Divide and Orthogonalize: Efficient Continual Learning with Local Model Space Projection

Jin Shang[1] **Simone Shao**[1] **Tian Tong**[1] **Fan Yang**[1] **Yetian Chen**[1] **Yang Jiao**[1] **Jia Liu**[2,1] **Yan Gao**[1]

[1]Amazon.com, Seattle, WA, USA
[2]The Ohio State University, Columbus, OH, USA,
[1]{imjshang, simengsh, tongtn, fnam, yetichen, jaoyan, yanngao}@amazon.com
[2] liu@ece.osu.edu

Abstract

Continual learning (CL) has gained increasing interest in recent years due to the need for models that can continuously learn new tasks while retaining knowledge from previous ones. However, existing CL methods often require either computationally expensive layer-wise gradient projections or large-scale storage of past task data, making them impractical for resource-constrained scenarios. To address these challenges, we propose a local model space projection (LMSP)-based continual learning framework that significantly reduces computational complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ while preserving both forward and backward knowledge transfer with minimal performance trade-offs. We establish a theoretical analysis of the error and convergence properties of LMSP compared to conventional global approaches. Extensive experiments on multiple public datasets demonstrate that our method achieves competitive performance while offering substantial efficiency gains, making it a promising solution for scalable continual learning.

1 INTRODUCTION

Humans have the unique ability to continuously learn new tasks throughout their lives without forgetting their previously learned knowledge. This remarkable capability has recently inspired the efforts in the machine learning community to develop similar capabilities for deep neural network (DNN)-based machine learning models, which is termed continual learning (CL). However, one of the most significant challenges in CL is that DNN models are known to suffer from the problem of "catastrophic forgetting", i.e., the performances of the learned old tasks decay after the model learns new tasks. In the literature, numerous strategies have been proposed to address the challenge of catastrophic forgetting in CL. Existing forgetting mitigation approaches can be classified into three major categories: i) experience replay, ii) regularization, and iii) orthogonal projection (see Section 2 for more in-depth discussions). Generally speaking, experience-replay-based methods constrain the gradient directions by replaying the data of old tasks during learning new tasks, in the format of either real data or synthetic data from generative models, while regularization-based methods penalize the modification on the most important weights of old tasks through model regularizations. Due to the mixed information of old and new tasks (model or data), some performance decay of the old tasks are inevitable under the experience replay and regularization-based approaches. In contrast, orthogonal-projection-based methods update the model in the direction *orthogonal* to the subspace of old tasks, which has demonstrated superior performance compared to other approaches [Saha et al., 2021] – a highly desirable feature for CL in practice.

We note, however, that due to a number of technical challenges, developing practical orthogonal-projection-based CL approaches remains highly non-trivial. The first major challenge of orthogonal-projection-based CL approaches stems from the projection operation, which typically relies on singular-value decomposition (SVD) [Lin et al., 2022a,b]. These methods perform **layer-wise** SVD after the training of each task. It is well-known that the SVD operation costs $O(n^3)$ complexity for a n-dimensional model, which grows rapidly as n increases. With the ever-increasing widths and depths of large and deep learning models, computing such layer-wise SVDs upon the completion of each new task's training also becomes more and more difficult.

Another key challenge of the standard orthogonal-projection-based CL approaches lies in the inherent difficulty in facilitating *forward and backward knowledge transfer* (i.e., the learning of new tasks benefiting from the acquired knowledge from old tasks, and the knowledge learnt from new tasks further improves the performance of old tasks), when new task has strong similarity with some old tasks. However, integrating the computational efficiency

Proceedings of the 41st Conference on Uncertainty in Artificial Intelligence (UAI 2025), PMLR 244:3766–3786.

into an orthogonal-projection-based continual learning (CL) framework—while preserving performance and enabling both forward and backward knowledge transfer—remains a significant challenge. This motivates us to pursue a new efficient orthogonal-projection-based CL design.

In this paper, we propose an efficient local low-rank orthogonal-projection-based CL method based on local model space projection (LMSP), which not only significantly reduces the complexity of SVD basis computation, but also enables forward and backward knowledge transfers without sacrificing too much performance. The main results and contributions of this paper are as follows:

- Our proposed LMSP-based orthogonal projection approach is based on the basic idea of "divide and orthogonalize" principle, where we approximate the per-layer parameter matrix by a set of local low-rank matrices defined by a set of anchor points, which significantly reduces the computational complexity from $\mathcal{O}(n^3)$ to $\mathcal{O}(n^2)$ in performing projections with a minor projection error.
- We theoretically show that our proposed LMSP-based orthogonal projection approach achieves an $\mathcal{O}(1/K)$ convergence rate performance under both convex and non-convex settings, where K is the number of iterations. Moreover, we further prove the forward and backward knowledge transfers of the proposed LMSP-based orthogonal projection approach. In addition, by characterizing the upper bound and lower bounds for the approximation error, we provide the approximation accuracy analysis for using LMSP-based orthogonal projection approach compared to the original full-rank based approach.
- Through extensive experiments, we demonstrate that our proposed LMSP-based orthogonal projection approach achieves performance comparable to state-of-the-art baselines on four public datasets in terms of training accuracy and forward/backward knowledge transfer. Moreover, our approach significantly enhances efficiency while maintaining competitive performance, even compared to the original full-rank approach. We further conduct ablation studies to validate the effectiveness and efficiency of each key component in our LMSP-based design.

2 RELATED WORK

In this section, we provide an overview on the continual learning and local low-rank model approximation literature to further motivate this research and put our work in comparative perspectives.

1) Continual Learning: A Primer. Continual learning (CL), also known as lifelong learning or incremental learning, is an emerging area in machine learning research that has attracted a significant amount of interests recently. CL addresses the challenge of enabling a machine learning model to accumulate knowledge and adapt to new tasks that arrive sequentially over time [Chen and Liu, 2018]. A key goal of CL is to avoid "catastrophic forgetting" [McCloskey and Cohen, 1989, Abraham and Robins, 2005], i.e., a model's performance on previously learned tasks decays upon learning new tasks. To mitigate catastrophic forgetting in CL, various methodologies and strategies have been proposed:

- *Regularization-Based Approaches:* Regularization approaches use regularization to prevent a learning model from over-fitting to training data. For example, elastic weight consolidation (EWC) [Kirkpatrick et al., 2017] regularizes the updates on weights based on their significance for previous tasks using the Fisher information matrix. Aljundi et al. [2018] used an unsupervised and online approach to evaluate the model output's sensitivity to the inputs and penalize changes to important parameters.
- *Replay-Based Approaches:* Replay-based approaches store and replay old tasks' data to help models retain knowledge. For example, generative replay [Shin et al., 2017] generates data samples from previous tasks. In experience replay [Chaudhry et al., 2019b], a model replays previous experiences in a controlled manner. Techniques such as experience replay with replay buffer (ER-RB) [Lillicrap et al., 2019] and generative adversarial networks (GANs) [Goodfellow et al., 2020] have also been developed to enhance the efficiency of these mechanisms.
- *Orthogonal-Projection-Based Approaches:* To eliminate the need of storing data of old tasks or tuning the regularization parameter, researchers have proposed to learn the the new tasks and update the model in the *orthogonal subspace* of the old tasks [Chaudhry et al., 2020], which has demonstrated superior performance compared to other approaches [Saha et al., 2021]. State-of-the-art orthogonal-projection-based approaches include, e.g., [Lin et al., 2022a], first characterizes the task correlation to identify the positively correlated old tasks in a layer-wise manner, and then selectively modifies the learned model of the old tasks when learning the new task. More recently, several new techniques such as those proposed in [Yang et al., 2024] and [Xu et al., 2024], have been applied to orthogonal-projection-based approaches, yielding significant improvements in both forward and backward knowledge transfer.
- *Prompt-Based Continual Learning Approaches:* As large language models (LLMs) continue to be explored in greater depth, prompt-based continual learning methods, such as L2P [Wang et al., 2022b], DualPrompt [Wang et al., 2022a], and HiDE [Wang et al., 2023], are gaining popularity. These approaches typically prepend task-specific prompts (e.g., learnable tokens or embeddings) to the input or internal activations. In such methods, the base model remains largely **frozen** or undergoes minimal

updates, with learning primarily occurring in the prompt space. Each task is associated with a distinct prompt, enabling the model to adapt dynamically based on the given prompt. In contrast, Orthogonal-Projection-Based (e.g., GPM [Saha et al., 2021], OWM [Zeng et al., 2019], and A-GEM variants [Chaudhry et al., 2018]) update the model weights directly. These techniques project gradients orthogonally to the subspaces corresponding to previously learned tasks, ensuring that the model parameters are **not frozen** and continue to evolve while preserving past knowledge.

2) Local Low-Rank Approximation: Due to the superior performance compared to other approaches, we focus on the orthogonal-projection-based approach for CL in this paper. However, a key challenge of the orthogonal-projection-based CL approach stems from the need for computing orthogonal subspace, which is highly expensive as the model size gets large. This motivates us to propose a local model space projection (LSMP) approach based on local low-rank approximation to lower the the orthogonal subspace computation complexity. Recent works, such as [Li et al., 2024], have focused on improving continual learning (CL) efficiency by optimizing gradient directions and mitigating gradient conflicts during training. In contrast, our work primarily aims to reduce the computational cost of orthogonal-projection-based CL while maintaining competitive performance. The key idea of our posed LMSP approach is based on the local low-rank approximation (LRA) of matrics. LRA techniques have been widely applied in the areas of matrix factorization [Billsus and Pazzani, 1998, Mnih and Salakhutdinov, 2007, Salakhutdinov and Mnih, 2008, Candes and Plan, 2010]. The basic idea of these existing works is to represent a given matrix by a product of lower-rank matrices that capture the essential structure of the original matrix.

Local low-rank approximation (LLRA) extends LRA to preserve low-rank structures in localized regions of matrices. LLRA has been applied in various applications, such as recommendation [Beutel et al., 2017, Sarwar et al., 2002, Christakopoulou and Karypis, 2018], collaborative filtering [George and Merugu, 2005, Lee et al., 2014, Koren, 2008]. For example, Lee et al. [2013] proposed a local low-rank matrix approximation (LLORMA) method, which finds anchor points of the matrix and estimates local low-rank matrices in the neighborhood surrounding each anchor point. Then, a weighted sum of the local matrices is used to approximate the original matrix, where the weight is the similarity between the pair of anchor points. Lee et al. [2014] later used this method in collaborative filtering to estimate the user-item rating matrix with a weighted combination of local matrices. To our knowledge, our work is the first to leverage the local low-rank approximation approach for CL.

3 A LOCAL MODEL SPACE PROJECTION APPROACH

In this section, we first introduce the basic idea of local representation and task subspace construction in Section 3.1, based on which we define task similarity with local projection in Section 3.2. These key notions allow us to further propose update rules based on local representations and task subspaces in Section 3.3. Lastly, we conduct theoretical performance analysis for our proposed LMSP-based orthogonal projection approach in Section 3.4.

3.1 LOCAL REPRESENTATION AND TASK SPACE CONSTRUCTION

1) The Basic Idea: As mentioned in Section 1, to lower the SVD computational costs in full-rank orthogonal-projection-based CL approaches, the basic idea of our local model space projection (LMSP) approach is based on a "divide and orthogonalize" principle. Our LMSP approach is built upon the following key notion of local model representation.

Given N^j samples in an old task $j \in [0, t-1]$, we construct a representation matrix $\mathbf{R}_j^l = [\mathbf{r}_{j,1}^l, ...\mathbf{r}_{j,N^j}^l] \in \mathbb{R}^{M \times N^j}$ for layer l, where M is the representation dimension and each $\mathbf{r}_{j,i}^l \in \mathbb{R}^M, i = 1, 2, ..., N^j$ is the representation of layer l by forwarding the sample data point $\mathbf{x}_{j,i}$ through the model. Instead of directly applying SVD to the representation matrix $\mathbf{R}_j^l$, we approximate the matrix by a set of low-rank matrices defined by a set of anchor points.

