面向云边个性化模型解耦的聚类联邦学习方法

杜 甜[1)] 陈星延[1),3)] 寇 纲[1)] 赵 宇[1),3)] 许长桥[2)]

[1)](西南财经大学金融智能与金融工程四川省重点实验室 成都 611130)

[2)](北京邮电大学网络与交换技术全国重点实验室 北京 100876)

[3)](西南财经大学计算机与人工智能学院 成都 611130)

摘 要 联邦学习是一种前沿的分布式学习范式,该技术允许多个边缘客户端协作训练全局模型,而无需客户端共享数据至中央服务器,有效缓解了深度模型训练的"数据孤岛"和数据隐私安全问题。生成式人工智能的发展推动了更大规模的模型和数据集应用,加剧了联邦学习所面临的现实挑战,特别是由客户端本地数据高度异构导致的训练效率低和通信成本高等问题。本文提出了一种基于云边模型解耦的联邦学习创新框架,FedCPMD,该框架通过为客户端动态选择最优个性化层来应对数据异质性。动机实验表明,具有异构数据分布的客户端在选用不同神经网络层作为个性化层时,其性能存在明显差异。基于此,本文设计了一种逐层知识化表征方法,通过独立量化每一层神经网络对最终模型效果的影响,实现对个性化层的选择。FedCPMD还引入了一种基于知识表征的客户端聚类策略,通过将具有相同选层结果的客户端聚类到同一集群,来提升异构联邦学习的模型性能。本文在九个真实数据集上开展实验,结果表明与现有十余种先进方案相比,FedCPMD具有明显优势。针对CIFAR100、CINIC10、SVHN和Tiny ImageNet等复杂数据集,FedCPMD在LeNet5架构上的准确率平均提升2.450%($\alpha=0.1$),在VGG11架构上平均提升3.963%($\alpha=0.1$)。

关键词 聚类;个性化联邦学习;模型解耦;云边系统;绿色通信

中图法分类号 TP18 **DOI号** 10.11897/SP.J.1016.2025.00407

Clustered Federated Learning with Cloud-Edge Personalized Model Decoupling

DU Tian[1)] CHEN Xing-Yan[1),3)] KOU Gang[1)] ZHAO Yu[1),3)] XU Chang-Qiao[2)]

[1)](Financial Intelligence and Financial Engineering Key Laboratory of Sichuan Province, Southwestern University of Finance and Economics, Chengdu 611130)

[2)](National Key Laboratory of Network and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876)

[3)](School of Computer Science and Artificial Intelligence, Southwestern University of Finance and Economics, Chengdu 611130)

Abstract Federated Learning (FL) is an advanced distributed learning paradigm that enables multiple edge clients to collaboratively train a global model without sharing their data with a central server, effectively addressing the challenges of data silos and data privacy concerns in deep learning. With the rapid development of generative artificial intelligence, the scale laws have driven the

收稿日期:2024-06-24;在线发布日期:2024-12-10。本课题得到国家自然科学基金青年基金项目(62302400)、国家自然科学基金面上项目(62376227)、国家杰出青年科学基金项目(62225105)、四川省自然科学基金青年基金项目(2023NSFSC0114)、四川省自然科学基金重点项目(2023NSFSC5094)、中央高校基本科研业务费项目(JBK2406080)资助。 **杜 甜**,博士研究生,主要研究领域为异构联邦学习、时间序列聚类。E-mail:dtian.nora@gmail.com。**陈星延**(通信作者),博士,副教授,硕士生导师,主要研究领域为算网融合、分布式学习。E-mail:xychen@swufe.edu.cn。**寇 纲**(通信作者),博士,长江学者特聘教授,博士生导师,主要研究领域为大数据与金融智能、数据科学与智能决策、商务智能、信息系统。E-mail:kougang@swufe.edu.cn。**赵 宇**,博士,教授,博士生导师,主要研究领域为数据挖掘、自然语言处理、图形学习和机器学习。**许长桥**,博士,教授,博士生导师,主要研究领域为未来互联网技术、多媒体通信、网络安全和人工智能。

application of increasingly larger models and datasets, which exacerbate the FL challenges, particularly those arising from the high heterogeneity of local data across clients, leading to reduced training efficiency and increased communication costs. While existing solutions, such as knowledge distillation, loss function design and similarity aggregation, have alleviated these challenges to some extent, they each have inherent limitations. For instance, knowledge distillation often relies on auxiliary public datasets, the introduction of additional loss functions increases computational overhead, and similarity aggregation based on client features may pose a risk to privacy.

In this paper, we identify two key findings from motivating experiments: (1) Clients with different data distributions achieve optimal performance with distinct personalized layers, suggesting that a uniform personalized layer strategy for all edge clients may not be effective. (2) Using these optimal personalized layers as a basis for clients clustering result in several well-balanced clusters, indicating that appropriate model decoupling can aid the central server in identifying similar edge clients. Building on these insights, we propose a knowledge representation-based approach for personalized layer selection, which accurately measures the alignment between neural network layers and the heterogeneous data distributions of edge clients. We further introduce a FL framework, FedCPMD, which dynamically selects the most suitable layer for each client based on data distribution, and clusters edge clients accordingly. Within each cluster, clients perform global aggregation and parameter updates, enhancing both the efficiency and performance of the model. We conduct experiments on nine real-world datasets, and the results demonstrate that FedCPMD significantly outperforms a series of existing state-of-the-art approaches in model performance. On complex datasets such as CIFAR100, CINIC10, SVHN, and Tiny ImageNet, FedCPMD achieves an average accuracy improvement of 2.450%on the LeNet5 and 3.963% ($\alpha=0.1$) on the VGG11 compared to the baseline methods.

In conclusion, the main contributions of this paper are as follows: (1) Motivating experiments were conducted on six real-world datasets, leading to two key findings and the proposal of a layer-wise knowledge representation method for multi-layer neural network models. This approach quantifies the independent impact of each neural network layer on the final model performance, thereby optimizing the personalized layer selection strategy. (2) A FL framework based on model decoupling, named FedCPMD, is introduced. This framework dynamically clusters clients based on the personalized layer selection results, without the need to predefine the number of clusters. The performance of FedCPMD is further discussed under various distribution distance metrics, validating its generalizability and effectiveness. (3) Experimental results demonstrate the advantages of the proposed method in model training. Based on tests on nine real-world datasets, FedCPMD achieves an average accuracy improvement of approximately 4% compared to over ten state-of-the-art methods on complex classification datasets, including CIFAR100, CINIC10 and Tiny ImageNet.

Keywords clustering; personalized federated learning; model decoupling; cloud-edge system; green communication

1 引 言

近年来，以GPT-4和SORA等为代表的生成式人工智能取得了重大突破，人类社会开启了探索通用人工智能的新纪元。生成式模型的训练通常依赖超大规模的现实数据，并在集中式云数据中心进行。然而，中心化的训练方式无法避免数据流通和共享使用，导致训练过程中存在严重的隐私泄露风险，例如著名的Facebook用户数据泄露事件①和普渡大学学生信息泄露事件②。联邦学习[1]是一种分

①腾讯网．“5.53亿Facebook用户个人信息被泄露”，2021-04-04.

② CYWARE SOCIAL. “Purdue data leak: Personal information of over 26000 prospective students exposed”, 2018-07-13.

布式模型训练框架，允许多个边缘客户端在不直接交换原始数据的情况下协同训练共享模型。客户端仅需利用自身数据训练本地模型，将更新后的参数或梯度上传至中央服务器聚合，便能有效降低训练过程中数据隐私的泄露风险。

以FedAvg[1]为代表的传统联邦学习方法在边缘客户端数据呈现独立同分布（Independent and Identically Distributed，IID）时性能良好。但现实场景中边缘客户端数据往往具有非独立同分布（Non-Independent and Identically Distributed，Non-IID）特征，即不同客户端存在数据类别不平衡、样本数量不均匀等问题。上述数据分布异质性问题将显著影响模型的训练效率。另外，当前基于Transformer架构的生成式模型[2]通常具有超大规模参数量，这导致模型训练的时间开销和单次模型参数聚合的固有通信成本愈发高昂。边缘客户端的数据异质性和海量模型参数已成为制约联邦学习模型训练效率提升的关键因素。

为解决上述挑战，研究者们基于不同研究路径开拓出许多有效解决方案，包括基于知识蒸馏[3-5]、损失正则化[6-8]、相似性聚合[9-11]以及模型解耦[12-14]等方法。其中，模型解耦作为一种具有潜力的新兴策略在近些年获得了学者的广泛关注，该技术将神经网络模型分割为两部分，即用于提取共享特征的主体层和用于处理数据异质性的个性化层。如图1所示，目前大部分研究工作[12-15]将神经网络模型的前$(n-1)$层作为提取数据共享特征表示的主体（Body），而将神经网络模型的最后一层作为个性化头（Head）。主体层参数会上传中央服务器进行全局聚合，而个性化头参数则仅在边缘客户端本地更新，无须参数共享。基于模型解耦的联邦框架不仅能够减少聚合过程需要上传的模型参数量，降低模型训练的通信开销，还能有效避免数据异质性对模型更新造成的性能损害[14]，有助于模型在异构条件下提高训练效率和性能表现。

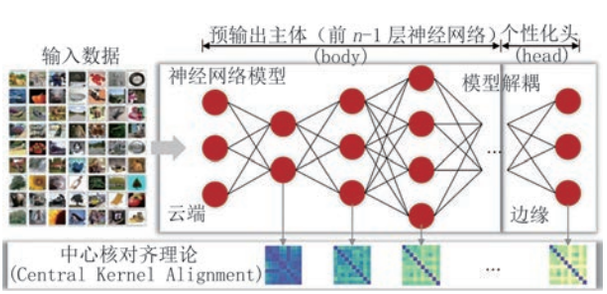

图1　基于CKA的模型解耦原理示意图

由于被解耦的神经网络具有黑盒特性，其内部结构和计算过程与输入输出的关系难以解释，模型解耦方式依然是亟待研究的重要问题。除了基于中心核对齐理论（Central Kernel Alignment，CKA）的Body-Head解耦方法，研究者们仍在探索更多灵活的模型分解方案，例如选择最后单层[13-15]、最后多层[16-17]或特定层[18]作为个性化头的解耦方案。虽然上述方法在性能上取得了一定提升，但本文认为模型解耦依然缺乏一套标准化分析方法。为探究其中规律，本文在六个真实数据集上对两种不同神经网络架构进行动机实验，分析得出了两个关键发现：（1）对于具有不同数据分布的客户端，它们取得最佳性能的个性化层往往不同，这表明以往统一为所有边缘客户端选择相同个性化层的策略并非最佳。（2）如果将取得最佳性能的个性化层作为客户端聚类的依据，可以形成多个数量相对均匀的集群，这表明恰当的模型解耦方式有助于中央服务器识别相似边缘客户端。

基于上述发现，本文提出了一种基于模型解耦的云边协同聚类联邦学习框架，名为FedCPMD。该框架将联邦学习分为两个阶段：第一阶段是聚类准备阶段，负责为各客户端选择候选个性化层，做好聚类的前期准备。第二阶段是聚类联邦学习阶段，根据个性化层的选择结果对客户端进行聚类，并在各集群内进行基于模型解耦的个性化联邦学习。

在聚类准备阶段，为量化分析神经网络每一层对最终结果的独立影响，本文提出了面向多层神经网络的逐层知识化表征方法。基于此，本文还设计了一种逐层对比的个性化层选择机制，为不同数据分布的边缘客户端选择最佳个性化层。

在聚类联邦学习阶段，根据最佳个性化层选择结果对边缘客户端进行聚类，并在各集群内部实现基于云边模型解耦的个性化联邦学习模型训练。

本文主要贡献如下：

（1）在六个真实数据集上进行动机实验，得出了两个关键发现，并基于此提出了一种面向多层神经网络模型的逐层知识化表征方法。本研究通过量化每层神经网络对最终模型性能的独立影响，优化个性化层选择方案。

（2）提出了一种基于模型解耦的聚类联邦学习框架，名为FedCPMD。该框架能够根据个性化层的选择结果实现动态聚类，无需预先设定集群数目。本文还讨论了FedCPMD在多种分布距离指标下的性能表现，检验了该方法的通用性与有效性。

