

# KAC:用于持续学习的Kolmogorov-Arnold分类器

胡煜松<sup>1</sup>, 梁紫琛<sup>1</sup>, 杨飞<sup>1,2</sup>, 侯淇彬<sup>1,2</sup>, 刘夏雷<sup>1,2\*</sup>, 程明明<sup>1,2</sup>

<sup>1</sup>VCIP, CS, 南开大学    <sup>2</sup>NKIARI, Shenzhen Futian

{ethanhu, liangztc}@mail.nankai.edu.cn

{feiyang, houqb, xialei, cmm}@nankai.edu.cn

## Abstract

持续学习要求模型能够在连续任务上不断训练而不会遗忘。目前大多数方法采用线性分类器，这些分类器在学习新任务时难以维持稳定的分类空间。受Kolmogorov-Arnold网络(KAN)在简单持续回归任务中保持学习稳定性方面取得成功的启发，我们着手探索其在更复杂持续学习场景中的潜力。在本文中，我们介绍了基于KAN结构的Kolmogorov-Arnold分类器(KAC)，这是一种为持续学习开发的新型分类器。我们深入研究了KAN样条函数的影响，并引入了径向基函数(RBF)，以提高与持续学习的兼容性。我们在几种近期的方法中用KAC替换了线性分类器，并在各种持续学习基准上进行了实验，所有实验均表明性能有所提升，突出了KAC在持续学习中的有效性和鲁棒性。代码可在<https://github.com/Ethanuhuhu/KAC>中获取。

## 1. 引言

深度学习模型通常基于固定数据集进行单次训练，在各种静态任务中展现出卓越性能。然而，现实场景通常在不断变化，这要求模型能够通过数据流实现增量式学习。在此类动态学习场景中，模型常面临重大挑战——灾难性遗忘[12]。持续学习[2, 8, 18, 47]致力于研究如何通过序列化数据输入有效训练模型，旨在使模型能够持续适应并避免遗忘，从而在动态环境中实现长期知识积累。

类增量学习(Class Incremental Learning, CIL) [50]作为持续学习领域的关键挑战，已引发广泛研究关注。该方法要求模型在持续任务流中不断引入新类别，且在新任务训练后需对所有遇到的类别进行分类。当前主流CIL方法通过保留样本并采用例如知识蒸馏[9, 50, 59]或动态架构[6, 10, 31, 60]等技术，从而缓解模型的遗忘问题。随着预训练模型的发展，诸多研究[44, 63]尝试探索预训练模型在CIL中的应用，并取得了显著成果。在此背景下，基于提示的方法[16, 52, 57, 58]凭借其独特优势引发了广泛关注。

\*Corresponding author.

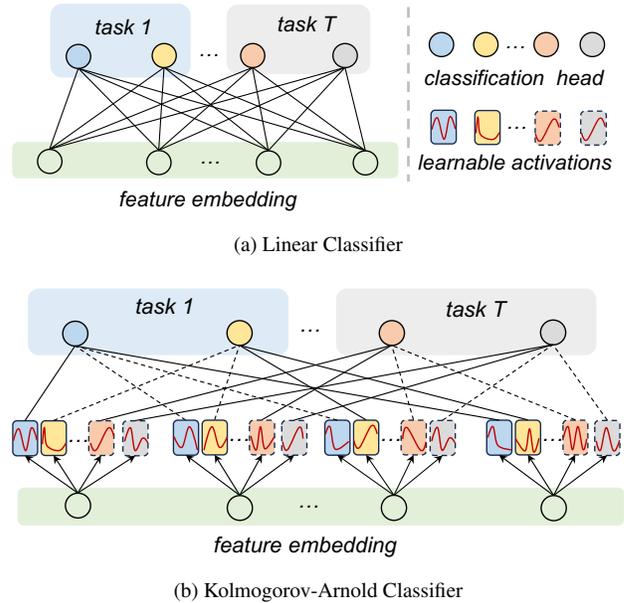


Figure 1. 传统线性分类器与我们提出的Kolmogorov-Arnold分类器之间的简明对比。实线表示激活的权重，虚线表示抑制的权重。(a)传统的分类器在所有任务当中都均等的激活每一个权重，导致新任务中无关权重被同步更新；(b)我们的Kolmogorov-Arnold分类器为每一个通道在每一个类别上学习一个面向类别的可调节激活模式，最大程度减小了无关权重变化导致的遗忘问题。

现有方法中，部分研究[19, 44, 62]通过精心设计的分类器和训练或者推理策略，专注于特征空间的构建，取得了优异性能。这些研究表明，结构合理的特征空间能够有效缓解遗忘问题：稳定的分布对持续分类任务至关重要，而分类器设计对特征空间的构建及连续任务中遗忘的减少具有决定性作用。然而，现有方法[16, 52, 64]大多采用线性分类器或最近类均值分类器(Nearest Class Mean Classifier, NCM) [50]，而针对CIL开发专用分类器以有效缓解模型灾难性遗忘的研究仍较为有限，这一领域值得进一步探索。

最新研究表明，Kolmogorov-Arnold网络(KAN) [41]作为一种新型网络架构，在持续学

习任务中展现出优异的性能表现。研究团队通过构建一个需要顺序拟合5个高斯峰的持续一维回归基准测试，将KAN与传统多层感知机(MLP) [25]进行了系统性对比。KAN表现出更优的性能，能有效缓解灾难性遗忘，这得益于其样条函数的局部特性与内在的局部可塑性机制。这种局部性使KAN在序列任务中能够识别需要重组的相关区域，同时保持其他区域的稳定性 [41]。这些发现激励我们探索KAN在更具挑战性的CIL任务中的应用。

本文提出了一种基于Kolmogorov-Arnold网络(KAN)架构的持续学习即插即用分类器——Kolmogorov-Arnold分类器(KAC)。通过运用Kolmogorov-Arnold表示定理 [34]，我们在分类器的边缘集成了可学习的激活函数。我们发现，传统的B样条函数KAN在处理高维数据时存在困难，导致模型的可塑性不足，这可能会降低模型用作分类器时的适应性。这种限制使得模型在学习新任务时进行过度更新，导致显著遗忘。

为了解决这一问题，我们探索了样条函数，并确定径向基函数(Radial Basis Functions, RBF)是持续学习的有效替代方案。通过在KAC中使用RBF，我们增强了模型适应CIL的能力，同时最大限度地减少了遗忘。由于这些可学习的样条激活，KAC允许模型为每个通道选择感兴趣的特定激活范围，同时保留其他部分的分布，而RBF则使其与CIL更加兼容。如图 1b 所示，这些可学习的激活函数帮助模型选择每个通道中有趣的部分并激活它们用于判定，而不是像图 1a 中的简单线性分类器那样激活所有边缘。这为类别增量学习带来了显著的优势。当新任务到来时，可学习的激活函数可以帮助模型选择每个通道的相关部分进行更新。这可以防止在新任务的训练过程中不相关的特征发生偏移。同时，旧任务中已停用的部分不会受到这些更新的影响，从而减少对旧任务的遗忘。