Following a similar token as in [Lee et al., 2013], we define a *smoothing kernel* $K_h(s_1, s_2)$ with bandwidth h, where $(s_1, s_2) \in [M] \times [N^j]$ is an entry in the representation matrix $\mathbf{R}_j^l$. For convenience, we also denote this kernel matrix by $\mathbf{K}_h^{(a,b)}$. Also, the (i, j)-th entry in $\mathbf{K}_h^{(a,b)}$ is denoted as $K_h((a, b), (i, j))$. Simply speaking, the smoothing kernel is a non-negative symmetric unimodal function parameterized by the bandwidth parameter $h > 0$. Generally, the larger the value of h, the wider the spread of the kernel [Wand and Jones, 1994].

To obtain a set of local representation matrices, we first sample m *"anchor points"* from the global representation matrix $\mathbf{R}_j^l$, which are denoted as $\{s_q \triangleq (i_q, j_q)\}_{q=1}^m$, where $(i_q, j_q) \in [M] \times [N^j]$ is the entry location of the q-th anchor point. In [Wand and Jones, 1994, Lee et al., 2013], it has been shown that the global representation matrix $\mathbf{R}_j^l$ has a locally low-rank structure and thus could be approximated by the local representation matrices $\{\hat{\mathbf{R}}_j^l(s_q)\}_{q=1}^m$ corresponding to these anchor points (i.e., Nadaraye-Waston regression, note that $\hat{\mathbf{R}}_j^l(s_q)$ is depended on the specific anchor point s_q) as follows:

$$\mathbf{R}_j^l \approx \hat{\hat{\mathbf{R}}}_j^l \triangleq \sum_{q=1}^{m} \frac{K_h(s_q, s)}{\sum_{p=1}^{m} K_h(s_p, s)} \hat{\mathbf{R}}_j^l(s_q). \tag{1}$$

To obtain the local representation matrices $\{\hat{\mathbf{R}}_j^l(s_q)\}_{q=1}^m$ in Eq. (1), we adopt a product form for the general kernel function $K_h(s_1, s_2) = K_h((a,b),(c,d)) = K_{h_1}(a,c)K'_{h_2}(b,d)$, where $s_1, s_2 \in [M] \times [N^j]$ and K, K' are kernels on the spaces $[M]$ and $[N^j]$, respectively. We summarize several popular smoothing kernels in Appendix D. In this paper, we use the Gaussian kernel for both K, K' (we will conduct ablation studies on the choice of smoothing kernels later in Section 4).

In the literature, there are two widely used ways to choose the anchor points $\{s_q \triangleq (i_q, j_q)\}_{q=1}^m$: 1) sample uniformly at random from the representation matrix in $[M] \times [N^j]$; and 2) use K-means or other clustering methods to pre-cluster the representation matrix and then use their centers as the anchor points. Even though using pre-clustering to find centroids as anchor points may provide a more distinct and diverse representation and it is also proved by some works such as [Zhang et al., 2017], our numerical studies later show that the improvements are marginal. More specifically, we found that as long as the choices of random anchor points are relatively uniform, the empirical difference between two selection methods is not significant. Since the basis of the task are extracted at each layer, considering the huge additional computational costs introduced by this layer-wise clustering methods (e.g., k-means), we elect to use the random sample strategy in our experiments for simplicity in this paper.

Next, with local representations, we will show how the local model spaces are constructed for task j at layer l. For an old task $j \in [0, t-1]$, to obtain the basis $\mathbf{S}_j^l$ at layer l, traditional methods [Saha et al., 2021, Lin et al., 2022b] adopted the standard singular value decomposition (SVD) for the representation matrix of each layer, which incurs a high computation cost of $\mathcal{O}(MN^j \min(M, N^j)) = \mathcal{O}(n^3)$. In contrast, by using a low-rank structure for each local model, the computation can be significantly reduced. Specifically, we first obtain the local decomposed matrices $\mathbf{A}$ and $\mathbf{B}$ for each anchor point s_q by minimizing the following global least square loss in Eq. (2):

$$\{(\mathbf{A}^{(q)}, \mathbf{B}^{(q)})\}_{q=1}^m := \\ \underset{\mathbf{A}^{(q)},\mathbf{B}^{(q)}}{\arg\min} \sum_{x,y\in\Omega} \left[\sum_{q=1}^m \Big(\frac{K_h^{(q)} \odot [\mathbf{A}^{(q)}\mathbf{B}^{(q)^\top}]}{\sum_{p=1}^m K_h^{(p)}} - \mathbf{R}_j^l \Big)^2 \right]_{x,y} \\ + \sum_{q=1}^m [\lambda_A^{(q)} \|\mathbf{A}^{(q)}\|_F^2 + \lambda_B^{(q)} \|\mathbf{B}^{(q)}\|_F^2], \qquad (2)$$

where Ω is the observed set of indices of the matrices, $K_h^{(q)} = K_h^{s_q} = K_h^{(i_q, j_q)}$ is the kernel matrix whose (a,b)-th entry is

$$K_h((i_q, j_q), (a,b)) = K_{h_1}(i_q, a) K'_{h_2}(j_q, b)$$

and $\odot$ is the Hadamard product. We also add ℓ_2 regularization as is standard in conventional SVD. Similar to [Lee et al., 2013], we can obtain $(\mathbf{A}^{(q)}, \mathbf{B}^{(q)})$ in a parallel fashion as follows:

$$(\mathbf{A}^{(q)}, \mathbf{B}^{(q)}) := \underset{\mathbf{A},\mathbf{B}}{\arg\min} \sum_{x,y\in\Omega} [K_h^{(q)} \odot ([\mathbf{A}\mathbf{B}^\top] - \mathbf{R}_j^l)^2]_{x,y} \\ + \lambda_A \|\mathbf{A}\|_F^2 + \lambda_B \|\mathbf{B}\|_F^2. \qquad (3)$$

Being a variant of low-rank matrix completion, this problem can be solved efficiently via various methods, including AltMin [Jain et al., 2013, Hastie et al., 2015], singular value projection [Netrapalli et al., 2014, Jain et al., 2010], Riemannian GD [Wei et al., 2016], ScaledGD [Tong et al., 2021, Xu et al., 2023], etc; see [Chen and Chi, 2018, Chi et al., 2019] for recent overviews. In this paper, we use the AltMin method to find the optimizer and obtain the basis for each local model.

2) Computation Complexity Analysis: Denote the rank for each local model as $r \ll \min(M, N^j)$, and $\mathbf{A} \in \mathbb{R}^{M\times r}, \mathbf{B} \in \mathbb{R}^{N^j \times r}$. Later, we adopt QR decomposition for $\mathbf{A} = \hat{\mathbf{U}}\mathbf{\Psi}_A, \mathbf{B} = \hat{\mathbf{V}}\mathbf{\Psi}_B$, where $\mathbf{\Psi}_A, \mathbf{\Psi}_B \in \mathbb{R}^{r\times r}$, and then perform SVD on the $r \times r$ matrix to achieve: $\mathbf{\Psi}_A\mathbf{\Psi}_B^\top = \mathbf{U}_\Psi \mathbf{\Sigma} \mathbf{V}_\Psi^\top$. The final basis for local model space q can be constructed as $\{\mathbf{S}_j^{l,(q)} \triangleq \hat{\mathbf{U}}_{\Psi,j}^{l,(q)} \mathbf{U}_{\Psi,j}^{l,(q)}\}_{q=1}^m \in \mathbb{R}^{M\times r}$.

Then, for a new task t, we treat all m local model spaces as m old tasks. As a result, we have a total of tm old tasks as candidates for new task t to find the top-k correlated ones. Since the AltMin algorithm has the complexity of $\mathcal{O}(MN^j r) = \mathcal{O}(n^2)$, the total complexity can be reduced to $\mathcal{O}(n^2 m) = \mathcal{O}(n^2)$, as the total number of anchor points $m \ll \min(M, N^j)$. Thus the computation cost in LMSP is significantly reduced.

3.2 TASK SIMILARITY WITH LOCAL PROJECTION

With the local representations in Section 3.1, we are now in a position to introduce the following definitions on task gradients to formally characterize the task similarity. Toward this end, we introduce the following definitions, which generalize Definition 1 and Definition 2 from [Lin et al., 2022a] to local settings.

Definition 3.1 (Local Sufficient Projection)**.** For any new task $t \in [1, T]$, we say that it has local sufficient gradient projection on the local subspace $q \in [1, m]$ of old task $j \in [0, t-1]$ if for some $\lambda_1 \in (0,1)$: $\|\mathrm{Proj}_{K_h^{(q)}D_j}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}))\|_2 \geq \lambda_1 \|\nabla\mathcal{L}_t(\mathbf{W}_{t-1})\|_2$.

Definition 3.2 (Local Positive Correlation)**.** For any new task $t \in [1, T]$, we say that it has local positive correlation with the local subspace $q \in [1, m]$ of old task $j \in [0, t-1]$ if for some $\lambda_2 \in (0,1)$: $\langle \nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{(q)}), \nabla\mathcal{L}_t(\mathbf{W}_{t-1}) \rangle \geq \lambda_2 \|\nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{(q)})\|_2 \|\nabla\mathcal{L}_t(\mathbf{W}_{t-1})\|_2$.

Here, for any matrix $\mathbf{A}$, $\mathrm{Proj}_{K_h^{(q)}D_j}(\mathbf{A}) \triangleq \mathbf{S}_j^{(q)}\mathbf{S}_j^{(q)\top}\mathbf{A}$ defines the projection on the input local model space for anchor point q of old task j, and $\mathbf{S}_j^{(q)}$ is the basis for this local model space.

Compared to global sufficient definition, which is the Definition 1 in [Lin et al., 2022a], the projection space in Definition 3.1 is changed to the q-th local model basis rather than the global basis for task j.

Definition 3.1 implies that task t and the q-th local model of task j have sufficiently common basis and are strongly correlated since the gradient lies in the span of the input [Zhang et al., 2021]. Also, similar to positive correlation definiation, which is Definition 2 in [Lin et al., 2022a], Definition 3.2 goes one step further to characterize the task similarity.

In addition to the local sufficiency projection and positive correlation conditions, we introduce a *new* concept, termed *"local relative orthogonality"*, specifically tailored for our LMSP-based method, defined as follows:

Definition 3.3 (Local Relative Orthogonality)**.** For any new task $t \in [1, T]$, we say that it is more locally relatively orthogonal to local subspace $q \in [1, m]$ of old task $j \in [0, t-1]$ than the global subspace old task $j \in [0, t-1]$ for some $\lambda_3 \in (0, 1)$ if the following condition holds:

$$\|\mathrm{Proj}_{K_h^{(q)}D_j}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}))\|_2 =$$
$$\lambda_3\|\mathrm{Proj}_{D_j}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}))\|_2 \le \|\mathrm{Proj}_{D_j}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}))\|_2.$$

The local relative orthogonality means that the input of the q-th local model space for old task j is more orthogonal to the new task t than the global one, which indicates that updating the model along the $\nabla\mathcal{L}_t(\mathbf{W})$ direction would introduce *less inference* to old task j, thus mitigating the forgetting problem. Note that Definitions 3.2–3.3 characterize the similarity based on the old model weights $\mathbf{W}_{t-1}$, hence they allow the task similarity detection before learning the new task t.

3.3 LOW-COMPLEXITY CONTINUAL LEARNING WITH LOCAL MODEL SPACE PROJECTION

With the local representations and the associated task similarity, we propose the following LMSP-based orthogonal projection approach, which aims to avoid forgetting while enabling backward knowledge transfer. Toward this end, based on Definitions 3.1 and 3.2, we establish the following regimes and update rules, which correspond to the global settings described in [Lin et al., 2022a].

