(4):32–55, 2018.

René Vidal and Paolo Favaro. Low rank subspace clustering. *Pattern Recognition Letters*, 43:47–61, 2014.

Deng-Bao Wang, Lei Feng, and Min-Ling Zhang. Rethinking calibration of deep neural networks: Do not be afraid of overconfidence. *Advances in Neural Information Processing Systems*, 34:11809–11820, 2021.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning, 2023a. URL `https://arxiv.org/abs/2310.14152`.

Zhi-Yong Wang, Xiao Peng Li, Hing Cheung So, and Abdelhak M. Zoubir. Adaptive rank-one matrix completion using sum of outer products. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4868–4880, 2023b. doi: 10.1109/TCSVT.2023.3250651.

Per-Åke Wedin. Perturbation bounds in connection with singular value decomposition. *BIT*, 12(1):99–111, March 1972. ISSN 0006-3835. doi: 10.1007/BF01932678. URL `https://doi.org/10.1007/BF01932678`.

H. Weyl. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). *Mathematische Annalen*, 71:441–479, 1912. URL `http://eudml.org/doc/158545`.

Martin Wistuba, Prabhu Teja Sivaprasad, Lukas Balles, and Giovanni Zappella. Continual learning with low rank adaptation, 2023. URL `https://arxiv.org/abs/2311.17601`.

Taiqiang Wu, Jiahao Wang, Zhe Zhao, and Ngai Wong. Mixture-of-subspaces in low-rank adaptation. *arXiv preprint arXiv:2406.11909*, 2024.

Peng Xu, Bryan He, Christopher De Sa, Ioannis Mitliagkas, and Chris Re. Accelerated stochastic power iteration. In Amos Storkey and Fernando Perez-Cruz (eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 58–67. PMLR, 09–11 Apr 2018. URL `https://proceedings.mlr.press/v84/xu18a.html`.

Bin Yang. Projection approximation subspace tracking. *IEEE Transactions on Signal processing*, 43(1): 95–107, 1995.

Sohee Yang, Jonghyeon Kim, Joel Jang, Seonghyeon Ye, Hyunji Lee, and Minjoon Seo. Improving probability-based prompt selection through unified evaluation and analysis. *Transactions of the Association for Computational Linguistics*, 12:664–680, 2024.

Zitong Yang, Yaodong Yu, Chong You, Jacob Steinhardt, and Yi Ma. Rethinking bias-variance trade-off for generalization of neural networks. In *International Conference on Machine Learning*, pp. 10767–10777. PMLR, 2020.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2017.

Fuzhen Zhang. *The Schur complement and its applications*, volume 4. Springer Science & Business Media, 2006.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. *arXiv preprint arXiv:2303.10512*, 2023.

Tuo Zhao, Zhaoran Wang, and Han Liu. A nonconvex optimization framework for low rank matrix estimation. In *Advances in Neural Information Processing Systems*, pp. 559–567, 2015.

Xianbing Zhao, Lizhen Qu, Tao Feng, Jianfei Cai, and Buzhou Tang. Learning in order! a sequential strategy to learn invariant features for multimodal sentiment analysis. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 9729–9738, 2024a.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, Kun Kuang, and Fei Wu. Merging LoRAs like playing LEGO: Pushing the modularity of LoRA to extremes through rank-wise clustering. *arXiv preprint arXiv:2409.16167*, 2024b.

Qinqing Zheng and John Lafferty. A convergent gradient descent algorithm for rank minimization and semidefinite programming from random linear measurements. In *Advances in Neural Information Processing Systems*, pp. 109–117, 2015.

Daquan Zhou, Zhiding Yu, Enze Xie, Chaowei Xiao, Animashree Anandkumar, Jiashi Feng, and Jose M Alvarez. Understanding the robustness in vision transformers. In *International Conference on Machine Learning*, pp. 27378–27394. PMLR, 2022.

A Proof of Theorem 1

The proof focuses on bounding two crucial quantities: $\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F$ and $\|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$. The first quantity measures the difference between the true *k-th* rank-1 approximation of Y and the leading rank-1 estimation of $\mathbf{Y}_k$ derived from the *k-th* deflation step, which minimizes the loss specified in (1). The second quantity evaluates the distance between the "ground-truth" deflation matrices, $\mathbf{Y}_k^\star$, and the deflation matrices, $\mathbf{Y}_k$, obtained in practice. Together, these bounds provide a comprehensive understanding of the approximation accuracy and the effectiveness of the deflation process.

The difference between $\sum_{k=1}^r \mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}$, the sum of rank-1 approximations returned by Algorithm (2), and $\mathbf{Y}$, the training data label matrix, consists of three components:

- **Ground-truth approximation error**: $\left\|\mathbf{Y} - \sum_{k=1}^r \mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}\right\|_F$, which measures the deviation between the true product $\mathbf{Y}$ and the sum of the exact top r rank-1 approximations of $\mathbf{Y}$.
- **Propagation error**: $\left\|\sum_{k=1}^r \mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X} - \sum_{k=1}^r \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F$, which accumulates the differences between the exact top rank-1 estimates of each "ground-truth" deflated matrix, $\mathbf{Y}_k^\star$, and the corresponding empirical deflated matrix, $\mathbf{Y}_k$.
- **Optimization error**: $\left\|\sum_{k=1}^r \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X} - \sum_{k=1}^r \mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F$, which captures the cumulative numerical error from the sub-routine `rank-1` not being solved exactly.

Combining these, we can express the overall difference $\left\|\mathbf{Y} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F$ as follows:

$$\begin{aligned}\left\|\mathbf{Y} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F &\leq \left\|\mathbf{Y} - \sum_{k=1}^{r} \mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X}\right\|_F + \left\|\sum_{k=1}^{r} \mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \sum_{k=1}^{r} \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F \\ &\quad + \underbrace{\left\|\sum_{k=1}^{r} \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F}_{\leq \sum_{k=1}^{r} \|\mathbf{\Psi}_k \mathbf{X}\|_F}\end{aligned} \tag{15}$$

The last inequality follows from the definition in (7). The norm $\|\mathbf{\Psi}_k\|_F$ largely depends on the sub-routine `rank-1` and can be controlled as long as t, the number of sub-routine iterations, is sufficiently large. Also, ttThe first component equals $\sum_{k=r+1}^{p} \sigma_k^\star$, which is simply the difference $\sum_{k=1}^{p} \sigma_k^\star - \sum_{k=1}^{r} \sigma_k^\star$. By the triangle inequality:

$$\begin{aligned}\left\|\mathbf{Y} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F &\leq \sum_{k=r+1}^{p} \sigma_k^\star + \left\|\sum_{k=1}^{r} \mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \sum_{k=1}^{r} \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F + \sum_{k=1}^{r} \|\mathbf{\Psi}_k \mathbf{X}\|_F \\ &\leq \sum_{k=r+1}^{p} \sigma_k^\star + \sum_{k=1}^{r} \left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F + \sum_{k=1}^{r} \|\mathbf{\Psi}_k \mathbf{X}\|_F .\end{aligned} \tag{16}$$

Therefore, our analysis will primarily focus on upper-bounding $\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F$. Intuitively, this depends on the difference between $\mathbf{Y}_k$ and $\mathbf{Y}_k^\star$. As a foundational step in our proof, we will first provide a characterization of $\|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$.

Lemma 2. *Let $\mathbf{Y}_k^\star$'s and $\mathbf{Y}_k$'s be defined as generated by Algorithm 1 and Algorithm 2, respectively. Then we have that for all $k \in [r]$,*

$$\|\mathbf{Y}_{k+1} - \mathbf{Y}_{k+1}^\star\|_F \leq \|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F + \|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\|_F + \|\boldsymbol{\delta}_k \mathbf{X}\|_F. \tag{17}$$

Lemma 2 upper-bounds $\left\|\mathbf{Y}_{k+1} - \mathbf{Y}_{k+1}^\star\right\|_F$ in terms of $\|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$, $\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F$, and $\|\boldsymbol{\delta}_k \mathbf{X}\|_F$. To establish a recursive characterization of $\|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$, we first need to derive an upper bound for $\left\|\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X} - \mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X}\right\|_F$.

Lemma 3. *Let $\mathcal{T}_k := \min\{\min_{j>k} |\sigma_k^\star - \sigma_{j_k}|, \sigma_k^\star\}$. If $\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_2 < \min_{j>k} |\sigma_k^\star - \sigma_j^\star|$, then for all $k \in [r]$, we have:*

$$\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top} \mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F \leq \left(\frac{3\sigma_k^\star}{\mathcal{T}_k} + 1\right) \|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F . \tag{18}$$

Plugging 18 into 17 gives a recurrence that depends purely on $\|\mathbf{\Psi}_k\|_F$ and the spectrum of $\mathbf{Y}$:

$$\begin{aligned}\left\|\mathbf{Y}_{k+1} - \mathbf{Y}_{k+1}^\star\right\|_F &\leq \|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F + \left(\frac{3\sigma_k^\star}{\mathcal{T}_k} + 1\right) \|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F + \|\boldsymbol{\delta}_k \mathbf{X}\|_F \\ &\leq \left(\frac{3\sigma_k^\star}{\mathcal{T}_k} + 2\right) \|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F + \|\boldsymbol{\delta}_k \mathbf{X}\|_F .\end{aligned} \tag{19}$$

Unrolling this recurrence gives a closed-form upper bound for $\|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$. Combining this upper bound with 3 and plugging the result into 16 gives us and upper bound for $\|\mathbf{Y} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\|_F$.