为了证明KAC的优越性，我们对几种基于提示的持续学习方法进行了实验，这些方法建立在预先训练好的主干网络上，分类器在这些方法中起着关键作用。采用我们方法的模型只需用KAC替换线性分类器，无需进行任何其他修改或超参数调整，即可在多个数据集的各种持续学习(CIL)场景下实现显著提升。此外，在领域增量学习(Domain Incremental Learning, DIL) [56] 设置下进行的实验表明，我们的方法也能提升性能，证明了其有效性和鲁棒性。

我们的主要贡献可以概括如下：

- 我们探索了Kolmogorov-Arnold网络 (KAN) 在持续学习中的应用，并分析了它在持续学习中的弱点以及如何增强其与此类任务的兼容性。
- 我们提出了Kolmogorov-Arnold分类器 (KAC)，这是一种基于KAN结构的新型连续分类器，以径向基函数 (RBF) 作为基函数。KAC增强了CIL方法的稳定性和可塑性。
- 我们将KAC集成到各种方法中，并在多个持续学习基准上验证了它们的性能。结果表明，KAC可以有效减少这些方法中的遗忘。

## 2. 相关工作

类别增量学习旨在按顺序学习一系列分类任务，其中类别数量随着每个任务的进行而增加。其中的主要挑战是灾难性遗忘[43]。目前已有多项研究对此进行探讨，大致可分为三大类：基于正则化、基于结构和基于重放的方法。基于正则化的方法通过采用知识蒸馏技术[9, 59, 61] 或对关键模型参数施加约束[29, 32, 40] 来减少遗忘。基于结构的方法[6, 10, 27, 54] 通过动态网络架构来缓解遗忘。基于重放的方法保留一小部分旧数据[28, 50] 或使用辅助模型[14, 30, 51] 来生成合成数据，并与新类数据相结合以更新模型。

基于预训练模型的持续学习(CIL) 因其强大的迁移能力，在类增量学习中展现出卓越的性能。LAE [13] 和SLCA [63] 等技术通过基于EMA的更新和动态分类器调整增强了模型自适应能力。RanPAC [44] 采用随机投影来提升持续学习效果，而EASE [64] 则专注于优化特定任务的可扩展适配器，以增强知识保留。得益于自然语言处理(NLP) 中参数高效的调优技术，基于提示的方法在类增量学习中取得了令人瞩目的成果。这些方法利用自适应提示来引导冻结的Transformer模型，从而在无需修改编码器参数的情况下实现高效的特定任务学习。L2P [58]、DualPrompt [57]、S-Prompts [56]、CODA-Prompt [52]、HiDe-Prompt [55] 和CPrompt [15] 等技术引入了多样化的提示选择策略，以改进任务学习、知识保留和模型鲁棒性。

**Kolmogorov-Arnold网络 (KAN)** [37] 是一种基于Kolmogorov-Arnold表示定理 [34] 的新型网络架构。它将多变量函数表示为多个单变量函数的组合，并使用非线性样条函数进行近似。一些研究专注于如何应用KAN解决科学问题 [3, 26, 33]，而另一些研究则致力于探索各种基函数以提升性能 [1, 3, 37]。许多研究 [4, 7, 17, 42] 将KAN应用于各个领域，并探讨如何在这些领域有效发挥其优势。这些研究鼓励我们探索KAN在持续学习中的应用。

## 3. 方法

### 3.1. 前置工作

类别增量学习 在类别增量学习(CIL) 中，模型需要逐步学习类别。在每个步骤 $t$ ，模型需要学习特定于该步骤的类别，表示为 $\mathcal{Y}_t$ ，并且只能访问当前数据集 $D_t = \{(\mathbf{x}_t^i, y_t^i)\}_{i=1}^{n_t}$ ，其中 $\mathbf{x}_t^i$ 表示输入图像， $y_t^i$ 是其对应的标签。CIL中的一个关键挑战是如何在学习新任务时保持模型的稳定性以避免灾难性遗忘 [12]。该模型由一个主干网络 $F$  和一个分类器 $h \in \mathbb{R}^{n \times C}$  组成，其中 $n$ 表示嵌入维度， $C$ 表示已学习类别的总数，该模型的任务是预测来自新类别的测试样本以及来自以前遇到的任务的样本的类标签 $y = h(F(\mathbf{x})) \in \mathcal{Y}$ 。

**Kolmogorov-Arnold网络** Kolmogorov-Arnold网络(KAN) [41] 是一种新颖的模型架构，有望成

为多层感知器(MLP) [21, 25] 的替代方案。MLP 依赖于通用近似定理(UAT) [25], 而KAN 则受到Kolmogorov–Arnold表示定理(KAT) [34] 的启发。KAT 假设, 任何定义在有界域上的多元连续函数 $f(\mathbf{x})$  都可以通过加法表示为由单变量连续函数组成的有限组合。Kolmogorov–Arnold表示定理可以写成:

$$f(\mathbf{x}) = f(x_1, x_2, \dots, x_n) = \sum_{q=1}^{2n+1} \Phi_q \left( \sum_{p=1}^n \phi_{q,p}(x_p) \right), \quad (1)$$

其中 $\Phi_q$  和 $\phi_{q,p}$  是每个变量的单变量函数。KAN 将 $\phi_{q,p}$  和 $\Phi_q$  参数化为B样条曲线, 其可学习系数为局部B样条基函数 $B(\mathbf{x})$  [49]。实际应用中, 残差连接由激活函数 $b(\mathbf{x}) = \text{silu}(\mathbf{x}) = \mathbf{x}/(1+e^{-\mathbf{x}})$  组成的线性函数, 与B样条曲线 $\text{spline}(\mathbf{x}) = \sum_i \omega_i B_i(\mathbf{x})$  线性组合, 形成最终的 $\phi$ :

$$\phi(\mathbf{x}) = \omega_b b(\mathbf{x}) + \omega_s \text{spline}(\mathbf{x}), \quad (2)$$

其中 $\omega_b$ 和 $\omega_s$ 表示控制激活函数整体幅值的线性函数。因此, KAN层可以表示为:

$$\mathbf{x}_{l+1} = \underbrace{\begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix}}_{\Phi_l} \mathbf{x}_l. \quad (3)$$

$\mathbf{x}_l$  和 $\mathbf{x}_{l+1}$  分别表示KAN层的输入和输出, 而 $\Phi_l$  是每层的一维单变量函数矩阵。KAN网络由多层KAN层堆叠而成。

### 3.2. 传统的KAN层不是一个好的连续分类器

在[41]中, 作者展示了一个简单一维回归任务的实验结果, 证明了样条函数的局部性可以固有地避免灾难性遗忘。这一发现启发我们将KAN引入连续学习(CIL)。利用KAN局部性的一种直接方法是直接利用KAN层来开发连续分类器, 以替换CIL方法中的线性分类器。为此, 我们只需将线性分类器 $h(\mathbf{x})$  替换为输入维度为 $d$ 、输出维度为 $C$ 的KAN层即可。我们比较了它们在几种基准方法上的性能。实验结果如图2所示, 表明用KAN层替换线性分类器的简单替换并未带来任何改进, 甚至性能反而更差。