Regime 1 (Forget Mitigation): For a new task t's layer l, if $\|\mathrm{Proj}_{K_h^{(q)}D_j^l}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l))\|_2 < \lambda_1\|\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\|_2$, we say that the q-th local model of old task j falls in Regime 1.

Note that in this case, since task t and task $j^{(q)}$ are *relatively orthogonal,* we update the model in the direction of orthogonal projection to avoid forgetting:

$$\nabla\mathcal{L}_t(\mathbf{W}^l) \leftarrow \nabla\mathcal{L}_t(\mathbf{W}^l) - \mathrm{Proj}_{K_h^{(q)}D_j^l}(\nabla\mathcal{L}_t(\mathbf{W}^l)). \quad (4)$$

Regime 2 (Forward Knowledge Transfer): For a new task t's layer l, if it holds that

$$\|\mathrm{Proj}_{K_h^{(q)}D_j^l}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l))\|_2 \ge \lambda_1\|\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\|_2,$$
$$\langle\nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{l,(q)}), \nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\rangle <$$
$$\lambda_2\|\nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{l,(q)})\|_2\|\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\|_2,$$

we say the q-th local model of old task j falls into Regime 2.

In this case, since task t and task $j^{(q)}$ are strongly correlated on gradient norm projection but negatively correlated on gradient direction, we still update the model on the orthogonal projection and use a scalar matrix $\mathbf{Q}$ to facilitate forward knowledge similar to the idea in [Lin et al., 2022b]:

$$\nabla\mathcal{L}_t(\mathbf{W}^l) \leftarrow \nabla\mathcal{L}_t(\mathbf{W}^l) - \mathrm{Proj}_{K_h^{(q)}D_j^l}(\nabla\mathcal{L}_t(\mathbf{W}^l)), \quad (5)$$
$$\mathbf{Q}_{j,t}^{l,(q)} \leftarrow \mathbf{Q}_{j,t}^{l,(q)} - \beta\nabla_{\mathbf{Q}}\mathcal{L}_t(\mathbf{W}^l - \mathrm{Proj}_{K_h^{(q)}D_j^l}(\mathbf{W}^l)$$
$$- \mathbf{W}^l\mathbf{S}_j^{l,(q)}\mathbf{Q}_{j,t}^{l,(q)}\mathbf{S}_j^{l,(q)\top}).$$

Regime 3 (Backward Knowledge Transfer): For a new task t's layer l, if it holds that

$$\|\mathrm{Proj}_{K_h^{(q)}D_j^l}(\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l))\|_2 \ge \lambda_1\|\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\|_2,$$
$$\langle\nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{l,(q)}), \nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\rangle \ge$$
$$\lambda_2\|\nabla\mathcal{L}_j^{(q)}(\mathbf{W}_j^{l,(q)})\|_2\|\nabla\mathcal{L}_t(\mathbf{W}_{t-1}^l)\|_2,$$

we say the q-th local model of old task j falls into Regime 3.

In this case, since task t and task $j^{(q)}$ are positively correlated in both norm and direction, updating the model directly along with $\nabla\mathcal{L}_t(\mathbf{W}^l)$ not only leads to a better model for continual learning, but also improves the performance of old task j. Since the weight projection is frozen, i.e., $\mathrm{Proj}_{K_h^{(q)}D_j^l}(\mathbf{W}_{t-1}^l) = \mathrm{Proj}_{K_h^{(q)}D_j^l}(\mathbf{W}_j^l)$, we update the model as follows:

$$\mathbf{W}^l \leftarrow \mathbf{W}^l - \alpha\nabla[\mathcal{L}_t(\mathbf{W}^l) + \theta\|\mathrm{Proj}_{K_h^{(q)}D_j^l}(\mathbf{W}^l - \mathbf{W}_{t-1}^l)\|].$$

In summary, the optimization problem for learning a new task t can be written as follows:

$$\min_{\mathbf{W},\{\mathbf{Q}_{j,t}^{l,(q)}\}_{l,j^{(q)}\in\mathrm{Reg}_{t,2}^l\bigcup\mathrm{Reg}_{t,3}^l}} \mathcal{L}_t(\{\tilde{\mathbf{W}}^l\}_l) +$$
$$\theta\sum_l\sum_{j^{(q)}\in\mathrm{Reg}_{t,3}^l}\|\mathrm{Proj}_{K_h^{(q)}D_j^l}(\mathbf{W}^l - \mathbf{W}_{t-1}^l)\|, \quad (6)$$

$$s.t.\ \tilde{\mathbf{W}}^l = \mathbf{W}^l + \sum_{j^{(q)}\in\mathrm{Reg}_{t,2}^l\bigcup j^{(q)}\in\mathrm{Reg}_{t,3}^l}[\mathbf{W}^l\mathbf{S}_j^{l,(q)}\mathbf{Q}_{j,t}^{l,(q)}\mathbf{S}_j^{l,(q)\top} - \quad (7)$$

Algorithm 1 Efficient Continual Learning with Local Model Space Projection (LMSP).

```
Input: task sequence $\mathbb{T} = \{t\}_{t=0}^{T}$;
Learn first $j \in [0, t-1]$ task using vanilla stochastic gradient descent;
for each old task $j$ do
    Sample $m$ anchor point
    Extract basis $\mathbf{S}_j^{l,(q)}$ for each local model space $q$ using the learnt model $\mathbf{W}_j$
end for
for each new task $t$ do
    Calculate gradient $\nabla \mathcal{L}_t(\mathbf{W}_{t-1})$;
    Evaluate the local sufficient projection and local positive correlation conditions for layer-wise correlation computation to determine its membership in $\mathrm{Reg}_{t,1}^{l}$, $\mathrm{Reg}_{t,2}^{l}$ or $\mathrm{Reg}_{t,3}^{l}$;
    for $k = 1, 2, ...$ do
        Update the model and scaling matrices by solving Eq. (6);
    end for
end for
Output: The learnt model $\mathbf{W}_t$, scaling matrices $\{\mathbf{Q}_{j,t}^{l,(q)}\}_{l, j^{(q)} \in \mathrm{Reg}_{t,3}^{l} \bigcup \mathrm{Reg}_{t,3}^{l}}$;
```

$$\mathrm{Proj}_{K_h^{(q)} D_j^l}(\mathbf{W}^l)],$$

$$\nabla \mathcal{L}_t(\mathbf{W}^l) = \nabla \mathcal{L}_t(\mathbf{W}^l) - \sum_{j^{(q)} \in \mathrm{Reg}_{t,1}^{l} \bigcup j^{(q)} \in \mathrm{Reg}_{t,2}^{l}} \mathrm{Proj}_{K_h^{(q)} D_j^l}(\nabla \mathcal{L}_t(\mathbf{W}^l)).$$

In simple language, the optimization problem in Eqs. (6–7) can be interpreted as follows: First, note that task similarity has been calculated before learning the new task t, we can first determine the regimes of different local model spaces of old task j and then construct the task $j^{(q)}$, which is the old task j projected onto local model space corresponding to anchor point s_q. Next, we conservatively update the model for task $j^{(q)}$ in Regime 3 while using orthogonal projection to preserve the knowledge for the rest (cf. the objective function in (6). The scaled weight projection is used for old tasks in both Regime 2 and Regime 3 to facilitate *forward knowledge transfer* (cf. the constraint in (7)). Note that one can always strike a good balance between adapting the model to new task while not forgetting the knowledge of the learnt tasks by adjusting the regularization parameter θ. The overview of our LMSP-based efficient continual learning framework is described in Algorithm 1 (see next page).

3.4 THEORETICAL PERFORMANCE ANALYSIS

In this subsection, we will establish the convergence rate and backward knowledge transfer of our proposed LMSP-based orthogonal projection approach. Without loss of generality, consider the scenario of learning two consecutive tasks 1 and 2. Note that since [Lin et al., 2022a] has already conducted theoretical analysis for the vanilla GD-type update (cf. Rule #2 in [Lin et al., 2022a]), which is also applicable in our work, we will only focus on the major difference in our work, which lies in the analysis for the local low-rank and full-rank orthogonal-projection-based updates.

For simplicity, consider the scenario with a sequence of two tasks 1 and 2. Let $\mathcal{F}(\mathbf{W}) = \mathcal{L}(\mathbf{W}, \mathcal{D}_1) + \mathcal{L}(\mathbf{W}, \mathcal{D}_2)$, $\boldsymbol{g}_1(\mathbf{W}) = \nabla_{\mathbf{W}} \mathcal{L}(\mathbf{W}, \mathcal{D}_1)$ and $\boldsymbol{g}_2(\mathbf{W}) = \nabla_{\mathbf{W}} \mathcal{L}(\mathbf{W}, \mathcal{D}_2)$. Note that $\bar{\boldsymbol{g}}(\mathbf{W}^{(k)}) = \boldsymbol{g}(\mathbf{W}^{(k)}) - \mathrm{Proj}_{K_h^{(q)} D_j}(\boldsymbol{g}(\mathbf{W}^{(k)}))$ as the gradients for the local low-rank orthogonal-projection-based updates in Eq. (4) as well as Eq. (5), and $\ddot{\boldsymbol{g}}(\mathbf{W}^{(k)}) = \boldsymbol{g}(\mathbf{W}^{(k)}) - \mathrm{Proj}_{D_j}(\boldsymbol{g}(\mathbf{W}^{(k)}))$ as the gradients for the full-rank orthogonal-projection-based updates under Regime 1 and Regime 2 in [Lin et al., 2022a]. Thus, we let $k \in [0, K-1]$ denote the step index and use $\mathbf{W}_1$ to denote the model parameters for task 1, with $\mathbf{W}_1 = \mathbf{W}^{(0)}$ for the initialization of the new task model weights. We first state our major convergence rate result for the local low-rank orthogonal-projection-based update as follows:

Theorem 3.4. *Suppose the loss function $\mathcal{L}$ is B-Lipschitz and $\frac{H}{2}$-smooth. Let $\alpha \leq \min\{\frac{1}{H}, \frac{\gamma \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}\|}{HBK}\}$ and $\lambda_1 \geq \sqrt{1 - 2\frac{2\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| - \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{\gamma^2 \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}\|}}$ for some $\gamma \in (0,1)$. Then, the following results hold:*

(1) if $\mathcal{L}$ is convex, the local low-rank orthogonal-projection-based update in Regimes 1 and 2 for task 2 converges to the optimal model $\mathbf{W}^{\star} = \arg\min \mathcal{F}(\mathbf{W})$;

(2) if $\mathcal{L}$ is non-convex, the local low-rank orthogonal-projection-based update in Regimes 1 and 2 for task 2 converges to a first-order stationary point:

$$\min_k \|\nabla \mathcal{F}(\mathbf{W}^{(k)})\|^2 \leq \frac{2}{\alpha K} \sum_{k=0}^{K-1} [\mathcal{F}(\mathbf{W}^{(k)}) - \mathcal{F}(\mathbf{W}^{\star})] + \frac{[2 + \gamma^2(5 - \lambda_1^2)]}{2} \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + 4 \sum_{i=1}^{2} \|\boldsymbol{g}_i(\mathbf{W}^{(0)})\|^2.$$

Theorem 3.4 characterizes the convergence of the joint objective function $\mathcal{F}(\mathbf{W})$ when updating the model with local low-rank orthogonal-projection-based updates in the convex setting, as well as the convergence to a first-order stationary point in the non-convex setting when the q-th local model of task 1 and task 2 satisfy the local sufficient projection definition with certain λ_1. Hence, it benefits the joint learning of tasks 1 and 2. The proof of Theorem 3.4 is relegated to Appendix A due to space limitations. The next result establishes the backward knowledge transfer of our CL approach:

Theorem 3.5. *Suppose loss $\mathcal{L}$ is B-Lipschitz and $\frac{H}{2}$-smooth. Then, the following results hold:*

(1) Let $\mathbf{W}^s$ and $\mathbf{W}^c$ be the model parameters after one update to an initial model $\mathbf{W}$ by using local

low-rank and full-rank orthogonal-projection-based updates, respectively. Suppose the new task satisfy local relative orthogonality for a $\lambda_3 \in (0,1)$*, i.e.,* $\|Proj_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}^{(i)}))\|_2 = \lambda_3 \|Proj_{D_1}(\boldsymbol{g}_2(\mathbf{W}^{(i)}))\|_2$ *for* $i \in [0, k-1]$, $\alpha \leq \min\{\frac{1}{H}, \frac{\gamma\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{HBK}\}$ *and* $\lambda_1 \geq \max\{ \sqrt{1-2\frac{2\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| - \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{\gamma^2\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}}, \sqrt{1-\frac{(1-\lambda_3^2)(2+\alpha H)\lambda'^2_1}{1+2\alpha H}}\}$*, then we have* $\mathcal{F}(\mathbf{W}^s) \leq \mathcal{F}(\mathbf{W}^c)$*;*

(2) Let $\mathbf{W}^{(k)}$ *be the* k*-th iterate for task 2 with the* θ*-regularized update in Regime 3. If* $\alpha \leq \frac{4\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{HBk^{1.5}}$*, then* $\mathcal{L}_1(\mathbf{W}^{(k)}) \leq \mathcal{L}_1(\mathbf{W}_1) = \mathcal{L}_1(\mathbf{W}^{(0)})$.