B Missing proofs from Appendix A and a generalized theorem

B.1 Proof of Lemma 1

Proof. Let $\{(\mathbf{a}_k^\star, \mathbf{b}_k^\star)\}_{k=1}^{r}$ denote the outputs of Algorithm 1. According to the third line in Algorithm 1, starting from $k = 1$, we have:

$$(\mathbf{a}_1^\star, \mathbf{b}_1^\star) = \underset{\mathbf{a}\in\mathbb{R}^d, \mathbf{b}\in\mathbb{R}^m}{\arg\min} \frac{1}{2} \left\|\mathbf{Y}_1^\star - \mathbf{b}\mathbf{a}^\top \mathbf{X}\right\|_F^2 .$$

Given that $\mathbf{b}\mathbf{a}^\top$ is a rank-1 matrix, the product $\mathbf{b}\mathbf{a}^\top\mathbf{X}$ is also rank-1. Therefore, according to the Eckart-Young-Mirsky theorem, the best rank-1 approximation of $\mathbf{Y}_1^\star = \mathbf{Y} = \sum_{i=1}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top}$, based on its Singular Value Decomposition, is $\sigma_1^\star \mathbf{u}_1^\star \mathbf{v}_1^{\star\top}$. Thus, $\mathbf{b}_1^\star \mathbf{a}_1^{\star\top}\mathbf{X} = \sigma_1^\star \mathbf{u}_1^\star \mathbf{v}_1^{\star\top}$. Next, the deflation step in line 4 yields:

$$\mathbf{Y}_2^\star := \mathbf{Y}_1^\star - \mathbf{b}_1^\star \mathbf{a}_1^{\star\top}\mathbf{X} = \mathbf{Y}_1^\star - \sigma_1^\star \mathbf{u}_1^\star \mathbf{v}_1^{\star\top} = \sum_{i=2}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top}.$$

Now, assuming by induction that $\mathbf{Y}_k^\star = \sum_{i=k}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top}$ and $\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} = \sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^{\star\top}$, we obtain for $k+1$:

$$\mathbf{Y}_{k+1}^\star = \mathbf{Y}_k^\star - \mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} = \sum_{i=k}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top} - \sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^{\star\top} = \sum_{i=k+1}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top}.$$

Returning to line 3, we have:

$$(\mathbf{a}_{k+1}^\star, \mathbf{b}_{k+1}^\star) = \underset{\mathbf{a}\in\mathbb{R}^d, \mathbf{b}\in\mathbb{R}^m}{\arg\min} \frac{1}{2}\left\|\mathbf{Y}_{k+1}^\star - \mathbf{b}\mathbf{a}^\top\mathbf{X}\right\|_F^2.$$

By the same reasoning as for $k=1$, $\mathbf{b}_{k+1}^\star \mathbf{a}_{k+1}^{\star\top}\mathbf{X}$ is equal to the first rank-1 approximation of $\mathbf{Y}_{k+1}^\star = \sum_{i=k+1}^p \sigma_i^\star \mathbf{u}_i^\star \mathbf{v}_i^{\star\top}$, which is $\sigma_{k+1}^\star \mathbf{u}_{k+1}^\star \mathbf{v}_{k+1}^{\star\top}$. □

B.2 Proof of Lemma 2

Proof. As we defined in (7), we'll have:

$$\mathbf{b}_k \mathbf{a}_k^\top = \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top + \mathbf{\Psi}_k$$

Plugging this into (2) we'll get:

$$\mathbf{Y}_{k+1} = \mathbf{Y}_k - (\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top + \mathbf{\Psi}_k)\mathbf{X}$$

Now let $\mathbf{\Delta}_k = \|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F$

$$\begin{aligned}
\mathbf{\Delta}_{k+1} =& \|\mathbf{Y}_{k+1} - \mathbf{Y}_{k+1}^\star\|_F \\
=& \|\mathbf{Y}_k - (\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top + \mathbf{\Psi}_k)\mathbf{X} - \mathbf{Y}_k^\star + \mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X}\|_F \\
=& \|\mathbf{Y}_k - \mathbf{Y}_k^\star - (\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top - \mathbf{b}_k^\star \mathbf{a}_k^{\star\top})\mathbf{X} - \mathbf{\Psi}_k\mathbf{X}\|_F \\
\leq& \|\mathbf{Y}_k - \mathbf{Y}_k^\star\|_F + \|(\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top - \mathbf{b}_k^\star \mathbf{a}_k^{\star\top})\mathbf{X}\|_F + \|\mathbf{\Psi}_k\mathbf{X}\|_F \\
\leq& \mathbf{\Delta}_k + \|\overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top\mathbf{X} - \mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X}\|_F + \|\mathbf{\Psi}_k\mathbf{X}\|_F
\end{aligned}$$

□

B.3 Proof of Lemma 3

Proof. We start by observing:

$$\begin{aligned}
\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F &= \|\sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \sigma_{1_k}\mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F \\
&= \|\sigma_k^\star(\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top) + (\sigma_k^\star - \sigma_{1_k})\mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F \\
&\leq \|\sigma_k^\star\|_F \cdot \|\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F + \|\sigma_k^\star - \sigma_{1_k}\|_F \cdot \underbrace{\|\mathbf{u}_{1_k}\|_2}_{=1} \cdot \underbrace{\|\mathbf{v}_{1_k}^\top\|_2}_{=1} \\
&\leq |\sigma_k^\star| \cdot \|\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F + |\sigma_k^\star - \sigma_{1_k}|.
\end{aligned}$$

Therefore,

$$\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top\mathbf{X}\|_F \leq \sigma_k^\star \cdot \|\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F + |\sigma_k^\star - \sigma_{1_k}|. \tag{20}$$

Now, we express the term $\|\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F$ as follows:

$$\begin{aligned}\|\mathbf{u}_k^\star \mathbf{v}_k^{\star\top} - \mathbf{u}_{1_k}\mathbf{v}_{1_k}^\top\|_F &= \|\mathbf{u}_k^\star(\mathbf{v}_k^{\star\top} - \mathbf{v}_{1_k}^\top) + (\mathbf{u}_k^\star - \mathbf{u}_{1_k})\mathbf{v}_{1_k}^\top\|_F \\ &\leq \underbrace{\|\mathbf{u}_k^\star\|_2}_{=1} \cdot \|\mathbf{v}_k^{\star\top} - \mathbf{v}_{1_k}^\top\|_2 + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2 \cdot \underbrace{\|\mathbf{v}_{1_k}^\top\|_2}_{=1} \\ &\leq \|\mathbf{v}_k^{\star\top} - \mathbf{v}_{1_k}^\top\|_2 + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2\end{aligned}$$

Substituting this result into inequality equation 20 gives:

$$\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\|_F \leq \sigma_k^\star\left(\|\mathbf{v}_k^{\star\top} - \mathbf{v}_{1_k}^\top\|_2 + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2\right) + \underbrace{|\sigma_k^\star - \sigma_{1_k}|}_{\leq\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F}. \tag{21}$$

The last inequality follows from Weyl's theorem. To complete the bound, we will now find an upper bound for $\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\| + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|$. To do so, we will define two angles:

$$\alpha := \angle\{\mathbf{v}_k^\star, \mathbf{v}_{1_k}\} \quad \text{and} \quad \beta := \angle\{\mathbf{u}_k^\star, \mathbf{u}_{1_k}\}$$

We know that

$$\cos\alpha = \frac{\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k}}{\|\mathbf{v}_k^\star\|_2\|\mathbf{v}_{1_k}\|_2} = \mathbf{v}_k^{\star\top}\mathbf{v}_{1_k} \leq 1$$

since $\|\mathbf{v}_k^\star\|_2 = 1$ and $\|\mathbf{v}_{1_k}\|_2 = 1$. Therefore

$$\sin^2\alpha = 1 - \cos^2\alpha = 1 - (\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k})^2 \Rightarrow (\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k})^2 = 1 - \sin^2\alpha$$

We use the expansion of the square of $\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\|_2$ to get:

$$\begin{aligned}\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\|_2^2 &= \underbrace{\|\mathbf{v}_k^\star\|_2^2}_{=1} + \underbrace{\|\mathbf{v}_{1_k}\|_2^2}_{=1} - 2(\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k}) \\ &= 2 - 2(\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k}) \\ &\leq 2 - 2(\mathbf{v}_k^{\star\top}\mathbf{v}_{1_k})^2 = 2 - 2(1 - \sin^2\boldsymbol{\alpha}) \\ &= 2\sin^2\alpha\end{aligned}$$

Thus

$$\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\|_2^2 \leq 2\sin^2\alpha \tag{22}$$

Following the same procedure for β, we start with:

$$\cos\beta = \frac{\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k}}{\|\mathbf{u}_k^\star\|_2\|\mathbf{u}_{1_k}\|_2} = \mathbf{u}_k^{\star\top}\mathbf{u}_{1_k} \leq 1$$

since $\|\mathbf{u}_k^\star\|_2 = 1$ and $\|\mathbf{u}_{1_k}\|_2 = 1$. Therefore

$$\sin^2\beta = 1 - \cos^2\beta = 1 - (\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k})^2 \Rightarrow (\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k})^2 = 1 - \sin^2\beta$$

We use the expansion of the square of $\|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2$ to get:

$$\begin{aligned}\|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2^2 &= \underbrace{\|\mathbf{u}_k^\star\|_2^2}_{=1} + \underbrace{\|\mathbf{u}_{1_k}\|_2^2}_{=1} - 2(\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k}) \\ &= 2 - 2(\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k}) \\ &\leq 2 - 2(\mathbf{u}_k^{\star\top}\mathbf{u}_{1_k})^2 \\ &= 2 - 2(1 - \sin^2\beta) = 2\sin^2\beta\end{aligned}$$

Thus

$$\|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2^2 \leq 2\sin^2\beta \tag{23}$$

Using the fact that $a + b \leq \sqrt{2a^2 + 2b^2}$, 22, and 23 we'll get:

$$\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\|_2 + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2 \leq \sqrt{2\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\|_2^2 + 2\|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\|_2^2} \leq 2\sqrt{\sin^2\alpha + \sin^2\beta} \tag{24}$$

To proceed, we observe that under the assumption $\|Y_k^\star - Y_k\|_2 < \min_{j>k}|\sigma_k^\star - \sigma_j^\star|$, Weyl's inequality implies that

$$\min_{j>k}|\sigma_k^\star - \sigma_{j_k}| \geq \min_{j>k}|\sigma_k^\star - \sigma_j^\star| - |\sigma_{j_k} - \sigma_j^\star| > \|Y_k^\star - Y_k\|_2 - \|Y_k^\star - Y_k\|_2 = 0.$$

We know $\sigma_k^\star > 0$. Since we define $\mathcal{T}_k := \min\{\min_{j>k}|\sigma_k^\star - \sigma_{j_k}|, \sigma_k^\star\}$, and both terms are positive, $\mathcal{T}_k$ is also positive. This satisfies the conditions of Wedin's theorem, allowing us to apply it.