我们将KAN层分解为两部分: 残差连接 $b(\mathbf{x})$  和B样条曲线 $\text{spline}(\mathbf{x})$ , 并用这两个部分分别替换线性分类器, 以探究直接引入KAN层为何会增加遗忘。令人惊讶的发现是, B样条函数会导致所有基线模型的性能严重下降。

要理解为何用B样条曲线取代传统的线性分类器会导致严重的遗忘, 我们需要深入研究线性层和样条曲线之间的差异。在高维复杂数据中, 样条函数会遭遇

维数灾难(Curse Of Dimensionality, COD) [20]; 随着数据维数的增加, 模型难以进行数据逼近 [22, 35, 45]。这是因为样条曲线无法有效地建模数据中存在的组成结构, 而线性分类器则受益于其全连接结构, 使其能够有效地学习这种结构 [23]。尽管KAN网络通过堆叠KAN层, 利用近似理论 [41] 来缓解COD, 但对于单个样条层来说, 近似高维函数仍然是一个具有挑战性的问题, 而对于传统的线性分类器来说, 这相对简单。

正是由于B样条函数在高维数据上的弱拟合能力, 将其引入CIL时会导致严重的遗忘。在CIL中, 网络通常由一个主干网络 $F$  (将图像编码为特征嵌入) 和一个分类头 $h$  (作为将嵌入映射到类别概率的高维投影) 组成。大多数方法通过添加分类器来适应新的类别, 同时所有任务之间共享主干网络。最终的对数函数 $l$  总是按如下方式计算:

$$l = h(F(\mathbf{x})), h = [h_1, h_2, \dots, h_t]. \quad (4)$$

为了防止主干网络变化影响特征空间而导致的严重遗忘, 模型必须在新任务训练期间保持主干网络参数的稳定。因此, 许多方法使用正则化技术来限制特征嵌入的变化 [31, 38, 59, 61]。然而, 由于单个B样条层的逼近能力有限, 与传统的线性分类器相比, 该模型需要对主干网络参数进行更广泛的更新, 才能在新任务上取得良好的性能。这种广泛的更新会严重扰乱特征空间, 导致明显的遗忘。

基于以上分析, 我们认为单一B样条函数拟合能力较弱, 阻碍了模型发挥KAN层的局部性, 因此需要增强样条函数的拟合能力, 使KAN结构适应CIL任务。[36, 39]表明, 在特定意义上, 一个基于KAT的浅层可以在通过具有特定组合结构的设计基函数近似高维函数时打破COD问题, 这促使我们探索与CIL适配的基函数类型。

### 3.3. 径向基函数非常适合类增量学习

一些研究 [44, 62, 65]假设分类空间遵循高斯空间, 并在此前提下开发方法, 取得了优异的性能。这表明, 构建高斯分类空间可以帮助模型有效地学习新任务, 同时对抗灾难性遗忘。我们能否找到一种这样的基函数, 使基于KAT的层函数能够作为连续分类器, 从而解决COD问题并有利于CIL? 答案是肯定的!

FastKAN [37]证明了KAN [41]中的B样条基函数可以被径向基函数(RBF) [5, 46]很好地替代。我们发现, 当KAN被引入作为连续分类器时, 这种替代会给CIL带来更多优势, 如下文所示。带有RBF的KAN层表示为:

$$f(\mathbf{x}) = \sum_{p=1}^n \Phi_p \sum_{i=1}^N \omega_{p,i} \phi(\|\mathbf{x}_p - \mathbf{c}_i\|), \quad (5)$$

其中 $\mathbf{c}_i$  表示一系列均匀分布在特定范围内的中心点,  $N$  表示 $\mathbf{c}_i$  的总数。 $\phi(\cdot)$  是一个用作基函数的RBF, 其值仅取决于输入 $\mathbf{x}_p$  和中心点 $\mathbf{c}_i$  之间的距离

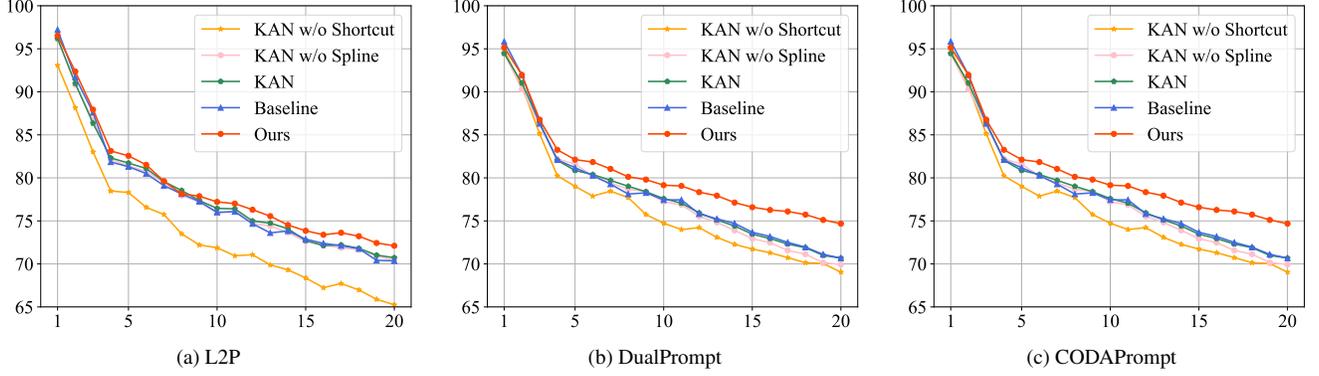


Figure 2. 在ImageNet-R 20步场景下，三种近期方法与不同分类器的准确率曲线比较。x轴表示任务数量的增加，y轴表示每一步对应的测试准确率。基线表示使用传统线性分类器的性能，其他曲线表示使用简化KAN分类器和我们的Kolmogorov-Arnold分类器的结果。