The first claim in Theorem 3.5 indicates that updating the model using the local low-rank orthogonal-projection-based updates achieves lower loss value than the full-rank orthogonal-projection-based updates when the q-th local model of task 1 and task 2 satisfy the sufficient projection with some λ_1 and the local relative orthogonality in Definition 3.3 with some λ_3, hence implying *backward knowledge transfer.* The second claim in Theorem 3.5 suggests that the local low-rank orthogonal-projection-based update results in a better model for task 1 with respect to $\mathcal{L}_1$. The proofs of Theorem 3.5 is also relegated to Appendix B due to space limitation.

Next, we provide the approximation accuracy analysis and comparison of the loss functions between applying local low-rank and full-rank orthogonal-projection-based updates.

Without loss of generality, for any anchor point s_q, we let $B_h(s_q)$ denote the neighborhood of indices near that anchor point, which is defined as $B_h(s_q) \overset{\text{def}}{=} \{\forall s' \in [M] \times [N] : d(s_q, s') < h\}$ and we use $M(h, s_q)$ and $N(h, s_q)$ to denote the number of unique row and column indices in $\mathcal{B}_h(s_q)$. Also, we denote $n_q = \min(M(h, s_q), N(h, s_q))$. Then we have the following theorem for approximation accuracy:

Theorem 3.6. *Suppose loss* $\mathcal{L}$ *is* B*-Lipschitz and* $\frac{H}{2}$*-smooth. Let* $\mathbf{W}^s$ *and* $\mathbf{W}^c$ *be the model parameters after one update to an initial model* $\mathbf{W}$ *by using local low-rank and full-rank orthogonal-projection-based updates, respectively. Given the mapping function* $T(s) = \mathcal{T}^2 = \mathbf{R}_j^l \mathbf{R}_j^{l\top}$ *which represents the gram matrix of the original matrix, is Hölder continuous with parameter* $Z, \beta > 0$*. Let* $\alpha \leq \min\{\frac{1}{H}, \frac{\gamma\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|}{HBK}\}$ *for some* $\gamma \in (0,1)$*. Then, the value of loss discrepancy between full-rank and local low-rank orthogonal-projection-based updates corresponding with anchor point* s_q*, i.e.,*

$$\mathcal{E}(\mathcal{F})(s_q, h) = \mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s), \tag{8}$$

is upper bounded as:

$$\begin{aligned}\mathcal{E}(\mathcal{F})(s_q, h) \leq & HZ^2 h^{2\beta}(24n_q + 9)B^2 \\ & + Zh^\beta(4\sqrt{3n_q}+2)\Big[\frac{2+\gamma^2}{4}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 \\ & + \|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\| + \frac{3}{2}B^2\Big], \end{aligned} \tag{9}$$

and lower bounded as:

$$\begin{aligned}\mathcal{E}(\mathcal{F})(s_q, h) \geq & -HZ^2 h^{2\beta}(24n_q + 9)B^2 \\ & + Zh^\beta(4\sqrt{3n_q+2})\Big[-\frac{2+\gamma^2}{4}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 \\ & + \|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\| + \frac{1}{2}B^2\Big]. \end{aligned} \tag{10}$$

Theorem 3.6 provides the approximation accuracy bounds between using original full-rank and local low-rank updates, which indicates that error introduced by the local model space projection can be bounded and the bound is mostly influenced by the first squared term in both Eqs. (9) and (10). Noting that this term is the squared bound for matrix completion based on [Lee et al., 2013], the LMSP loss error bound is roughly on the order of the square of the matrix completion bound due to the inner product calculation of basis in projection computation. The proof of Theorem 3.6 is also relegated to Appendix C due to space limitation.

4 NUMERICAL RESULTS

In this section, we conduct experiments to verify the efficacy of our proposed research. We will first discuss our experiment settings, including datasets, baselines, and evaluation metrics, which are followed by experimental results.

1) Datasets: We evaluate the performance of our LMSP on four public datasets for CL: (1) Permuted MNIST [LeCun et al., 2010]; (2) CIFAR-100 Split [Krizhevsky et al., 2009]; (3) 5-Datasets [Lin et al., 2022a,b]; and (4) MiniImageNet [Vinyals et al., 2016]. Due to space limitations, the detailed dataset information is relegated to Appendix E.

2) Baseline Methods: We compare our LMSP method with the following baseline methods:

(1) *EWC* [Kirkpatrick et al., 2017]: EWP adopts the Fisher information matrix for weights importance evaluation.

(2) *HAT* [Serra et al., 2018]: HAT preserves the knowledge of an old task by learning a hard attention mask;

(3) *Orthogonal Weight Modulation (OWM)* [Zeng et al., 2019]: OWM projects the gradient of a new task to the orthogonal direction of the input subspace of an old task by learning a projector matrix;

(4) *Gradient Projection Memory (GPM)* [Saha et al., 2021]: GPM first stores old tasks' basis of the input subspace, and then uses the gradient projection orthogonal to the subspace spanned by stored basis to update the model;

(5) *TRGP* [Lin et al., 2022b]: TRGP uses a scaled weight projection to facilitate the forward knowledge transfer from related old tasks to the new task;

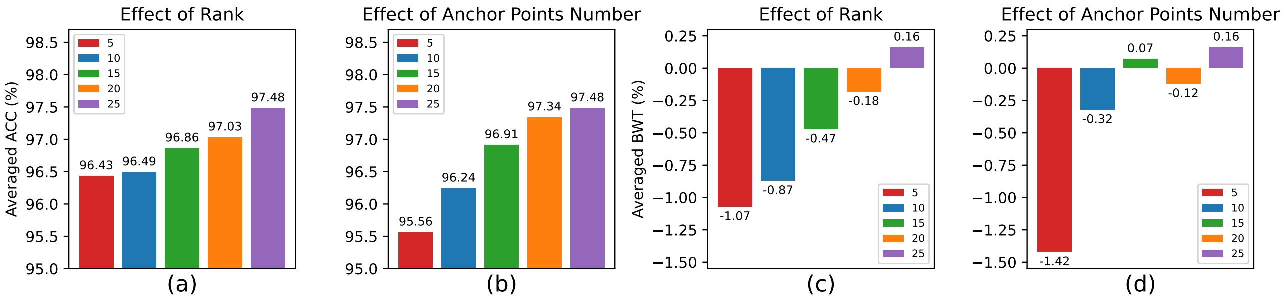

Figure 1: Ablation studies on rank and number of anchor points.

Table 1: The ACC and BWT performance comparisons between LMSP (ours) and baselines.

Method	PMNIST		CIFAR-100 Split		5-Dataset		MiniImageNet	
	ACC(%)	BWT(%)	ACC(%)	BWT(%)	ACC(%)	BWT(%)	ACC(%)	BWT(%)
Multitask	96.70	-	79.58	-	91.54	-	69.46	-
OWM	90.71	-1	50.94	-30	-	-	-	-
EWC	89.97	-4	68.80	-2	88.64	-4	52.01	-12
HAT	-	-	72.06	0	91.32	-1	59.78	-3
A-GEM	83.56	-14	63.98	-15	84.04	-12	57.24	-12
ER-Res	87.24	-11	71.73	-6	88.31	-4	58.94	-7
GPM	93.91	-3	72.48	-0.9	91.22	-1	60.41	-0.7
TRPG	96.26	-1.01	74.98	-0.15	92.41	-0.08	**64.46**	-0.89
CUBER	97.04	-0.11	**75.29**	0.14	92.85	-0.13	63.67	0.11
LMSP($r = 25$)	**97.48**	**0.16**	74.21	**0.94**	**93.78**	**0.07**	64.2	**1.55**

(6) *CUBER* [Lin et al., 2022a]: CUBER categorizes the tasks as strong projection and positive correlation.

(7) *Averaged GEM (A-GEM)* [Chaudhry et al., 2018]: A-GEM stores and incorporates old tasks' data in computing gradients for the new task's learning;

(8) *Experience Replay with Reservoir sample (ER-Res)* [Chaudhry et al., 2019a]: ER-Res uses a small episodic memory to store old task samples to address the forgetting problem;

(9) *Multitask* [Saha et al., 2021]: Multitask jointly learns all tasks once with a single network using all datasets.

3) Evaluation Metrics: We use the following two metrics to evaluate the learning performance of the baseline models and our model: (1) Accuracy (ACC), which is the final averaged accuracy over all tasks; (2) Backward transfer (BWT), which is the average accuracy change of each task after learning the new task.

$$ACC = \frac{1}{T}\sum_{i=1}^{T} A_{T,i},$$

$$BWT = \frac{1}{T-1}\sum_{i=1}^{T-1}(A_{T,i} - A_{i,i}),$$

where $A_{i,j}$ denotes the testing accuracy of task j upon the completion of learning task i.

4) Experimental Results: We can see from Table 1 that our LMSP method outperforms other baseline methods in both ACC and BWT. It is worth noting that the BWT performance in our method is generally better than CUBER. This improvement stems from our approach of dividing old tasks into multiple local tasks, making it easier to identify highly correlated local tasks for the new task. To understand the efficacy of the proposed techniques, we further conduct ablation studies. We show the effects with different rank values and number of anchor points for our approach in Fig. 1. Due to space limitation, we relegate the ablation study results with different kernel types to the Appendix F.

4-1) Effect of Low Rank: Fig. 1(a) and (c) show the results of our method using a different low-rank value r. We can see that, as expected, the model's performance becomes better when the rank becomes higher. In general, a higher rank value implies less information loss during the base construction. Further, as the rank value becomes sufficiently high, the performance improvement becomes insignificant since most of the information has already been included.

4-2) Effect of Anchor Point Number: Fig. 1(b) and (d) illustrate the performance of our LMSP method with a different number of anchor points. We can see that more anchor points lead to better performance since more candidate old tasks are generated, thus it would be easier to find more correlated old tasks with the new task. However, as the number of anchor points increases, the computation cost also increases correspondingly, which implies a trade-off between performance and cost.

Table 2: Training time comparison on CIFAR-100 Split, 5-Datasets and MiniImageNet. Here the training time is normalized with respect to the value of GPM. Please refer [Saha et al., 2021] for more specific time.