Now since $\alpha := \angle\{\mathbf{v}_k^\star, \mathbf{v}_{1_k}\}$ and $\beta := \angle\{\mathbf{u}_k^\star, \mathbf{u}_{1_k}\}$, and because of the fact that $\mathbf{v}_k^\star$ and $\mathbf{u}_k^\star$ are the first right and left singular vectors of $\mathbf{Y}_k^\star$, and $\mathbf{v}_{1_k}$ and $\mathbf{u}_{1_k}$ are the first right and left singular vectors of $\mathbf{Y}_k$, we can apply Wedin's theorem to get:

$$\begin{aligned}
\sin^2\alpha + \sin^2\beta &\leq \frac{\|\mathbf{v}_k^{\star\top}(\mathbf{Y}_k^\star - \mathbf{Y}_k)\|_F^2 + \|(\mathbf{Y}_k^\star - \mathbf{Y}_k)\mathbf{u}_k^\star\|_F^2}{\mathcal{T}_k^2} \\
&\leq \frac{\|\mathbf{v}_k^{\star\top}\|_F^2\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F^2 + \|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F^2\|\mathbf{u}_k^\star\|_F^2}{\mathcal{T}_k^2} \\
&\leq \frac{2\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F^2}{\mathcal{T}_k^2}
\end{aligned}$$

Plugging this in 24, we'll have:

$$\|\mathbf{v}_k^\star - \mathbf{v}_{1_k}\| + \|\mathbf{u}_k^\star - \mathbf{u}_{1_k}\| \leq 2\sqrt{2}\frac{\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F}{\mathcal{T}_k} \leq 3\frac{\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F}{\mathcal{T}_k}$$

Plugging this to 21 and using Weyl's inequality, we'll get:

$$\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\|_F \leq \sigma_k^\star\left(3\frac{\|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F}{\mathcal{T}_k}\right) + \|\mathbf{Y}_k^\star - \mathbf{Y}_k\|_F = \left(\frac{3\sigma_k^\star}{\mathcal{T}_k} + 1\right)\|\mathbf{Y}_k^\star - \mathbf{Y}_k\| \tag{25}$$

□

B.4 Proof of a generalization of Theorem 1

In this section, we proof a more general form of Theorem 1, stated below.

Theorem 7. *Let $\{(\mathbf{a}_k, \mathbf{b}_k)\}_{k=1}^r$ be the output of Algorithm 2. Let $\mathbf{\Psi}_k$ be given as in Definition 1 with $\|\mathbf{\Psi}_k\|_F > 0$. Let $\mathbf{Y} = \sum_{k=1}^{r^\star}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^{\star\top}$ be the SVD of $\mathbf{Y}$, with $\sigma_1^\star \geq \cdots \geq \sigma_{r^\star}^\star$ and define $\mathbf{Y}^{(r')} = \sum_{k=1}^{r'}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^{\star\top}$. Define the minimum singular value gap $\mathcal{T}_k^\star := \min\left\{\min_{j>k}|\sigma_k^\star - \sigma_j^\star|, \sigma_k^\star\right\}$. Define error bound $E(k)$ as:*

$$E(k) := \sigma_{\max}(\mathbf{X})\sum_{k'=0}^{k-1}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k-1}\left(\frac{6\sigma_j^\star}{\mathcal{T}_j^\star} + 2\right)$$

If $\|\mathbf{\Psi}_k\|_F$'s are small enough such that $E(k) < \frac{1}{2}\min_{j>k}|\sigma_k^\star - \sigma_j^\star|$, then for any $r \leq r' \leq r^\star$, the output of Algorithm 2 satisfies:

$$\left\|\mathbf{Y}^{(r')} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \sum_{k=r+1}^{r'}\sigma_k^\star + \sigma_{\max}(\mathbf{X})\sum_{k=1}^{r}\sum_{k'=0}^{k}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k}\left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right) \tag{26}$$

Notice that Theorem 7 naturally implies Theorem 1 by taking $r' = r^\star$. We give the proof of Theorem 7 below.

Proof. We decompose the approximation error as

$$\left\|\mathbf{Y}^{(r')}-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \left\|\mathbf{Y}^{(r')}-\sum_{k=1}^{r}\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}\right\|_F + \left\|\sum_{k=1}^{r}\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\sum_{k=1}^{r}\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F + \left\|\sum_{k=1}^{r}\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \tag{27}$$

By Lemma 1, we must have that

$$\left\|\mathbf{Y}^{(r')}-\sum_{k=1}^{r}\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}\right\|_F = \left\|\sum_{k=1}^{r'}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^\star-\sum_{k=1}^{r}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^\star\right\|_F \leq \sum_{k=r+1}^{r'}\sigma_k^\star \tag{28}$$

Moreover, by Definition 1, we have

$$\left\|\sum_{k=1}^{r}\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \sum_{k=1}^{r}\left\|\mathbf{\Psi}_k\mathbf{X}\right\|_F \leq \sigma_{\max}(\mathbf{X})\sum_{k=1}^{r}\left\|\mathbf{\Psi}_k\right\|_F \tag{29}$$

Therefore, it suffice to study the second term in (27). Towards this end, we use the result in Lemma 2 and Lemma 3 to obtain

$$\left\|\mathbf{Y}_{k+1}-\mathbf{Y}_{k+1}^\star\right\|_F \leq \left\|\mathbf{Y}_k-\mathbf{Y}_k^\star\right\|_F + \left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F + \left\|\boldsymbol{\delta}_k\mathbf{X}\right\|_F \tag{30}$$

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F \leq \left(\frac{3\sigma_k^\star}{\mathcal{T}_k}+1\right)\left\|\mathbf{Y}_k^\star-\mathbf{Y}_k\right\|_F. \tag{31}$$

Combining (30) and (31), we have that

$$\left\|\mathbf{Y}_{k+1}-\mathbf{Y}_{k+1}^\star\right\|_F \leq \left(\frac{3\sigma_k^\star}{\mathcal{T}_k}+2\right)\left\|\mathbf{Y}_k^\star-\mathbf{Y}_k\right\|_F + \left\|\boldsymbol{\delta}_k\mathbf{X}\right\|_F$$

Let the sequence $\{Q_k\}_{k=1}^r$ be defined as

$$Q_{k+1}=\mathbf{a}_kQ_k+\mathbf{b}_k;\quad Q_0=0;\quad \mathbf{a}_k=2+\frac{3\sigma_k^\star}{\mathcal{T}_k};\quad \mathbf{b}_k=\left\|\boldsymbol{\delta}_k\mathbf{X}\right\|$$

Then by inequality 19 we must have that $\|\mathbf{Y}_k-\mathbf{Y}_k^\star\|_F \leq Q_k$ for all k. Invoking lemma 5 gives:

$$\begin{aligned}\|\mathbf{Y}_k-\mathbf{Y}_k^\star\|_F &\leq \sum_{k'=0}^{k-1}\|\mathbf{\Psi}_{k'}\mathbf{X}\|_F\prod_{j=k'+1}^{k-1}\left(2+\frac{3\sigma_j^\star}{\mathcal{T}_k}\right)\\ &\leq \underbrace{\sigma_{\max}(\mathbf{X})\sum_{k'=0}^{k-1}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k-1}\left(2+\frac{3\sigma_j^\star}{\mathcal{T}_j}\right)}_{:=\hat{E}(k)}\end{aligned} \tag{32}$$

We define the right hand side to be equal to $\hat{E}(k)$. Enforcing $\hat{E}(k) \leq \frac{1}{2}\min_{j>k}|\sigma_k^\star-\sigma_j^\star|$ gives that

$$\|\mathbf{Y}_k-\mathbf{Y}_k^\star\|_2 \leq \frac{1}{2}\min_{j>k}|\sigma_k^\star-\sigma_j^\star| \leq \min_{j>k}|\sigma_k^\star-\sigma_j^\star| \tag{33}$$

So the condition in Lemma 3 is met. Combining (31) and (32), and notice that $\|\mathbf{\Psi}_k\mathbf{X}\|_F \leq \sigma_{\max}(\mathbf{X})\|\mathbf{\Psi}_k\|_F$, we have:

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\overline{\mathbf{b}}_k\overline{\mathbf{a}}_k^\top\mathbf{X}\right\|_F \leq \sigma_{\max}(\mathbf{X})\left(\frac{3\sigma_k^\star}{\mathcal{T}_k}+1\right)\sum_{k'=0}^{k-1}\|\mathbf{\Psi}_{k'}\|_F\prod_{j=k'+1}^{k-1}\left(2+\frac{3\sigma_j^\star}{\mathcal{T}_k}\right) \tag{34}$$