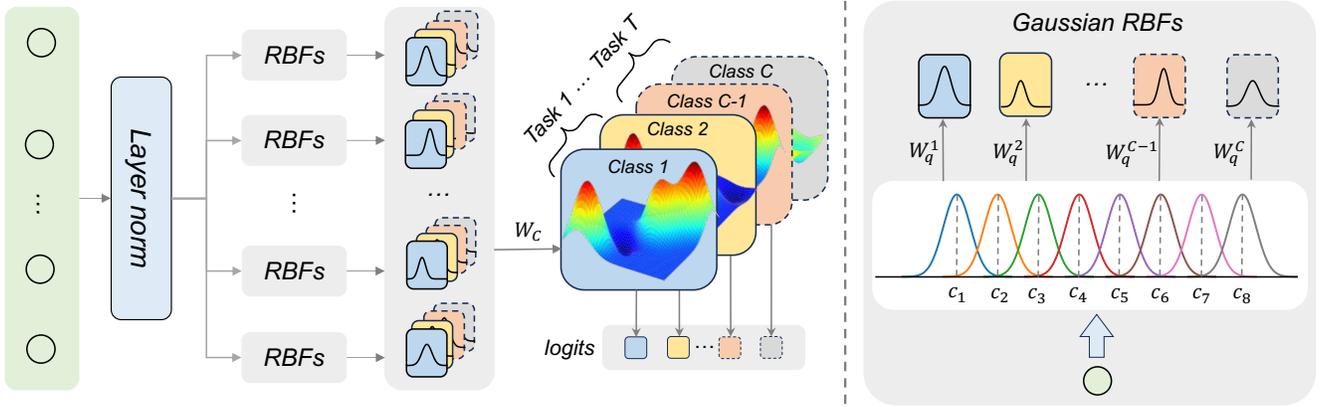


Figure 3. Kolmogorov-Arnold分类器流程概览。对于输入特征嵌入，我们首先使用层归一化对其进行归一化，然后将其传入一组径向基函数(RBF)，激活它们使其服从可学习的高斯分布。最后，我们用 $W_C$ 加权所有通道，以获得每个类别的决策空间。右侧展示了高斯RBF的处理过程，它将单变量映射到以不同点为中心的不同高斯分布，并用 $W_q^c$ 加权这些分布，从而得出所有类别中每个通道的最终激活分布。输出对数函数根据每个类别分布内的通道值进行采样。随着任务的增加，只需扩展 $W_C$ 即可容纳新的类别。

离。项 $\omega_{p,i}$ 表示每个 $\phi$ 的权重。可以选择具有协方差 $\sigma_i$ 的高斯函数作为 $\phi$ ，其定义为：

$$\phi(\|x_p - c_i\|) = \exp\left(-\frac{(x_p - c_i)^2}{2\sigma_i^2}\right). \quad (6)$$

引入高斯RBF函数作为KAN的基函数，虽然可以实现更快的评估速度和更高的性能，如[37]所示，但是同时也建立了固有的高斯结构，可以作为CIL场景的有效组合结构。以 $c = [c_1, c_2, \dots, c_N]$ 为中心，设置一系列高斯分布 $\mathcal{N}$ ，每一维的激活函数由 $N$ 个高斯分布组合而成，而每一维的分布可以表示为一个高斯混合模型：

$$\begin{aligned} \sum_{i=1}^N \omega_{p,i} \phi(\|x_p - c_i\|) &\sim \sum_{i=1}^N \omega_{p,i} \mathcal{N}(c_i, \sigma_i^2) \\ &= \omega_{p,1} \mathcal{N}(c_1, \sigma_1^2) + \omega_{p,2} \mathcal{N}(c_2, \sigma_2^2) \\ &\quad + \dots + \omega_{p,N} \mathcal{N}(c_N, \sigma_N^2) \end{aligned} \quad (7)$$

该混合模型保留了原始高斯分量的多模态特性。每个类的最终预测值可以表示为这些高斯混合模型的维度加权组合：

$$f(\mathbf{x}) = \sum_{p=1}^n \Phi_p \left[ \sum_{i=1}^N \omega_{p,i} \exp\left(-\frac{(x_p - c_i)^2}{2\sigma_i^2}\right) \right] \quad (8)$$

我们可以很容易地得出，由于高斯RBF函数的引入，KAN层中第 $p$ 维的特征在激活函数之后遵循高斯

混合分布。当我们简单地将 $\Phi_p$ 定义为每个维度的可学习权重时，显然所得的函数形式符合[11]中定义的具有一阶加性核的高斯过程（GP）。该结构始终易于拟合分类任务，并且具有强大的长程结构，可以有效解决近似高维函数时的COD问题 [11]。使用此类函数作为连续分类器的基函数，它不仅将特征的每个通道投影到高斯空间，还允许模型为每个通道选择针对不同类别的兴趣范围。

### 3.4. 面向CIL的Kolmogorov-Arnold分类器

上述分析表明，带有RBF的KAN层可以使CIL受益，这促使我们引入Kolmogorov-Arnold分类器(KAC)，它可以通过替换传统的线性分类器集成到任何CIL方法中。

图3展示了KAC的概览。KAC首先使用层正则化 $\mathcal{LN}$ 对特征分布进行正则化，得到正则化的嵌入 $\mathcal{LN}(F(\mathbf{x})) = [x'_1, x'_2, \dots, x'_n]$ 。之后，它加入一个KAN层，该层包含 $N$ 个以 $c = [c_1, c_2, \dots, c_N]$ 为中心的高斯径向基函数。基函数 $\phi$ 的定义如方程6所示，然后logit  $l$  计算如下：

$$l = \text{KAC}(F(\mathbf{x})) = \text{diag}\left(W_C \cdot \Phi\left(\mathcal{LN}(F(\mathbf{x}))\right) \cdot W_q\right), \quad (9)$$

其中 $\text{diag}(\cdot)$ 表示提取矩阵的对角线元素，而 $\Phi\left(\mathcal{LN}(F(\mathbf{x}))\right)$ 是可学习的高斯RBF，其计算方式如下：

$$\begin{pmatrix} \phi(\|x'_1 - c_1\|) & \phi(\|x'_1 - c_2\|) & \cdots & \phi(\|x'_1 - c_N\|) \\ \phi(\|x'_2 - c_1\|) & \phi(\|x'_2 - c_2\|) & \cdots & \phi(\|x'_2 - c_N\|) \\ \vdots & \vdots & \ddots & \vdots \\ \phi(\|x'_n - c_1\|) & \phi(\|x'_n - c_2\|) & \cdots & \phi(\|x'_n - c_N\|) \end{pmatrix}, \quad (10)$$

其中 $n$ 是输入嵌入的维数， $W_C \in \mathbb{R}^{C \times n}$ 是一个可学习的权重矩阵，作为输出线性函数来预测每个类别的概率，对应于传统KAN中的 $\Phi_p$ ； $W_q \in \mathbb{R}^{N \times C}$ 对应于传统KAN中的 $\phi_{p,q}$ ，作为每个类别每个通道的单变量可学习激活。实际上， $W_C$ 和 $W_q$ 可以合并为一个权重矩阵 $W \in \mathbb{R}^{C \times (N \times n)}$ ，然后使用基函数 $\phi$ 直接预测最终的logit。KAC表示为：

$$\text{KAC}(F(\mathbf{x})) = W \cdot \text{Reshape}\left(\Phi\left(\mathcal{LN}(F(\mathbf{x}))\right)\right). \quad (11)$$

$\text{reshape}(\cdot)$ 函数将 $N \times n$ 矩阵展平为一维向量，以方便与 $W$ 进行计算。

在CIL场景中， $T$ 个任务按顺序到达，类别计数为 $[C_1, C_2, \dots, C_T]$ 。KAC会扩展 $W$ 以容纳新的类别，类似于传统的分类器[52]。在第 $t$ 步，存在一个旧的分类型矩阵 $W^{t-1} \in \mathbb{R}^{(N \times n) \times C_{old}}$ ，其中 $C_{old} = C_1 + C_2 + \dots + C_{t-1}$ ，以及一个新的矩阵 $W^t \in \mathbb{R}^{(N \times n) \times C_t}$ ，第 $t$ 步之后的最终 $W$ 是这两个矩阵的串联。