(3) 实验验证了所提方案在模型训练方面具有优势。根据九个真实数据集的测试结果，FedCPMD与十余种先进方案相比，在CIFAR100、CINIC10、SVHN和Tiny ImageNet等复杂数据集上分类准确率能够平均提升约4%(VGG11，$\alpha=0.1$)。

2 相关工作

2.1 异构联邦学习

由谷歌团队提出的FedAvg[1]方法被视为联邦学习开创性算法，允许跨分布式边缘设备在协作训练全局模型时保护数据隐私安全。然而，当该方法应用于Non-IID场景时，其收敛速度显著变慢，最终性能也明显降低。对此，研究者们提出异构联邦学习，一种专门应对客户端数据分布异质性的联邦学习方法。常见策略包括：(1)知识蒸馏[3-5]，一种优化中央服务器和边缘客户端之间模型参数或梯度交换方式的方法，能够增强模型的泛化能力；(2)正则化项，在目标损失函数中使用正则化项[6-8]纠正全局模型的更新漂移，提高全局模型的鲁棒性和普适性；(3)相似性聚合，例如聚类[9-11]和多任务学习[8,19]等方法，旨在促进相似客户端之间的模型参数高效共享。然而，上述方法都存在固有缺陷，例如依赖额外数据集[3-5]、增加了模型计算复杂度[6-8]和共享参数规模、易引发数据隐私泄漏风险[8-11,19]。当前一个具有潜力的方案是(4)模型解耦，该方法将神经网络模型分解为共享主体层(Body)和个性化头部(Head)。共享主体层的参数更新遵循传统联邦学习范式，全局模型参数由参与联邦的所有边缘客户端的参数加权平均计算得出；个性化头部的参数则不提交至中央服务器，而是由每个边缘客户端基于本地数据独立训练得出。

2.2 模型解耦

模型解耦的主要方法包括FedRep[12]、pFedSim[14]、FedPer[20]和FedPav[21]等。上述方法将主体层参数上传至中央服务器进行全局聚合，而将个性化层参数保留在边缘客户端进行本地更新，以应对数据异质性问题。FedBN[22]和FedAP[23]则将批量归一化层(Batch-Normalization Layer)作为个性化层，将其他层作为主体层，以应对特征分布随时间变化而引发的模型性能下降问题，但它们只适用于包含批量归一化层的模型架构。此外，FedBabu[13]将分类器层作为个性化头部，将其前面所有层视为主体层。头部参数采用随机初始化，并在整个训练过程中固定不变，以减少模型的计算成本。然而，该方法对初始参数的选择较为敏感。

现有研究工作主要存在以下两方面缺陷：一是解耦方式单一，现有研究主要以分类器层作为个性化层，缺乏解耦的理论依据。二是解耦缺乏灵活性，现有研究大多为所有边缘客户端统一定制个性化层，这种固有的解耦方式过于僵化，无法灵活适应并利用客户端之间的数据分布特性差异。本文将最能表征客户端异质性的神经网络层作为个性化层，并给出了选层机制及原理，以更好地应对客户端数据异质性，避免因数据差异过大而造成训练不稳定。

2.3 聚类联邦学习

聚类联邦学习(Clustered Federated Learning，CFL)[9]将具有数据同质性的边缘客户端分组到相同集群中，以缓解数据异质性问题。FedGroup[24]通过分解余弦相似度来构建欧几里得距离，并使用模型参数作为函数输入来量化客户端间优化方向的相似性，进而实现客户端的静态聚类。FeSEM[10]通过提取客户端的空间表征，并使用空间表征计算该客户端与各集群聚类中心距离，来解决目标聚类问题。IFCA[25]给所有边缘客户端分发K个全局模型，帮助边缘客户端计算模型更新损失并根据损失情况执行聚类。CGPFL[26]则通过为每个客户端保留个性化模型来解决聚类问题，中央服务器通过软正则化方法加速集群内客户端相互学习。尽管聚类联邦学习已得到大量研究，但仍面临以下挑战：(1)异质性环境下，边缘客户端之间的数据分布差异显著，如何准确识别并聚类边缘客户端以提升模型性能。(2)客户端的随机采样机制，导致部分客户端在不同通信轮次下，动态加入或退出训练。当新一轮被采样客户端的数据分布与现有聚类集群不一致时，如何动态调整聚类结构以保持训练稳定。(3)如何根据实际情况自动确定合适的聚类集群数目。

本文通过量化神经网络各层输出特征与边缘客户端数据分布的差异性，捕捉最能表征客户端异质性的神经网络层。通过追踪记录各客户端差异性最小的神经网络层，FedCPMD能自动确定聚类集群数目，并根据记录的层信息，在后续轮次中动态将新加入的客户端分配到对应集群中。所提方法能够显著提升联邦学习的稳定性，有效应对上述三个挑战。

3 基础知识

本节主要介绍联邦学习云边系统模型、引入基

于模型解耦的联邦学习工作流程。关于符号表示，本文遵循标准的数学符号规范，其中$|\cdot|$和$\|\cdot\|$分别表示1-范数和2-范数。为便于读者查阅，表1列出了本文使用的所有数学符号及其对应含义。

表1　数学符号及其含义

符号	定义
$\mathrm{C};N$	边缘客户端集合；边缘客户端的总数量
$\mathrm{D}_i;\mathrm{X}_i,\mathrm{Y}_i$	第i个客户端的本地数据集；输入数据集和标签集
$n_i;n$	第i个客户端的样本量；所有客户端的总样本数量
$K;K_p$	总通信轮次数量；聚类准备阶段的通信轮次数量
$\mathrm{C}^k;\mathrm{M}_m^k;N^k$	参与第k轮联邦的客户端集合；属于第m个集群且参与第k轮联邦的客户端集合；参与第k轮联邦的客户端数量
$n^k;n_m^k$	参与第k轮联邦的客户端样本数量；属于第m个集群且参与第k轮联邦的客户端样本数量
γ	每轮参与联邦的客户端比例
$\theta_G^k;\theta_i^k$	第k轮通信的全局模型参数和第i个客户端参数
$\omega;\phi;\boldsymbol{\Phi}$	主体层参数；个性化层参数；客户端相似度矩阵
$z^{o_l};l^*$	第l层输出的特征分布；最佳个性化层
$\mathrm{M};M$	聚类所得的客户端集群及其数量
S	存放候选个性化层的字典

3.1　联邦系统模型

令C为边缘客户端集合，$N=|\mathrm{C}|$表示边缘客户端数量。设第i个客户端的本地数据集为D_i，进一步地，$\mathrm{D}_i=\{\mathrm{X}_i,\mathrm{Y}_i\}=\{x_j^{(i)},y_j^{(i)}\}_{j=1}^{n_i}$，其中$n_i$表示第$i$个客户端的数据样本量，$x_j$表示第$j$个输入数据，$y_j$表示对应标签，$\mathrm{X}_i,\mathrm{Y}_i$则分别表示输入数据集和标签集。所有客户端的样本总数为$n=\sum_{i\in\mathrm{C}}n_i$。

3.2　基于模型解耦的聚类联邦学习

（1）联邦学习工作流程：联邦学习训练过程包含K轮通信，对于每一轮通信$k\in[0,K]$，都有一部分客户端被随机选中参与联邦训练。记参与第k轮训练的客户端为C^k，$\mathrm{C}^k\subset\mathrm{C}$，其数量可表示为$N^k=|\mathrm{C}^k|=\gamma\cdot|\mathrm{C}|$，其中$\gamma$是客户端参与比例。根据联邦学习的典型流程，首先服务器将全局模型参数θ_G^k分发给一组选定的客户端，首次训练的初始参数为θ_G^0；接着，客户端接收到全局模型参数后，将利用本地数据集D_i训练更新本地模型参数。联邦学习的全局目标函数可表示如下：

$$\arg\min_{\theta}\frac{1}{N}\sum_{i\in\mathrm{C}}f_i(\theta)\tag{1}$$

其中$f_i(\theta)=\mathrm{E}_{(x,y)\sim\mathrm{D}_i}[l(\theta;(x,y))]$，$l$表示第$i$个客户端的局部经验损失函数。完成当前轮次的本地训练后，客户端将其本地模型参数θ_i^k回传至服务器。接收到客户端参数后，服务器将根据样本数量进行加权平均聚合，数学表达为

$$\theta_G^k=\sum_{i\in\mathrm{C}^k}\frac{n_i}{n^k}\theta_i^k\tag{2}$$

其中$n^k=\sum_{i\in\mathrm{C}^k}n_i$，表示参与第$k$轮联邦训练的所有客户端的总样本数量。

（2）模型解耦：形式上，模型参数θ可解耦为$\theta=\omega\circ\phi$，其中ω表示公共主体层，ϕ表示个性化层。基于模型解耦的联邦学习目标函数可对应表示为

$$\arg\min_{\omega,\{\phi_i\}}\frac{1}{N}\sum_{i\in\mathrm{C}}f_i(\omega,\phi_i)\tag{3}$$

这里$\omega\circ\phi_i=\theta_i$。在每个通信轮次下，共享组件$\omega$将在所有客户端上保持初始化一致并参与服务器端全局聚合；而ϕ_i则表示客户端i包含异质性的唯一个性化层，其参数仅保留本地更新不参与全局聚合。

（3）基于解耦的聚类联邦学习：假设将总边缘客户端集合C聚类为M个集群，记为M，则有$\mathrm{C}=\sum_{m=1}^{M}\mathrm{M}_m$。基于模型解耦的聚类联邦学习目标函数可进一步表示为

$$\arg\min_{\{\omega_m\},\{\phi_i\}}\frac{1}{N}\sum_{m\in\mathrm{M}}\sum_{i\in\mathrm{C}_m}f_i(\omega_m,\phi_i)\tag{4}$$

其中ω_m表示第m个集群中客户端的公共主体层参数。此时所有边缘客户端仅在所属集群内参与聚合。第k轮通信下，集群M_m中边缘客户端的参数聚合更新公式可表述如下：

$$\omega_m^k=\sum_{i\in\mathrm{M}_m^k}\frac{n_i}{n_m^k}\omega_i^k\tag{5}$$

这里ω_m^k表示第k个通信轮次下第m个集群中客户端聚合后的公共主体层参数；M_m^k表示属于第m个集群且参与第k轮通信的客户端集合；n_m^k则表示属于第m个集群且参与第k轮通信的客户端样本数量。

4 动机实验

本节主要从两方面阐述该项工作的研究动机：(1) 探究现有模型解耦工作所采取的“为所有客户端选取相同个性化层”的解耦方案是否为最优方案，是否存在其它更灵活高效的个性化层选择机制。(2) 探讨利用聚类机制区分不同客户端数据异质性类型的必要性，即基于个性化层选择结果的聚类机制能否进一步提升模型性能。

动机实验参数设置如下：联邦训练的通信轮数设置为200，客户端参与比例γ为0.1。边缘客户端数量设置为100，各客户端的数据分布遵循Dirichlet分布$Dir(\alpha)$，其中$\alpha=0.1$。所有客户端都采用SGD作为模型优化器，其中本地迭代轮数设置为5，批量大小为32，学习率为0.01。

4.1 基于聚类和统一个性化层的模型性能比较

包括FedRep[12]、FedBadu[13]和pFedSim[14]在内的现有大部分模型解耦方法均致力于统一边缘客户端的个性化层。为探究聚类是否有益于提高联邦模型性能，本文基于LeNet5①架构进行了以下探索。首先记录所有边缘客户端统一选择某层作为个性化层时的准确率结果。针对各客户端，根据其在某层取得的最高准确率，将该客户端归入该层对应的集群。由于神经网络浅层侧重于局部特征提取，深层则用于提取全局特征，而模型差异主要体现在最后几层，因此本研究聚焦于全连接层。以ID为0～9的客户端为例，表2展示了10个客户端在CIFAR10数据集上各层的准确率。以ID为0的客户端为例，其选择Classifier时准确率最高，因此被分到Classifier对应的集群中。按照该方式，100个客户端将依据其最佳表现层被分配到相应集群。为说明客户端数据分布对于聚类选层的影响，本节新增$\alpha=0.5$的情况，并将准确率汇总在表3中。对比表2和表3可以发现，当数据分布发生变化时，客户端1、3、4、6、7、8的选层结果均发生改变，这表明数据分布的变化会对客户端的选层结果造成影响。

表2 $\alpha=0.1$时，10个边缘客户端关于CIFAR10数据集的各层分类准确率(%，粗体表示最优结果)