Training time	OWM	EWC	HAT	A-GEM	ER-Res	GPM	TRPG	CUBER	**LMSP** (r=5)	**LMSP** (r=20)	**LMSP** (r=25)
PMNIST	-	1.49	1.23	2.57	1.34	1	1.37	1.52	**0.37**	0.48	0.52
Cifar-100 Split	2.41	1.76	1.62	3.48	1.49	1	1.65	1.86	**0.24**	0.41	0.46
5-Dataset	-	1.52	1.47	2.41	1.40	1	1.21	1.55	**0.42**	0.63	0.67
MiniImageNet	-	1.22	0.91	1.79	0.82	1	1.34	1.61	**0.18**	0.30	0.33

4-3) Results of training time: We show the results of forward knowledge transfer(FWT) in Table 2. As shown in the table, we summarize the normalized wall-clock training times of our LMSP algorithm and several baselines with respect to the wall-clock training time of GPM (additional wall-clock training time results can also be found in [Saha et al., 2021]). Here, we set the rank r to $5, 20, 25$ for each local model. We can see that the wall-clock time of our LMSP($r = 5$) method with *only one anchor point* can already reduce the total wall-clock training time of CUBER by 86% on average. Moreover, thanks to the fact that our LMSP approach endows distributed implementation that can run different local models in a parallel fashion, the total walk-clock training time with m anchor points is similar to the single-anchor-point case above.

It is worth mentioning that LMSP achieves comparable training time even with $r = 25$. Furthermore, the results in Fig. 1 and Table 1 indicate that LMSP with $r = 5$ performs on par with TRPG and outperforms GPM in terms of average ACC and BWT, suggesting that a lower rank does not significantly compromise the performance of our method. Our local low-rank-based methods also demonstrate improved efficiency, particularly when compared to CUBER, which relies on a full-rank setting without leveraging local low-rank strategies. In conclusion, results in Table 2 demonstrate the effectiveness of our LMSP approach in terms of computation cost reductions comparing to the original layer-wise full-rank orthogonal-projection-based approach.

Due to space limitation, we relegate the results of forward knowledge transfer for our LMSP approach in Appendix G.

5 CONCLUSION

In this paper, we proposed a new efficient local low-rank orthogonal-projection-based continual learning strategy based on local model space projection (LMSP), which not only reduces the complexity of basis computation but also enables forward and backward knowledge transfers. We conducted a theoretical analysis to show that the new task's performance could benefit from the local old tasks more than just using the global old task under certain circumstances. We also provided a training loss error analysis and showed that the approximation accuracy of LMSP compared to the original full-rank orthogonal-projection-based approach can be both upper and lower bounded. Our extensive experiments on public datasets demonstrated the efficacy of our approach. Future work includes deploying our efficient CL method to some popular deep learning structures such as transformers and large language models (LLMs) and extending our approach to more general CL settings.

References

Wickliffe C Abraham and Anthony Robins. Memory retention–the synaptic stability versus plasticity dilemma. *Trends in neurosciences*, 28(2):73–78, 2005.

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *Proceedings of the European conference on computer vision (ECCV)*, pages 139–154, 2018.

Alex Beutel, Ed H Chi, Zhiyuan Cheng, Hubert Pham, and John Anderson. Beyond globally optimal: Focused learning for improved recommendations. In *Proceedings of the 26th International Conference on World Wide Web*, pages 203–212, 2017.

Daniel Billsus and Michael J. Pazzani. Learning collaborative information filters. In *Proceedings of the Fifteenth International Conference on Machine Learning*, volume 98, pages 46–54, 1998.

Yaroslav Bulatov. Notmnist dataset. *Google (Books/OCR), Tech. Rep.[Online]. Available: http://yaroslavvb. blogspot. it/2011/09/notmnist-dataset. html*, 2, 2011.

Emmanuel J Candes and Yaniv Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, P Dokania, P Torr, and M Ranzato. Continual learning with tiny episodic memories. In *Workshop on Multi-Task and Lifelong Reinforcement Learning*, 2019a.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019b.

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 Yuejie Chi. Harnessing structures in big data via guaranteed low-rank matrix estimation: Recent theory and fast algorithms via convex and nonconvex optimization. *IEEE Signal Processing Magazine*, 35(4): 14–31, 2018.

Zhiyuan Chen and Bing Liu. *Lifelong machine learning*. Morgan & Claypool Publishers, 2018.

Yuejie Chi, Yue M Lu, and Yuxin Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67 (20):5239–5269, 2019.

Evangelia Christakopoulou and George Karypis. Local latent space models for top-n recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1235–1243, 2018.

Thomas George and Srujana Merugu. A scalable collaborative filtering framework based on co-clustering. In *Fifth IEEE International Conference on Data Mining (ICDM'05)*, pages 4–pp. IEEE, 2005.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Trevor Hastie, Rahul Mazumder, Jason D Lee, and Reza Zadeh. Matrix completion and low-rank svd via fast alternating least squares. *The Journal of Machine Learning Research*, 16(1):3367–3402, 2015.

Prateek Jain, Raghu Meka, and Inderjit Dhillon. Guaranteed rank minimization via singular value projection. *Advances in Neural Information Processing Systems*, 23, 2010.

Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

Yehuda Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434, 2008.

Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

Yann LeCun, Corinna Cortes, Chris Burges, et al. Mnist handwritten digit database, 2010.

Joonseok Lee, Seungyeon Kim, Guy Lebanon, and Yoram Singer. Local low-rank matrix approximation. In *International conference on machine learning*, pages 82–90. PMLR, 2013.

Joonseok Lee, Samy Bengio, Seungyeon Kim, Guy Lebanon, and Yoram Singer. Local collaborative ranking. In *Proceedings of the 23rd international conference on World wide web*, pages 85–96, 2014.

Wei Li, Tao Feng, Hangjie Yuan, Ang Bian, Guodong Du, Sixin Liang, Jianhong Gan, and Ziwei Liu. Unigradfs: Unified gradient projection with flatter sharpness for continual learning. *IEEE Transactions on Industrial Informatics*, 2024.

Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning, 2019.

Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. Beyond not-forgetting: Continual learning with backward knowledge transfer. *Advances in Neural Information Processing Systems*, 35:16165–16177, 2022a.

Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. Trgp: Trust region gradient projection for continual learning. In *The Tenth International Conference on Learning Representations*, 2022b.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. *Advances in neural information processing systems*, 30, 2017.

Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

Andriy Mnih and Russ R Salakhutdinov. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20, 2007.

Praneeth Netrapalli, Niranjan UN, Sujay Sanghavi, Animashree Anandkumar, and Prateek Jain. Non-convex robust pca. *Advances in neural information processing systems*, 27, 2014.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*. Granada, 2011.

Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021.

Ruslan Salakhutdinov and Andriy Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *Proceedings of the 25th international conference on Machine learning*, pages 880–887, 2008.

Badrul M Sarwar, George Karypis, Joseph Konstan, and John Riedl. Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering. In *Proceedings of the fifth international conference on computer and information technology*, volume 1, pages 291–324, 2002.

Joan Serra, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *International conference on machine learning*, pages 4548–4557. PMLR, 2018.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

Tian Tong, Cong Ma, and Yuejie Chi. Accelerating ill-conditioned low-rank matrix estimation via scaled gradient descent. *The Journal of Machine Learning Research*, 22(1):6639–6701, 2021.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

Matt P Wand and M Chris Jones. *Kernel smoothing*. CRC press, 1994.

Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. *Advances in Neural Information Processing Systems*, 36:69054–69076, 2023.

Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648, 2022a.

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022b.

Ke Wei, Jian-Feng Cai, Tony F Chan, and Shingyu Leung. Guarantees of riemannian optimization for low rank matrix recovery. *SIAM Journal on Matrix Analysis and Applications*, 37(3):1198–1222, 2016.

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

Xingyu Xu, Yandi Shen, Yuejie Chi, and Cong Ma. The power of preconditioning in overparameterized low-rank matrix sensing. In *International Conference on Machine Learning*, pages 38611–38654. PMLR, 2023.

Zhaopeng Xu, Qi Qin, Bing Liu, and Dongyan Zhao. Disentangled representations for continual learning: Overcoming forgetting and facilitating knowledge transfer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 143–159. Springer, 2024.

Chengyi Yang, Mingda Dong, Xiaoyue Zhang, Jiayin Qi, and Aimin Zhou. Introducing common null space of gradients for gradient projection methods in continual learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 5489–5497, 2024.

Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continual learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1(8):364–372, 2019.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Menghao Zhang, Binbin Hu, Chuan Shi, and Bai Wang. Local low-rank matrix approximation with preference selection of anchor points. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1395–1403, 2017.

Appendix

Jin Shang[1] **Simone Shao**[1] **Tian Tong**[1] **Fan Yang**[1] **Yetian Chen**[1] **Yang Jiao**[1] **Jia Liu**[2,1] **Yan Gao**[1]

[1]Amazon.com, Seattle, WA, USA
[2]The Ohio State University, Columbus, OH, USA,
[1]{imjshang, simengsh, tongtn, fnam, yetichen, jaoyan, yanngao}@amazon.com
[2] liu@ece.osu.edu

A PROOF OF THEOREM 3.4

Proof. For a $\frac{H}{2}$-smooth loss function $\mathcal{L}$, it can be easily shown that $\mathcal{F}$ is H-smooth. (1) For any $k \in [0, K]$, we can have:

$$\begin{aligned}\mathcal{F}(\mathbf{W}^{(k+1)}) &\leq \mathcal{F}(\mathbf{W}^{(k)}) + \nabla\mathcal{F}(\mathbf{W}^{(k)})^\top(\mathbf{W}^{(k+1)} - \mathbf{W}^{(k)}) + \frac{H}{2}\|\mathbf{W}^{(k+1)} - \mathbf{W}^{(k)}\|^2 \\ &= \mathcal{F}(\mathbf{W}^{(k)}) + (\boldsymbol{g}_1(\mathbf{W}^{(k)}) + \boldsymbol{g}_2(\mathbf{W}^{(k)}))^\top(-\alpha\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})) + \frac{\alpha^2 H}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \\ &= \mathcal{F}(\mathbf{W}^{(k)}) - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 - \alpha\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle, \end{aligned} \tag{11}$$

since:

$$\langle\boldsymbol{g}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle = \langle\mathrm{Proj}_{K_h^{(q)}D_1}(\boldsymbol{g}_1(\mathbf{W}^{(k)})), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle + \langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle, \tag{12}$$

$$\langle\boldsymbol{g}_2(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle = \langle\mathrm{Proj}_{K_h^{(q)}D_1}(\boldsymbol{g}_2(\mathbf{W}^{(k)})), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle + \langle\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle, \tag{13}$$

and:

$$\langle\mathrm{Proj}_{K_h^{(q)}D_1}(\boldsymbol{g}_1(\mathbf{W}^{(k)})), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle = 0, \tag{14}$$

$$\langle\mathrm{Proj}_{K_h^{(q)}D_1}(\boldsymbol{g}_2(\mathbf{W}^{(k)})), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle = 0. \tag{15}$$

For the term $\langle\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle$, it follows that:

$$\begin{aligned}&\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\ =&\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}) + \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\ =&\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle + \langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\ =&\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle + \langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle + \langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle. \end{aligned} \tag{16}$$

Considering

$$\begin{aligned}&2\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle + \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \\ =&\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}) + \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \geq 0, \end{aligned} \tag{17}$$

we have:

$$\langle\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \geq -\frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2, \tag{18}$$

and similarly:

$$\langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle \geq -\frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2. \tag{19}$$