## 4. 实验

### 4.1. Benchmarks & Implementations

**Benchmarks.** 我们评估了CIL场景，并进一步验证了我们方法在领域增量学习(DIL) [56]中的鲁棒性。对于CIL，我们在两个常用数据集ImageNet-R [24]和CUB200 [53]上进行了实验，每个数据集包含200个类别。从0个基类开始，所有类别被分为5、10、20和40个步骤，并按顺序输入模型进行训练。对于DIL，我们遵循Sprompt [56]的方法，将DomainNet [48]数据集划分为6个领域，在所有任务中共分类出345个类别。所有实验均在非样本环境下进行，没有用于新训练的保存旧样本。不同种子样本的实验结果在补充材料中展示。

**实现细节** 所有实验均基于ViT-B/16主干网络进行。RBF函数个数设为4，中心 $[c_1, c_2, \dots, c_N]$ 均匀分布在-2至2之间，高斯函数中的 $\sigma$ 设为1，以便对范围进行平均划分。为了验证KAC的有效性，我们选择了四种基于提示的CIL方法L2P [58]、DualPrompt [57]、CODAPrompt [52]和CPrompt [16]作为基准，这些方法在各种CIL基准测试中均取得了优异的性能。这些方法利用可学习的提示从预训练的主干网络中提取信息，并使用线性分类器对提取出的嵌入进行分类。我们直接用KAC替换这些方法的分类器，并使用其原始超参数来训练模型，以便比较不同分类器之间的差异。我们实现了所有比较方法，并保留了它们的官方代码和原始选定的超参数。对于所有实验，我们给出平均增量准确度（所有任务的平均准确度）和最后一个任务的准确度（学习最终任务后的总体准确度）。

### 4.2. 实验结果

在ImageNet-R上的实验。Tab. 1比较了ImageNet-R基准测试中基线方法和采用KAC的准确率。用KAC替换线性分类器后，所有方法的准确率均有所提升，尤其是在具有挑战性的长序列场景中，大多数情况下准确率提升3到5个百分点。这表明KAC能够有效帮助模型在每一步中缓解遗忘。此外，通过比较CODAPrompt和CPrompt，我们发现，虽然两者在使用线性分类器时表现相似，但是换成KAC后CODAPrompt明显超过了CPrompt。这表明KAC与不同方法的兼容性有所不同。

在CUB200上的实验。表2展示了CUB200设置下各项指标的比较，在长序列场景中观察到了令人惊讶的提升，提升幅度达到10%到25%。由于CUB200是一个细粒度的鸟类分类数据集，我们相信KAC在该数据集上的表现将非常出色。

在DomainNet上的实验。我们在DomainNet上进行了领域增量学习实验，旨在验证KAC扩展到其他连续分类任务的能力。如表3所示，当所有方法都使用KAC实现时，平均增量准确率提升了约1%，最终准确率提升

Table 1. ImageNet-R 数据集上的结果。我们报告了在5、10、20 和40 步CIL 场景下的平均增量准确率和最终准确率，并与各种方法进行了比较，并使用线性分类器（基线）和我们的KAC 评估了结果。结果表明，我们的KAC 持续提升了性能，尤其是在长序列任务中。准确率旁边显示了变化情况，蓝色表示下降，红色表示提升。

Method	5 steps		10 steps		20 steps		40 steps	
	Avg	Last	Avg	Last	Avg	Last	Avg	Last
L2P	78.42	73.57	79.58	73.10	77.93	70.35	74.28	66.02
w/ KAC	77.98 (-0.44)	73.56 (-0.01)	79.22 (-0.36)	73.14 (+0.04)	78.94 (+1.01)	72.11 (+1.76)	76.34 (+2.06)	69.74 (+3.72)
DualPrompt	79.75	74.57	79.50	72.48	78.35	70.68	74.51	66.31
w/ KAC	79.96 (+0.21)	76.37 (+1.80)	80.72 (+1.22)	75.67 (+3.19)	80.40 (+2.05)	74.68 (+4.00)	76.87 (+2.36)	71.24 (+4.93)
CODAPrompt	82.27	77.62	82.49	77.01	80.92	74.40	76.80	69.34
w/ KAC	83.75 (+1.48)	80.14 (+2.52)	84.43 (+1.94)	79.24 (+2.23)	83.59 (+2.67)	77.94 (+3.54)	79.79 (+2.99)	74.31 (+4.97)
CPrompt	84.07	78.68	83.13	76.80	81.83	74.32	78.98	70.07
w/ KAC	84.51 (+0.44)	79.08 (+0.40)	83.97 (+0.84)	78.07 (+1.27)	82.56 (+0.73)	75.73 (+1.41)	80.89 (+1.91)	72.05 (+1.98)

Table 2. CUB200 数据集上的结果。报告了平均增量准确率和最终准确率。KAC 在所有基线上都取得了显著的提升，尤其是在长序列任务中，凸显了其在细粒度数据集上的卓越性能。

Method	5 steps		10 steps		20 steps		40 steps	
	Avg	Last	Avg	Last	Avg	Last	Avg	Last
L2P	80.05	76.04	74.02	65.28	63.31	51.78	46.84	35.41
w/ KAC	84.42 (+4.37)	83.80 (+7.76)	81.54 (+7.52)	79.77 (+14.49)	73.70 (+10.39)	70.13 (+18.35)	66.08 (+19.24)	60.43 (+25.02)
DualPrompt	81.84	76.38	75.10	64.60	66.89	54.68	50.61	37.55
w/ KAC	86.20 (+4.36)	85.03 (+8.65)	82.18 (+7.08)	79.61 (+14.01)	76.93 (+10.04)	71.91 (+17.23)	71.31 (+20.70)	64.69 (+27.14)
CODAPrompt	83.09	78.73	79.30	71.87	69.49	58.00	52.57	37.81
w/ KAC	86.56 (+3.47)	85.61 (+6.88)	85.04 (+5.74)	82.59 (+10.72)	77.23 (+7.74)	73.32 (+15.32)	71.36 (+18.79)	64.56 (+26.75)
CPrompt	88.62	82.02	85.77	76.80	83.97	72.99	77.34	64.80
w/ KAC	89.60 (+0.98)	83.08 (+1.06)	89.04 (+3.27)	80.75 (+3.95)	87.06 (+3.09)	78.54 (+5.55)	85.11 (+7.77)	76.51 (+11.71)