ID	0	1	2	3	4	5	6	7	8	9
FC1	78.664	**77.616**	74.246	**84.635**	**82.249**	**85.545**	85.429	90.009	**93.244**	85.429
FC2	79.186	71.161	**76.995**	76.565	79.709	**85.545**	**85.719**	91.693	87.796	**85.719**
Classifier	**82.866**	68.558	75.532	70.207	80.508	83.706	81.868	**92.171**	92.112	81.868

表3 $\alpha=0.5$时，10个边缘客户端关于CIFAR10数据集的各层分类准确率(%，粗体表示最优结果)

ID	0	1	2	3	4	5	6	7	8	9
FC1	62.767	59.867	56.229	58.245	66.748	66.548	66.197	66.657	70.017	67.065
FC2	67.440	**66.165**	**62.664**	**66.689**	**73.245**	**72.705**	70.250	**73.649**	**74.922**	**71.466**
Classifier	**68.130**	61.372	60.382	62.566	70.886	70.463	**70.984**	68.737	72.974	69.222

本文对基础LeNet5模型进行了拓展，设计了两个神经网络架构，详见附录I表16和表17。本研究在六个真实数据集上应用两种不同架构对边缘客户端进行实验，结果汇总在表4和表5中。为便于对比，本研究增加传统解耦方案为基线方法，即所有客户端选择相同层作为个性化层。其中X-full表示所有边缘客户端选择神经网络的X层作为个性化层时的性能表现；Mean-cluster则表示聚类后各集群的平均分类准确率。X-full要求所有边缘客户端统一选择X层作为个性化层，而Mean-cluster则针对各集群内部的客户端使用相同个性化层进行联邦学习，不同集群之间相互独立。

实验结果显示，对于LeNet5拓展模型，在模型解耦的基础上，采用聚类联邦学习方法的平均分类精度(Mean-cluster)在所有数据集上均优于统一个性化层的传统模型解耦方法(FC1-full，Classifier-full)。尤其对于CIFAR100数据集，Mean-cluster比传统解耦方法的最优结果Classifier-full高出15.691%，CINIC10数据集则以90.060%的平均精度超过统一选层9.790%。值得注意的是，随着模型架构中全连接层数量增加，基于聚类联邦学习的模型性能有所下降，如表5中模型平均分类精度仅在3个数据集上高于传统解耦方法。导致该结果的原因有两方面：一是

① Maintainers T. TorchVision: PyTorch's computer vision library. November 2016. https://pytorch.org/vision/stable/index.html

表4　双全连接层LeNet5拓展模型关于六个数据集的分类准确率(%,粗体和下划线表示最优和次优结果)

架构1	CIFAR100	CINIC10	EMNIST	MNIST	SVHN	T. ImageNet
Mean-cluster	**59.338(+15.691)**	**90.06(+9.79)**	**97.282(+3.647)**	**99.516(+0.18)**	**95.254(+1.381)**	**31.524(+3.323)**
FC1-full	40.367	80.270	92.968	98.827	92.990	22.345
Classifier-full	43.647	78.792	93.635	99.336	93.873	28.201

表5　LeNet5模型关于六个数据集的分类准确率(%,粗体和下划线表示最优和次优结果)

架构2	CIFAR100	CINIC10	EMNIST	MNIST	SVHN	T. ImageNet
Mean-cluster	**48.41(+0.869)**	**81.365(+0.79)**	**95.038(+1.044)**	99.367(−0.054)	94.218(−0.436)	26.900(−4.156)
FC1-full	42.698	80.500	93.188	98.956	93.302	21.740
FC2-full	47.541	80.575	93.994	**99.421**	**94.654**	**31.056**
Classifier-full	42.925	78.983	93.554	99.363	94.240	28.061

神经网络层数增加导致聚类复杂性增加。随着层数增长,系统中边缘客户端面临的聚类决策环境更复杂,聚类错误的概率会相应提升,进而导致模型性能下降。二是集群数量增长导致单个集群内部客户端数量减少,引发模型稳定性问题。客户端数量减少可能导致训练数据不足,进而影响模型更新的稳定性和收敛速度,导致模型最终性能下降。

4.2　聚类机制的必要性探讨

为探讨聚类机制的必要性,本节给出了关于CINIC10和SVHN两个数据集的选层聚类结果,并统计了各集群内的客户端数量,见图2。结果显示,在LeNet5扩展模型架构下,对于CINIC10,选择FC1和Classifier作为最优个性化层的客户端数量分别为79和21。若采用统一选层策略,即所有客户端选择相同个性化层,许多客户端无法选择其最优层,将导致其模型性能下降。尤其在SVHN数据集上,共有35个边缘客户端的最优个性化层为Classifier,但在传统解耦方案中,这些客户端的个性化层将被强制设定为FC1,严重限制了模型的性能优势。因此,统一选层难以满足各客户端的个性化需求,而聚类机制通过将客户端分组,能够针对性缓解统一选层带来的性能下降问题。

5　对比层选择机制FedCPMD

本节介绍了一种基于度量特征分布转移程度的新型个性化选层策略,并详细阐述了其工作原理。在此基础上,本文设计了一种名为FedCPMD的对比选层机制,旨在分析特征分布的转移情况,实现训练过程中个性化层的动态选择。为便于理解和应用,本文提供了FedCPMD在联邦学习云边系统中的具体实现算法。

5.1　对比层选择机制原理

Chen等人[18]通过观察CIFAR10数据集下LeNet5模型各输出层的特征高斯分布可视化结果发现:随着网络层级加深,中间网络层的特征分布逐渐朝零均值分布移动,并且移动程度在不同层之间并不均匀。为了量化这种特征分布转移程度,沿着该项工作的研究路径,本文使用特征分布转移距离s_l来衡量神经网络框架下每一层网络的低维特征变换与从输入到标签的数据分布传输的对齐情况。

特征分布转移距离s_l的核心原理在于:神经网络的中间隐藏层能够捕获数据的低维特征信息。具体地,给定输出o_l和o_{l-1},它们表示神经网络所提取的低维特征。当客户端数据分布发生变化时,中间层所提取的低维特征信息也将相应改变。本文将最能准确捕捉客户端数据异质性的层定义为最佳个性化层。特征分布转移距离s_l通过量化神经网络各层的输出特征与客户端数据分布的对齐程度,帮助识别出最能反映客户端数据异质性的层,从而指导最佳个性化层的选择。

对比层选择机制的关键机理在于:各边缘客户端的最佳个性化层应当与其独特的异构数据分布特征密切对齐。本文通过构建s_l来评估低维特征分布z^{o_l}和异构数据分布z^x和z^y的对齐程度。与传统基于CKA的、仅关注分类器输出的解决方案不同,s_l量化神经网络每一层的输入和输出,能够更全面地评估每一层的低维特征与客户端数据分布的对齐情况。其数学表达式如下所示:

$$s_l = \min d\left(\left(z^{o_l} - z^{o_{l-1}}\right), \left(z^y - z^x\right)\right) \tag{6}$$

其中函数$d(\cdot)$表示通用分布距离,适用于常见的分布距离函数。符号o_l和o_{l-1}分别指神经网络第l层和第$l-1$层输出,其对应的低维特征分布由z^{o_l}和$z^{o_{l-1}}$表示。变量x和y代表输入数据样本及其对应

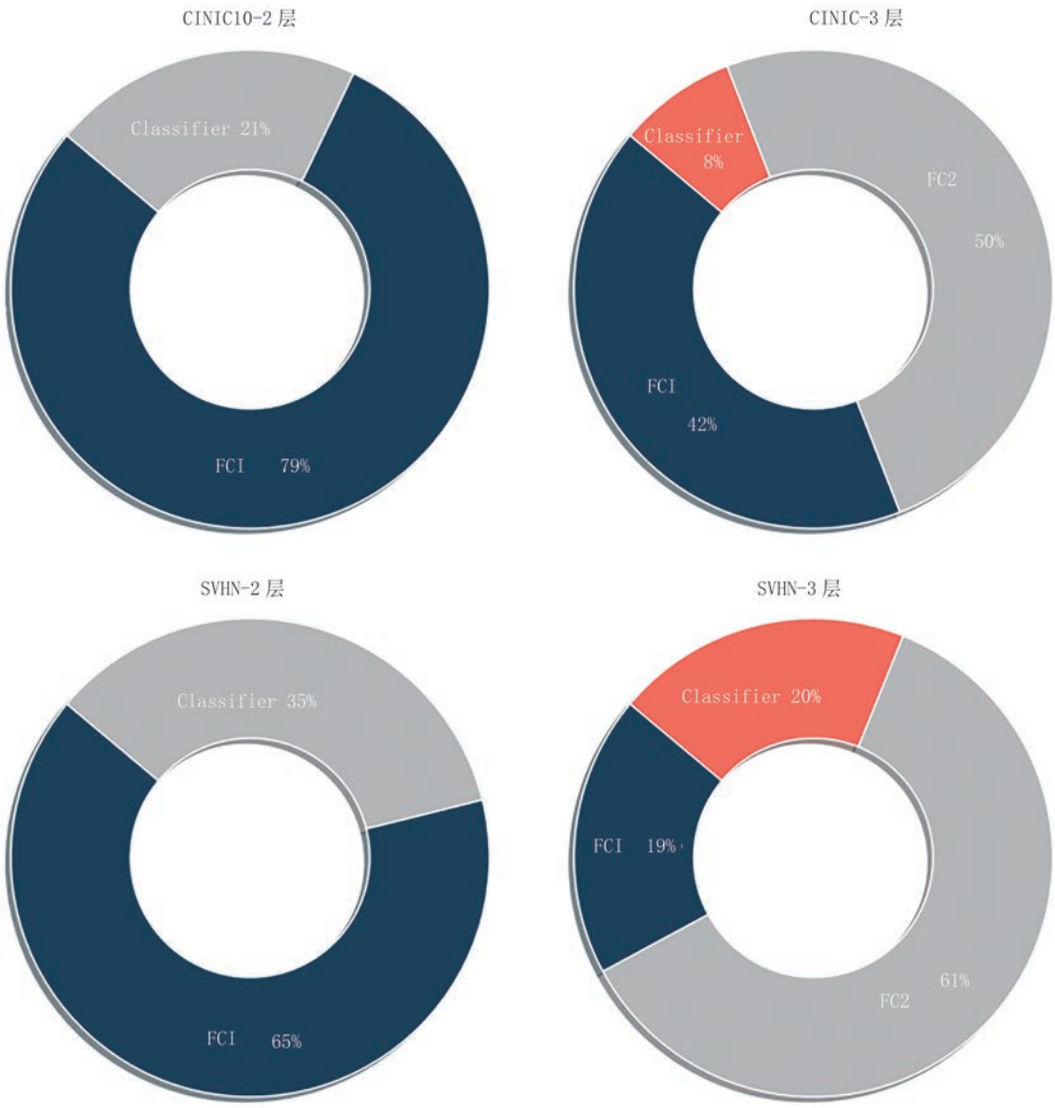

图2 CINIC10和SVHN关于各集群的客户端数量统计

的数据标签，它们根据数据分布捕获输入和标签的低维表示，其特征分布由z^x和z^y表示。为简化模型，本文将特征分布z建模为一个高斯分布。

鉴于神经网络的黑盒性质，精确计算每个隐藏层的特征分布转移距离极具挑战性，本文采用一种估计方法来解决该问题。将x和y作为参考，利用当前层输出与原始数据x、原始标签y的分布距离以及前一层输出与原始数据x、原始标签y的分布距离之间的差值来估计当前层的特征分布转移情况。基于此，公式(6)可以进一步写为：

$$\begin{aligned} s_l &= \min d\left(\left(z^{o_l}-z^{o_{l-1}}\right),\left(z^{y}-z^{x}\right)\right) \\ &\approx \min\left|\underbrace{\left(d\left(z^{o_l},z^{y}\right)-d\left(z^{o_l},z^{x}\right)\right)}_{\text{Prat A}}\right. \\ &\quad \left.-\underbrace{\left(d\left(z^{o_{l-1}},z^{y}\right)-d\left(z^{o_{l-1}},z^{x}\right)\right)}_{\text{Part B}}\right| \end{aligned} \tag{7}$$

由于分布距离恒为非负数，这里使用l_1范数来避免结果产生负值。Part A表示第l层输出与原始数据之间的特征分布转移距离，Part B表示第$l-1$层输出与原始数据之间的特征分布转移距离，二者之差用于估计第l层特征分布转移的增量。

