模式分析与机器智能
工业和信息化部重点实验室
MIIT Key Laboratory of
Pattern Analysis & Machine Intelligence

ParNeC 模式识别与神经计算研究组
PAttern Recognition and NEural Computing

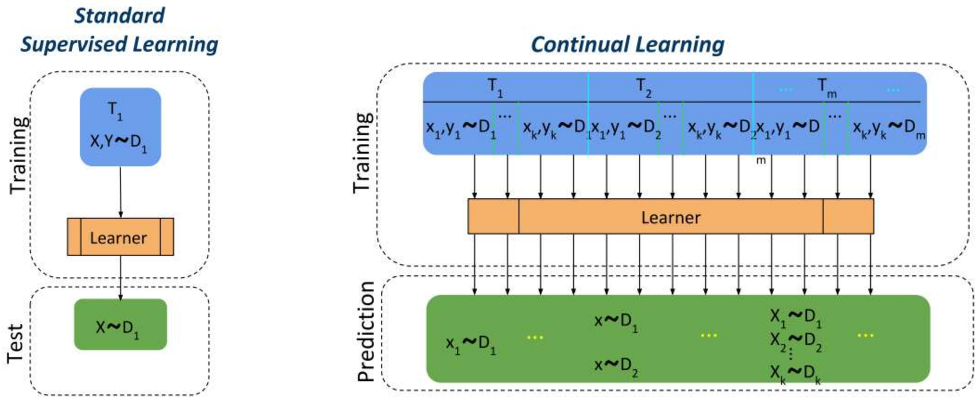
Rethinking Gradient Projection Continual Learning: Stability / Plasticity Feature Space Decoupling

Zhen Zhao[1], Zhizhong Zhang[1,†], Xin Tan[1], Jun Liu[2], Yanyun Qu[3], Yuan Xie[1,†], Lizhuang Ma[1]
[1]School of Computer Science and Technology
East China Normal University, Shanghai, China
[2]Tencent Youtu Lab [3]School of Informatics, Xiamen University, Fujian, China
{51255901056}@stu.ecnu.edu.cn, {zzzhang,xtan,yxie,lzma}@cs.ecnu.edu.cn
{junsenselee}@tencent.com, {yyqu}@xmu.edu.cn

CVPR 2023

Background

- As humans live, they acquire more knowledges through new experiences.
- In machine perspective, continual learning focuses on **how to learn new knowledge from new incoming data with preserving previous knowledge**.

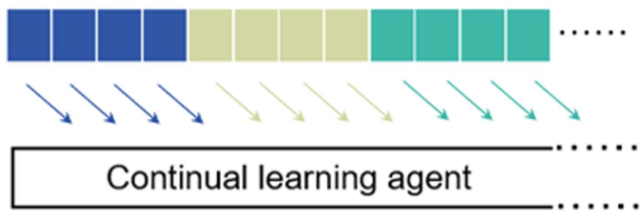


Non-stationary data comes one example at a time in a stream: $(x_1, y_1, t_1), \ldots, (x_i, y_i, t_i), \ldots, (x_{i+j}, y_{i+j}, t_{i+j})$

Data is locally i.i.d. - Samples for a task are drawn from the same unknown joint probability distribution $x_i, y_i \sim P_t(x, y)$

- **Online vs Offline**
  - **Online:** allow only **once** to see the incoming data **except** for examples in the memory.
  - **Offline**: allow to see the examples in current task and episodic memory **many times.**