Notice that $1 + \frac{3\sigma_k^\star}{\mathcal{T}_k} \leq 2 + \frac{3\sigma_k^\star}{\mathcal{T}_k}$. Therefore, 34 becomes:

$$\left\|\mathbf{b}_k^\star \mathbf{a}_k^{\star\top}\mathbf{X} - \overline{\mathbf{b}}_k \overline{\mathbf{a}}_k^\top \mathbf{X}\right\|_F \leq \sigma_{\max}(\mathbf{X}) \sum_{k'=0}^{k-1} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{3\sigma_j^\star}{\mathcal{T}_k}\right) \tag{35}$$

Combining (28), (29), and (35) gives

$$\begin{aligned}
\left\|\mathbf{Y}^{(r')} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F &\leq \sum_{k=r+1}^{r'} \sigma_k^\star + \sigma_{\max}(\mathbf{X}) \sum_{k=1}^{r} \|\mathbf{\Psi}_k\|_F \\
&\quad + \sigma_{\max}(\mathbf{X}) \sum_{k=1}^{r} \sum_{k'=0}^{k-1} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{3\sigma_j^\star}{\mathcal{T}_k}\right) \\
&= \sum_{k=r+1}^{r'} \sigma_k^\star + \sigma_{\max}(\mathbf{X}) \sum_{k=1}^{r} \sum_{k'=0}^{k} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{3\sigma_j^\star}{\mathcal{T}_k}\right)
\end{aligned}$$

Finally, due to (33) and the Weyl's inequality, we must have that $|\sigma_{j_k} - \sigma_k^\star| \leq \frac{1}{2} \min_{j>k} \left|\sigma_k^\star - \sigma_j^\star\right|$. Thus, we have that $\mathcal{T}_k \geq \frac{1}{2}\mathcal{T}_k^\star$. This allows us to define

$$E(k) = \sigma_{\max}(\mathbf{X}) \sum_{k'=0}^{k-1} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k-1} \left(\frac{6\sigma_j^\star}{\mathcal{T}_j^\star} + 2\right)$$

and obtain that $E(k) \geq \hat{E}(k)$. Thus, enforcing $E(k) \leq \frac{1}{2} \min_{j>k} \left|\sigma_k^\star - \sigma_j^\star\right|$ suffices. Moreover, we have

$$\left\|\mathbf{Y}^{(r')} - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top \mathbf{X}\right\|_F \leq \sum_{k=r+1}^{r'} \sigma_k^\star + \sigma_{\max}(\mathbf{X}) \sum_{k=1}^{r} \sum_{k'=0}^{k} \|\mathbf{\Psi}_{k'}\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)$$

which completes the proof. □

C Proof of Theorem 5

Proof overview. A rough idea of showing this theorem can build upon our characterization of the training error. Let $\hat{\mathbf{b}}_k^\star$'s be output of Algorithm 1 when using $\mathbf{Y}^\star$ as the label. We consider the following orthonormal basis of $\mathbb{R}^m$ extended from $\hat{\mathbf{b}}_k^\star$'s:

$$\hat{\mathbf{b}}_1, \ldots, \hat{\mathbf{b}}_m; \;\; \hat{\mathbf{b}}_i = \hat{\mathbf{b}}_k^\star / \|\hat{\mathbf{b}}_k^\star\|_2 \text{ if } k \leq r^\star$$

Let $\hat{\mathbf{B}} \in \mathbb{R}^{m \times r^\star}$ consist of $\hat{\mathbf{b}}_1, \ldots \hat{\mathbf{b}}_{r^\star}$, and let $\hat{\mathbf{B}}^\perp \in \mathbb{R}^{m \times (m-r^\star)}$ consist of $\hat{\mathbf{b}}_{r^\star+1}, \ldots \hat{\mathbf{b}}_m$. Then, we can write $\mathbf{Y}^\star$ as:

$$\mathbf{Y} = \mathbf{W}^\star \mathbf{X} + \hat{\mathbf{B}}\hat{\mathbf{B}}^\top \boldsymbol{\mathcal{E}} + \hat{\mathbf{B}}^\perp \hat{\mathbf{B}}^{\perp\top} \boldsymbol{\mathcal{E}} = \hat{\mathbf{B}}(\mathbf{\Sigma}\hat{\mathbf{A}}^\top \mathbf{X} + \boldsymbol{\mathcal{E}}_1) + \hat{\mathbf{B}}^\perp \boldsymbol{\mathcal{E}}_2,$$

where $\boldsymbol{\mathcal{E}}_1 \in \mathbb{R}^{r^\star \times n}$ and $\boldsymbol{\mathcal{E}}_2 \in \mathbb{R}^{(m-r^\star) \times n}$ are noise matrices with I.I.D. Gaussian entries. Therefore, based on the above decomposition, $\boldsymbol{\mathcal{E}}_1$ can be seen as the unavoidable noise, which will adds up to the training error, and $\boldsymbol{\mathcal{E}}_2$ is the error that can be avoided if we solve for only the top $r^\star$ components. Of course, $\hat{\mathbf{B}}(\mathbf{\Sigma}\hat{\mathbf{A}}^\top \mathbf{X} + \boldsymbol{\mathcal{E}}_1)$ is not the truncated top-$r^\star$ SVD of $\mathbf{Y}$ since $\mathbf{\Sigma}\hat{\mathbf{A}}^\top \mathbf{X} + \boldsymbol{\mathcal{E}}_1$ does not have orthogonal rows. However, when $\boldsymbol{\mathcal{E}}_1$ is small, this term approximates the truncated SVD well enough. Base on this intuition, we have the following lemma:

Lemma 4. *Let $\mathbf{Y}^\star$ to have the SVD $\mathbf{Y}^\star = \sum_{k=1}^{r} \hat{\sigma}_k \hat{\mathbf{u}}\hat{\mathbf{v}}^\star$, and let $\mathbf{Y} = \mathbf{Y}^\star + \boldsymbol{\mathcal{E}}$ to have SVD $\mathbf{Y} = \sum_{k=1}^{m} \sigma_k^\star \mathbf{u}_k^\star \mathbf{v}_k^\star$. Let $\mathbf{Y}^{\star(\hat{m})}$ and $\mathbf{Y}^{(\hat{m})}$ be the truncated rank-$\hat{m}$ SVD of $\mathbf{Y}^\star$ and $\mathbf{Y}$, respectively. Then with probability at least $1 - \delta$ we have that*

$$\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F \leq O\left(\varepsilon\sqrt{n \log 1/\delta}\left(\hat{m} + \sqrt{\frac{\min\{r, \hat{m}\}}{\mathcal{T}_{\min}^\star}}\right)\right)$$

The proof of Lemma 4 is given in Appendix C.2. With the help of Lemma 4, the proof of Theorem 6 involves choosing a reference label $\mathbf{Y}^{(r)}$ that involves only the relevant noise that will be fitted by Algorithm 2. We then control the generalization error by estimating the difference between $\mathbf{W}^\star\mathbf{X}$ and $\mathbf{Y}^{\star(r)}$, and the difference between $\mathbf{Y}^{\star(r)}$ and $\mathbf{Y}^{(r)}$ using Lemma 4.

C.1 More details in the proof of Theorem 5

Utilizing Lemma 3, we could get that

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \le \left(2+\frac{6\sigma_k^\star}{\mathcal{T}_k^\star}\right)+\sigma_{\max}(\mathbf{X})\left\|\mathbf{\Psi}_k\right\|_F$$

Plugging in (32), we have that

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \le \sigma_{\max}(\mathbf{X})\sum_{k'=0}^{k}\left\|\mathbf{\Psi}_{k'}\right\|_F\prod_{j=k'+1}^{k}\left(2+\frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)$$

Since $\sigma_{\min}(\mathbf{X})>0$, we can then have

$$\begin{aligned}&\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \ge \sigma_{\min}(\mathbf{X})\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}-\mathbf{b}_k\mathbf{a}_k^\top\right\|_F\\ \Rightarrow &\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}-\mathbf{b}_k\mathbf{a}_k^\top\right\|_F \le \frac{1}{\sigma_{\min}(\mathbf{X})}\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}\mathbf{X}-\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F\end{aligned}$$

This implies that

$$\left\|\mathbf{b}_k^\star\mathbf{a}_k^{\star\top}-\mathbf{b}_k\mathbf{a}_k^\top\right\|_F \le \kappa(\mathbf{X})\sum_{k'=0}^{k}\left\|\mathbf{\Psi}_{k'}\right\|_F\prod_{j=k'+1}^{k}\left(2+\frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)$$

which proves the first statememt. To prove the second statement, we directly use Theorem 1 to get that

$$\begin{aligned}\left\|\mathbf{W}^\star-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\right\|_F &\le \frac{1}{\sigma_{\min}(\mathbf{X})}\left\|\mathbf{W}^\star\mathbf{X}-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F\\ &= \frac{1}{\sigma_{\min}(\mathbf{X})}\left\|\mathbf{Y}-\sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F\\ &\le \sum_{k=r+1}^{r^\star}\frac{\sigma_k^\star}{\sigma_{\min}(\mathbf{X})}+\kappa(\mathbf{X})\sum_{k=1}^{r}\sum_{k'=1}^{k}\left\|\mathbf{\Psi}_{k'}\right\|_F\prod_{j=k'+1}^{k}\left(2+\frac{6\sigma_j^\star}{\mathcal{T}_j^\star}\right)\end{aligned}$$

C.2 Proof of Lemma 4

Proof. By Lemma 6, we have that with probability $1-\delta$

$$\sigma_{\max}\left(\boldsymbol{\mathcal{E}}\right) \le O\left(\varepsilon\left(\sqrt{n}+\sqrt{\log\frac{1}{\delta}}\right)\right)$$

To start, by Weyl's inequality, we have that

$$\left|\hat{\sigma}_k-\sigma_k^\star\right| \le \sigma_{\max}\left(\boldsymbol{\mathcal{E}}\right) \le O\left(\varepsilon\left(\sqrt{n}+\sqrt{\log\frac{1}{\delta}}\right)\right)$$