本小节介绍四种分布距离$d(\cdot)$用于后文实验：JS、Wasserstein、Hellinger和Bhattacharyya距离。

(1) Jensen-Shannon距离

JS(Jensen-Shannon)距离[27]是一种衡量两个概率分布之间相似度的统计度量，通过计算两个分布的KL(Kullback-Leibler)散度的加权平均值，来量化二者的差异程度。JS距离通常被应用在文本分类[28]、图像匹配[29]等领域。数学公式如下：

$$JS(P\|Q)=\frac{1}{2}\times(D_{KL}(P\|M)+D_{KL}(Q\|M)) \tag{8}$$

其中，D_{KL}表示KL散度，P和Q是两个概率分布，M是P和Q的平均分布。

（2）Wasserstein距离

Wasserstein距离[30]用于衡量两个概率分布之间的差异，量化从一个分布到另一个分布的转换过程。它考虑将一个分布中的点移动到另一个分布中相应点的最佳匹配方式，以及为此移动所需付出的最小成本。对于分布μ和ν，Wasserstein距离被定义为将μ转换为ν所需的最低预期成本，可表达为

$$W_p(\mu,\nu)=\left(\inf_{\gamma\in\Gamma(\mu,\nu)}\int_{U\times V}c(\mu,\nu)^p\mathrm{d}\gamma(\mu,\nu)\right)^{\frac{1}{p}} \tag{9}$$

其中，U和V是定义分布μ和ν的空间。$\Gamma(\mu,\nu)$表示所有可能的μ和ν之间联合分布的集合。$c(\mu,\nu)$是衡量分布μ和ν中元素距离的成本函数，p表示Wasserstein 距离的阶数，本文设置$p=2$。该指标广泛应用于图像处理[31]和生成建模[32]等领域。

（3）Hellinger距离

Hellinger距离基于概率密度函数的相对差异来衡量两个分布之间的距离，具有对称性和非负性，其值介于0和1之间。Hellinger距离在统计分析[33]、模式识别[34]等领域有着广泛应用。其公式如下：

$$H(P,Q)=\sqrt{1-\sum_i\sqrt{p_i\cdot q_i}} \tag{10}$$

其中，P和Q是两个概率分布，p_i和q_i分别是分布P和Q中的第i个元素。

（4）Bhattacharyya距离

Bhattacharyya距离计算两个概率分布的重叠程度，衡量两个分布之间重叠部分的累积量。该距离在统计学[35]和信息论[36]中应用广泛，常用于比较两个概率分布之间的相似性。其公式可表述如下：

$$BC(P,Q)=-\ln\left(\sum_{i=1}^{n}\sqrt{p_i\cdot q_i}\right) \tag{11}$$

5.2　FedCPMD框架

联邦系统涉及两种类型节点，包括一个中央服务器和多个边缘客户端。本文提出一种基于模型解耦的聚类联邦学习框架，FedCPMD，旨在促进联邦系统各组件间协作，详细流程见图3。该框架包括两个主要阶段，依次为聚类准备阶段和聚类联邦学习阶段，两阶段中边缘客户端均采用模型解耦方案。

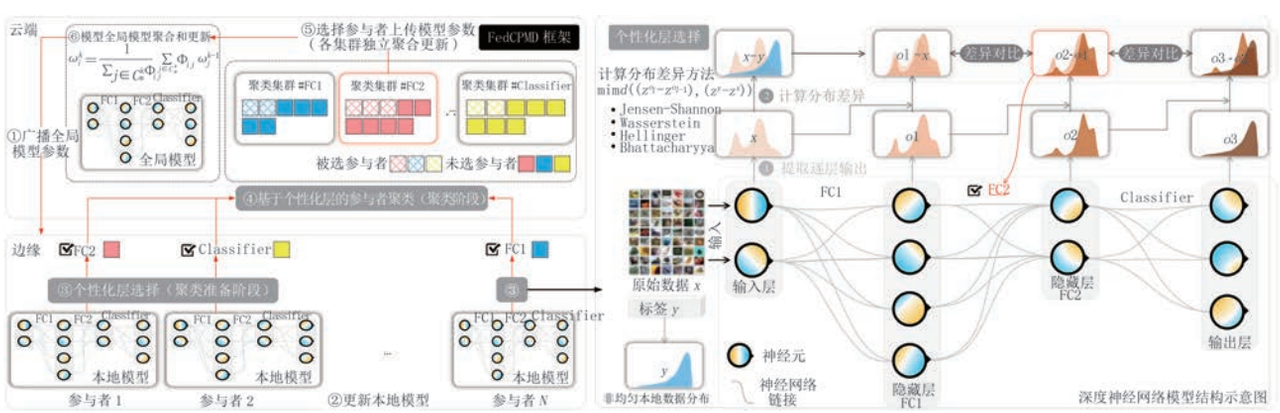

图3　基于动态个性化层选择的聚类联邦学习框架FedCPMD

在聚类准备阶段，Classifier被作为边缘客户端的个性化层保留在本地更新，而共享主体层被上传至中央服务器以样本加权的方式进行全局聚合。所有边缘客户端的候选个性化层由基于特征分布转移的层选择机制确定。边缘客户端需多次确认候选个性化层的选择，并基于确认结果确定聚类联邦学习阶段的个性化层。第一阶段具体流程如下：

（1）广播全局参数：中央服务器初始化全局模型参数并将其下发给所有边缘客户端。

（2）更新本地模型：接收到全局模型参数后，各边缘客户端使用本地数据训练更新客户端模型。

（3）选择候选个性化层：根据公式(7)，在各边缘客户端的训练过程中，为其从所有神经网络层中选择最佳候选个性化层。

（4）本地梯度通信：所有边缘客户端向中央服务器发送本地模型主体层参数的梯度更新。

（5）服务器端处理：中央服务器采用标准聚合方法更新主体层的全局模型参数，并统计当前累计通信轮次下各边缘客户端的选层结果。

聚类准备阶段需保证所有边缘客户端至少被采样一次。根据候选个性化层的选择结果，将选中次数最多的层作为该客户端的个性化层，用于聚类联邦学习阶段。在聚类联邦学习阶段，各边缘客户端被聚类到对应集群中，具有相同个性化层的客户端

被分配至同一集群。客户端的个性化层参数仅在本地训练更新，不参与集群内聚合。客户端的共享主体层参数将上传至对应集群服务器，并基于个性化层距离加权聚合更新参数，以优化异构数据分布下的模型性能。由于不同集群选择了不同的个性化层，导致各集群间主体层参数不同，无法进行全局模型聚合。聚类联邦学习阶段的流程概括如下：

（1）聚类客户端：根据所有边缘客户端的个性化层信息，将客户端聚类到相应集群，其中每个集群对应一个集群服务器。

（2）广播全局参数：每个集群服务器将全局模型参数下放至该集群内的所有边缘客户端。

（3）更新本地模型：接收到全局模型参数后，各边缘客户端利用本地数据对模型训练更新。

（4）本地梯度通信：各边缘客户端将其本地模型参数的梯度更新返回至各自对应的集群服务器。

（5）集群服务器处理：集群服务器利用个性化层参数计算集群内各边缘客户端间的相似度矩阵，并利用该矩阵对全局模型参数加权聚合以更新主体层参数。

6 算法设计

6.1 FedCPMD算法

FedCPMD为聚类准备和聚类联邦学习定制了两阶段算法，主要涉及三个关键子算法：客户端更新、个性化层选择和基于解耦的聚类联邦学习。

（1）客户端更新：每个客户端i使用本地数据D_i通过随机梯度下降方法更新本地模型参数(ω^k, ϕ_i^k)。该过程伪代码见算法1。

算法1. 客户端更新 $ClientUpdate(i, \theta_i^k)$

输入：服务器全局模型参数θ_G^k

输出：客户端本地模型参数(ω_i^k, ϕ_i^k)

1. 接收服务器下放的全局模型参数θ_G^k；
2. 将θ_G^k赋值为θ_i^k并解耦为(ω^k, ϕ_i^k)；
3. FOR /each local epoch/ DO
4. 使用随机梯度下降$SGD_i(\omega^k, \phi_i^k; D_i)$更新模型参数$(\omega_i^k, \phi_i^k)$；
5. END FOR
6. RETURN(ω_i^k, ϕ_i^k)

（2）聚类准备：令总通信轮数为K，聚类准备阶段通信轮数为K_p。对于每个通信轮次k，利用公式(7)计算各层$z^{o_l}-z^{o_{l-1}}$与z^y-z^x之间的对齐程度。接着，识别各客户端最小分布转移距离所对应的层，并将其加入候选字典S中。将最频繁出现的层作为该客户端的个性化层l^*。具体过程参见算法2。

算法2. 聚类准备 $ClusterPre(\theta_G^0, C, K_p, S)$

输入：初始参数θ_G^0，客户端集合C，当前阶段通信轮次总数K_p，初始候选字典S

输出：各客户端的个性化层l^*

1. 服务器下放初始化参数θ_G^0到所有客户端；
2. 各客户端计算本地数据的均值与方差，拟合数据分布z^x, z^y；
3. FOR $k=1, 2, \cdots, K_p$ DO
/*运行于客户端*/
4. 从C中随机挑选$\gamma\cdot|C|$个客户端组成集合C^k参与联邦 训练；
5. FOR 客户端$i\in C^k$ DO
6. 更新$\theta_i^k \leftarrow ClientUpdate(i, \theta_i^{k-1})$；
7. FOR /输出层o_l/ DO
8. 计算输出层l的均值和方差并拟合分布$z^{o_l}, z^{o_{l-1}}$；
9. 使用公式(7)计算s_l并将对应层记录到候选字典S中；
10. END FOR
11. 将参数θ_i^k和字典S上传至服务器；
12. END FOR
/*运行于服务器*/
13. 将S中各客户端出现最频繁的层作为该客户端的个性化层，解耦后仅聚合更新主体参数$\omega^k \leftarrow \sum_{i\in C^k}(|D_i|/|D|)\omega_i^k$，其中$|D|=\sum_{i\in C^k}|D_i|$；
14. END FOR

（3）聚类联邦学习：在该阶段$K-K_p$个通信轮次中，各边缘客户端将根据个性化层结果l^*被聚类到相应集群中，所有客户端将在本地更新其个性化层参数。基于个性化层选择结果，本文进一步使用公式(12)、(13)对模型公共主体层进行加权聚合，详细过程见算法3。

算法3. 聚类联邦学习 $ClusterFed(K, K_p, l^*)$

输入：总通信轮次K，前一阶段通信轮次数K_p，个性化层l^*

输出：客户端主体层参数$\{\omega_i^K\}$

1. 将具有相同个性化层的所有客户端聚类到对应集群中；
2. FOR /集群m/ DO
3. FOR /$k=K_p+1, \cdots, K$/ DO
/*运行于客户端*/
4. 接收集群服务器参数θ_m^k(初始为θ_G^k)，将其解耦为ω_m^k, ϕ_m^k并赋值给集群中所有客户端(ω_i^k, ϕ_i^k)；
5. 从集群M_m中随机挑选$\gamma\cdot|M_m|$个客户端组成M_m^k参与联邦训练；
6. FOR /客户端$i\in M_m^k$/ DO
7. 更新$(\omega_i^k, \phi_i^k) \leftarrow ClientUpdate(i, \theta_i^{k-1})$；

8. 上传ω_i^k到所属集群服务器；
9. END FOR
/*运行于各集群服务器*/
10. FOR /任意两个不同客户端$i,j\in \mathrm{M}_m^k$/ DO
11. 使用公式(13)计算相似度矩阵$\boldsymbol{\Phi}_{ij}$；使用公式(12)更新ω_i^k并广播ω_i^k到第i个客户端；
12. END FOR
13. END FOR
14. END FOR

FedCPMD利用上述3个子算法能够在JS、Wasserstein、Hellinger和Bhattacharyya距离4种指标下进行异构联邦学习并得到训练后的模型结果。