Offline Continual Learning Setting



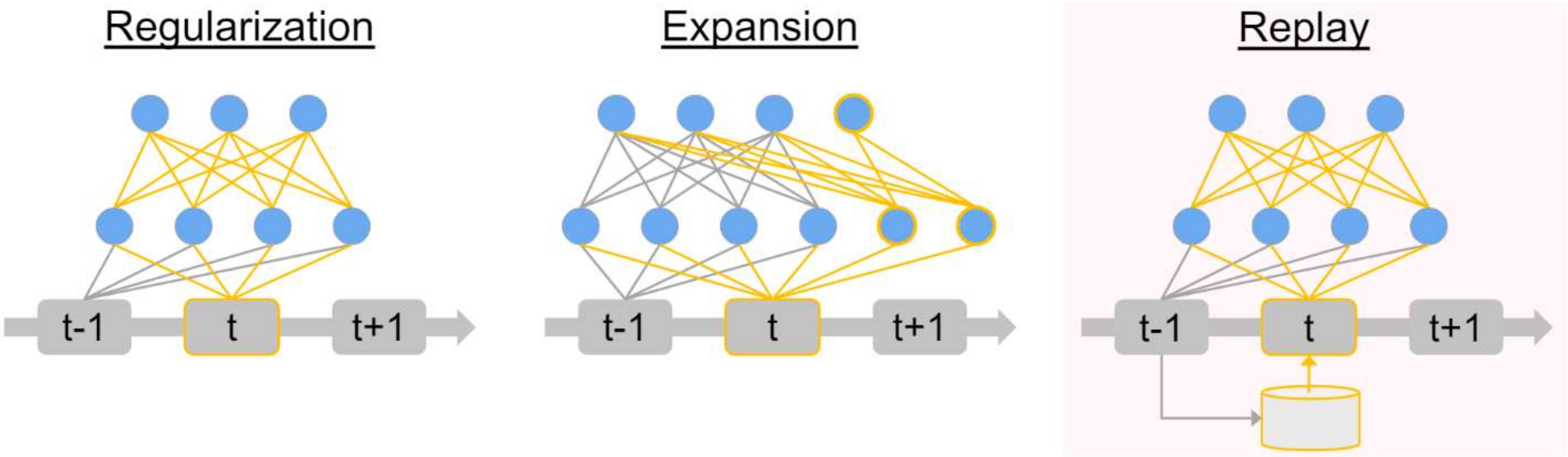
Online Continual Learning Setting

Background

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Previous Works for Continual Learning Methods

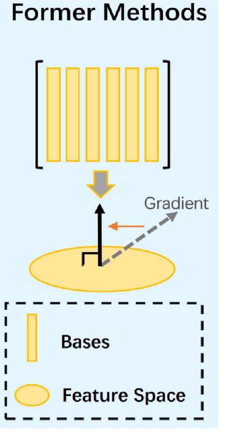
- Regularization based CL
  - **Trade-off:** keeping weights for maintaining previous knowledges vs. changing weights for learning new knowledges.
- Parameter isolation based CL
  - To prevent forgetting previous tasks, an intuition is to construct sufficiently large models, and for each task construct a subset of the larger model. This approach can be achieved by fixing Backbone and adding new branches for new tasks
- Rehearsal/Replay based CL
  - Assume that we can use small amount of data from previous tasks
  - **Key Point:** How to get informative data from the previous tasks?



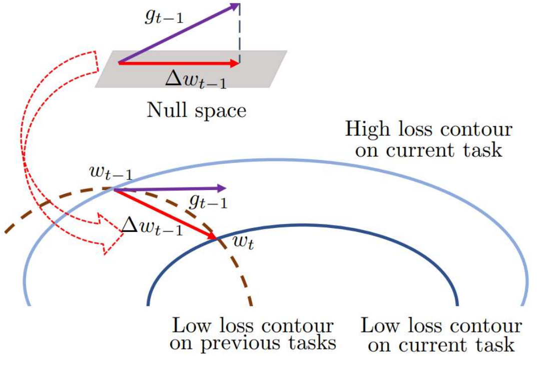
Background

Methods of Gradient Projection

Core idea: Project the gradient of the new task onto the **orthogonal space** of the **input space** of the old task



- **Orthogonal-based**: OWM / GPM / TRGP / etc
  (NMI'19) (ICLR 21) (ICLR 22)



- **Null space-based:** Adam-NSCL / AdNS/etc
  (CVPR'21) (ECCV'22)

Background

Methods of Gradient Projection

Core idea: Project the gradient of the new task into the **orthogonal space** of the **input space** of the old task

$$\boldsymbol{R}_1^l = \left[\left(\boldsymbol{X}_{1,1}^l\right)^T, \left(\boldsymbol{X}_{2,1}^l\right)^T, \ldots, \left(\boldsymbol{X}_{n_s,1}^l\right)^T \right]$$

$$\boldsymbol{R}_1^l = \boldsymbol{U}_1^l \boldsymbol{\Sigma}_1^l \left(\boldsymbol{V}_1^l\right)^T$$

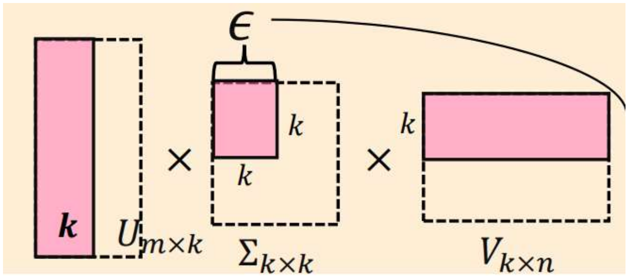
$$\left\| \left(\boldsymbol{R}_1^l\right)_k \right\|_F^2 \geq \epsilon_{th}^l \left\| \boldsymbol{R}_1^l \right\|_F^2$$

$$\boldsymbol{M}^l = \left[\boldsymbol{u}_{1,1}^l, \boldsymbol{u}_{2,1}^l, \ldots, \boldsymbol{u}_{k,1}^l \right]$$

$$\nabla_{\boldsymbol{W}_2^l} L_2 = \nabla_{\boldsymbol{W}_2^l} L_2 - \boldsymbol{M}^l \left(\boldsymbol{M}^l\right)^T \left(\nabla_{\boldsymbol{W}_2^l} L_2 \right)$$