Therefore, taking $\varepsilon \le O\left(\frac{\mathcal{T}_{\min}^\star}{\sqrt{n}+\sqrt{\log\frac{1}{\delta}}}\right)$ ensures that $\min\left\{\min_{j\neq k}\left\{\left|\sigma_j-\sigma_k^\star\right|\right\},\sigma_k^\star\right\} \ge \frac{1}{2}\mathcal{T}_{\min}^\star$. Thus, by Wedin's Theorem, we have that

$$\hat{\mathbf{u}}_k^\top\mathbf{u}_k^\star+\hat{\mathbf{v}}_k^\top\mathbf{v}_k^\star \le 2-\frac{2}{\mathcal{T}_k^\star}\left(\left\|\boldsymbol{\mathcal{E}}^\top\mathbf{u}_k^\star\right\|_2^2+\left\|\boldsymbol{\mathcal{E}}\mathbf{v}_k^\star\right\|_2^2\right)$$

We will consider two cases. First, when $\hat{m} \leq r$, we have

$$\begin{aligned}
\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F &= \left\|\sum_{k=1}^{\hat{m}} \left(\sigma_k^\star \mathbf{u}^\star \mathbf{v}^\star - \hat{\sigma}_k \hat{\mathbf{u}}_k \hat{\mathbf{v}}_k\right)\right\|_F \\
&\leq \left\|\sum_{k=1}^{\hat{m}} \left(\hat{\sigma}_k - \sigma_k^\star\right) \mathbf{u}_k \mathbf{v}_k\right\|_F + \left\|\hat{\mathbf{U}}_{\hat{m}} \mathbf{\Sigma}_{\hat{m}}^\star \hat{\mathbf{V}}_{\hat{m}}^\top - \mathbf{U}_{\hat{m}}^\star \mathbf{\Sigma}_{\hat{m}}^\star \hat{\mathbf{V}}_{\hat{m}}^{\top\star}\right\|_F \\
&\leq \sum_{r=1}^{\hat{m}} |\hat{\sigma}_k - \sigma_k^\star| + \left\|\left(\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right) \mathbf{\Sigma}_{\hat{m}}^\star\right\|_F + \left\|\left(\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right) \mathbf{\Sigma}_{\hat{m}}^\star\right\|_F \\
&\leq \sum_{r=1}^{\hat{m}} |\hat{\sigma}_k - \sigma_k^\star| + \sigma_1^\star \left(\left\|\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right\|_F + \left\|\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right\|_F\right) \\
&\leq \sum_{r=1}^{\hat{m}} |\hat{\sigma}_k - \sigma_k^\star| + 2\sigma_1^\star \left(\left\|\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right\|_F^2 + \left\|\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right\|_F^2\right)^{\frac{1}{2}}
\end{aligned}$$

Notice that

$$\begin{aligned}
\left\|\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right\|_F^2 &= 2\hat{m} - 2\left\langle \hat{\mathbf{U}}_{\hat{m}}, \mathbf{U}_{\hat{m}}^\star \right\rangle = 2\hat{m} - 2\sum_{k=1}^{\hat{m}} \hat{\mathbf{u}}_k^\top \mathbf{u}_k^\star \\
\left\|\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right\|_F^2 &= 2\hat{m} - 2\left\langle \hat{\mathbf{V}}_{\hat{m}}, \mathbf{V}_{\hat{m}}^\star \right\rangle = 2\hat{m} - 2\sum_{k=1}^{\hat{m}} \hat{\mathbf{v}}_k^\top \mathbf{v}_k^\star
\end{aligned}$$

Therefore

$$\begin{aligned}
\left\|\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right\|_F^2 + \left\|\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right\|_F^2 &\leq 4k - 2\sum_{k=1}^{\hat{m}} \left(\hat{\mathbf{u}}_k^\top \mathbf{u}_k^\star + \hat{\mathbf{v}}_k^\top \mathbf{v}_k^\star\right) \\
&\leq \frac{4}{\mathcal{T}_{\min}^\star} \sum_{k=1}^{\hat{m}} \left(\left\|\boldsymbol{\mathcal{E}}^\top \mathbf{u}_k^\star\right\|_2^2 + \|\boldsymbol{\mathcal{E}} \mathbf{v}_k^\star\|_2^2\right) \\
&= \frac{4}{\mathcal{T}_{\min}^\star} \left(\left\|\boldsymbol{\mathcal{E}}^\top \mathbf{U}_{\hat{m}}^\star\right\|_F^2 + \|\boldsymbol{\mathcal{E}} \mathbf{V}_{\hat{m}}^\star\|_F^2\right)
\end{aligned}$$

Since $\boldsymbol{\mathcal{E}} \in \mathbb{R}^{m \times n}$ contains I.I.D. Gaussian entries from $\mathcal{N}(0, \varepsilon^2)$, we must have that $\mathbf{U}_k^\star \boldsymbol{\mathcal{E}} \in \mathbb{R}^{\hat{m} \times n}$ and $\boldsymbol{\mathcal{E}} \mathbf{V}_k^\star \in \mathbb{R}^{m \times \hat{m}}$ contains I.I.D. Gaussian entries from $\mathcal{N}(0, \varepsilon^2)$. By Lemma 6, we have that with probability at least $1 - \delta$, it holds that

$$\left\|\mathbf{U}_{\hat{m}}^{\star\top} \boldsymbol{\mathcal{E}}\right\|_F^2 + \|\boldsymbol{\mathcal{E}} \mathbf{V}_{\hat{m}}^\star\|_F^2 \leq O\left(\varepsilon^2 (m+n) \hat{m} \log {}^1\!/\delta\right)$$

Thus, we have

$$\left\|\hat{\mathbf{U}}_{\hat{m}} - \mathbf{U}_{\hat{m}}^\star\right\|_F^2 + \left\|\hat{\mathbf{V}}_{\hat{m}} - \mathbf{V}_{\hat{m}}^\star\right\|_F^2 \leq O\left(\frac{\varepsilon^2}{\mathcal{T}_{\min}^\star} \hat{m}(m+n) \log {}^1\!/\delta\right)$$

Combining the results above, we have

$$\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F \leq O\left(\varepsilon \hat{m} \left(\sqrt{n} + \sqrt{\log {}^1\!/\delta}\right)\right) + O\left(\frac{\varepsilon}{\sqrt{\mathcal{T}_{\min}^\star}} \sqrt{r(m+n) \log {}^1\!/\delta}\right)$$

Next, we consider the case $\hat{m} \geq r$. In this case, we have

$$\begin{aligned}
\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F &\leq \left\|\mathbf{Y}^\star - \mathbf{Y}^{(r)}\right\|_F + \left\|\sum_{k=r+1}^{\hat{m}} \sigma_k \mathbf{u}_k \mathbf{v}_k^\top\right\|_F \\
&\leq O\left(\varepsilon r \left(\sqrt{n} + \sqrt{\log {}^1\!/\delta}\right)\right) + O\left(\frac{\varepsilon}{\sqrt{\mathcal{T}_{\min}^\star}} \sqrt{r(m+n) \log {}^1\!/\delta}\right) + \sum_{k=r+1}^{\hat{m}} \sigma_k
\end{aligned}$$

Notice that by Weyl's inequality, for all $k \geq r$

$$\sigma_k \leq |\sigma_k - 0| \leq \sigma_{\max}(\boldsymbol{\mathcal{E}}) \leq O\left(\varepsilon\left(\sqrt{n} + \sqrt{\log\frac{1}{\delta}}\right)\right)$$

This gives

$$\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F \leq O\left(\varepsilon\hat{m}\left(\sqrt{n} + \sqrt{\log 1/\delta}\right)\right) + O\left(\frac{\varepsilon}{\sqrt{\mathcal{T}^\star_{\min}}}\sqrt{r(m+n)\log 1/\delta}\right)$$

Combining the two cases, and using $m \leq n$, we have that

$$\left\|\mathbf{Y}^{\star(\hat{m})} - \mathbf{Y}^{(\hat{m})}\right\|_F \leq O\left(\varepsilon\sqrt{n\log 1/\delta}\left(\hat{m} + \sqrt{\frac{\min\{r,\hat{m}\}}{\mathcal{T}^\star_{\min}}}\right)\right)$$

□

C.3 Proof of Theorem 6

Given the SVD of $\mathbf{Y}$ and $\mathbf{Y}^\star$ as $\mathbf{Y} = \sum_{k=1}^{p}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^\star$ and $\mathbf{Y}^\star = \sum_{k=1}^{r^\star}\hat{\sigma}_k\hat{\mathbf{u}}_k\hat{\mathbf{v}}_k^\top$, we define

$$\mathbf{Y}^{(r)} = \sum_{k=1}^{r}\sigma_k^\star\mathbf{u}_k^\star\mathbf{v}_k^\star;\ \ \mathbf{Y}^{\star(r)} = \sum_{k=1}^{\min\{r,r^\star\}}\hat{\sigma}_k\hat{\mathbf{u}}_k\hat{\mathbf{v}}_k^\top$$

Then we can decompose the error $\left\|\mathbf{W}^\star\mathbf{X} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F$ as

$$\left\|\mathbf{W}^\star\mathbf{X} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \left\|\mathbf{W}^\star\mathbf{X} - \mathbf{Y}^{\star(r)}\right\|_F + \left\|\mathbf{Y}^{\star(r)} - \mathbf{Y}^{(r)}\right\|_F + \left\|\mathbf{Y}^{(r)} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F$$