6.2　基于个性化层的加权全局聚合

本文借鉴pFedSim[14]中“分类器距离”理念，采用余弦相似度计算不同边缘客户端个性化层参数的相似性，并据此对主体层参数进行加权平均更新。具体计算公式如下：

$$\theta_i^k=\boldsymbol{\Phi}_i^k\circ\omega_i^k=\begin{cases}\omega_i^k=\dfrac{1}{\sum_{j\in \mathrm{C}_m^k}\boldsymbol{\Phi}_{ij}}\sum\limits_{j\in \mathrm{C}_m^k}\boldsymbol{\Phi}_{ij}\omega_j^{k-1},\\ \boldsymbol{\Phi}_i^k=\boldsymbol{\Phi}_i^{k-1}.\end{cases}\tag{12}$$

其中$\boldsymbol{\Phi}$表示相似度矩阵，$\boldsymbol{\Phi}_{ij}\in[0,1]$，其值越大意味着客户端$i$和$j$越相似，这里进一步写为公式(13)。

$$\boldsymbol{\Phi}_{ij}=\mathrm{cosine}[\theta_{l^*}(i),\theta_{l^*}(j)]=\frac{\boldsymbol{\Phi}_i\cdot\boldsymbol{\Phi}_j}{//\boldsymbol{\Phi}_i//\cdot//\boldsymbol{\Phi}_j//+\epsilon}\tag{13}$$

$\theta_{l^*}(i)$表示客户端i所选择的个性化层ϕ_i的参数。ϵ定义为一个极小的正值以避免产生极值，实验中设置为10^{-8}。FedCPMD在通信过程中仅涉及交换模型参数，无需传递任何额外信息，能够降低通信成本并保护数据隐私，提升联邦系统的安全性和可靠性。

6.3　时间复杂度分析

首先分析在FedCPMD的两个阶段中都会使用到的共用子算法客户端更新的复杂度。假设每个随机梯度下降的复杂度为$O(1)$，*ClientUpdate*(•)的时间复杂度主要受到本地迭代轮数E和客户端i的样本数量n_i两个因素影响，该子算法的总时间复杂度为$O(E\cdot n_i)$。

接下来分析聚类准备阶段的时间复杂度。客户端更新通过从总池C中随机选择一组客户端C^k开始每一轮通信，其复杂度为$O(|C|)$。接着，C^k中的各客户端利用本地数据使用*ClientUpdate*(•)更新参数。假设各客户端神经网络模型共L层，其计算和拟合高斯分布$\{z^{o_i}\}$的复杂度为$O(L)$，以Wasserstein距离为例，根据公式(7)计算分布距离的复杂度为$O(n_i)$，那么总复杂度为$O\left(|C^k|\cdot(E\cdot n_i+L+n_i)\right)$。由于$L$远小于$n_i$，该时间复杂度可简化为$O\left(|C^k|\cdot E\cdot n_i\right)$。服务器参数更新的时间复杂度为$O\left(|C^k|\right)$。算法2重复上述过程$K_p$轮，*ClusterPre*(•)的总时间复杂度为$O\left(K_p\cdot(E\cdot n_i)\cdot|C^k|\right)$。

接着分析聚类联邦学习阶段的时间复杂度。该阶段客户端涉及聚类、模型解耦和本地参数更新，其中聚类和解耦的时间复杂度均为$O(|C|)$，客户端参数更新*ClientUpdate*(•)为$O\left(|M_m|\cdot E\cdot n_i\right)$。此外，还需计算表示客户端之间相似性的矩阵$\Phi$，假设公式(13)的时间复杂度为$O(1)$，基于公式(12)的加权全局聚合时间复杂度为$O\left(|M_m|\right)$。算法3中步骤10至12则为$O\left(|M_m|^2+|M_m|\right)$。*ClusterFed*(•)的总时间复杂度为$O\left(\left(K-K_p\right)\cdot\left(|M_m|^2+|M_m|+|M_m|\cdot E\cdot n_i\right)\right)$，这里进一步简化为$O\left(K\cdot\left(|M_m|^2+|M_m|\cdot E\cdot n_i\right)\right)$。

由上述分析可知，FedCPMD算法的时间复杂度受客户端数量$|C^k|$和$|M_m|$、客户端本地数据集大小n_i、迭代次数E以及通信轮次K影响。

7　实验设置与性能评估

本节主要介绍实验设置、实验数据集、基线对比方法及其参数配置。为了验证所提算法的性能，本文围绕联邦学习模型的准确率、收敛性、系统可拓展性、算法超参数设置、通信开销等方面与其他十余种基线对比，其中收敛性分析置于附录Ⅱ中。本研究还通过消融实验进一步验证了聚类和模型解耦分别带来的性能提升效果。

7.1　实验设置和数据集

本研究基于开源联邦学习基线框架FL-bench①。所有实验由一台高性能服务器部署实现，配备GPU（NVIDIA GeForce RTX 4090 24GB）和CPU（Intel i9-13900K，24核，3.00 GHz）。

（1）数据集。实验涉及九个标准数据集：CIFAR10、CIFAR100、CINIC10、EMNIST、FMNIST、MedmNistA、MNIST、SVHN和TinyImageNet，其详细

① https://github.com/KarhouTam/FL-bench

表6 边缘客户端所使用的数据集统计信息

数据集	样本数	类别数	描述	分辨率	年份
CIFAR10/100	60 000	10/100	包括飞机、汽车、鸟类等10/100类别图像	32 × 32	2009
CINIC10	270 000	10	包括飞机、汽车、鸟类等10类别图像	32 × 32	2018
EMNIST	805 263	62	带有字母和数字的62类别扩展MNIST数据集	28 × 28	2017
FMNIST	70 000	10	包括T恤、裤子、套衫等时尚单品的10类别图像	28 × 28	2017
MNIST	70 000	10	10类别手写数字图像	28 × 28	1998
MedmNistA	58 850	11	腹部CT上的10类生物医学图像	28 × 28	2019
SVHN	600 000	10	10类别街景房屋门牌号	32 × 32	2011
TinyImageNet	110 000	200	200类别的图像识别挑战数据集	64 × 64	2015

信息见表6。每个数据集被划分为100个子集(即$N=100$),数据分布遵循Dirichlet分布$Dir(\alpha)$,其中α被设置为集合$\{0.1, 0.5, 1.0\}$中的值,以模拟Non-IID数据分布场景。数据分布的异构程度由超参数α确定,α值越小,数据异质性越显著。例如,当$\alpha=0.1$时,不同客户端的数据分布异质性将非常显著,几乎所有边缘客户端样本都不可能涵盖所有类别,即$|Y_i|\leqslant|Y|$,Y_i表示第i个边缘客户端的数据样本的标签空间。

(2)基线方法。为全面评估所提方法的泛化性能和个性化性能,实验将经典的FedAvg方法[1]以及本地训练方法Local-Only作为对比基线。此外,本研究对比了以下十余种前沿的联邦学习方案:

① FedProx[6]通过在目标损失函数中引入一个近端项,来处理客户端间的异质性问题。该近端项使局部模型更新更接近全局模型更新,有效防止边缘客户端模型偏离全局模型。

② CFL[9]通过捕捉边缘客户端模型梯度更新的几何特性,评估客户端之间的相似性,将具有相似梯度特性的客户端动态聚类到同一集群。

③ FedRep[12]提出包含全局共享表示和本地个性化表示的混合模型,其中全局共享表示在所有边缘客户端上共同优化,而本地个性化表示则仅根据各客户端的特定数据独立更新。

④ FedBabu[13]通过实验发现个性化联邦学习算法性能下降的原因主要来自个性化头部,并据此提出在联邦训练过程中随机初始化个性化头部并保持其固定,仅聚合更新主体层参数。

⑤ pFedSim[14]在Body-Head的解耦方式之上,利用分类器参数计算各边缘客户端之间的相似性,并将其作为主体层全局聚合的权重参数。

⑥ FedCMD[18]提出一种基于Wasserstein距离的神经网络逐层特征分布转移度量,以量化逐层特征变化幅度,实现个性化层选择和模型解耦。

⑦ FedPer[20]通过将深度前馈神经网络划分为基础层和个性化层来实现个性化联邦学习。该方法将神经网络最后一层作为个性化头部进行模型解耦,以对抗统计异质性的不良影响。

⑧ FedBN[22]在边缘客户端模型中保留批量归一化层以稳定联邦训练过程,并将其余网络层参数上传至中央服务器进行全局聚合。

⑨ FedAP[23]在FedBN框架之上,使用基于批量归一化层统计的Wasserstein距离度量边缘客户端之间的相似性,以优化模型聚合。

⑩ FedDyn[37]引入动态正则化器,根据每一轮通信结果动态调整正则化参数,从而保证各边缘客户端与中央服务器的解决方案保持一致。

⑪ FedFomo[38]量化评估每个边缘客户端在参数共享阶段能够从其它客户端获得的增益,并基于此计算每个客户端最佳的加权参数组合模型。

上述对比方法所涉及的部分超参数设置如下,超参数设置与对比方法原文保持一致。

① FedProx[6]设置超参数$\mu=1$。

② FedRep[12]设置训练特征提取器通信轮次为1。

③ pFedSim[14]设置泛化比例$\rho_{general}=0.5$。

④ FedCMD[18]设置划分比例$\rho=0.1$。

⑤ FedAP[23]设置模型动能$\mu=0.5$。

⑥ FedDyn[37]设置超参数$\alpha=0.01$。

⑦ FedFomo[38]设置$M=5$,验证集比例为0.2。

⑧ FedCPMD(Ours)设置聚类准备阶段的通信轮次数为60。

(3)实验设置。所有方法均基于LeNet5和VGG11[39]模型架构,本文实现了在所有数据集上的性能评估,架构细节见附录I表16和表18。均等划分每个边缘客户端数据为训练集和测试集$|D_i^{train}|=|D_i^{test}|$,两者无交集样本。采用SGD作为模型优化器,

设置学习率为0.01，客户端参与比例$\gamma=0.1$，通信轮次$K=200$，本地迭代轮数$E=5$，批量大小为32。

7.2　分类准确率比较

（1）不同方案在LeNet5上的分类准确率比较

表7～表9展示了FedCPMD与其他十三种方法（含FedAvg和Local-Only基线）在LeNet5架构上的准确率比较结果。表7展示了$\alpha=0.1$时的结果，FedCPMD在大多数据集上表现优异，尤其针对复杂数据集CIFAR100和Tiny ImageNet，其准确率分别比次优方案FedCMD提升了0.766%和1.991%，这表明FedCMD在高异质性和复杂任务场景中具有优势。相比之下，基于全局聚合的FedAvg和FedProx表现较弱，在复杂数据集上准确率较低。这是因为客户端数据分布呈高度异构，这种异质性会损害全局模型性能；而FedCPMD通过将具有相似数据分布（即具有相同个性化层）的客户端聚类到同一集群，有效缓解了异质性带来的不利影响。

表7　$\alpha=0.1$时，关于LeNet5模型的分类准确率（%，粗体和下划线表示最优和次优结果）

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	85.356	36.058	78.876	92.712	95.528	91.808	96.555	89.402	21.424
FedAvg[1]	24.539	13.845	29.516	75.648	77.701	67.475	94.668	68.894	9.350
CFL[9]	25.166	13.912	29.581	75.714	78.344	69.484	94.492	73.332	9.413
FedBN[22]	41.496	13.803	42.703	77.749	79.463	76.026	95.503	73.858	10.721
FedAP[23]	84.092	30.184	75.670	92.829	95.383	94.598	98.452	92.907	14.412
FedBabu[13]	29.015	5.718	23.864	73.819	74.808	64.034	94.320	67.330	3.467
FedFomo[38]	84.081	35.754	76.455	92.411	95.341	91.351	96.561	88.517	20.513
FedProx[6]	28.893	12.897	29.333	74.547	79.285	66.074	93.470	70.442	10.116
FedRep[12]	85.129	36.500	<u>82.914</u>	94.379	95.501	91.691	97.852	91.531	26.066
pFedSim[14]	81.712	38.452	74.771	<u>94.552</u>	96.171	94.113	99.062	93.371	30.179
FedDyn[37]	27.538	13.475	30.278	75.569	78.536	66.563	94.369	72.487	9.423
FedPer[20]	81.779	36.777	78.905	93.949	95.709	93.903	98.800	93.176	26.083
FedCMD[18]	<u>87.252</u>	<u>48.061</u>	80.773	94.245	<u>96.569</u>	<u>96.464</u>	**99.441**	**95.023**	<u>31.107</u>
FedCPMD	**90.267 (+3.015)**	**48.827 (+0.766)**	**89.079 (+6.165)**	**95.518 (+0.966)**	**97.803 (+1.234)**	**97.740 (+1.276)**	<u>99.415</u> (−0.026)	<u>94.861</u> (−0.162)	**33.098 (+1.991)**