Combining Eq. (16), Eq. (18) and Eq. (19) gives a lower bound on $\langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle$, i.e.,

$$\begin{aligned}
&\langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\
\geq& -\frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \\
&- \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle \\
\geq& -\frac{H^2(1-\lambda_1^2)}{8}\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \\
&- \frac{H^2(1-\lambda_1^2)}{8}\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle \\
\geq& -\frac{H^2(1-\lambda_1^2)}{4}\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle,
\end{aligned} \tag{20}$$

where the second inequality is true due to the smoothness of the loss function and:

$$\begin{aligned}
\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 &= \|\boldsymbol{g}_1(\mathbf{W}^{(k)}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 - \|\mathrm{Proj}_{K_h^{(q)}D_1}(\boldsymbol{g}_1(\mathbf{W}^{(k)}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}))\|^2 \\
&\leq (1-\lambda_1^2)\|\boldsymbol{g}_1(\mathbf{W}^{(k)}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2,
\end{aligned} \tag{21}$$

as well as

$$\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)}) - \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\|^2 \leq (1-\lambda_1^2)\|\boldsymbol{g}_2(\mathbf{W}^{(k)}) - \boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2. \tag{22}$$

Based on the local low-rank orthogonal-projection-based update, it can be seen that:

$$\mathbf{W}^{(k)} = \mathbf{W}^{(0)} - \alpha\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}). \tag{23}$$

Therefore, continuing with Eq. (11), we have:

$$\begin{aligned}
&\mathcal{F}(\mathbf{W}^{(k+1)}) \\
\leq& \mathcal{F}(\mathbf{W}^{(k)}) - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 - \alpha\langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\
\leq& \mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha^3 H^2(1-\lambda_1^2)}{4}\|\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2 + \frac{\alpha}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 \\
&- \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\|,
\end{aligned} \tag{24}$$

where the last term is based on the definition of projection. Since

$$\alpha \leq \frac{\gamma\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{HBK} \leq \frac{\gamma\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{H\|\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|}, \tag{25}$$

thus

$$\begin{aligned}
&\frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \frac{\alpha^2 H^2(1-\lambda_1^2)}{4}\|\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2 \\
\leq& \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \frac{\gamma^2(1-\lambda_1^2)\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2}{4H^2\|\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2}H^2\|\sum_{i=0}^{k-1}\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2 \\
=& \frac{2+\gamma^2(1-\lambda_1^2)}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2.
\end{aligned} \tag{26}$$

Therefore, we can obtain that:

$$\begin{aligned}
&\mathcal{F}(\mathbf{W}^{(k+1)}) \\
\le&\mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha[2+\gamma^2(1-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
\le&\mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 \\
\le&\mathcal{F}(\mathbf{W}^{(k)}),
\end{aligned} \tag{27}$$

where the second inequality is true because:

$$\begin{aligned}
&\lambda_1 \ge \sqrt{1 - 2\frac{2\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| - \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{\gamma^2\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}} \\
\Longrightarrow &\frac{\alpha[2+\gamma^2(1-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \le 0.
\end{aligned} \tag{28}$$

This sufficient decrease of the objective function value indicates that the optimal $\mathcal{F}(\mathbf{W}^\star)$ can be obtained for convex loss functions.

(2) For a non-convex loss function $\mathcal{L}$, as $\nabla\mathcal{F}(\mathbf{W}^{(k)}) = \boldsymbol{g}_1(\mathbf{W}^{(k)}) + \boldsymbol{g}_2(\mathbf{W}^{(k)})$ we have Eq. (27):

$$\begin{aligned}
&\mathcal{F}(\mathbf{W}^{(k+1)}) \\
\le&\mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha[2+\gamma^2(1-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
&- \frac{\alpha}{2}[\|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 - \|\boldsymbol{g}_1(\mathbf{W}^{(k)})\|^2 - \|\boldsymbol{g}_2(\mathbf{W}^{(k)})\|^2 - 2\langle\boldsymbol{g}_1(\mathbf{W}^{(k)}), \boldsymbol{g}_2(\mathbf{W}^{(k)})\rangle] \\
\le&\mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha[2+\gamma^2(1-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
&- \frac{\alpha}{2}[\|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 - 2\|\boldsymbol{g}_1(\mathbf{W}^{(k)})\|^2 - 2\|\boldsymbol{g}_2(\mathbf{W}^{(k)})\|^2].
\end{aligned} \tag{29}$$

From Eq. (23) we have

$$\begin{aligned}
\|\boldsymbol{g}_1(\mathbf{W}^{(k)})\|^2 &= \|\boldsymbol{g}_1(\mathbf{W}^{(k)}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}) + \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 \le 2\|\boldsymbol{g}_1(\mathbf{W}^{(k)}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + 2\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 \\
&\le \frac{\alpha^2 H^2}{2}\|\sum_{i=0}^{k-1}\boldsymbol{g}_2(\mathbf{W}^{(i)})\|^2 + 2\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 \\
&\le \frac{\gamma^2}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + 2\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2,
\end{aligned} \tag{30}$$

and

$$\begin{aligned}
\|\boldsymbol{g}_2(\mathbf{W}^{(k)})\|^2 &= \|\boldsymbol{g}_2(\mathbf{W}^{(k)}) - \boldsymbol{g}_2(\mathbf{W}^{(0)}) + \boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 \le 2\|\boldsymbol{g}_2(\mathbf{W}^{(k)}) - \boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 + 2\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 \\
&\le \frac{\alpha^2 H^2}{2}\|\sum_{i=0}^{k-1}\boldsymbol{g}_2(\mathbf{W}^{(i)})\|^2 + 2\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 \\
&\le \frac{\gamma^2}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + 2\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2,
\end{aligned} \tag{31}$$

where the last inequality holds as

$$\alpha \le \frac{\gamma\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{HBK} \le \frac{\gamma\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{H\|\sum_{i=0}^{k-1}\boldsymbol{g}_2(\mathbf{W}^{(i)})\|} \tag{32}$$

Therefore

$$\mathcal{F}(\mathbf{W}^{(k+1)})$$

$$
\begin{aligned}
&\leq \mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha[2+\gamma^2(1-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
&\quad - \frac{\alpha}{2}\|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 + 2\alpha\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + 2\alpha\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 + \alpha\gamma^2\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}\|^2 \\
&\leq \mathcal{F}(\mathbf{W}^{(k)}) - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + \frac{\alpha[2+\gamma^2(5-\lambda_1^2)]}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - \alpha\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
&\quad - \frac{\alpha}{2}\|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 + 2\alpha\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + 2\alpha\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2. \qquad (33)
\end{aligned}
$$

Thus,

$$
\begin{aligned}
&\min_k \|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 \\
&\leq \frac{1}{K}\sum_{k=0}^{K-1}\|\nabla\mathcal{F}(\mathbf{W}^{(k)})\|^2 \\
&\leq \frac{2}{\alpha K}\sum_{k=0}^{K-1}[\mathcal{F}(\mathbf{W}^{(k)}) - \mathcal{F}(\mathbf{W}^{(k+1)})] + \frac{[2+\gamma^2(5-\lambda_1^2)]}{2(K-1)}\sum_{k=1}^{K-1}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 - 2\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| \\
&\quad - \frac{1-\alpha H}{K}\sum_{k=0}^{K-1}\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\|^2 + 4\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + 4\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 \\
&\leq \frac{2}{\alpha K}[\mathcal{F}(\mathbf{W}^{(0)}) - \mathcal{F}(\mathbf{W}^{\star})] + \frac{[2+\gamma^2(5-\lambda_1^2)]}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + 4\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + 4\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2, \qquad (34)
\end{aligned}
$$

where the last inequality holds due to $\mathcal{F}(\mathbf{W}^{\star}) \leq \mathcal{F}(\mathbf{W}^{(K)})$. □

B PROOF OF THEOREM 3.5

Proof. (1) For local low-rank orthogonal-projection-based update, we have

$$
\mathbf{W}^s = \mathbf{W} - \alpha[\boldsymbol{g}_2(\mathbf{W}) - \mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}))] = \mathbf{W} - \alpha\bar{\boldsymbol{g}}_2(\mathbf{W}). \qquad (35)
$$

For full-rank orthogonal-projection-based update, we have

$$
\mathbf{W}^c = \mathbf{W} - \alpha[\boldsymbol{g}_2(\mathbf{W}) - \mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))] = \mathbf{W} - \alpha\ddot{\boldsymbol{g}}_2(\mathbf{W}). \qquad (36)
$$

Based on Eq. (11) and the smoothness of the objective function, we have an upper bound on $\mathcal{F}(\mathbf{W}^s)$:

$$
\mathcal{F}(\mathbf{W}^s) \leq \mathcal{F}(\mathbf{W}) - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 - \alpha\langle\bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle, \qquad (37)
$$

and a lower bound on $\mathcal{F}(\mathbf{W}^c)$:

$$
\mathcal{F}(\mathbf{W}^c) \geq \mathcal{F}(\mathbf{W}) + \nabla\mathcal{F}(\mathbf{W})^\top(\mathbf{W}^c - \mathbf{W}) - \frac{H}{2}\|\mathbf{W}^c - \mathbf{W}\|^2. \qquad (38)
$$

Combining Eq. (37) and Eq. (38), we have

$$
\begin{aligned}
&\mathcal{F}(\mathbf{W}^s) \\
&\leq \mathcal{F}(\mathbf{W}^c) - \nabla\mathcal{F}(\mathbf{W})^\top(\mathbf{W}^c - \mathbf{W}) + \frac{H}{2}\|\mathbf{W}^c - \mathbf{W}\|^2 - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 - \alpha\langle\bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle \\
&= \mathcal{F}(\mathbf{W}^c) - \langle\boldsymbol{g}_1(\mathbf{W}) + \boldsymbol{g}_2(\mathbf{W}), -\alpha\ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle + \frac{\alpha^2 H}{2}\|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 \\
&\quad - \alpha\langle\bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle \\
&= \mathcal{F}(\mathbf{W}^c) + \alpha\langle\boldsymbol{g}_1(\mathbf{W}), \alpha\ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle + \alpha\langle\boldsymbol{g}_2(\mathbf{W}), \ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle + \frac{\alpha^2 H}{2}\|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 \\
&\quad - \alpha\langle\bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle
\end{aligned}
$$

$$=\mathcal{F}(\mathbf{W}^c) + [\alpha + \frac{\alpha^2 H}{2}]\|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 - [\alpha - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 - \alpha\langle \bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle, \tag{39}$$

where the last equality is true because

$$\langle \boldsymbol{g}_2(\mathbf{W}), \ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle = \langle \mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W})), \ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle + \langle \ddot{\boldsymbol{g}}_2(\mathbf{W}), \ddot{\boldsymbol{g}}_2(\mathbf{W})\rangle, \tag{40}$$

and both $\boldsymbol{g}_1(\mathbf{W})$ and $\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))$ are orthogonal to $\ddot{\boldsymbol{g}}_2(\mathbf{W})$. Based on Eq. (20), the last term has:

$$\begin{aligned}
&\langle \bar{\boldsymbol{g}}_1(\mathbf{W}), \bar{\boldsymbol{g}}_2(\mathbf{W})\rangle \\
\geq& -\frac{H^2(1-\lambda_1^2)}{4}\|\mathbf{W} - \mathbf{W}^{(0)}\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 - \frac{1}{2}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\rangle.
\end{aligned} \tag{41}$$

Suppose that $\mathbf{W}$ is the model update at n-th iteration where $n \leq K$. For the local low-rank orthogonal-projection-based update,

$$\begin{aligned}
\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 &= \alpha^2 \|\sum_{i=0}^{n} \bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2 \\
&\leq \frac{\gamma^2 \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2}{H^2 B^2 K^2} n \sum_{i=0}^{n} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|^2 \\
&\leq \frac{\gamma^2 n^2 \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2}{H^2 K^2} \\
&\leq \frac{\gamma^2 \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2}{H^2},
\end{aligned} \tag{42}$$

and similarly for full-rank orthogonal-projection-based update, we also have

$$\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 \leq \frac{\gamma^2 \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2}{H^2}. \tag{43}$$