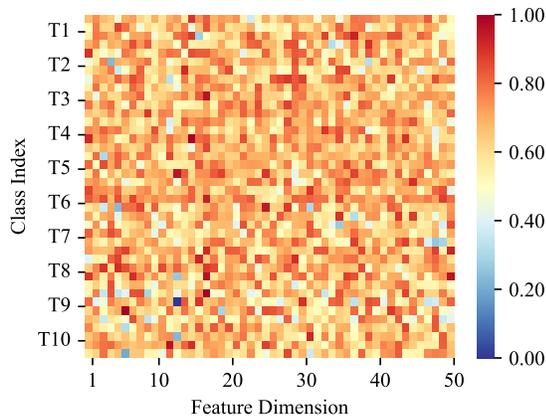


Figure 4. 不同类别在不同通道上的激活图。x 轴表示从特征嵌入中随机选择的50 个通道，y 轴表示来自不同任务的类别。颜色表示不同的关注程度。

了约0.5%，这证明了KAC 的稳定性。

激活图的可视化 图 4 展示了不同类别如何激活不同

的通道，以及不同类别在不同通道上的激活差异。每个类别仅激活一部分通道，并且更新仅应用于这些通道，从而避免对其他通道产生任何影响，并凸显了局部性在缓解灾难性遗忘方面的优势。

### 4.3. 消融研究与分析

基函数数量的消融 基函数数量 $N$  是KAC 的一个关键超参数。过多的基函数数量可能会导致额外的计算，并导致 $W$  的维数过高。相反，较小的 $N$  值可能会损害KAC 的近似能力。为了探索合适的 $N$  值，我们对其进行了消融研究。图 5 显示了在ImageNet-R 上进行20 步实验时，四种使用不同基函数数量的KAC 方法的平均增量准确率。结果表明，单纯增加基函数数量并不能有效缓解遗忘，并进一步证明了性能提升并非源于增加嵌入空间的维数。大多数方法在 $N = 4$  或 $N = 8$  时表现出更好的性能，因此我们在实验中将 $N$  设置为4。

KAN 结构起着关键作用 为了证明KAC 的优势在于引入的KAN 结构，而非额外的计算量，我们用MLP 层替换了RBF 结构，并将其输出维度设置为类数，隐藏维

Table 3. DomainNet 上的结果。我们在6个增量域（共345个类别）上进行了域增量学习实验。我们报告了平均增量准确率和最后一个任务的准确率。结果表明，KAC在DIL设置下也能正常工作。

Method	Avg	Last
L2P	57.78	49.22
w/ KAC	59.79 (+2.01)	51.10 (+1.88)
DualPrompt	60.96	51.83
w/ KAC	62.06 (+1.10)	52.76 (+0.93)
CODAPrompt	61.61	53.12
w/ KAC	62.78 (+1.17)	53.54 (+0.42)
CPrompt	61.32	52.49
w/ KAC	62.13 (+0.81)	53.02 (+0.53)

Table 4. 对分类器结构进行消融研究。我们用多层感知器(MLP)替换KAC中的样条函数，以验证KAN结构的有效性。其中，w/ MLP表示与模型一起训练的MLP，而w/ MLP (fixed)表示随机初始化的MLP投影，不进行任何更新。实验在20步ImageNet-R场景中进行。

	CODAPrompt	w/ KAC	w/ MLP	w/ MLP (fixed)
Avg	80.92	83.59	80.56	65.87
Last	74.40	77.94	73.59	51.03

度设置为 $N \times n$ ，以使参数数量与使用RBF的KAC保持一致，从而使我们能够对两种结构进行公平的比较。表4展示了在CODAPrompt上实现的将RBF替换为传统线性分类器并添加MLP结构的性能。比较后我们发现，无论附加的MLP结构是否随模型更新，都不会产生任何积极的影响。这表明KAC的优势源于其KAN结构，而非简单地增加分类空间的维数。

复杂度分析与传统的线性分类器相比，我们的KAC在分类器层引入的计算成本和参数数量几乎可以忽略不计。KAC在每个维度上应用固定的高斯激活函数，几乎不会引入额外的计算。对于嵌入维度为768的ViT网络来说，要对100个类别进行分类，KAC引入的额外参数仅为0.23M，与主干网络的86M参数相比可以忽略不计。

## 5. 总结

本文探讨了Kolmogorov-Arnold网络(KAN)在持续学习中的应用，并提出了一种新型持续学习器——Kolmogorov-Arnold分类器(KAC)。该分类器利用KAN固有的局部性能力，有效缓解学习新任务过程中的特征偏移。分析表明，KAN中B样条函数的逼近能力有限，当应用于高维数据时，会迫使模型主干引入更多偏移以适应新类别，从而显著降低持续学习的性能。这加剧了模型的遗忘效应，掩盖了局部性能力相对于传统线性分类器的优势。为了解决这个

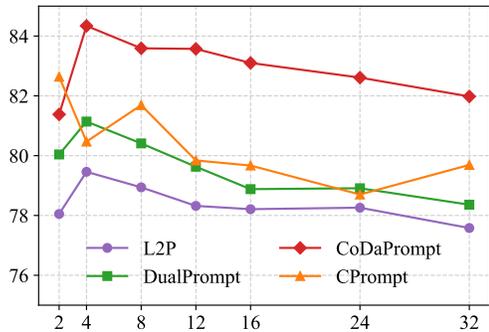


Figure 5. 在20步ImageNet-R场景中对不同基函数数量进行消融研究。x轴表示基函数数量，y轴表示不同数量下的平均增量准确率。

问题，我们将KAN中的B样条函数替换为径向基函数(RBF)，从而提升了性能。KAC在各种持续学习场景中展现出显著优势，彰显了其有效性和稳健性。未来，我们计划探索KAN在持续学习中的更多可能性，充分利用其固有优势。

## 6. 致谢

本研究由国家自然科学基金（编号：62206135, 62225604）、中国科学院青年拔尖人才支持计划（编号：2023QNRC001）、“科技甬江2035”重大科技突破计划项目（编号：2024Z120）、深圳市科技计划（编号：JCYJ20240813114237048）以及中央高校基本科研业务费专项资金（南开大学，编号：070-63233085）资助。计算由南开大学超算中心提供支持。

## References

- [1] Alireza Afzal Aghaei. fkan: Fractional kolmogorov-arnold networks with trainable jacobi basis functions. *arXiv preprint arXiv:2406.07456*, 2024. 2
- [2] Eden Belouadah, Adrian Popescu, and Ioannis Kanellos. A comprehensive study of class incremental learning algorithms for visual tasks. *Neural Networks*, 135:38–54, 2021. 1
- [3] Zavareh Bozorgasl and Hao Chen. Wav-kan: Wavelet kolmogorov-arnold networks. *arXiv preprint arXiv:2405.12832*, 2024. 2
- [4] Roman Bresson, Giannis Nikolentzos, George Panagopoulos, Michail Chatzianastasis, Jun Pang, and Michalis Vazirgiannis. Kagnns: Kolmogorov-arnold networks meet graph learning. *arXiv preprint arXiv:2406.18380*, 2024. 2
- [5] Martin Dietrich Buhmann. Radial basis functions. *Acta numerica*, 9:1–38, 2000. 3
- [6] Xiuwei Chen and Xiaobin Chang. Dynamic residual classifier for class incremental learning. In *ICCV*, pages 18743–18752, 2023. 1, 2