Addressing catastrophic forgetting

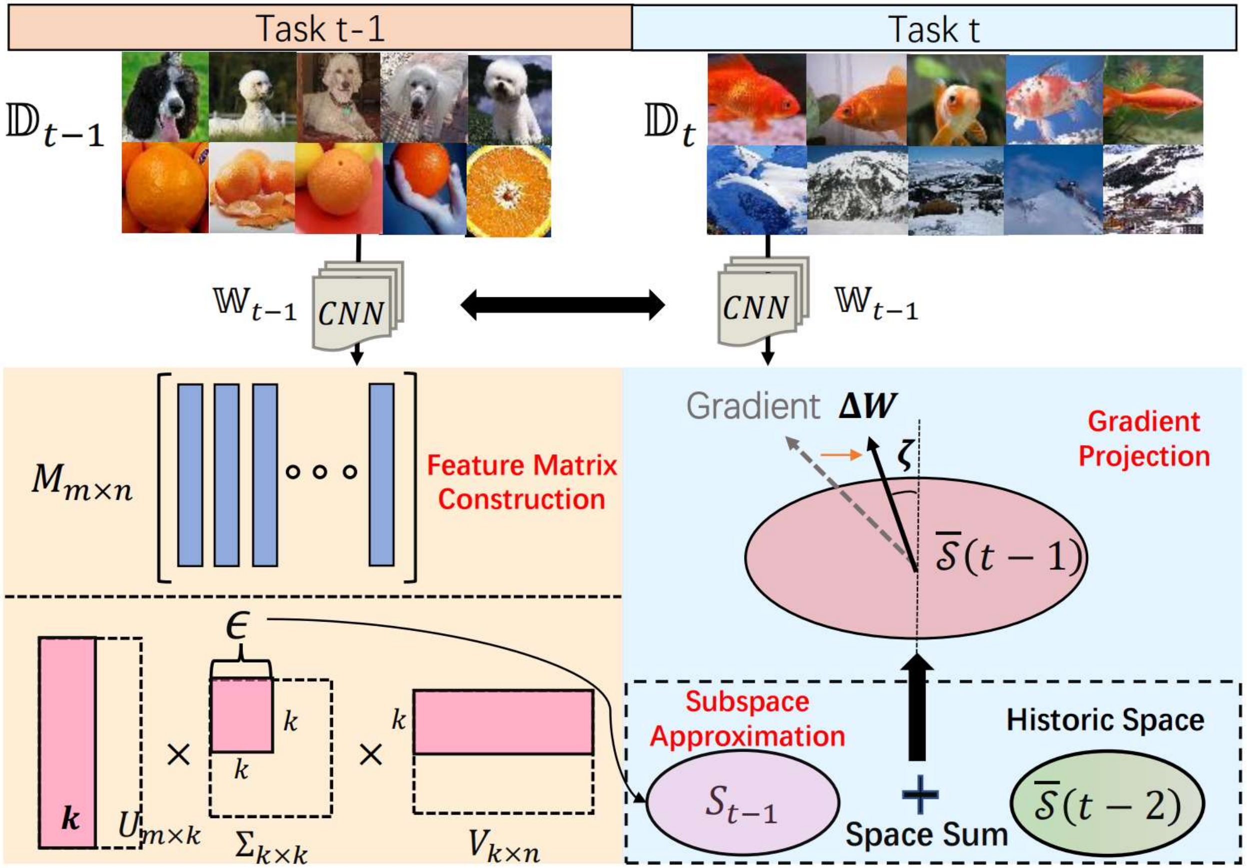
$$\begin{aligned} \boldsymbol{\theta}_t^l x_{j,i}^l &= (\boldsymbol{\theta}_{t-1}^l + \Delta\boldsymbol{\theta}_{t-1}^l) x_{j,i}^l \\ &= \boldsymbol{\theta}_{t-1}^l x_{j,i}^l + \Delta\boldsymbol{\theta}_{t-1}^l x_{j,i}^l \\ &= \boldsymbol{\theta}_{t-1}^l x_{j,i}^l. \end{aligned}$$



Background

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Feature Space Continual Learning Paradigm

