Rethinking Gradient Projection Continual Learning: Stability / Plasticity Feature Space Decoupling (Supplementary Material)

In this supplementary material, we first prove the equivalence between updating the model in the null space and updating the model in the direction orthogonal to the feature space. Then we give the proof of the given Subspace Intersection Algorithm and Subspace Sum Algorithm (*i.e.*, Algorithm 1 and Algorithm 2 in body text). After that, we give a proof of the applicability of the proposed Feature Space Paradigm to gradient projection methods [1–4]. After that, we present the implementation details of TRGP+SD and Adam-NSCL+SD. Finally, we provide additional data of our Space Decoupling (SD) algorithm.

1. Appendix A

Theorem 1. *Given a feature matrix $\boldsymbol{M}$ and the new gradient $\boldsymbol{g}$, updating $\boldsymbol{g}$ in the null space of $\boldsymbol{M}$ and updating $\boldsymbol{g}$ in the direction orthogonal to the feature space of $\boldsymbol{M}$ are equivalent.*

Proof. Denote the feature space and the null space of $\boldsymbol{M}$ as $\mathcal{S}$ and $\mathcal{N}$ respectively. By applying SVD to $\boldsymbol{M}$, we have

$$U, \Sigma, V^T = SVD(\boldsymbol{M}) \tag{1}$$

where $U = [U_1, U_2]$ and $\Sigma = \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix}$. All singular values of zero are in Σ_2 [4], thus we have $\mathcal{S} = span\{U_1\}$ and $\mathcal{N} = span\{U_2\}$, which implies that $\mathcal{S}$ and $\mathcal{N}$ are a pair of orthogonal complementary subspaces. Thus we have

$$\begin{aligned} \boldsymbol{g} &= \text{Proj}_{\mathcal{N}}(\boldsymbol{g}) + \text{Proj}_{\mathcal{S}}(\boldsymbol{g}) \\ &= \boldsymbol{g}U_2(U_2^{\text{T}}) + \boldsymbol{g}U_1(U_1^{\text{T}}). \end{aligned} \tag{2}$$

By constraining the gradient update $\boldsymbol{g}^{\mathcal{N}}$ to lie in the null space $\mathcal{N}$, we have

$$\boldsymbol{g}^{\mathcal{N}} = \text{Proj}_{\mathcal{N}}(\boldsymbol{g}) = \boldsymbol{g}U_2(U_2^{\text{T}}) \tag{3}$$

By constraining the gradient update $\boldsymbol{g}^{\mathcal{S}}$ to be orthogonal to $\mathcal{S}$, we have

$$\boldsymbol{g}^{\mathcal{S}} = \boldsymbol{g} - \text{Proj}_{\mathcal{S}}(\boldsymbol{g}) = \boldsymbol{g} - \boldsymbol{g}U_1(U_1^{\text{T}}) = \boldsymbol{g}U_2(U_2^{\text{T}}). \tag{4}$$

It is clear that $\boldsymbol{g}^{\mathcal{N}} = \boldsymbol{g}^{\mathcal{S}}$, which proves the equivalence between updating the model in the null space and updating the model in the direction orthogonal to the feature space. □

2. Appendix B

In this section we give the proof of the given Subspace Intersection Algorithm and Subspace Sum Algorithm (i.e., Algorithm 1 and Algorithm 2 in body text).

2.1. Subspace Intersection

Consider two subspaces $\mathcal{P} = span\{\boldsymbol{P}\}, \mathcal{Q} = span\{\boldsymbol{Q}\}$ in a d-dimensional space, where $\boldsymbol{P} = [\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_{\boldsymbol{k_1}}] \in \mathbb{R}^{d \times k_1}, \boldsymbol{Q} = [\boldsymbol{q}_1, \boldsymbol{q}_2, \cdots, \boldsymbol{q}_{\boldsymbol{k_2}}] \in \mathbb{R}^{d \times k_2}$. According to the definition of the subspace intersection, *i.e.*,

$$\mathcal{P} \cap \mathcal{Q} = \{\boldsymbol{\alpha} | \boldsymbol{\alpha} \in \mathcal{P}, \boldsymbol{\alpha} \in \mathcal{Q}\} \tag{5}$$

we have

$$\begin{aligned} \boldsymbol{\alpha} &= \gamma_1 \cdot \boldsymbol{p}_1 + \gamma_2 \cdot \boldsymbol{p}_2 + \cdots \gamma_{k_1} \cdot \boldsymbol{p}_{k_1} \\ &= \beta_1 \cdot \boldsymbol{q}_1 + \beta_2 \cdot \boldsymbol{q}_2 + \cdots \beta_{k_2} \cdot \boldsymbol{q}_{k_2} \end{aligned} \tag{6}$$

which is equivalent to the following homogeneous linear equation:

$$[\boldsymbol{P}, -\boldsymbol{Q}] \cdot [\gamma_1, \cdots, \gamma_{k_1}, \beta_1, \cdots, \beta_{k_2}]^{\text{T}} = \boldsymbol{0}. \tag{7}$$

As a result, by calculating the basic solutions $\boldsymbol{N} \in \mathbb{R}^{(k_1+k_2) \times k}$, we have

$$\mathcal{P} \cap \mathcal{Q} = span\{\boldsymbol{P} \cdot \boldsymbol{N}[0 : k_1]\} \tag{8}$$

where k is the rank of the basic solutions.

2.2. Subspace Sum

The correctness of the Subspace Sum Algorithm is very obvious. We first remove the common bases in $\boldsymbol{Q}$ between $\boldsymbol{P}$ and $\boldsymbol{Q}$, which is achieved by calculating $\hat{\boldsymbol{Q}} = \boldsymbol{Q} - (\boldsymbol{P}\boldsymbol{P}^T)\boldsymbol{Q}$. Then we orthogonalize $\hat{\boldsymbol{Q}}$ and append the new bases to $\boldsymbol{P}$.