We will analyze each of the three terms individually. To start, for the first term, we notice that $\mathbf{Y}^{\star(r)}$ is precisely the truncated SVD of $\mathbf{W}^\star\mathbf{X}$ when $r < r^\star$. Therefore

$$\left\|\mathbf{W}^\star\mathbf{X} - \mathbf{Y}^{\star(r)}\right\|_F \leq \sum_{k=r+1}^{r^\star}\sigma_k\left(\mathbf{W}^\star\mathbf{X}\right) \leq \sigma_{\max}\left(\mathbf{X}\right)\sum_{k=r+1}^{r^\star}\sigma_r\left(\mathbf{W}^\star\right)$$

For the second term, by Lemma 4, we have that with probability at least $1-\gamma$

$$\left\|\mathbf{Y}^{\star(r)} - \mathbf{Y}^{(r)}\right\|_F \leq O\left(\varepsilon\sqrt{n\log 1/\gamma}\left(r + \sqrt{\frac{\min\{r^\star,r\}}{\mathcal{T}^\star_{\min}}}\right)\right)$$

Lastly, by Theorem 7, we have that

$$\left\|\mathbf{Y}^{(r)} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \sigma_{\max}(\mathbf{X})\sum_{k=1}^{r}\sum_{k'=0}^{k}\left\|\boldsymbol{\Psi}_{k'}\right\|_F\prod_{j=k'+1}^{k}\left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)$$

Combining the above equations, we have that

$$\left\|\mathbf{W}^\star\mathbf{X} - \sum_{k=1}^{r}\mathbf{b}_k\mathbf{a}_k^\top\mathbf{X}\right\|_F \leq \sigma_{\max}\left(\mathbf{X}\right)\sum_{k=r+1}^{r^\star}\sigma_r\left(\mathbf{W}^\star\right) + \sigma_{\max}(\mathbf{X})\sum_{k=1}^{r}\sum_{k'=0}^{k}\left\|\boldsymbol{\Psi}_{k'}\right\|_F\prod_{j=k'+1}^{k}\left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)$$
$$+ O\left(\kappa\sqrt{n\log 1/\gamma}\left(r + \sqrt{\frac{\min\{r^\star,r\}}{\mathcal{T}^\star_{\min}}}\right)\right)$$

This gives that

$$\left\|\mathbf{W}^\star - \sum_{k=1}^{r} \mathbf{b}_k \mathbf{a}_k^\top\right\|_F \leq \kappa(\mathbf{X}) \left(\sum_{k=r+1}^{r^\star} \sigma_r\left(\mathbf{W}^\star\right) + \sum_{k=1}^{r} \sum_{k'=0}^{k} \left\|\mathbf{\Psi}_{k'}\right\|_F \prod_{j=k'+1}^{k} \left(2 + \frac{6\sigma_j^\star}{\mathcal{T}_k^\star}\right)\right) + O\left(\frac{\varepsilon\sqrt{n\log{}^1\!/\gamma}}{\sigma_{\min}(\mathbf{X})}\left(r + \sqrt{\frac{\min\{r^\star, r\}}{\mathcal{T}_{\min}^\star}}\right)\right)$$

which completes the proof.

D Supporting theorems and lemmas

Lemma 5. *Consider a sequence of quantities $\{Q_k\}_{k=1}^{\infty}$ satisfying*

$$Q_{k+1} = \mathbf{a}_k Q_k + \mathbf{b}_k$$

with some $\mathbf{a}_k, \mathbf{b}_k \geq 0$ for all $k \in \mathbb{Z}^+$. Set $b_0 = Q_1$. Then we have that

$$Q_k = \sum_{k'=0}^{k-1} b_{k'} \prod_{j=k'+1}^{k-1} a_j$$

Proof. We shall prove by induction. For the base case, let $k = 1$. In this case, we have that

$$Q_1 = \sum_{k'=0}^{0} b_{k'} \prod_{j=k'+1}^{0} a_j = b_0 = Q_1$$

For the inductive case, assume that the property holds for k. Then we have that

$$Q_{k+1} = \mathbf{a}_k Q_k + \mathbf{b}_k = \mathbf{a}_k \cdot \sum_{k'=0}^{k-1} b_{k'} \prod_{j=k'+1}^{k-1} a_j + \mathbf{b}_k = \sum_{k'=0}^{k} b_{k'} \prod_{j=k'+1}^{k} a_j$$

This proves the inductive step and finishes the proof. □

Lemma 6. *Let $\mathbf{M} \in \mathbb{R}^{m\times n}$ be a matrix containing I.I.D. Gaussian entries from $\mathcal{N}(0,1)$. Then we have that with probability at least $1-\delta$, the following holds*

- $\sigma_{\max}(\mathbf{M}) \leq O\left(\sqrt{m} + \sqrt{n} + \sqrt{\log{}^1\!/\delta}\right)$
- $\|\mathbf{M}\|_F \leq O\left(\sqrt{mn\log{}^1\!/\delta}\right)$

Proof. By standard results of Gaussian random matrices and vectors, we have that

- $\mathbb{P}\left(\sigma_{\max}(\mathbf{M}) \leq O\left(\sqrt{m} + \sqrt{n} + t_1\right)\right) \geq 1 - \exp\left(-t_1^2\right)$
- $\mathbb{P}\left(\|\mathbf{M}\|_F \leq t_2\right) \geq 1 - 2\exp\left(-\frac{t_2^2}{2mn}\right)$

Take $t_1 = \sqrt{\log\frac{2}{\delta}}$ and $t_2 = \sqrt{2mn\log\frac{4}{\delta}}$ finishes the proof. □

Theorem 8 (Eckart-Young-Mirsky Theorem)**.** *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be a matrix with singular value decomposition $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$, where $\mathbf{U}$ and $\mathbf{V}$ are orthogonal matrices and $\mathbf{\Sigma}$ is a diagonal matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(m,n)} \geq 0$. For any integer $k \leq \min(m,n)$, let $\mathbf{A}_k = \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^\top$ be the best rank-k*

approximation of $\mathbf{A}$*, where* $\mathbf{U}_k$ *and* $\mathbf{V}_k$ *consist of the first* k *columns of* $\mathbf{U}$ *and* $\mathbf{V}$*, and* $\mathbf{\Sigma}_k$ *is the diagonal matrix of the largest* k *singular values.*

Then $\mathbf{A}_k$ *minimizes the approximation error in both the Frobenius norm and the spectral norm:*

$$\mathbf{A}_k = \underset{\mathbf{B}, rank(\mathbf{B})=k}{\arg\min} \|\mathbf{A} - \mathbf{B}\|_F$$

Theorem 9 (Wedin Theorem ((Wedin, 1972))). *Let* $\mathbf{M}$*,* $\tilde{\mathbf{M}} \in \mathbb{R}^{m\times n}$ *be two matrices with rank-r SVDs:*

$$\mathbf{M} = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{\Sigma}_1 & 0\\ 0 & \mathbf{\Sigma}_2\end{bmatrix}\begin{bmatrix}\mathbf{V_1}^\top\\ \mathbf{V_2}^\top\end{bmatrix}, \quad \textit{and} \quad \tilde{\mathbf{M}} = \mathbf{M} + \mathbf{\Delta} = \begin{bmatrix}\tilde{\mathbf{U}}_1 & \tilde{\mathbf{U}}_2\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{\Sigma}}_1 & 0\\ 0 & \tilde{\mathbf{\Sigma}}_2\end{bmatrix}\begin{bmatrix}\tilde{\mathbf{V}}_1^\top\\ \tilde{\mathbf{V}}_2^\top\end{bmatrix}.$$

If $\delta = \min\left\{\min_{1\le i\le r, r+1\le j\le n} |\sigma_i - \tilde{\sigma}_j|, \min_{1\le i\le r} \sigma_i\right\} > 0$*, then:*

$$\left\|\sin\theta(\tilde{\mathbf{U}}_1, \mathbf{U}_1)\right\|_F^2 + \left\|\sin\theta(\tilde{\mathbf{V}}_1, \mathbf{V}_1)\right\|_F^2 \le \frac{\left\|\mathbf{U}_1^\top \mathbf{\Delta}\right\|_F^2 + \|\mathbf{\Delta}\mathbf{V}_1\|_F^2}{\delta^2}$$

Theorem 10 (Weyl's Theorem for Singular Values((Weyl, 1912))). *Let* $\mathbf{M}$ *and* $\mathbf{\Delta}$ *be* $m \times n$ *matrices. If* $\tilde{\mathbf{M}} = \mathbf{M} + \mathbf{\Delta}$*, then the singular values* σ_i *of* $\mathbf{M}$ *and the singular values* $\tilde{\sigma}_i$ *of* $\tilde{\mathbf{M}}$ *satisfy the following inequality for all* $i = 1, 2, \ldots, \min(m, n)$*:*

$$|\tilde{\sigma}_i - \sigma_i| \le \|\mathbf{\Delta}\|_2,$$

E Experimental analysis on linear matrix regression.

We present experiments that validate our theory on error propagation in sequential rank-1 learning. Our experiments aim to demonstrate how the distribution of computational resources across rank-1 components affects the overall approximation quality, particularly focusing on how errors in early components propagate to later stages of the sequential learning process.