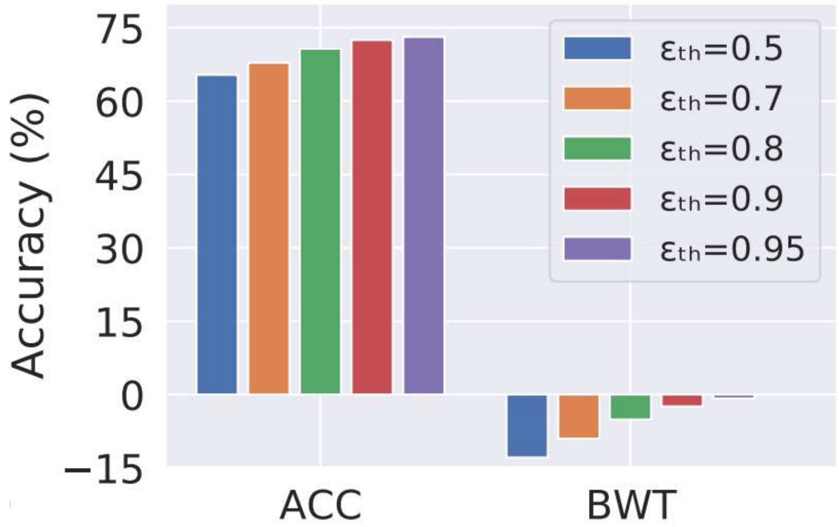


- OWM
- GPM
- TRGP
- Adam-NSCL
- AdNS

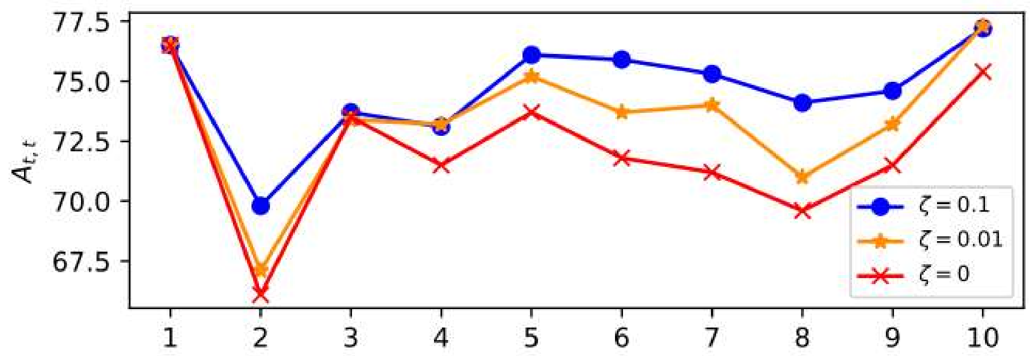
…

Background

Stability-Plasticity Dilemma in Gradient Projection



ϵ increases, stability increases, plasticity decreases



ζ increases, plasticity increases, stability decreases

Motivation

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Space Decoupling: Stability and Plasticity

Inspired by GPM: When updating the feature space, GPM directly **removes duplicate bases** between tasks.

At the end of the task 2 training, we update the GPM with new task-specific bases (of CGS). To obtain such bases, we construct $\boldsymbol{R}_2^l = [\boldsymbol{x}_{1,2}^l, \boldsymbol{x}_{2,2}^l, ..., \boldsymbol{x}_{n_s,2}^l]$ using data from task 2 only. However, before performing SVD and subsequent k-rank approximation, from $\boldsymbol{R}_2^l$ we eliminate the common directions (bases) that are already present in the GPM so that newly added bases are unique and orthogonal to the existing bases in the memory. To do so, we perform the following step :