表8展示了异质性参数$\alpha=0.5$时上述方法在9个数据集上的分类准确率，此时数据异质性相比于$\alpha=0.1$更弱。分析表8可见，FedCPMD在复杂数据集上的表现尤为突出，例如在CINIC10数据集上，其准确率达到76.260%，比次优方案FedRep高出10.602%。针对CIFAR10等中等复杂数据集，FedCPMD以9.786%的绝对优势领先FedCMD。对于MNIST等简单数据集，所有方法均能取得较高准确率。

表9则展示了所有方法在$\alpha=1.0$时的模型分类准确率，此时每个客户端的数据分布异质性相比于$\alpha=0.1$和$\alpha=0.5$更低，各客户端数据呈现均匀分布。表9显示，FedCPMD在8个数据集上仍具有最优或者次优性能。例如在CINIC10数据集上，它以61.556%的准确率超次优方案FedRep 3.695%。综合表7～表9来看，随着异质性程度逐渐降低，FedaAvg和FedProx等传统联邦方案的准确率有所提升，但采用解耦方案的FedCPMD、FedPer和pFedSim等方法性能均有所下降。这表明传统算法对数据异质性更为敏感，解耦方案则更适合处理异质性数据。

（2）不同方案在VGG11上的分类准确率比较

表10～表12展示了FedCPMD与其他11种方法（含FedAvg和Local-Only基线）在VGG11模型上关于9个数据集的分类准确率比较结果。整体而言，在不同异质性水平下，各算法在所有数据集上的表现普遍优于LeNet5架构下的结果。这主要得益于VGG11架构的复杂性，其神经网络层数更多，参数规模更大，能够更有效地学习客户端数据特征，从而获得更好的分类性能。

表10显示了$\alpha=0.1$时的结果。FedCPMD在多个数据集，如CIFAR10、CIFAR100、CINIC10和SVHN上表现优异，特别是在CINIC10上比次优方案FedCMD提升3.844%。整体来看，pFedSim、

表8 $\alpha=0.5$时，关于LeNet5模型的分类准确率(%，粗体和下划线表示最优和次优结果)

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	57.366	14.370	55.863	84.462	87.024	79.494	92.377	74.408	6.646
FedAvg[1]	44.085	17.393	41.550	82.636	84.491	81.696	96.908	82.540	10.207
CFL[9]	44.443	16.931	41.580	82.758	84.401	81.702	96.962	82.418	10.219
FedBN[22]	50.654	16.845	46.284	82.859	84.982	85.712	97.306	83.514	11.26
FedAP[23]	62.987	18.439	56.187	86.864	89.945	90.249	98.016	87.571	6.948
FedBabu[13]	44.387	8.754	39.173	82.477	84.024	81.109	96.963	81.801	3.318
FedFomo[38]	58.419	13.300	55.029	83.659	86.611	78.381	93.385	72.330	7.951
FedProx[6]	45.004	16.330	41.051	82.07	83.582	80.560	96.781	82.273	10.640
FedRep[12]	64.459	15.594	65.658	88.551	89.622	85.346	96.606	84.967	10.325
pFedSim[14]	61.379	21.335	58.836	**90.055**	89.908	89.114	98.055	88.214	**16.725**
FedDyn[37]	43.818	16.861	40.815	82.711	84.110	81.278	97.021	82.038	10.085
FedPer[20]	61.908	16.908	62.104	88.032	90.034	88.544	97.670	87.592	10.615
FedCMD[18]	71.331	**27.531**	64.409	89.596	92.26	**93.982**	98.636	90.039	14.561
FedCPMD	**84.117** (**+9.786**)	27.138 (−0.393)	**76.260** (**+10.602**)	89.653 (−0.402)	**94.631** (**+2.371**)	93.480 (−0.502)	**98.729** (**+0.093**)	**90.788** (**+0.749**)	14.565 (−2.160)

表9 $\alpha=1.0$时，关于LeNet5模型的分类准确率(%，粗体和下划线表示最优和次优结果)

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	48.019	9.270	44.726	79.627	82.121	74.161	90.564	70.358	4.037
FedAvg[1]	45.101	17.318	42.239	83.254	84.613	86.537	96.927	82.969	10.274
CFL[9]	45.638	17.288	41.555	83.235	84.472	87.020	97.027	83.907	10.552
FedBN[22]	47.935	16.756	44.147	83.012	84.841	87.967	97.218	84.108	11.462
FedAP[23]	56.092	15.853	47.755	83.921	86.848	90.233	97.590	87.33	5.847
FedBabu[13]	44.735	9.025	43.988	82.806	84.436	85.716	96.776	82.582	3.630
FedFomo[38]	46.625	8.840	44.022	78.865	82.282	73.107	93.311	68.423	5.376
FedProx[6]	44.853	16.665	43.028	82.443	84.515	86.496	96.880	82.792	10.637
FedRep[12]	55.345	10.167	57.861	84.444	86.088	83.942	95.976	84.163	6.770
pFedSim[14]	56.454	17.055	52.834	**87.203**	88.276	88.296	97.587	87.828	**12.905**
FedDyn[37]	45.964	17.016	42.826	83.217	84.754	86.851	97.042	83.229	10.271
FedPer[20]	56.753	12.477	55.777	84.258	88.018	87.846	97.198	87.066	7.910
FedCMD[18]	**64.508**	**22.178**	57.468	86.454	89.837	**93.312**	98.271	89.464	12.537
FedCPMD	64.034 (−0.474)	21.260 (−0.918)	**61.556** (**+3.695**)	87.086 (−0.117)	**90.594** (**+0.757**)	92.375 (−0.937)	**98.441** (**+0.170**)	**90.172** (**+0.708**)	11.730 (−1.175)

FedPer和FedRep等基于解耦的方案相比其他方法更加优越，表明模型解耦能够有效缓解数据异质性问题。对于复杂数据集，如CIFAR100、CINIC10和Tiny ImageNet，FedCPMD的性能表现更好。

表11展示了$\alpha=0.5$时不同方法在各数据集上的准确率结果。FedCPMD在CIFAR10、CIFAR100和CINIC10等6个数据集上获得最佳表现。对于CIFAR10、CINIC10，其分类准确率比次优方案分别提高7.400%和11.656%。在EMNIST、FMNIST和TinyImageNet数据集上，FedCPMD与最优方案准确率性能差距小于1%。相较而言，Local-Only在大多数据集上表现较差，这说明联邦学习能够有效提升模型的分类准确率。

表12则呈现了$\alpha=1.0$时的分类准确率结果，此时数据异质性被设置为最低。FedCPMD仍然在多个数据集上取得最佳性能，尤其针对复杂数据集CIFAR100、CINIC10和SVHN，其准确率相对次优方案FedCMD分别提高了2.270%、3.032%和5.918%。这表明基于个性化层的聚类机制能够有效提升模型分类准确率。分析对比表10～表12，可

表10　$\alpha=0.1$时，关于VGG11模型的分类准确率(%，粗体和下划线表示最优和次优结果)

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	82.567	46.785	85.038	92.979	94.852	79.984	96.512	76.185	51.933
FedAvg[1]	60.519	35.209	50.753	82.024	81.336	91.814	98.042	66.374	38.106
CFL[9]	62.955	35.813	52.942	80.060	78.808	92.413	95.157	69.327	38.371
FedBabu[13]	60.506	35.252	51.046	81.859	80.815	90.956	97.793	66.586	36.181
FedFomo[38]	84.295	45.086	84.666	93.025	95.424	91.422	96.241	82.668	50.148
FedProx[6]	60.951	35.035	51.309	82.244	77.735	90.803	97.788	66.122	38.082
FedRep[12]	89.277	56.639	86.245	95.088	95.604	94.732	98.850	86.454	58.713
pFedSim[14]	89.009	60.775	85.175	95.583	95.928	97.139	98.696	89.077	62.121
FedDyn[37]	57.612	27.568	49.866	81.276	83.389	92.313	97.890	33.747	34.950
FedPer[20]	91.704	60.885	87.089	95.346	96.769	97.637	99.435	91.676	61.943
FedCMD[18]	91.527	62.645	89.529	95.237	97.435	98.755	**99.554**	94.348	**64.622**
FedCPMD	**94.488** **(+2.784)**	**66.255** **(+3.160)**	**93.373** **(+3.844)**	**97.114** **(+1.531)**	**97.833** **(+0.398)**	**98.963** **(+0.208)**	99.291 (−0.263)	**96.291** **(+1.943)**	63.583 (−1.039)

表11　$\alpha=0.5$时，关于VGG11模型的分类准确率(%，粗体和下划线表示最优和次优结果)

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	59.926	19.822	64.336	85.040	86.250	82.767	91.509	59.251	25.732
FedAvg[1]	62.743	37.279	54.421	84.284	89.139	96.339	98.661	79.294	41.134
CFL[9]	62.872	37.130	59.996	84.280	86.796	96.343	72.205	79.713	41.418
FedBabu[13]	62.843	37.750	54.426	84.433	89.268	96.029	98.705	79.407	39.340
FedFomo[38]	61.093	27.101	63.307	84.617	85.429	86.837	94.221	60.038	24.068
FedProx[6]	62.505	37.700	54.220	84.319	89.093	95.902	97.633	79.330	41.164
FedRep[12]	71.948	34.375	71.360	90.408	91.319	91.667	98.156	78.002	39.826
pFedSim[14]	73.510	44.004	71.972	**91.145**	92.421	96.548	98.798	83.462	48.208
FedDyn[37]	48.372	31.793	43.379	83.816	89.153	96.196	98.574	48.895	39.722
FedPer[20]	76.219	41.899	73.531	90.602	92.777	96.600	98.943	86.088	45.510
FedCMD[18]	84.271	49.877	76.757	90.868	**94.052**	98.296	99.205	90.707	**49.664**
FedCPMD	**91.671** **(+7.400)**	**50.682** **(+0.805)**	**88.413** **(+11.656)**	91.008 (−0.137)	93.238 (−0.814)	**99.495** **(+1.199)**	**99.315** **(+0.110)**	**94.687** **(+3.980)**	49.076 (−0.588)

表12　$\alpha=1.0$时，关于VGG11模型的分类准确率(%，粗体和下划线表示最优和次优结果)

方法	CIFAR10	CIFAR100	CINIC10	EMNIST	FMNIST	MedmNistA	MNIST	SVHN	T. ImageNet
Local-Only	56.544	13.834	58.390	80.359	81.930	79.902	88.383	54.872	18.098
FedAvg[1]	71.431	38.559	62.464	84.084	86.394	97.162	98.665	81.292	41.413
CFL[9]	70.940	38.159	62.400	84.183	86.398	97.183	98.278	81.778	41.459
FedBabu[13]	70.867	38.399	62.429	84.153	86.473	96.893	98.620	81.333	39.256
FedFomo[38]	60.873	25.278	62.354	82.855	81.587	88.230	96.460	66.079	29.474
FedProx[6]	71.144	38.191	62.278	84.000	86.485	96.749	98.519	81.114	41.357
FedRep[12]	72.308	28.019	69.198	87.133	86.900	91.163	97.134	76.705	34.544
pFedSim[14]	76.014	40.566	70.446	**88.568**	89.014	96.875	98.634	83.673	44.054
FedDyn[37]	66.512	32.728	57.758	83.686	85.013	97.277	98.640	52.815	39.762
FedPer[20]	76.905	37.869	70.881	87.582	89.154	96.633	98.648	85.929	40.579
FedCMD[18]	**82.474**	46.674	74.845	87.965	90.209	98.290	**99.133**	90.177	45.356
FedCPMD	81.773 (−0.701)	**48.944** **(+2.270)**	**77.877** **(+3.032)**	88.448 (−0.120)	**90.434** **(+0.225)**	**98.803** **(+0.513)**	99.051 (−0.082)	**96.095** **(+5.918)**	**46.053** **(+0.697)**

以发现：基于全局聚合的传统联邦学习方法，如FedAvg、FedProx，随着数据异质性降低，分类性能逐渐提升；而异构联邦学习方案，如pFedSim、FedCPMD，则随着数据异质性降低而下降。这表明异构联邦学习方法在处理异质性数据时更具优势。