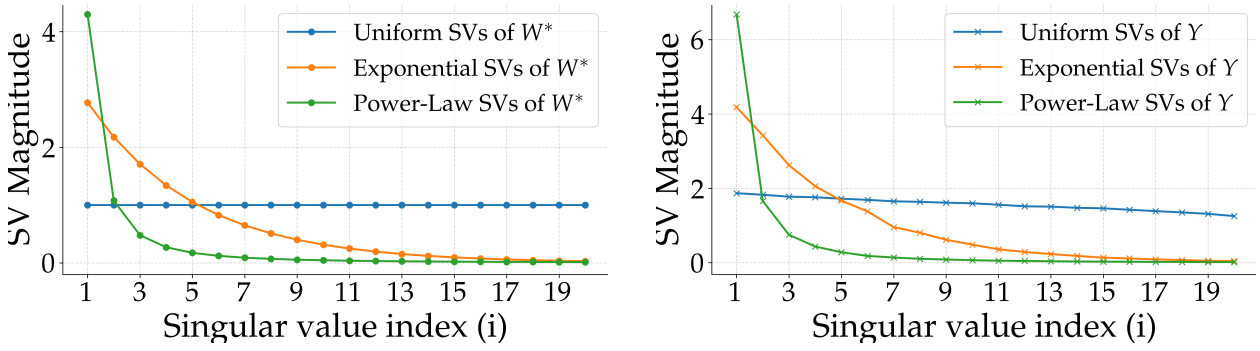
Problem setting. Per our theory, we consider the low-rank linear regression problem of finding $\mathbf{W}^\star \in \mathbb{R}^{m\times n}$ with rank $\le r$ such that $\mathbf{Y} = \mathbf{W}^\star\mathbf{X} + \boldsymbol{\mathcal{E}}$ where $\boldsymbol{\mathcal{E}}$ is the noise term. This corresponds to finding a low-rank approximation of $\mathbf{W}^\star$. We investigate the following settings:

1. *Singular Value Profiles:* We vary the singular value distribution of $\mathbf{W}^\star$ to analyze how the spectrum of ground truth influences error propagation.
2. *Noise Variations:* We introduce different types and levels of noise to assess the robustness of sequential rank-1 learning to perturbations.
3. *Iteration allocation strategies:* We evaluate three different iteration allocation strategies:
   (a) **Equal:** Same number of optimization iterations to each rank-1 component.
   (b) **More First:** More iterations allocated to the earlier components and fewer to later ones.
   (c) **Less First:** Fewer iterations allocated to the earlier components and more to later ones.

To ensure statistical robustness, all experiments are repeated across 5 independent trials. We report the mean performance across these trials, and visualize variability using shaded bands that represent the standard deviation.

We consider matrix dimensions $\mathbf{W}^\star \in \mathbb{R}^{500\times 1000}$; we observed that experiments varying the dimensions of $\mathbf{W}^\star$ do not introduce any additional value to the main messages of this section. We generate $\mathbf{W}^\star$ with different singular value profiles, as in:

- **Uniform:** $\sigma_i = 10$ for all $i = 1, \ldots, r^\star$;
- **Exponential decay:** $\sigma_i = 100 \cdot \left(\frac{1}{100}\right)^{\frac{i-1}{r^\star - 1}}$ for $i = 1, \ldots, r^\star$;



(a) Singular values of $\mathbf{W}^\star$. (b) Singular values of $\mathbf{Y} = \mathbf{W}^\star\mathbf{X}$.

Figure 6: Comparison of singular value decay under different profiles. *Left:* $\mathbf{W}^\star$. *Right:* $\mathbf{Y} = \mathbf{W}^\star\mathbf{X}$.

- **Power-law decay:** $\sigma_i = \frac{100}{i^2}$ for $i = 1, \ldots, r^\star$;

where $r^\star$ is the true rank of $\mathbf{W}^\star$. Without loss of generality, we fix the rank $r^\star$ to 20 as we did not observe unexpected behaviors in the performance of the algorithms for different rank values.

We also consider several noise scenarios to evaluate robustness: *i*) Noiseless; *ii*) Gaussian where $\boldsymbol{\mathcal{E}}$ has i.i.d. entries from $\mathcal{N}(0, \kappa)$ with $\kappa \in \{0.01, 0.05, 0.1\}$, and *iii*) Sparse where $\boldsymbol{\mathcal{E}}$ is a sparse matrix (in our case 5% of entries are non-zeros) with non-zero entries from $\mathcal{N}(0, \kappa)$ with $\kappa \in \{1, 10\}$. Per our theory, $\mathbf{X}$ is sampled from a standard Gaussian distribution, $\mathcal{N}(0, 1)$.

Effect of singular value profile. We study whether the singular value profile of $\mathbf{W}^\star$ has impact on error propagation through the term $\mathcal{T}_k^\star = \min\{\min_{j>k} |\sigma_k^\star - \sigma_j^\star|, \sigma_k^\star\}$ in our error bound. Figure 6 shows the singular value decay patterns of both $\mathbf{W}^\star$ and the resulting $\mathbf{Y}$ under different spectral profiles. Figure 7 illustrates the training and reconstruction errors under these three profiles.

To ensure a fair comparison across different spectral profiles, we normalize the singular values of $\mathbf{W}^\star$ such that all generated matrices have the same Frobenius norm. This avoids artificially inflating or deflating error magnitudes due to differences in matrix scale rather than the structure of singular value decay.

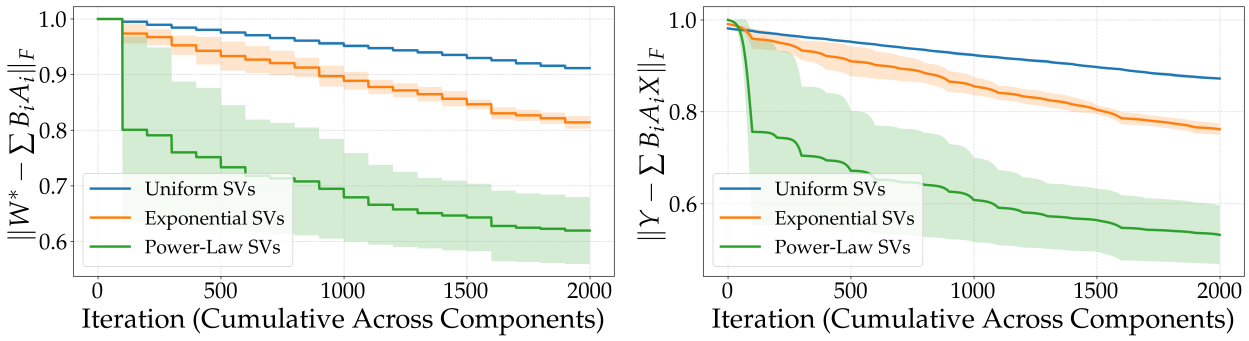


Figure 7: Effect of singular value profile on sequential learning performance. *Left*: $\mathbf{W}^\star$ reconstruction error. *Right*: Objective's training error.

Observations: The power-law decay profile shows the best performance, followed by the exponential decay, with the uniform profile performing worst. This matches the theoretical insight that large singular value gaps reduce the compounding of downstream error. Notably, power-law decay starts steep at the head—its first few singular values are significantly larger—creating large gaps for early components. In contrast, exponential decay is smoother initially and decays more evenly. Uniform singular values exhibit no decay, leading to minimal or zero gaps throughout.

Impact of noise level. Our theoretical analysis extends to noisy settings through Theorem 6, which characterizes how additive noise impacts generalization performance. Figure 8 illustrates the effect of increasing noise levels κ on both the training and reconstruction error under Gaussian and sparse noise levels.

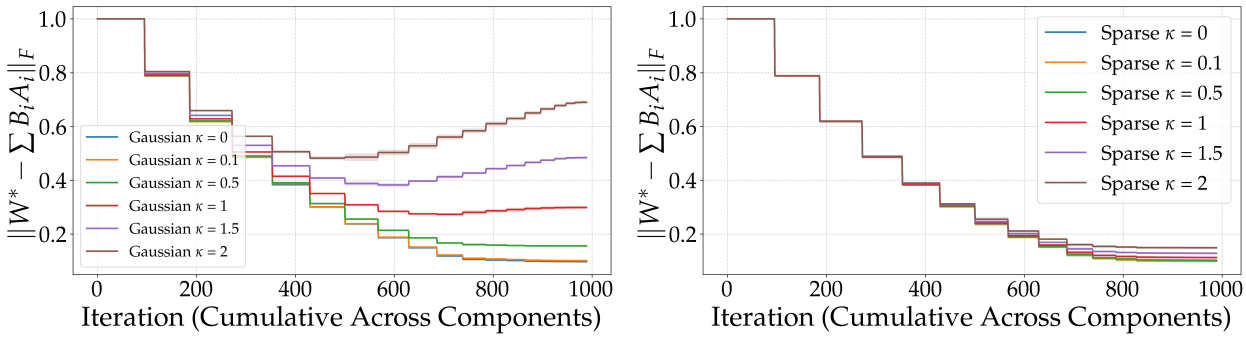


Figure 8: Impact of noise level. *Left:* Gaussian noise. *Right:* Sparse noise.

Observations: As expected, increasing the noise level κ leads to higher reconstruction error in both Gaussian and sparse settings. Higher noise levels tend to corrupt the smaller singular values of $\mathbf{Y}$, making it difficult to distinguish low-rank structure from noise. This can lead to overfitting in later components of the sequential learner, as the algorithm begins to capture noise rather than signal.

Effect of iteration allocation strategies in noisy settings. To investigate mitigation strategies, we first evaluate how different iteration allocation strategies perform under noisy conditions. Figure 9 shows that the "more-first" strategy consistently outperforms others across varying noise levels by concentrating effort where it matters most—early in the sequence.

Observation: Even in noisy settings, the *more-first* strategy consistently outperforms *equal*, which in turn outperforms *less-first*, across all noise levels κ. This highlights the importance of prioritizing early iterations to mitigate error amplification under noise.

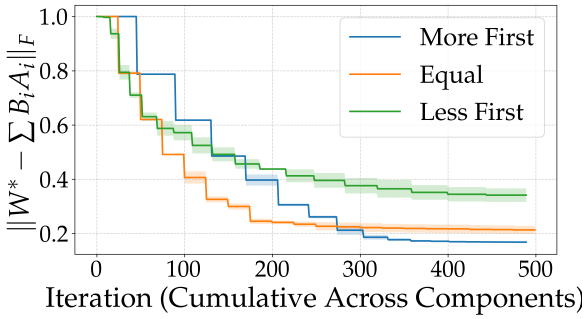
Effect of singular value profiles in noisy settings. We further examine how spectral decay influences robustness under noise. Using the *more-first* allocation strategy, Figure 10 shows that power-law decay consistently achieves lower reconstruction error compared to exponential and uniform profiles across all noise levels κ.