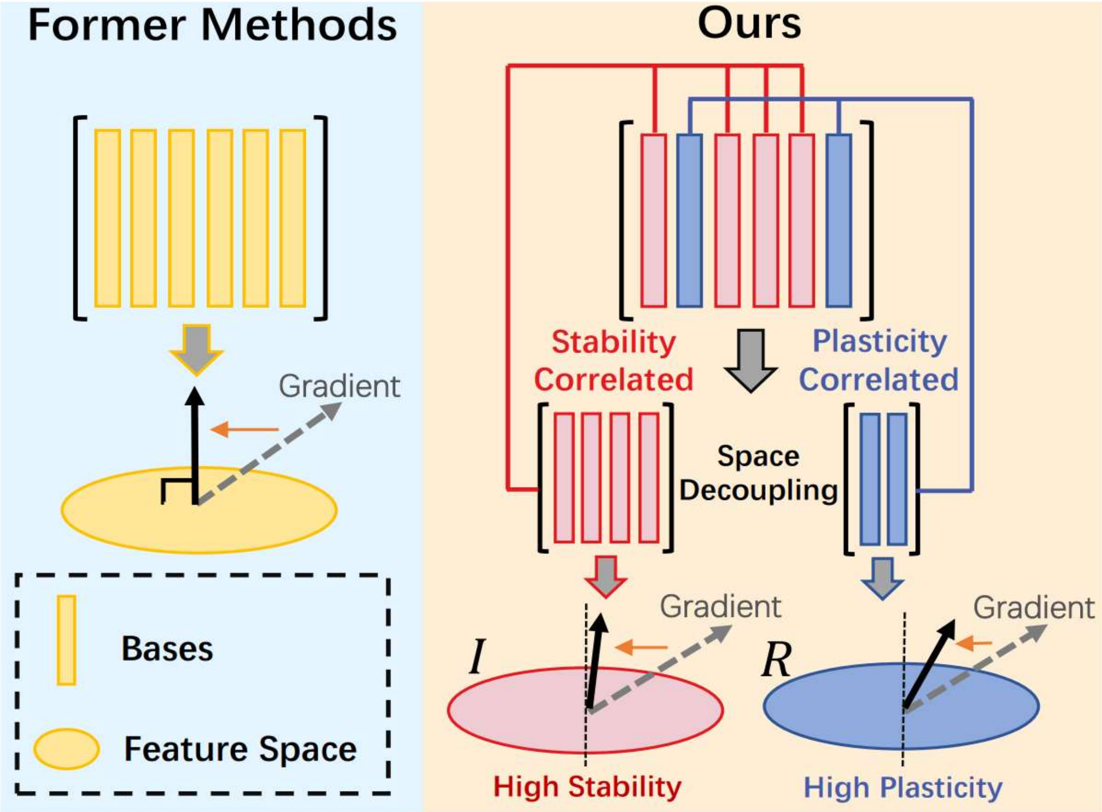
$$\hat{\boldsymbol{R}}_2^l = \boldsymbol{R}_2^l - \boldsymbol{M}^l(\boldsymbol{M}^l)^T(\boldsymbol{R}_2^l) = \boldsymbol{R}_2^l - \boldsymbol{R}_{2,Proj}^l. \tag{8}$$

The authors think **Task-Shared bases** are more important than **Task-Specific bases**

Method

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Space Decoupling Algorithm To Decouple The Feature Space



The stability space $\mathcal{I}$, The plasticity space $\mathcal{R}$

To balance stability and plasticity

$\mathcal{I} \longrightarrow$ More Bases Are Preserved & Stricter Gradient Constraints

$\mathcal{R} \longrightarrow$ Fewer Bases Are Preserved & Looser Gradient Constraints

Method

Space Decoupling Algorithm To Decouple The Feature Space

子空间的交和直和：

$$\mathcal{P} \cap \mathcal{Q} = \{\boldsymbol{\alpha} \mid \boldsymbol{\alpha} \in \mathcal{P}, \boldsymbol{\alpha} \in \mathcal{Q}\}$$

$$\mathcal{P} + \mathcal{Q} = \{\boldsymbol{\alpha} + \boldsymbol{\beta} \mid \boldsymbol{\alpha} \in \mathcal{P}, \boldsymbol{\beta} \in \mathcal{Q}\}$$

稳定性空间：

$$\mathcal{I}(t) = \sum_{1 < i \leq t} \overline{\mathcal{S}}(i-1) \cap \mathcal{S}_i$$

(特征子空间的交的直和) ⟶ **Task-Shared**

可塑性空间：

$$\boldsymbol{R}(t) = \overline{\boldsymbol{S}}(t) - \mathrm{Proj}_{\mathcal{I}(t)}(\overline{\boldsymbol{S}}(t))$$

(稳定性空间在特征空间下的正交补) ⟶ **Task-Specific**

Method

Empirical Evidence

Defining Gradient Update Influence Score:

- The degree of perturbation of the gradient with respect to the feature subspace of the old task.

$$\omega(\boldsymbol{g}) = \sum_{j=1}^{t} \left\| \mathrm{Proj}_{\mathcal{S}_j}(\boldsymbol{g}) \right\|_F^2$$

Further define the Influence Score of the feature subspace：

- The sum of the Influence Scores of the gradient components in that subspace.

$$\Omega(\mathcal{I}(t)) = \frac{\sum_i \omega\left(\boldsymbol{g}_i^{\mathcal{I}}\right)}{\dim(\mathcal{I}(t))} \qquad \Omega(\mathcal{R}(t)) = \frac{\sum_i \omega\left(\boldsymbol{g}_i^{\mathcal{R}}\right)}{\dim(\mathcal{R}(t))}$$