7.3 关于不同指标的实验结果

本小节通过实验比较FedCPMD在四种距离分布度量下的性能差异，包括Jensen-Shannon、Wasserstein、Hellinger和Bhattacharyya距离。实验的超参数设置如下：异质性参数$\alpha=0.1$，聚类准备阶段的通信轮次为60，客户端参与比例$\gamma=0.1$。表13展示了FedCPMD在四种指标下关于九个数据集的实验数据，粗体和下划线分别为最优和次优结果。

表13显示，各数据集在四种不同指标下的分类准确率性能差距较小，尤其在EMNIST、MedmNistA、MNIST和SVHN四个数据集下，准确率标准差均低于1。为验证个性化选层机制的有效性，本研究以最大、最小标准差对应的CINIC10和MNIST数据集为代表，在图4中绘制了其在四种距离指标下所有边缘客户端的选层结果。

表13 9个数据集关于不同指标的分类准确率(%)

距离指标	JS	Wass.	Hell.	Bhat.	Std.
CIFAR10	83.310	81.022	<u>84.145</u>	**90.267**	3.948
CIFAR100	**48.788**	<u>48.410</u>	47.805	42.506	2.942
CINIC10	<u>83.025</u>	81.365	79.667	**89.079**	*4.099*
EMNIST	**95.518**	95.038	<u>95.440</u>	94.901	0.301
FMNIST	93.505	<u>95.283</u>	95.271	**97.803**	1.768
MedmNistA	95.664	95.721	<u>96.122</u>	**97.740**	0.974
MNIST	99.283	<u>99.367</u>	**99.415**	99.354	*0.055*
SVHN	<u>94.419</u>	94.218	93.410	**94.861**	0.607
T. ImageNet	**33.098**	26.900	27.938	<u>32.813</u>	3.227

图4使用10×10方格展示了100个边缘客户端的选层结果，其中每格对应一个独立的边缘客户端。方格内呈现3种颜色，分别表示其选层结果，如果两个方格颜色相同，则意味着对应客户端选择了相同个性化层。特别地，该图最后一列子图反映了边缘客户端在四种指标下的聚类一致性，方格颜色越深表明该客户端在四种指标下的选层结果越一致。结合图4和表13分析，采用不同距离度量标准不会显著影响模型选层的最终结果，表明所提个性化选层机制较为稳定。

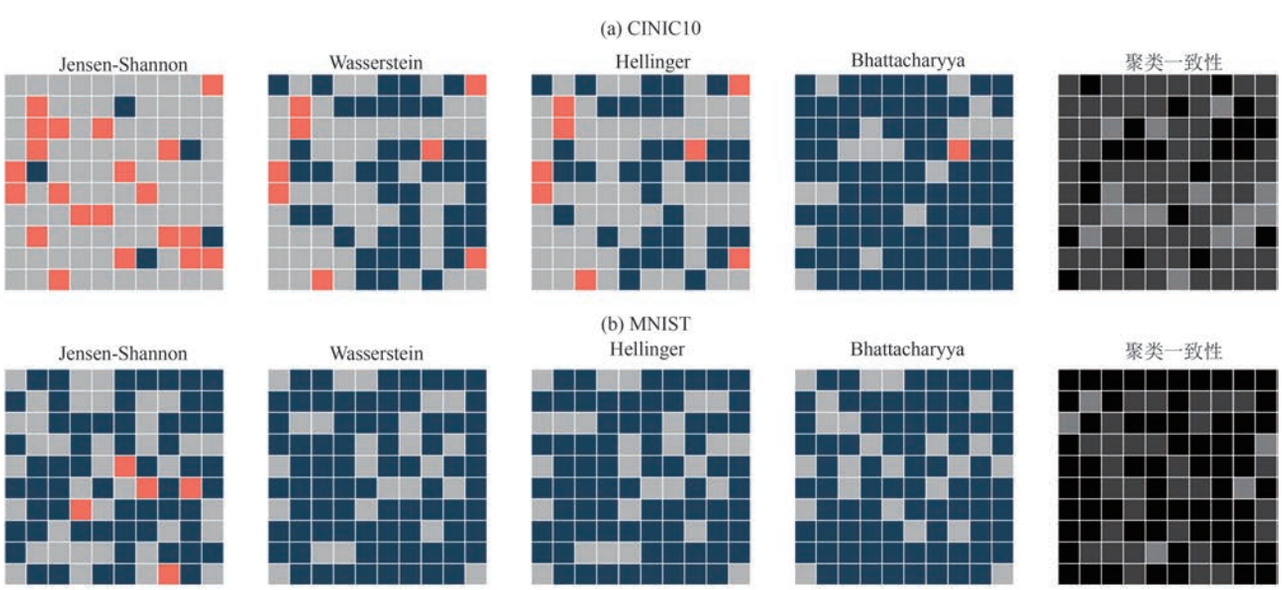

图4 CINIC10和MNIST在四种指标下的客户端选层结果

7.4 消融实验

本小节探讨聚类和模型解耦对于联邦性能的独立影响。消融实验比较了以下四种情况：(1) FedCPMD，包含模型解耦和聚类机制，(2)仅包含聚类而无模型解耦的FedCPMD退化算法，记作Only_cluster，(3)仅包含模型解耦而无聚类的FedCPMD退化算法，Only_decouling，以及(4)无模型解耦且无聚类的原始FedAvg方法。所有实验均在异质性参数$\alpha=0.1$、客户端参与比例$\gamma=0.1$的设置下完成。为简化实验，本研究将分布距离计算指标设置为JS，其实验结果如图5所示。

观察图5可知，FedCPMD在9个数据集上的准

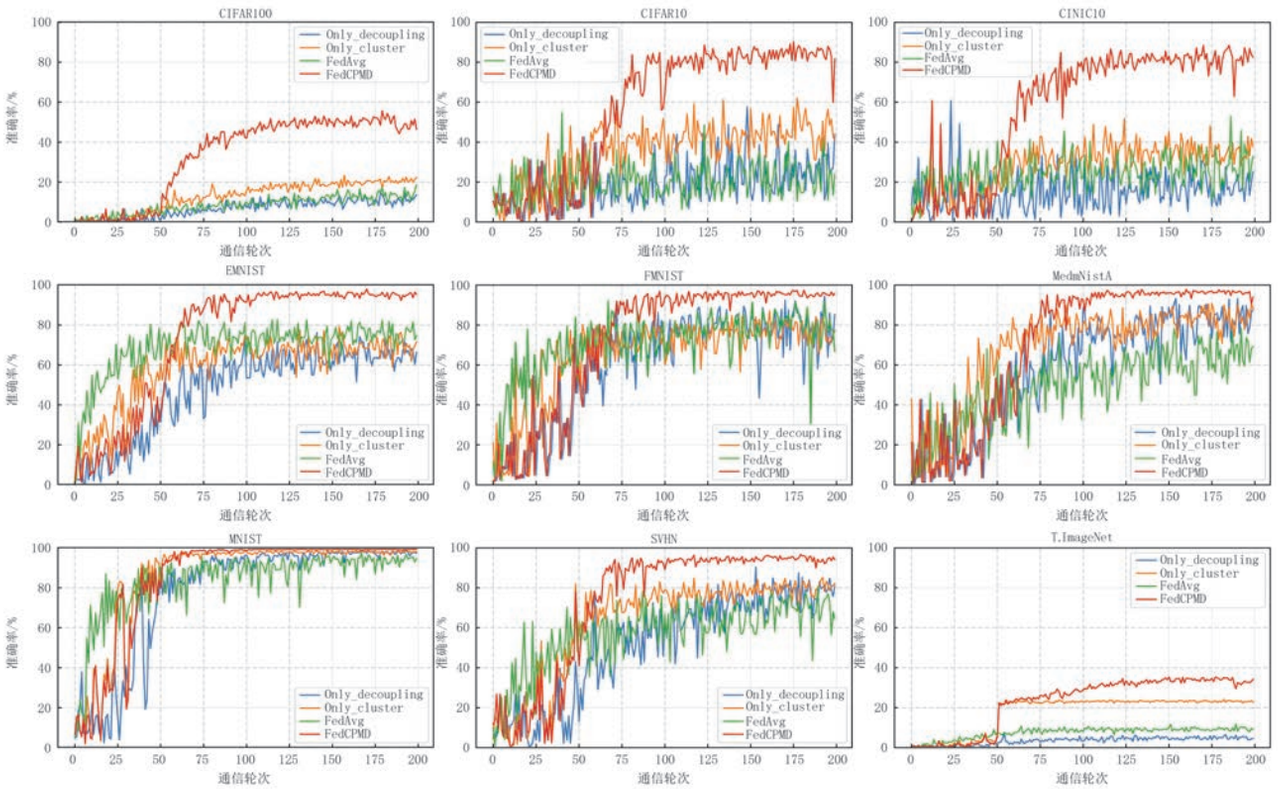

图5　FedAvg、FedCPMD及其两种退化算法在9个数据集上的准确率比较

确率显著优于其他三种算法，尤其在CIFAR100、CIFAR10、CINIC10和Tiny ImageNet四个复杂数据集上优势明显。将Only_cluster与基准算法FedAvg比较可以发现：对于大部分数据集如CIFAR100、MNIST和Tiny ImageNet等，聚类对联邦性能起促进作用。例如，对于CIFAR100，仅聚类不解耦的Only_cluster准确率比FedAvg提高近7%；在Tiny ImageNet上，其性能甚至表现出超10%的提升。

将Only_decoupling与基准算法FedAvg比较，可以发现：在大部分数据集上，如CIFAR10、CIFAR100、FMNIST、MedmNistA、MNIST和SVHN，两种算法效果基本相当或仅有小幅提升，这说明仅基于个性化层选择的模型解耦机制无法为联邦模型带来显著优势。尤其对于CINIC10、EMNIST和Tiny ImageNet数据集，仅采用模型解耦反而会对模型性能造成负向效应，导致Only_decoupling算法性能低于标准算法FedAvg。然而，当解耦与聚类机制结合使用时，即FedCPMD，其性能远超FedAvg。这表明，联邦学习模型性能的提升主要依赖于模型解耦与聚类机制协同作用。

7.5　可拓展性分析

本小节对所提算法FedCPMD在不同边缘客户端参与比例{0.1, 0.2, 0.3, 0.4}下进行性能评估。为简化实验，所涉及的超参数设计如下：$\alpha=0.1$，聚类准备阶段的通信轮次为60，距离指标为JS，其结果见表14，粗体和下划线分别表示最优和次优结果。本文还提供图6以进一步观察该算法的系统可拓展性。结合图表分析，FedCPMD在多个数据集上关于不同客户端参与比例的模型准确率变化趋势较为平稳。以MNIST为例，在$\gamma=0.1$时，其准确率为99.283%，随着参与比例增加，当$\gamma=0.2$和$\gamma=0.4$时，其准确率达到峰值99.294%，该数据集

表14　9个数据集关于不同客户端参与比例的准确率比较(%)

参与比例	0.1	0.2	0.3	0.4	Std.
CIFAR10	80.210	80.462	**87.087**	<u>85.850</u>	3.578
CIFAR100	48.788	**51.548**	<u>50.356</u>	48.868	1.320
CINIC10	83.025	84.255	**86.362**	<u>85.265</u>	1.424
EMNIST	**95.518**	<u>95.489</u>	94.059	95.189	0.686
FMNIST	95.004	**96.187**	<u>95.758</u>	95.321	0.519
MedmNistA	94.830	96.187	**97.214**	<u>97.134</u>	1.110
MNIST	<u>99.283</u>	**99.294**	99.264	**99.294**	0.014
SVHN	<u>94.419</u>	**95.557**	93.029	94.920	1.074
T. ImageNet	**33.445**	<u>27.567</u>	24.415	23.437	4.511