Observation: Spectral decay plays a critical role in robustness. Power-law decay, with its large leading singular values and wider gaps, allows early components to capture most of the signal, mitigating downstream error propagation. In contrast, uniform profiles lack this protective structure, making them more vulnerable to noise.

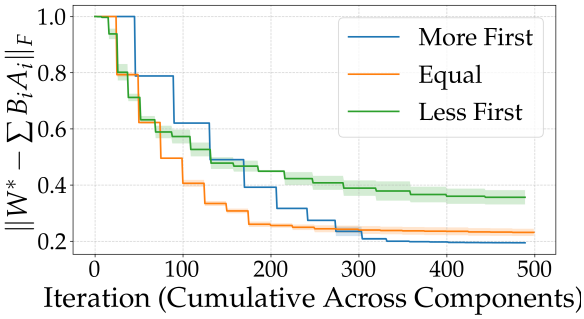
Implications for Practical Use: These results suggest that sequential learners can be more robust in the presence of noise by combining two strategies: allocating more iterations to early components and leveraging spectral decay. By front-loading optimization effort where it is impactful—at the beginning of the sequence—and favoring matrices with decaying singular values (especially power-law decay), models maintain lower reconstruction error despite increasing noise levels.

Computational efficiency analysis. Beyond approximation quality, we also analyze the computational efficiency of different iteration allocation strategies. Specifically, we investigate how quickly each strategy reduces the reconstruction error to a desired threshold. Figure 11 illustrates, for a range of target error thresholds, the number of iterations required by each allocation strategy to reach that threshold.

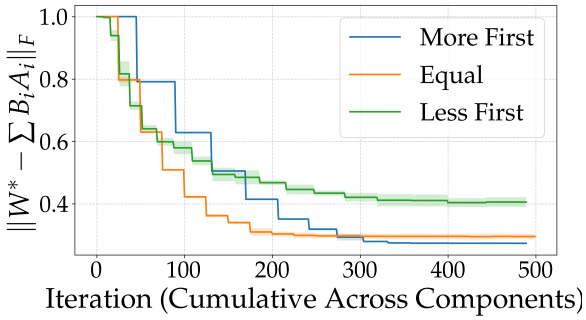
Observations: The more-first iterations strategy consistently reaches target reconstruction thresholds faster than the equal or less-first strategies. This aligns with our intuition that prioritizing the early components—those with the greatest influence on downstream error propagation—leads to quicker convergence. In contrast, less-first allocation delays learning the principal directions, requiring more total iterations to reach



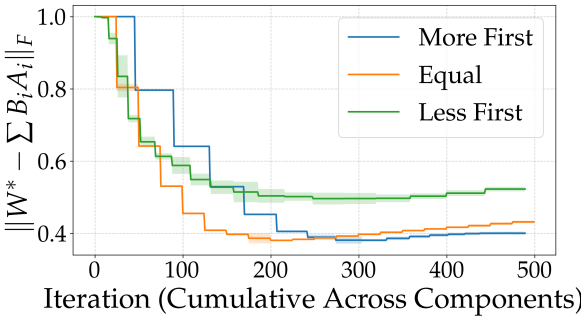
(a) $\kappa = 0.1$



(b) $\kappa = 0.5$



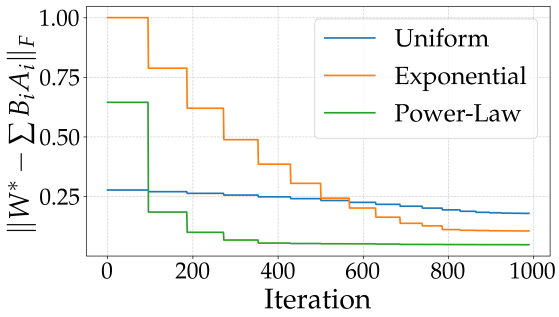
(c) $\kappa = 1$



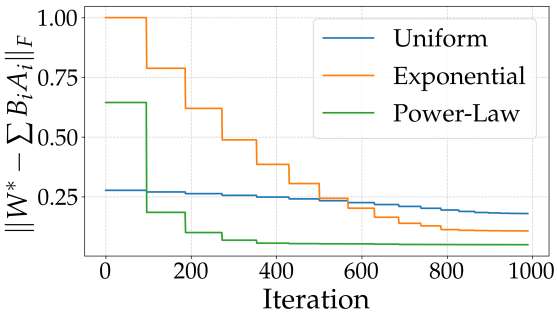
(d) $\kappa = 1.5$

Figure 9: Comparison of iteration allocation strategies under different noise levels. The "more-first" strategy achieves better reconstruction error across all κ values.

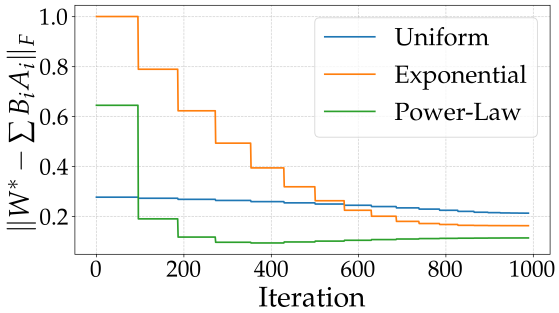
the same accuracy. This suggests that our theoretical insights can lead to more computationally efficient algorithms for low-rank approximation.



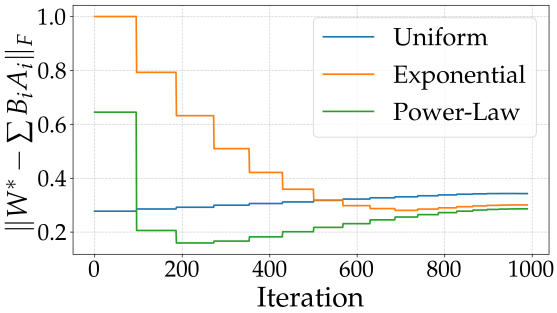
(a) $\kappa = 0.05$



(b) $\kappa = 0.1$

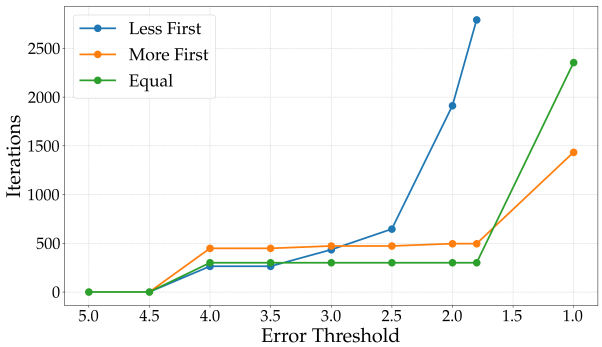


(c) $\kappa = 0.5$

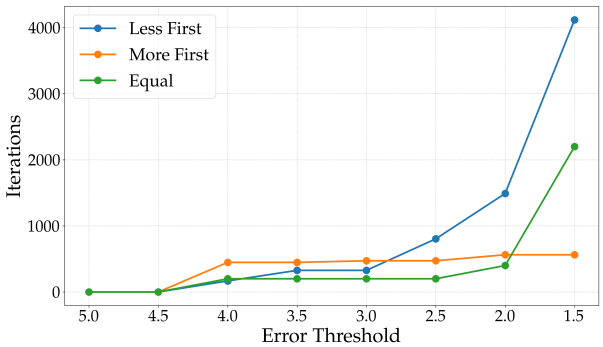


(d) $\kappa = 1$

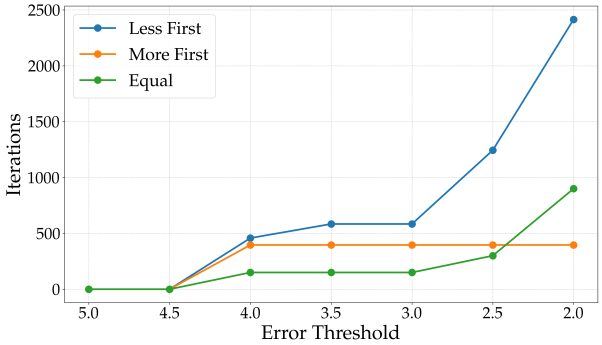
Figure 10: Effect of singular value profiles under noise. Power-law decay consistently achieves lower reconstruction error, followed by exponential and then uniform profiles, highlighting the benefit of spectral decay even in noisy settings.



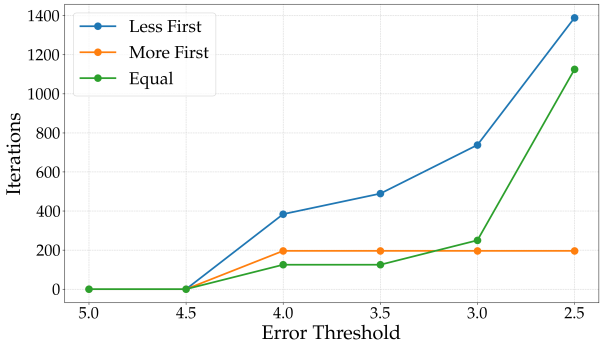
(a) Threshold: 1



(b) Threshold: 1.5



(c) Threshold: 2



(d) Threshold: 2.5

Figure 11: Number of iterations required to reach reconstruction error thresholds for different allocation strategies. Each subplot corresponds to a fixed error threshold. The "more-first" strategy consistently reaches the thresholds faster, especially for tighter reconstruction targets. In subplot (a), the "less-first" strategy fails to reach the threshold even after 10,000 iterations.