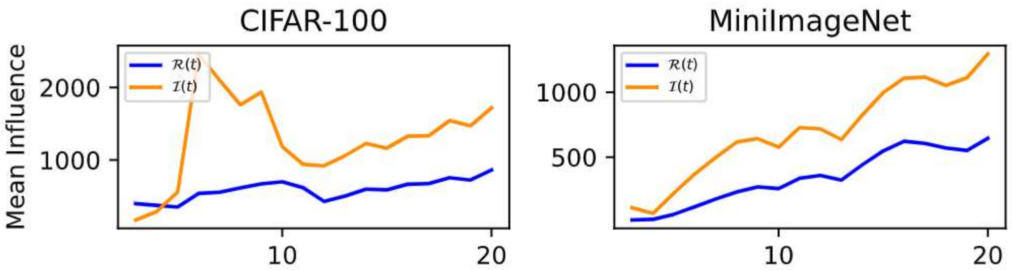
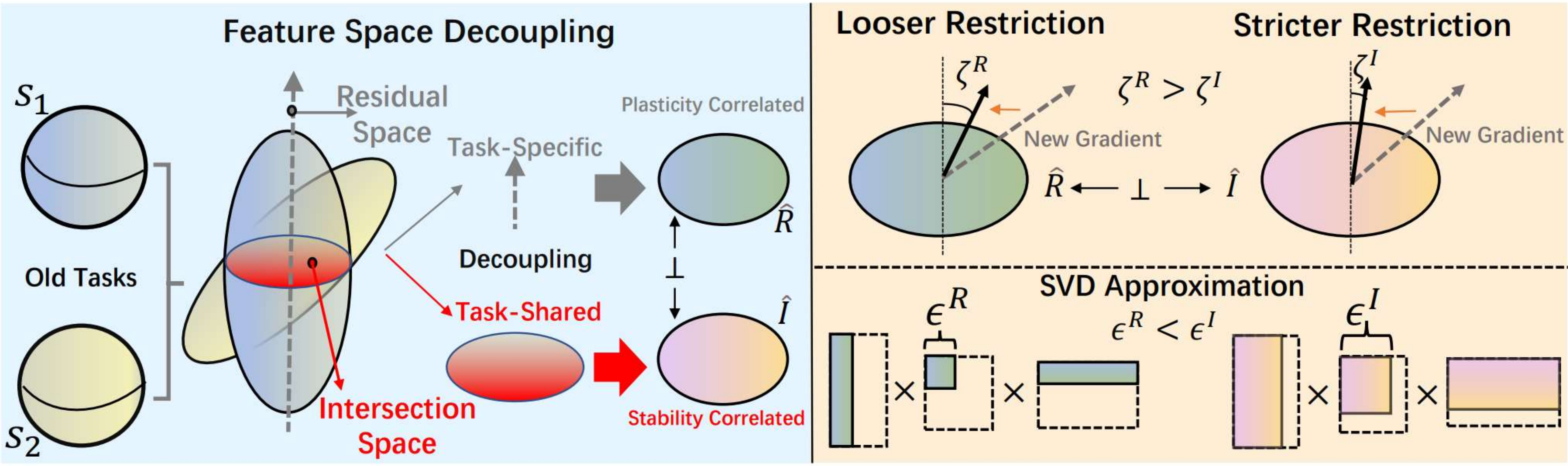


Figure 5. Mean knowledge interference of $\mathcal{I}(t)$ and $\mathcal{R}(t)$.

Method

Space Decoupling Algorithm To Decouple The Feature Space



$$\hat{\boldsymbol{I}}(t) = \mathcal{A}\left(\boldsymbol{I}(t); \epsilon^{\mathcal{I}}\right)$$
$$\hat{\boldsymbol{R}}(t) = \mathcal{A}\left(\boldsymbol{R}(t); \epsilon^{\mathcal{R}}\right)$$

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathcal{L}_{t+1} = &\nabla_{\boldsymbol{\theta}}\mathcal{L}_{t+1} \\ &- \nabla_{\boldsymbol{\theta}}\mathcal{L}_{t+1}\left(1-\zeta^{\mathcal{I}}\right)\hat{\boldsymbol{I}}(t)(\hat{\boldsymbol{I}}(t))^T \\ &- \nabla_{\boldsymbol{\theta}}\mathcal{L}_{t+1}\left(1-\zeta^{\mathcal{R}}\right)\hat{\boldsymbol{R}}(t)(\hat{\boldsymbol{R}}(t))^T\end{aligned}$$

Experiment

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Comparison with SOTA CL Methods