- [7] Gianluca De Carlo, Andrea Mastropietro, and Aris Anagnostopoulos. Kolmogorov-arnold graph neural networks. *arXiv preprint arXiv:2406.18354*, 2024. 2
- [8] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022. 1
- [9] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *ECCV*, pages 86–102. Springer, 2020. 1, 2
- [10] Arthur Douillard, Alexandre Ramé, Guillaume Couairon, and Matthieu Cord. Dytox: Transformers for continual learning with dynamic token expansion. In *CVPR*, pages 9285–9295, 2022. 1, 2
- [11] David K Duvenaud, Hannes Nickisch, and Carl Rasmussen. Additive gaussian processes. *Advances in neural information processing systems*, 24, 2011. 5
- [12] Robert M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999. 1, 2
- [13] Qiankun Gao, Chen Zhao, Yifan Sun, Teng Xi, Gang Zhang, Bernard Ghanem, and Jian Zhang. A unified continual learning framework with general parameter-efficient tuning. In *CVPR*, pages 11483–11493, 2023. 2
- [14] Rui Gao and Weiwei Liu. Ddgr: Continual learning with deep diffusion-based generative replay. pages 10744–10763. PMLR, 2023. 2
- [15] Zhanxin Gao, Jun Cen, and Xiaobin Chang. Consistent prompting for rehearsal-free continual learning. In *CVPR*, pages 28463–28473, 2024. 2
- [16] Zhanxin Gao, Jun Cen, and Xiaobin Chang. Consistent prompting for rehearsal-free continual learning. In *CVPR*, pages 28463–28473, 2024. 1, 5
- [17] Remi Genet and Hugo Inzirillo. Tkan: Temporal kolmogorov-arnold networks. *arXiv preprint arXiv:2405.07344*, 2024. 2
- [18] Lukasz Golab and M. Tamer Özsu. Issues in data stream management. *SIGMOD Rec.*, 32(2):5–14, 2003. 1
- [19] Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost van de Weijer. Fecam: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. *Advances in Neural Information Processing Systems*, 36, 2024. 1
- [20] PC Hammer. Adaptive control processes: a guided tour (r. bellman), 1962. 3
- [21] Simon Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998. 3
- [22] Juncai He. On the optimal expressive power of relu dnns and its application in approximation with kolmogorov superposition theorem. *arXiv preprint arXiv:2308.05509*, 2023. 3
- [23] Juncai He and Jinchao Xu. Deep neural networks and finite elements of any order on arbitrary dimensions. *arXiv preprint arXiv:2312.14276*, 2023. 3
- [24] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8340–8349, 2021. 5
- [25] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989. 2, 3
- [26] Amanda A Howard, Bruno Jacob, Sarah H Murphy, Alexander Heinlein, and Panos Stinis. Finite basis kolmogorov-arnold networks: domain decomposition for data-driven and physics-informed problems. *arXiv preprint arXiv:2406.19662*, 2024. 2
- [27] Yusong Hu, Zichen Liang, Xialei Liu, Qibin Hou, and Ming-Ming Cheng. Reformulating classification as image-class matching for class incremental learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(1):811–822, 2025. 2
- [28] Kishaan Jeeveswaran, Prashant Shivaram Bhat, Bahram Zonooz, and Elahe Arani. BiRT: Bio-inspired replay in vision transformers for continual learning. pages 14817–14835, 2023. 2
- [29] Minsoo Kang, Jaeyoo Park, and Bohyung Han. Class-incremental learning by knowledge distillation with adaptive feature consolidation. In *CVPR*, pages 16071–16080, 2022. 2
- [30] Junsu Kim, Hoseong Cho, Jihyeon Kim, Yihalem Yimolal Tiruneh, and Seungryul Baek. Sddgr: Stable diffusion-based deep generative replay for class incremental object detection. In *CVPR*, pages 28772–28781, 2024. 2
- [31] Taehoon Kim, Jaeyoo Park, and Bohyung Han. Cross-class feature augmentation for class incremental learning. In *AAAI*, pages 13168–13176, 2024. 1, 3
- [32] 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. 2
- [33] Benjamin C Koenig, Suyong Kim, and Sili Deng. Kan-odes: Kolmogorov–arnold network ordinary differential equations for learning dynamical systems and hidden physics. *Computer Methods in Applied Mechanics and Engineering*, 432: 117397, 2024. 2
- [34] Andrei Nikolaevich Kolmogorov. *On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables*. American Mathematical Society, 1961. 2, 3
- [35] Mario Köppen. On the training of a kolmogorov network. In *Artificial Neural Networks—ICANN 2002: International Conference Madrid, Spain, August 28–30, 2002 Proceedings 12*, pages 474–479. Springer, 2002. 3
- [36] Ming-Jun Lai and Zhaiming Shen. The kolmogorov superposition theorem can break the curse of dimensionality when approximating high dimensional functions. *arXiv preprint arXiv:2112.09963*, 2021. 3

- [37] Ziyao Li. Kolmogorov-arnold networks are radial basis function networks. *arXiv preprint arXiv:2405.06721*, 2024. 2, 3, 4
- [38] Zhizhong Li and Derek Hoiem. Learning without forgetting. *PAMI*, 40(12):2935–2947, 2017. 3
- [39] Henry W Lin, Max Tegmark, and David Rolnick. Why does deep and cheap learning work so well? *Journal of Statistical Physics*, 168:1223–1247, 2017. 3
- [40] Xialei Liu, Yu-Song Hu, Xu-Sheng Cao, Andrew D Bagdanov, Ke Li, and Ming-Ming Cheng. Long-tailed class incremental learning. In *ECCV*, pages 495–512. Springer, 2022. 2
- [41] Ziming Liu, Yixuan Wang, Sachin Vaidya, Fabian Ruele, James Halverson, Marin Soljačić, Thomas Y Hou, and Max Tegmark. Kan: Kolmogorov-arnold networks. *arXiv preprint arXiv:2404.19756*, 2024. 1, 2, 3
- [42] R. O. Malashin and M. A. Mikhalkova. Avoiding catastrophic forgetting via neuronal decay. In *2024 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*, pages 1–6, 2024. 2
- [43] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, pages 109–165. Elsevier, 1989. 2
- [44] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. 36, 2024. 1, 2, 3
- [45] Hadrien Montanelli and Haizhao Yang. Error bounds for deep relu networks using the kolmogorov-arnold superposition theorem. *Neural Networks*, 129:1–6, 2020. 3
- [46] Mark JL Orr et al. Introduction to radial basis function networks, 1996. 3
- [47] German I. Parisi, Ronald Kemker, Jose L. Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural Networks*, 113:54–71, 2019. 1
- [48] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1406–1415, 2019. 5
- [49] Kaihuai Qin. General matrix representations for b-splines. In *Proceedings Pacific Graphics’ 98. Sixth Pacific Conference on Computer Graphics and Applications (Cat. No. 98EX208)*, pages 37–43. IEEE, 1998. 3
- [50] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *CVPR*, pages 2001–2010, 2017. 1, 2
- [51] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *NeuIPS*, 30, 2017. 2
- [52] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *CVPR*, pages 11909–11919, 2023. 1, 2, 5
- [53] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 5
- [54] Fu-Yun Wang, Da-Wei Zhou, Liu Liu, Han-Jia Ye, Yatao Bian, De-Chuan Zhan, and Peilin Zhao. Beef: Bi-compatible class-incremental learning via energy-based expansion and fusion. In *ICLR*, 2022. 2
- [55] Liyuan Wang, Jingyi Xie, Xingxing Zhang, Mingyi Huang, Hang Su, and Jun Zhu. Hierarchical decomposition of prompt-based continual learning: Rethinking obscured sub-optimality. 36, 2024. 2
- [56] Yabin Wang, Zhiwu Huang, and Xiaopeng Hong. S-prompts learning with pre-trained transformers: An occam’s razor for domain incremental learning. 35:5682–5695, 2022. 2, 5
- [57] 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 *ECCV*, pages 631–648. Springer, 2022. 1, 2, 5
- [58] 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 *CVPR*, pages 139–149, 2022. 1, 2, 5
- [59] Haitao Wen, Lili Pan, Yu Dai, Heqian Qiu, Lanxiao Wang, Qingbo Wu, and Hongliang Li. Class incremental learning with multi-teacher distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 28443–28452, 2024. 1, 2, 3
- [60] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *CVPR*, pages 3014–3023, 2021. 1
- [61] Guanglei Yang, Enrico Fini, Dan Xu, Paolo Rota, Mingli Ding, Moin Nabi, Xavier Alameda-Pineda, and Elisa Ricci. Uncertainty-aware contrastive distillation for incremental semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2567–2581, 2022. 2, 3
- [62] Lu Yu, Bartłomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic drift compensation for class-incremental learning. In *CVPR*, 2020. 1, 3
- [63] Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. Slca: Slow learner with classifier alignment for continual learning on a pre-trained model. In *CVPR*, pages 19148–19158, 2023. 1, 2
- [64] Da-Wei Zhou, Hai-Long Sun, Han-Jia Ye, and De-Chuan Zhan. Expandable subspace ensemble for pre-trained model-based class-incremental learning. In *CVPR*, pages 23554–23564, 2024. 1, 2
- [65] Huiping Zhuang, Zhenyu Weng, Run He, Zhiping Lin, and Ziqian Zeng. Gkeal: Gaussian kernel embedded analytic learning for few-shot class incremental task. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7746–7755, 2023. 3