Therefore, continuing with Eq. (41), we obtain:

$$\begin{aligned}
&\langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(k)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(k)})\rangle \\
\geq& -\frac{2 + \gamma^2(1-\lambda_1^2)}{4}\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|^2 + \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|\|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(0)})\| - \frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 \\
\geq& -\frac{1}{2}\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2,
\end{aligned} \tag{44}$$

where the last inequality holds due to Eq. (28). Continuing with Eq. (39), we get:

$$\mathcal{F}(\mathbf{W}^s) \leq \mathcal{F}(\mathbf{W}^c) + [\alpha + \frac{\alpha^2 H}{2}]\|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 - [\frac{\alpha}{2} - \frac{\alpha^2 H}{2}]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2. \tag{45}$$

Based on assumption, we have

$$\|\mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}))\|_2 = \lambda_3 \|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\|_2 \leq \|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\|_2, \tag{46}$$

thus

$$\begin{aligned}
\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 &= \|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 + \|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\|^2 - \|\mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}))\|^2 \\
&= \|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2 + (1-\lambda_3^2)\|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\|^2.
\end{aligned} \tag{47}$$

Combining Eq. (45) and Eq. (47) on $\|\ddot{\boldsymbol{g}}_2(\mathbf{W})\|^2$, we have

$$\begin{aligned}
\mathcal{F}(\mathbf{W}^s) &\leq \mathcal{F}(\mathbf{W}^c) + [(\alpha + \frac{\alpha^2 H}{2}) - (\frac{\alpha}{2} - \frac{\alpha^2 H}{2})]\|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 - (1-\lambda_3^2)[\alpha + \frac{\alpha^2 H}{2}]\|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\|^2 \\
&\leq \mathcal{F}(\mathbf{W}^c) + [(\frac{\alpha}{2} + \alpha^2 H)(1-\lambda_1^2)]\|\boldsymbol{g}_2(\mathbf{W})\|^2 - (1-\lambda_3^2)[\alpha + \frac{\alpha^2 H}{2}]{\lambda_1'}^2\|\boldsymbol{g}_2(\mathbf{W})\|^2,
\end{aligned} \tag{48}$$

where the last inequality holds with global sufficient project definition (Definition 1 in [Lin et al., 2022a]) that $\|\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))\| \geq \lambda'_1 \|\boldsymbol{g}_2(\mathbf{W})\|$ and

$$\begin{aligned}
\|\boldsymbol{g}_2(\mathbf{W})\|^2 &= \|\mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W})) + \bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 \\
&= \|\mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}))\|^2 + \|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2 \\
&\geq \lambda_1^2 \|\boldsymbol{g}_2(\mathbf{W})\|^2 + \|\bar{\boldsymbol{g}}_2(\mathbf{W})\|^2. \qquad (49)
\end{aligned}$$

Considering

$$\begin{aligned}
&\lambda_1 \geq \sqrt{1 - \frac{(1-\lambda_3^2)(2+\alpha H)\lambda_1'^2}{1+2\alpha H}} \\
\Longrightarrow\ &\alpha(1-\lambda_1^2)(1+2\alpha H) \leq \alpha(1-\lambda_3^2)(2+\alpha H){\lambda_1'}^2, \qquad (50)
\end{aligned}$$

we get $\mathcal{F}(\mathbf{W}^s) \leq \mathcal{F}(\mathbf{W}^c)$.

(2) Base on the smoothness of the loss function, we have

$$\begin{aligned}
\mathcal{L}_1(\mathbf{W}^{(k)}) &\leq \mathcal{L}_1(\mathbf{W}^{(0)}) + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \mathbf{W}^{(k)} - \mathbf{W}^{(0)} \rangle + \frac{H}{4}\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 \\
&= \mathcal{L}_1(\mathbf{W}^{(0)}) + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), -\alpha \sum_{i=0}^{k-1} \bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}) \rangle + \frac{\alpha^2 H}{4}\| \sum_{i=0}^{k-1} \bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2 \\
&= \mathcal{L}_1(\mathbf{W}^{(0)}) - \alpha \sum_{i=0}^{k-1} \langle \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)}), \bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}) \rangle + \frac{\alpha^2 H}{4}\| \sum_{i=0}^{k-1} \bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2 \\
&\leq \mathcal{L}_1(\mathbf{W}^{(0)}) - \alpha \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\| [\sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)})\|] + \frac{\alpha^2 H k}{4} \sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2. \qquad (51)
\end{aligned}$$

Since $\alpha \leq \frac{4\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{HBk^{1.5}}$, we have

$$\begin{aligned}
\frac{\alpha H k}{4} \sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2 &\leq \frac{\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\|}{B\sqrt{k}} \sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2 \\
&\leq \frac{\|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\| (\sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2)}{\sqrt{\sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2}} \\
&\leq \|\bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\| \sqrt{\sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|^2} \\
&\leq \bar{\boldsymbol{g}}_1(\mathbf{W}^{(0)})\| [\sum_{i=0}^{k-1} \|\bar{\boldsymbol{g}}_2(\mathbf{W}^{(i)}\|]. \qquad (52)
\end{aligned}$$

Therefore, $\mathcal{L}_1(\mathbf{W}^{(k)}) \leq \mathcal{L}_1(\mathbf{W}^{(0)})$ □

C PROOF OF THEOREM 3.6

Proof. Suppose we define the updates as Eq. (35) and Eq. (36) for local low-rank and full-rank updates, for a $\frac{H}{2}$-smooth loss function $\mathcal{L}$, given $\mathcal{F}$ is H-smooth, we can have an upper bound on $\mathcal{F}(\mathbf{W}^c)$:

$$\mathcal{F}(\mathbf{W}^c) \leq \mathcal{F}(\mathbf{W}^s) + \nabla \mathcal{F}(\mathbf{W})^\top (\mathbf{W}^c - \mathbf{W}^s) + \frac{H}{2}\|\mathbf{W}^c - \mathbf{W}^s\|^2. \qquad (53)$$

As the projection consist of the basis and scaling matrices, we have:

$$\mathbf{W}^c - \mathbf{W}^s = \mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W})) - \mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W})) = (\mathbf{S}_j \mathbf{Q}_j {\mathbf{S}_j}^\top - \mathbf{S}_j^{(q)} \mathbf{Q}_j^{(q)} {\mathbf{S}_j^{(q)}}^\top) \boldsymbol{g}_2(\mathbf{W}), \qquad (54)$$

where $\mathrm{Proj}_{D_1}(\boldsymbol{g}_2(\mathbf{W}))$ defines the projection on the input local model space for anchor point q of old task j, and $\mathbf{S}_j$ is the basis for global model space and $\mathrm{Proj}_{K_h^{(q)} D_1}(\boldsymbol{g}_2(\mathbf{W}))$ defines the projection on the input local model space for anchor point q of old task j, and $\mathbf{S}_j^{(q)}$ is the basis for this local model space. $\mathbf{Q}_j$ and $\mathbf{Q}_j^{(q)}$ are the squared scaling matrices corresponding to the basis $\mathbf{S}_j$ and $\mathbf{S}_j^{(q)}$.

Without loss of generality, for any anchor point s_q, we denote by $B_h(s_q)$ the neighborhood of indices near that anchor point, $B_h(s_q) \overset{\text{def}}{=} \{\forall s' \in [M] \times [N] : d(s_q, s') < h\}$ and we use $M(h, s_q)$ and $N(h, s_q)$ to denote the number of unique row and column indices in $\mathcal{B}_h(s_q)$. Also, we denote $n_q = \min(M(h, s_q), N(h, s_q))$.

Based on the loss function Eq. (3), we denote the mapping function $\mathcal{T}(s) = \mathbf{R}_j^l$ as the original matrix where $s = (a, b) \in [M] \times [N_j]$. Then we can describe it locally with $\hat{\mathcal{T}}(s) = \hat{\mathcal{T}}(s_q) = \mathbf{A}^{(q)}\mathbf{B}^{(q)^\top}$ as the estimate for each anchor point s_q. Then following Proposition 1 of [Lee et al., 2013], given that if $|\Omega \cap \mathcal{B}_h(s_q)| \leq C\mu^2 r' n_q \log^6 n_q$, with probability greater than $1 - n_q^{-3}$, we have the total squared-error within a neighborhood of s_q bounded by the following:

$$\mathcal{E}(\mathcal{T})(q,h) = \|K_h^{(q)} \odot (\mathcal{T}(s) - \hat{\mathcal{T}}(s))\|_F \leq Z' h^\beta (4\sqrt{\frac{n_q(2+p)}{p}} + 2) = Z' h^\beta (4\sqrt{3n_q} + 2), \tag{55}$$

here, $\mathcal{T}$ is Hölder continuous $\|\mathcal{T}(x) - \mathcal{T}(x')\|_F \leq Z' d^\beta(x, x')$ with parameter $Z', \beta > 0$. $\mathcal{T}(s)$ is a rank r' matrix satisfies the strong incoherence property with parameter μ described by [Candes and Plan, 2010]. C is a constant. The kernel function K_h is a uniform kernel based on a product distance function. $p = \frac{|\Omega \cap \mathcal{B}_h(s_q)|}{|\mathcal{B}_h(s_q)|}$ is the density of observed samples. Given that the observed set of indices of the matrix is full in our case, thus $|\Omega \cap \mathcal{B}_h(s_q)| = |\mathcal{B}_h(s_q)|$ and $p = 1$.

Let $T = \mathcal{T}^2 = \mathbf{R}_j^l \mathbf{R}_j^{l^\top}$, which is the gram matrix of the original matrix, it is easy to prove that the functions is still Hölder continuous $\|T(x) - T(x')\|_F \leq Z d^\beta(x, x')$ with new parameter $Z > 0$ and the same parameter $\beta > 0$. Thus, the above inequality still hold given $|\Omega \cap \mathcal{B}_h(s_q)| \leq C\mu^2 r n_q \log^6 n_q$ with probability greater than $1 - n_q^{-3}$

$$\mathcal{E}(T)(q,h) = \|K_h^{(q)} \odot (T(s) - \hat{T}(s))\|_F \leq Z h^\beta (4\sqrt{3n_q} + 2), \tag{56}$$

since the number of indices n_q remains the same under squared matrix and the rank of the $r = rank(T(s)) = rank(\mathcal{T}(s)\mathcal{T}(s)^\top) \leq min(rank(\mathcal{T}(s)), rank(\mathcal{T}(s)^\top) = r'$.