Model	Venue	20-split-MiniImageNet		20-split-CIFAR-100		10-split-CIFAR-100	
		ACC(%)	BWT(%)	ACC(%)	BWT(%)	ACC(%)	BWT(%)
LWF [22]	PAMI'17	57.63	-8.72	74.38	-9.11	70.7	-6.27
EWC [17]	PANS'17	52.01	-12	71.66	-3.72	70.77	-2.83
MAS [2]	ECCV'18	50.12	-5.82	63.84	-6.29	66.93	-4.03
MUC-MAS [24]	ECCV'20	46.24	-3.79	67.22	-5.72	63.73	-3.38
GEM [25]	NIPS'17	-	-	68.89	-1.2	49.48	2.77
A-GEM [5]	ICLR'18	57.24	-12	61.91	-6.88	49.57	-1.13
$*$AdNS [18]	ECCV'22	60.82	-4.24	77.33	-3.25	**77.21**	-2.32
OWM [41]	NMI'19	47.48	-8.57	68.47	-3.37	68.89	-1.88
GPM [31]	ICLR'21	60.41$\pm$ 0.61	-0.7$\pm$0.4	77.53$\pm$0.83	-0.97$\pm$0.59	72.48$\pm$0.4	-0.9$\pm$0
Adam-NSCL [38]	CVPR'21	59.07$\pm$1.1	-4.9$\pm$1.32	75.81$\pm$0.93	-3.98$\pm$0.85	74.97$\pm$1.15	-2.64$\pm$0.91
TRGP [23]	ICLR'22	**63.51$\pm$0.74**	-0.76$\pm$0.25	**80.68$\pm$0.7**	-0.87$\pm$0.46	74.46$\pm$0.32	-0.9$\pm$0.01
GPM+SD		62.39$\pm$0.56	-0.61$\pm$0.11	80.71$\pm$0.82	-0.73$\pm$0.27	73.53$\pm$0.44	-0.83$\pm$0.31
Adam-NSCL+SD		60.38$\pm$0.75	-4.81$\pm$1	76.5$\pm$1.02	-3.99$\pm$0.96	**75.97$\pm$0.66**	-2.88$\pm$0.89
TRGP+SD		**65.8$\pm$0.16**	-0.49$\pm$0.08	**83.84$\pm$0.12**	-0.72$\pm$0.2	75.5$\pm$0.35	-0.96$\pm$0.09

Experiment

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

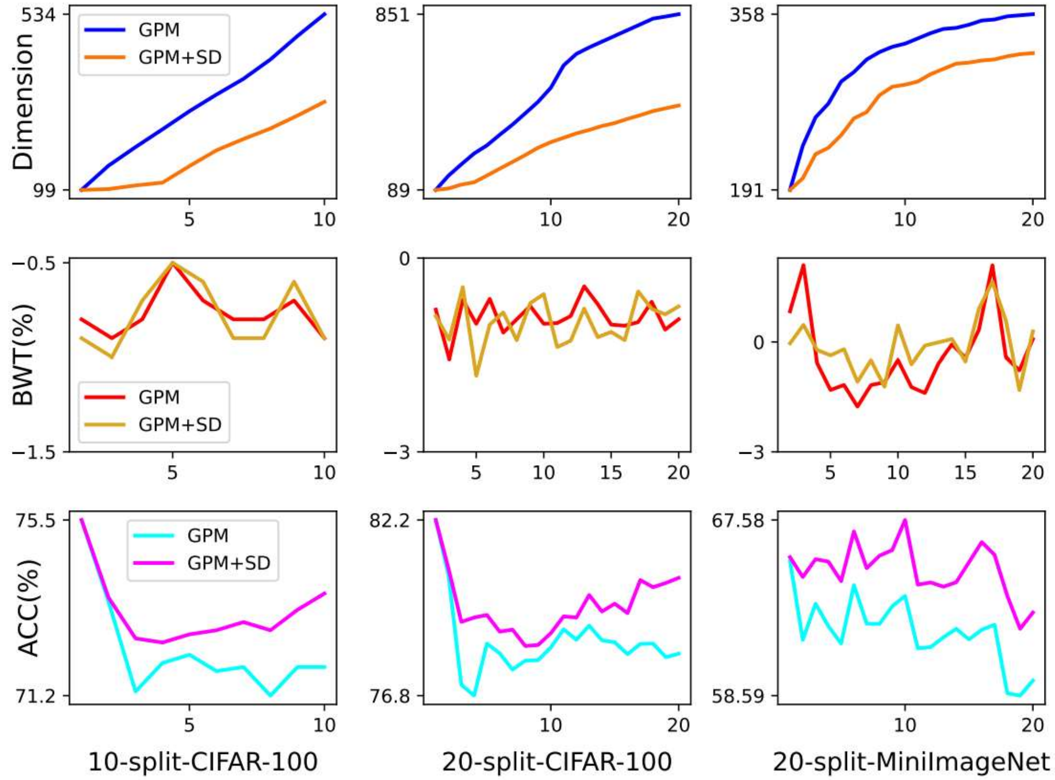
Analysis of I/R-Approximation and I/R-Projection Ablation

Model	$\mathcal{I}/\mathcal{R}$-A	$\mathcal{I}/\mathcal{R}$-P	ACC(%)	BWT(%)
GPM			77.53±0.83	-0.97±0.59
	✓		79.99±0.48	-0.7±0.45
		✓	78.41±1.3	-1.44±0.65
	✓	✓	80.71±0.82	-0.73±0.27
TRGP			80.68±0.7	-0.87±0.46
	✓		83.21±0.36	-0.4±0.02
		✓	81.15±1.22	-1.24±0.71
	✓	✓	83.84±0.12	-0.72±0.2