3. Appendix C

In this section we give a proof of the applicability of the proposed Feature Space Paradigm to gradient projection methods [1–4]. Since the applicability of the proposed

paradigm to Orthogonal based approaches like GPM [3] and TRGP [2] is obvious, we only prove the case of Null-space based approaches like Adam-NSCL [4] and AdNS [1].

Following Adam-NSCL [4], we denote X_p as the input feature of the p-th task, and $\bar{X}(t) = [X_1, X_2, \cdots, X_t]$ as the concatenation of input features of task $1, 2, \cdots, t$, ignoring the layer-wise notation. Null-space based approaches constrain the gradient of task $t+1$ to lie in the null space of $\bar{\chi}(t) = \sum_{i=1}^{t} \chi_i$, where $\chi_i = (X_i)^{\mathrm{T}} X_i$ is the uncentered feature covariance. This is equivalent to constraining the gradient to be orthogonal to the feature space of $\bar{\chi}(t)$ according to Section 1. We denote this feature space as $\bar{\mathcal{S}}^a(t)$, and the feature space generated by $\bar{X}(t)$ as $\bar{\mathcal{S}}^b(t)$. According to GPM [3] and TRGP [2] we have

$$\bar{\mathcal{S}}^b(t) = \bar{\mathcal{S}}^b(t-1) + \mathcal{S}_t^b \tag{9}$$

where S_t^b is the task-specific feature subspace generated by X_t. Similarly, we define the feature space generated by χ_t as $\mathcal{S}_t^a$. Next, to verify the applicability of the proposed paradigm, we only need to prove

$$\bar{\mathcal{S}}^a(t) = \bar{\mathcal{S}}^a(t-1) + \mathcal{S}_t^a. \tag{10}$$

According to Adam-NSCL [4] and AdNS [1], the null space of the input feature is equivalent to the null space of the uncentered feature covariance. Then, since the null space and the feature space are a pair of orthogonal complementary subspaces, we have $\mathcal{S}_t^a = \mathcal{S}_t^b$, thus Eq.(10) is proved according to Eq.(9). As a result, the applicability of the proposed paradigm to Null-space based approaches [1, 4] is proved.

4. Appendix D

In this section, we present the implementation details of TRGP+SD and Adam-NSCL+SD.

For TRGP+SD, we follow the implementation of TRGP [2]. On 10-split-CIFAR100 and 20-split-CIFAR100 we use a version of 5-layer AlexNet with an initial learning rate of 0.01, while on 20-split-MiniImageNet we consider a reduced ResNet18 with an initial learning rate of 0.1. The batch size is 64 for all datasets. In $\mathcal{I}/\mathcal{R}$ Approximation we set $\epsilon^{\mathcal{I}} = 0.99$ and $\epsilon^{\mathcal{R}} = 0.92$, while in $\mathcal{I}/\mathcal{R}$ Projection the constraint strength $\zeta^{\mathcal{I}}$ is set to $1e-6$ and $\zeta^{\mathcal{R}}$ is set to $5e-5$.

For Adam-NSCL+SD, we follow the implementation of Adam-NSCL [4]. We use ResNet18 with an initial learning rate of $5e-5$ for all datasets. The batch size is set to 32 for 10-split-CIFAR-100 and 16 for the other two datasets. In $\mathcal{I}/\mathcal{R}$ Approximation we set $\epsilon^{\mathcal{I}} = 5$ and $\epsilon^{\mathcal{R}} = 12$, while in $\mathcal{I}/\mathcal{R}$ Projection the constraint strength $\zeta^{\mathcal{I}}$ is set to $1e-6$ and $\zeta^{\mathcal{R}}$ is set to $5e-5$. Note that here the hyper-parameter $\epsilon^{\mathcal{I}}$ and $\epsilon^{\mathcal{R}}$ correspond to the hyper-parameter a adopted by Adam-NSCL's approximation strategy, the larger of which indicates the fewer dimension of the feature space.

5. Appendix E

In this section we provide additional data of our Space Decoupling (SD) algorithm. Shown in Table 1, we present the dimension of $\mathcal{R}(t)$, $\hat{\mathcal{R}}(t)$, $\mathcal{I}(t)$ and $\hat{\mathcal{I}}(t)$ in each task.

task	$\mathcal{R}(t)$	$\hat{\mathcal{R}}(t)$	$\mathcal{I}(t)$	$\hat{\mathcal{I}}(t)$
2	152	104	7	6
3	199	106	14	13
4	240	110	24	18
5	280	130	33	28
6	295	156	45	41
7	333	179	51	46
8	378	199	58	52
9	444	229	62	54
10	498	268	67	59

Table 1. Mean dimension of $\mathcal{R}(t)$, $\hat{\mathcal{R}}(t)$, $\mathcal{I}(t)$ and $\hat{\mathcal{I}}(t)$. The experiment is implemented by GPM+SD on 10-split-CIFAR-100 for a random seed. Here "mean dimension" is the average of the dimension of layer-wise feature spaces (rounded numbers).

References

[1] Yajing Kong, Liu Liu, Zhen Wang, and Dacheng Tao. Balancing stability and plasticity through advanced null space in continual learning. *arXiv preprint arXiv:2207.12061*, 2022. 1, 2

[2] Sen Lin, Li Yang, Deliang Fan, and Junshan Zhang. Trgp: Trust region gradient projection for continual learning. In *International Conference on Learning Representations*, 2021. 1, 2

[3] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2020. 1, 2

[4] Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 184–193, 2021. 1, 2