# KAC:用于持续学习的Kolmogorov-Arnold分类器

## Supplementary Material

### 7. 更多实验结果

在CIFAR-100 上的实验。 Tab. 5 比较了CIFAR-100 数据集上基线方法与采用KAC 方法的平均增量准确率。用KAC 替换线性分类器可以提升大多数方法的准确率，但CPrompt 和L2P 的准确率略有下降。由于CIFAR-100 的像素分辨率较低，它通常适合训练较小规模的网络。对于预训练的主干网络，性能趋于饱和，因此我们的方法在该数据集上没有表现出显著的提升。

Table 5. CIFAR-100 10 步场景的平均增量准确度。

Method	Linear	KAC
L2P	83.78	83.71 (-0.07)
DualPrompt	84.80	85.74 (+0.94)
CODAPrompt	86.65	87.26 (+0.61)
CPrompt	87.50	87.19 (-0.31)

Table 6. 在ImageNet-R 20 步场景中，KAC 与残差连接的KAC 的最终准确率比较。

Method	Baseline	KAC + Shortcut	KAC
L2P	70.35	71.09	72.11
DualPrompt	70.68	72.83	74.68
CODAPrompt	74.40	75.57	77.94
CPrompt	74.32	73.55	75.73

在CUB200 中多次实验的结果。为了进一步评估KAC 的鲁棒性，我们使用多个种子生成随机类别序列，并评估使用KAC 和传统线性分类器的各种方法的性能。具体而言，我们报告了CUB200 上每种设置的最终准确率。表 7 报告了这些指标的平均值和标准差。结果表明，KAC 在大多数实验中始终优于基线方法。

### 8. 更多的消融实验

在线性残差连接上的消融 在KAC 中，我们没有遵循传统的KAN 方法，即在样条函数中添加线性残差连接。在本节中，我们将说明线性残差连接并不能帮助KAC 获得更好的性能。表 6 报告了ImageNet-R 20 步场景中最后一个任务的准确率。结果表明，添加线性残差连接后，准确率反而下降，这致使我们移除线性残差连接。

### 9. 更多的可视化

CUB200 性能的可视化 为了探究KAC在CUB200上性能优异的原因，我们绘制并观察了CUB200在不同steps 的实验当中的精度曲线。如图 6 所示，随着任务数量的增加，KAC 的优势不断增强。在某些步骤中，基准模型经常出现显著的遗忘，而KAC 在这些步骤中通常表现出比线性分类器更少的遗忘，这有助于KAC 积累更高的最终准确率。

Table 7. 使用多个种子在CUB200数据集上进行多次运行的结果。报告了最后一次准确率的平均值和标准差。

Method	5 steps		10 steps		20 steps		40 steps	
	mean	std	mean	std	mean	std	mean	std
L2P	76.60	0.79	69.23	2.86	59.11	5.06	41.65	4.71
<i>w</i> KAC	80.35	2.60	77.07	1.92	73.32	0.57	65.89	4.09
DualPrompt	77.35	1.68	71.13	1.88	61.91	5.03	44.98	5.21
<i>w</i> KAC	82.39	1.95	79.55	1.88	75.94	1.76	69.86	4.28
CODAPrompt	75.62	2.65	70.77	0.71	62.58	3.44	45.14	5.49
<i>w</i> KAC	82.88	1.96	78.74	2.73	74.74	0.194	70.57	4.27

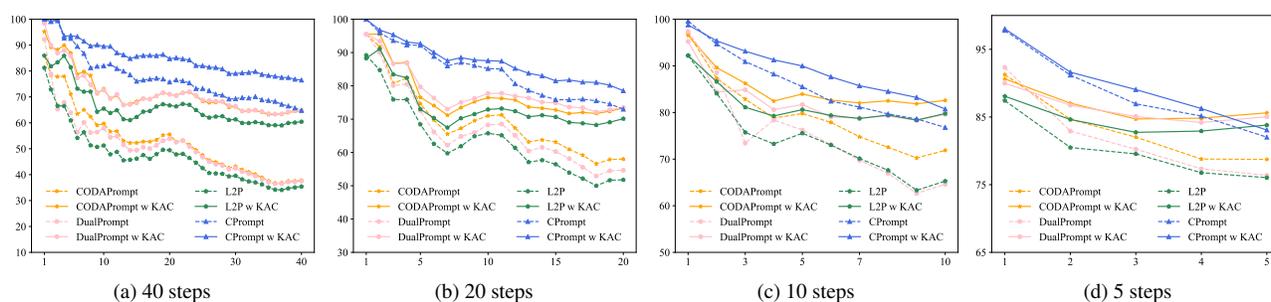


Figure 6. CUB200 数据集上不同步骤场景的准确率曲线。x 轴表示逐渐增加的任务，y 轴表示每一步的准确率。可以观察到，KAC 的趋势与基线相同，但每一步的遗忘程度有所降低。