Computational Complexity Analysis

Datasets	Methods					
	GPM	GPM+SD	TRGP	TRGP+SD	Adam-NSCL	Adam-NSCL+SD
10-split-CIFAR-100	0.25	0.27	0.41	0.45	2.71	2.93
20-split-CIFAR-100	0.31	0.36	0.48	0.53	4.14	4.53
20-split-MiniImageNet	0.58	0.66	0.81	0.92	11.28	12.95

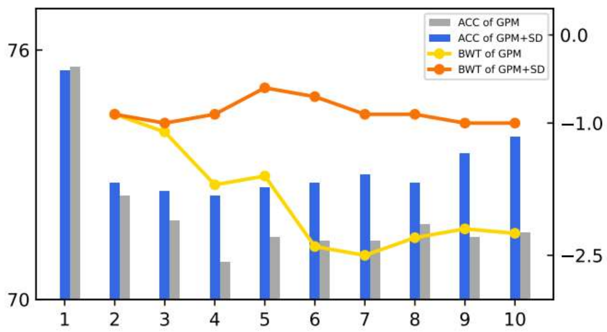
Comparison of feature space dimension



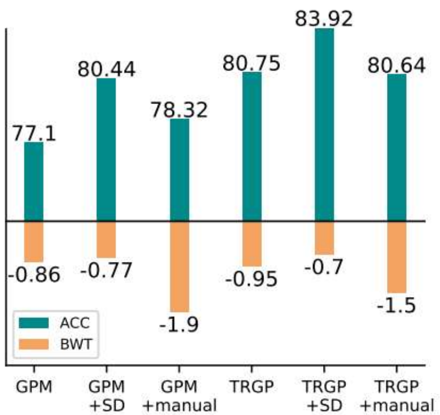
Experiment

ParNeC 模式识别与神经计算研究组 PAttern Recognition and NEural Computing

Forcing GPM to have the same feature dimension with GPM+SD.



Comparisons with different subspaces construction.



Stability and Plasticity analysis.

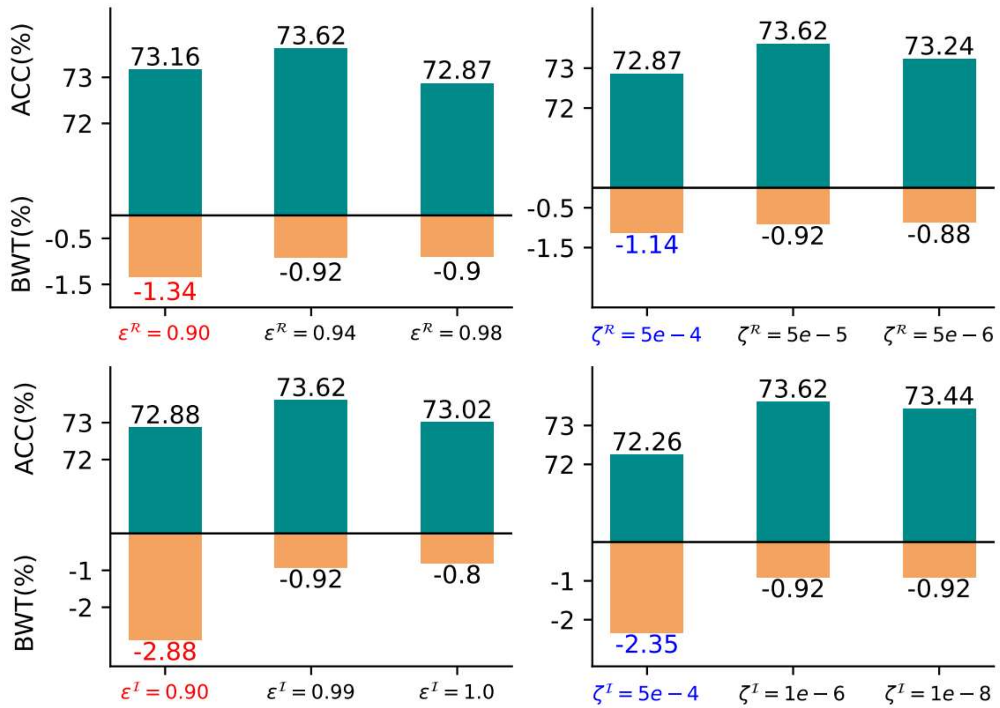


Figure 8. Stability and Plasticity analysis. Experiments are conducted by GPM+SD on 10-split-CIFAR-100. Red and blue values denote the comparison between $\mathcal{R}$ and $\mathcal{I}$.

Thanks