Note that for SVD, the left and right singular matrices are unitary matrices, i.e., $UU^\top = VV^\top = I$, and are actually the eigenvector of matrices $RR^\top$. $\mathbf{Q}_j$ and $\mathbf{Q}_j^{(q)}$ are diagonal matrices. Hence, the local space projection corresponding with anchor point s_q for $K_h^{(q)} T(s)$ and $K_h^{(q)} \hat{T}(s)$, we have:

$$\begin{aligned}\|\mathbf{S}_j \mathbf{Q}_j \mathbf{S}_j{}^\top - \mathbf{S}_j^{(q)} \mathbf{Q}_j^{(q)} \mathbf{S}_j^{(q)^\top}\|_F &\leq \|\mathbf{S}_j \sqrt{\mathbf{Q}_j} \mathbf{V}_j \mathbf{V}_j^\top \sqrt{\mathbf{Q}_j}^\top \mathbf{S}_j{}^\top - \mathbf{S}_j^{(q)} \sqrt{\mathbf{Q}_j^{(q)}} \mathbf{V}_j^{(q)} \mathbf{V}_j^{(q)^\top} \sqrt{\mathbf{Q}_j^{(q)}}^\top \mathbf{S}_j^{(q)^\top}\|_F \\ &= \|K_h^{(q)} \odot (T(s) - \hat{T}(s))\|_F.\end{aligned} \tag{57}$$

continuing with Eq. (53) and Eq. (54), we obtain:

$$\begin{aligned}\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s) &\leq \langle \boldsymbol{g}_1(\mathbf{W}) + \boldsymbol{g}_2(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W})\rangle Z h^\beta (4\sqrt{3n_q} + 2) + H\|\boldsymbol{g}_2(\mathbf{W})\|^2 Z^2 h^{2\beta}(24n_q + 9) \\ &= \langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W})\rangle Z h^\beta (4\sqrt{3n_q} + 2) + \|\boldsymbol{g}_2(\mathbf{W})\|^2 [Z h^\beta (4\sqrt{3n_q} + 2) + H Z^2 h^{2\beta}(24n_q + 9)].\end{aligned} \tag{58}$$

We follow the similar derivation of Eq. (16) for the term $\langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W})\rangle$ and have:

$$\begin{aligned}&\langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W})\rangle \\ =&\langle \boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}) + \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W})\rangle \\ =&\langle \boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W})\rangle + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W})\rangle \\ =&\langle \boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W})\rangle + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}) - \boldsymbol{g}_2(\mathbf{W}^{(0)})\rangle + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}^{(0)})\rangle.\end{aligned} \tag{59}$$

Considering:

$$-2\langle \boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W})\rangle + \|\boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \|\boldsymbol{g}_2(\mathbf{W})\|^2$$

$$= \|\boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}) + \boldsymbol{g}_2(\mathbf{W})\|^2 \geq 0, \tag{60}$$

we have:

$$\langle \boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}) \rangle \leq \frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W})\|^2, \tag{61}$$

and similarly:

$$\langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}) - \boldsymbol{g}_2(\mathbf{W}^{(0)}) \rangle \leq \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W}) - \boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 + \frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2. \tag{62}$$

Combining Eq. (60), Eq. (61) and Eq. (62) gives an upper bound on $\langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W}) \rangle$, i.e.,

$$\begin{aligned}
&\langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W}) \rangle \\
\leq &\frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}) - \boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W})\|^2 \\
&+ \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W}) - \boldsymbol{g}_2(\mathbf{W}^{(0)})\|^2 + \frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}^{(0)}) \rangle \\
\leq &\frac{H^2}{8}\|\mathbf{W} - \mathbf{W}^{(0)}\|^2 + \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W})\|^2 \\
&+ \frac{H^2}{8}\|\mathbf{W} - \mathbf{W}^{(0)}\|^2 + \frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}^{(0)}) \rangle \\
\leq &\frac{H^2}{4}\|\mathbf{W} - \mathbf{W}^{(0)}\|^2 + \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W})\|^2 + \frac{1}{2}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \langle \boldsymbol{g}_1(\mathbf{W}^{(0)}), \boldsymbol{g}_2(\mathbf{W}^{(0)}) \rangle,
\end{aligned} \tag{63}$$

Suppose that $\mathbf{W}$ is the model direct update without any projection at n-th iteration where $n \leq K$. Similar to Eq. (42) and Eq. (43), we always have:

$$\|\mathbf{W}^{(k)} - \mathbf{W}^{(0)}\|^2 \leq \frac{\gamma^2\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2}{H^2}. \tag{64}$$

Therefore, continuing with Eq. (63), we obtain:

$$\langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W}) \rangle \leq \frac{2+\gamma^2}{4}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\| + \frac{1}{2}\|\boldsymbol{g}_2(\mathbf{W})\|^2.$$

Combine Eq. (65) and Eq. (58), noted that the function is B-Lipschitz thus $\|\boldsymbol{g}_2(\mathbf{W})\| \leq B$:

$$\begin{aligned}
\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s) &= \langle \boldsymbol{g}_1(\mathbf{W}), \boldsymbol{g}_2(\mathbf{W}) \rangle Zh^\beta(4\sqrt{3n_q+2}) + \|\boldsymbol{g}_2(\mathbf{W})\|^2[Zh^\beta(4\sqrt{3n_q+2}) + HZ^2h^{2\beta}(24n_q+9)] \\
&\leq Zh^\beta(4\sqrt{3n_q+2})[\frac{2+\gamma^2}{4}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\| + \frac{3}{2}B^2] + HZ^2h^{2\beta}(24n_q+9)B^2.
\end{aligned} \tag{65}$$

The error of $\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s)$ has the above upper bound.

For the lower bound of $\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s)$, similar to Eq. (53), we first have the lower bound for error $\mathcal{F}(\mathbf{W}^c)$ as:

$$\mathcal{F}(\mathbf{W}^c) \geq \mathcal{F}(\mathbf{W}^s) + \nabla\mathcal{F}(\mathbf{W})^\top(\mathbf{W}^c - \mathbf{W}^s) - \frac{H}{2}\|\mathbf{W}^c - \mathbf{W}^s\|^2, \tag{66}$$

then following the similar proof above, we have:

$$\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s) \geq Zh^\beta(4\sqrt{3n_q+2})[-\frac{2+\gamma^2}{4}\|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|^2 + \|\boldsymbol{g}_1(\mathbf{W}^{(0)})\|\|\boldsymbol{g}_2(\mathbf{W}^{(0)})\| + \frac{1}{2}B^2] - HZ^2h^{2\beta}(24n_q+9)B^2. \tag{67}$$

which is the lower bound of error $\mathcal{F}(\mathbf{W}^c) - \mathcal{F}(\mathbf{W}^s)$. Since the proof is similar, we omit this in the paper. □

Table 3: Popular kernel functions and their efficiencies relative to Epanechnikov kernel.

Kernel Type	Kernel Function	Efficiency(%)
Uniform	$K_h(s_1, s_2) \propto \mathbf{1}[d(s_1, s_2) < h]$	92.9
Logistic	$K_h(s_1, s_2) \propto \frac{1}{\exp(d(s_1,s_2)/h)+2+\exp(-d(s_1,s_2)/h)}$	88.7
Gaussian	$K_h(s_1, s_2) \propto \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}h^{-2}d(s_1, s_2)^2)$	95.1
Triangular	$K_h(s_1, s_2) \propto (1 - d(s_1,s_2)/h)\mathbf{1}[d(s_1, s_2) < h]$	98.6
Cosine	$K_h(s_1, s_2) \propto \frac{\pi}{4} \cos \frac{\pi d(s_1,s_2)}{2h} \mathbf{1}[d(s_1, s_2) < h]$	99.9
Epanechnikov	$K_h(s_1, s_2) \propto \frac{3}{4}[1 - (d(s_1,s_2)/h)^2]\mathbf{1}[d(s_1, s_2) < h]$	100
Silverman	$K_h(s_1, s_2) \propto \frac{1}{2} \exp(-\frac{\lvert d(s_1,s_2)/h \rvert}{\sqrt{2}}) \cdot \sin(\frac{\lvert d(s_1,s_2)/h \rvert}{\sqrt{2}} + \frac{\pi}{4})$	N/A

D POPULAR KERNEL FUNCTIONS

We list the popular kernel functions in Table 3. The distance d can be computed by some standard distance measures such as ℓ_2 or cosine similarity. For example, for a global representation matrix $\mathbf{R}_j^l = [\mathbf{r}_{j,1}^l, ...\mathbf{r}_{j,N^j}^l] \in \mathbb{R}^{M \times N^j}$ for layer l task j, the distance between a and b on space $[N^j]$ is $d(a, b) = \arccos(\frac{\langle \mathbf{r}_{j,a}^l, \mathbf{r}_{j,b}^l \rangle}{\|\mathbf{r}_{j,a}^l\| \cdot \|\mathbf{r}_{j,b}^l\|})$, where $\mathbf{r}_{j,a}^l, \mathbf{r}_{j,b}^l$ are the a-th and b-th rows of the matrix $\mathbf{R}_j^l$.

E DATASETS INFORMATION

We evaluate the performance of our LMSP on four public datasets for CL: (1) Permuted MNIST [LeCun et al., 2010]: (PMNIST) is a variant of the MNIST dataset [LeCun et al., 2010], where the input pixels are randomly permuted. Following [Lopez-Paz and Ranzato, 2017, Saha et al., 2021], the dataset is divided into 10 tasks by different permutations and each task contains 10 classes; (2) CIFAR-100 Split [Krizhevsky et al., 2009]: the CIFAR-100 dataset [Krizhevsky et al., 2009] is divided into 10 different tasks, and each task is a 10-way multi-class classification problem; (3) 5-Datasets [Lin et al., 2022a,b]: we follow the setting of [Lin et al., 2022a,b] to use a sequence of 5 datasets, which are CIFAR-10, MNIST, SVHN [Netzer et al., 2011], not-MNIST [Bulatov, 2011], Fashion MNIST [Xiao et al., 2017], and the classification problem on each dataset is an individual task; and (4) MiniImageNet [Vinyals et al., 2016]: the MiniImageNet dataset [Vinyals et al., 2016] is divided into 20 tasks, and each task includes 5 classes.

F ABLATION STUDIES ON KERNEL TYPE

Figure 2 shows the influence of different kernels. We adopted five different kernels in our model and the result shows that the Gaussian kernel reach the best performance. Beside, the kernel effect is not that obvious and the overall performance are similar thus we could choose the simplest one in practise to reduce the computation.

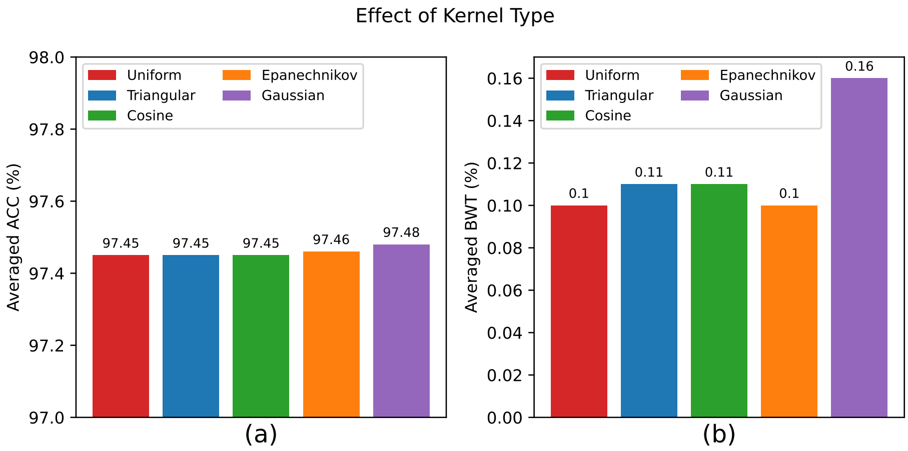

Figure 2: Ablation studies on kernel type.

G RESULTS OF FORWARD KNOWLEDGE TRANSFER.

We show the results of forward knowledge transfer(FWT) in the Table 4. We compared the FWT performance of our LMSP approach to those of the GPM, TRGP, and CUBER methods, which are the most related work to our paper. The value for GPM is zero because we treat GPM as the baseline and consider the relative FWT improvement over GPM. We compare them using four public datasets. We can see from the table that the FWT performance of our LMSP approach beats those of the TRGP and CUBER (two most related and state-of-the-art methods) on the PMNIST, Cifar-100 Split, and 5-Dataset datasets, and is comparable to those of the TRGP and CUBER on the MiniImageNet dataset. Clearly, this shows that the good BWT performance of our LMSP method is not achieved at the cost of sacrificing the FWT performance.

Table 4: Comparison of FWT among GPM, TRGP, CUBER and LMSP. The value for GPM is zero because we treat GPM as the baseline and consider the relative FWT improvement over GPM.

FWT (%)	PMNIST	Cifar-100 Split	5-Dataset	MiniImageNet
GPM	0	0	0	0
TRPG	0.18	2.01	1.98	2.36
CUBER	0.80	2.79	1.96	**3.13**
LMSP($r = 25$)	**0.92**	**2.89**	**2.43**	2.79