关于客户端参与比例的性能标准差仅为0.014。这说明FedCPMD对于客户端参与比例并不敏感。

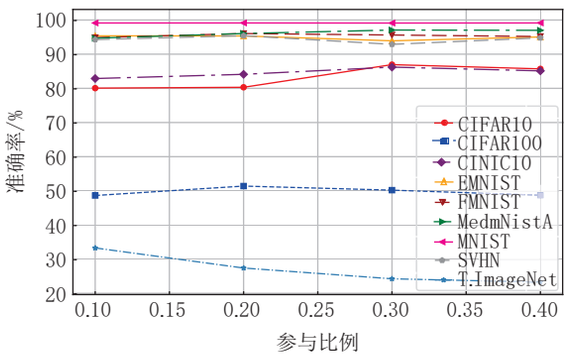

图6 FedCPMD关于不同客户端参与比例的准确率比较

7.6 关于聚类准备阶段的讨论

本小节主要探讨FedCPMD在聚类准备阶段中通信轮次对模型准确率的影响，数值结果见表15，粗体和下划线依次为最优和次优结果。实验所涉及的其他超参数设置如下：$\alpha=0.1$，客户端参与比例$\gamma=0.1$，距离指标为JS。表15表明，对于部分数据集，增加聚类准备阶段的通信轮次可以提升模型训练效果。例如，当聚类准备阶段的通信轮次为40时，CIFAR100的最终准确率仅为41.872%，随着通信轮次增加至80轮，CIFAR100的模型准确率达到54.382%。此外，FMNIST、MedmNistA和SVHN等数据集也呈现出相同趋势，它们在通信80轮时分别达到峰值95.269%、95.757%和94.870%。这表明，增加通信轮次有助于边缘客户端更准确地选择个性化层，从而提高聚类联邦学习的整体模型性能。

表15 9个数据集关于不同聚类准备轮次的分类准确率(%)

通信轮次	40	50	60	70	80
CIFAR10	76.264	76.915	**83.310**	77.191	<u>81.229</u>
CIFAR100	41.872	43.190	48.788	<u>50.147</u>	**54.382**
CINIC10	75.065	75.579	**83.025**	81.280	<u>81.520</u>
EMNIST	91.858	93.521	**95.518**	<u>95.220</u>	94.558
FMNIST	94.583	93.266	93.505	<u>94.813</u>	**95.269**
MedmNistA	93.210	94.240	<u>95.664</u>	94.327	**95.757**
MNIST	98.189	99.146	**99.283**	99.276	<u>99.280</u>
SVHN	90.230	92.574	94.419	<u>94.796</u>	**94.870**
T. ImageNet	27.229	**33.300**	<u>33.098</u>	32.698	23.956

然而，对于部分数据集，如CIFAR10、EMNIST和MNIST，FedCPMD在通信60轮时便分别达到最高准确率83.310%、95.518%和99.283%。对于Tiny ImageNet数据集，模型在仅50轮通信后便达到其峰值准确率33.300%。值得注意的是，当通信轮次为40轮时，模型在所有数据集上均未能取得最优性能，其原因在于此时通信轮次过少，无法保证每个边缘客户端至少被采样一次。

7.7 通信开销分析

已有大量研究致力于降低联邦学习的通信开销[18,20,40]。本小节通过理论实验比较了十二种方法的通信成本，并将结果可视化展示在图7中。图7描述了LeNet5架构下，不同方法的累计带宽消耗量随通信轮次增加的变化情况。结果显示，FedFomo通信开销最高，该方法允许每个客户端从其他客户端下载多个模型，频繁的模型交换导致通信开销显著增加。以FedAvg为比较基准，其他多数方法的通信开销与其相当甚至更低。例如，FedPer通过仅在边缘设备更新多层参数，避免将大量参数上传至云端聚合，有效降低了通信成本。FedCMD和FedCPMD通信开销最低，两者均选择其他全连接层而非分类器层作为个性化层，并将其保留在本地更新。由于其他全连接层的参数量比分类器层多，将其保留在本地更新后，上传云端服务器的参数量大幅下降，有效减少了带宽消耗。

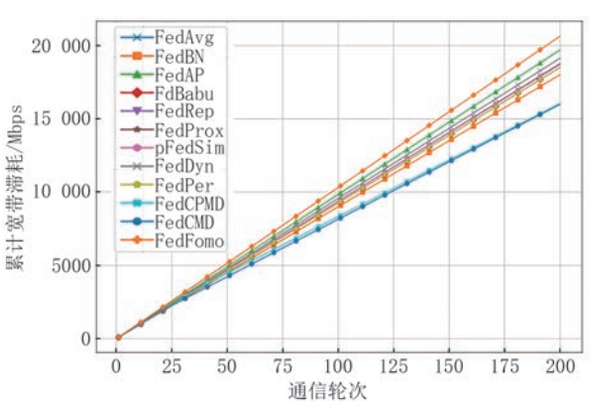

图7 LeNet5架构下12种算法的通信开销比较

8 结 论

本文提出了一种基于模型解耦的聚类联邦学习框架，FedCPMD，用于应对数据分布高度异构下的联邦学习问题。本研究引入了一种基于知识表征的个性化层选择方法，旨在准确评估各神经网络层与边缘客户端异构数据分布之间的特征对齐程度。FedCPMD选择与边缘客户端数据分布对齐程度最高的神经网络层作为该客户端的个性化层，并基于

该结果，将所有边缘客户端聚类为若干集群。每个集群内的客户端根据标准联邦学习框架进行全局聚合并更新参数，以提升模型的整体效率和性能。

FedCPMD采用两阶段结构设计，包括前期以个性化选层为主的聚类准备阶段和后期基于模型解耦的聚类联邦学习阶段。针对每个阶段，分别设计了聚类准备算法和聚类联邦学习算法。为验证所提方法的有效性，本文在九个真实数据集上进行了实验测试。结果表明，相比于其他十余种基线方案，FedCPMD算法在复杂分类任务上能够提供近4个百分点的性能提升(VGG11，$\alpha=0.1$)。

本文所提方法存在一定局限性，目前算法仍无法自适应选择最优距离度量指标。尽管实验结果表明，FedCPMD在4种距离度量指标下都能取得较好效果，但准确率性能依然存在差异。如何设计算法，使其能够自适应选择最优距离度量，提升其在异构数据分布环境下的适应性和鲁棒性，依然是未来亟待解决的重要问题。

致　谢　感谢国家自然科学基金项目、四川省自然科学基金项目和西南财经大学中央高校基本科研年度项目的资助。

参　考　文　献

[1] McMahan B., Moore E., Ramage D., et al. Communication-efficient learning of deep networks from decentralized data//Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Fort Lauderdale, USA, 2017: 1273-1282

[2] Lu Jia-Chen, Yao Jing-Han, Zhang Jun-Ge, et al. Soft: Softmax-free transformer with linear complexity//Proceedings of the Advances in Neural Information Processing Systems. Virtual, 2021(34): 21297-21309

[3] Zhang Jie, Guo Song, Guo Jing-Cai, et al. Towards data-independent knowledge transfer in model-heterogeneous federated learning. IEEE Transactions on Computers, 2023, 72(10): 2888-2901

[4] Chen Yi-Qiang, Lu Wang, Qin Xin, et al. Metafed: Federated learning among federations with cyclic knowledge distillation for personalized healthcare. IEEE Transactions on Neural Networks and Learning Systems, 2023, 35(11): 16671-16682

[5] Yao De-Zhong, Pan Wan-Ning, Dai Yu-Tong, et al. FedGKD: Towards heterogeneous federated learning via global knowledge distillation. IEEE Transactions on Computers, 2023, 73(1): 3-17

[6] Li Tian, Sahu A. K., Zaheer M., et al. Federated optimization in heterogeneous networks//Proceedings of the Machine Learning and Systems 2. Pasadena, USA, 2020(2): 429-450

[7] Dinh C. T., Tran N. Nguyen J. Personalized federated learning with moreau envelopes//Proceedings of the Advances in Neural Information Processing Systems 33. Novosibirsk, Russia, 2020(33): 21394-21405

[8] Li Tian, Hu Sheng-Yuan, Beirami A., et al. Ditto: Fair and robust federated learning through personalization//Proceedings of the International Conference on Machine Learning. Virtual, 2021: 6357-6368

[9] Sattler F., Müller K. R., Samek W. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. IEEE Transactions on Neural Networks and Learning Systems, 2021, 34(8): 3710-3722

[10] Long Guo-Dong, Xie Ming, Shen Tao, et al. Multi-center federated learning: Clients clustering for better personalization. World Wide Web, 2023, 26(1): 481-500

[11] Yang Lei, Huang Jia-Ming, Lin Wan-Yu, et al. Personalized federated learning on non-IID data via group-based meta-learning. ACM Transactions on Knowledge Discovery from Data, 2023, 17(4): 1-20

[12] Collins L., Hassani H., Mokhtari A., et al. Exploiting shared representations for personalized federated learning//Proceedings of the International Conference on Machine Learning. Virtual, 2021: 2089-2099

[13] Oh J., Kim S., Yun S. Y. Fedbabu: Towards enhanced representation for federated image classification//Proceedings of the International Conference on Learning Representations. Virtual, 2022

[14] Tan Jia-Hao, Zhou Yi-Peng, Liu Gang, et al. pFedSim: Similarity-aware model aggregation towards personalized federated learning. arXiv preprint arXiv:2305.15706, 2023

[15] Liang P. P., Liu T., Lin Zi-Yin, et al. Think locally, act globally: Federated learning with local and global representations. arXiv preprint arXiv:2001.01523, 2020

[16] Zhu Hang-Yu, Fan Yu-Xiang, Xie Zhen-Ping. Federated two stage decoupling with adaptive personalization layers. Complex & Intelligent Systems, 2024, 10: 3657-3671

[17] Su Rui-Zheng, Pang Xiong-Wen, Wang Hui. A novel parameter decoupling approach of personalized federated learning for image analysis. IET Computer Vision, 2023, 17(8): 913-924

[18] Chen Xing-Yan, Du Tian, Wang Mu, et al. Towards optimal customized architecture for heterogeneous federated learning with contrastive cloud-edge model decoupling. arXiv preprint arXiv:2403.02360, 2024

[19] He Chao-Yang, Ceyani E., Balasubramanian K., et al. Spreadgnn: Decentralized multi-task federated learning for graph neural networks on molecular data//Proceedings of the AAAI Conference on Artificial Intelligence. Vancouver, Canada, 2022, 36(6): 6865-6873

[20] Arivazhagan M. G., Aggarwal V., Singh A. K., et al. Federated learning with personalization layers. arXiv preprint arXiv:1912.00818, 2019

[21] Zhuang Wei-Ming, Wen Yong-Gang, Zhang Xue-Sen, et al. Performance optimization of federated person re-identification

via benchmark analysis//Proceedings of the 28th ACM International Conference on Multimedia. New York, USA, 2020: 955-963

[22] Li Xiao-Xiao, Jiang Mei-Rui, Zhang Xiao-Fei, et al. Fedbn: Federated learning on non-iid features via local batch normalization//Proceedings of the International Conference on Learning Representations. Vienna, Austria, 2021

[23] Lu Wang, Wang Jin-Dong, Chen Yi-Qiang, et al. Personalized federated learning with adaptive batchnorm for healthcare. IEEE Transactions on Big Data, 2022, 10(6): 915-945

[24] Duan M., Liu Duo, Ji Xin-Yuan, et al. FedGroup: Efficient federated learning via decomposed similarity-based clustering// Proceedings of the IEEE International Symposium on Parallel and Distributed Processing with Applications. New York, USA, 2021: 228-237

[25] Ghosh A., Chung J., Yin D., et al. An efficient framework for clustered federated learning//Proceedings of the 34th Annual Conference on Neural Information Processing Systems. Vancouver, Canada, 2020(33): 19586-19597

[26] Tang Xue-Yang, Guo Song, Guo Jing-Cai. Personalized federated learning with contextualized generalization//Proceedings of the 31st International Joint Conference on Artificial Intelligence. Vienna, Austria, 2022: 2241-2247

[27] MENÉNDEZ M. L., Pardo J. A., Pardo M. C. The Jensen-Shannon divergence. Journal of the Franklin Institute, 1997, 334(2): 307-318

[28] Tang Bo, He Hai-Bo. FSMJ: Feature selection with maximum Jensen-Shannon divergence for text categorization//Proceedings of the 12th World Congress on Intelligent Control and Automation (WCICA). Guilin, China, 2016: 3143-3148

[29] Ulger F., Yuksel S. E., Yilmaz A., et al. Fine-grained classification of solder joints with α-skew Jensen－Shannon divergence. IEEE Transactions on Components, Packaging and Manufacturing Technology, 2023, 13(2): 257-264

[30] Rüschendorf L. The Wasserstein distance and approximation theorems. Probability Theory and Related Fields, 1985, 70(1): 117-129

[31] She Qing-Shen, Chen Tie, Fang Feng, et al. Improved domain adaptation network based on Wasserstein distance for motor imagery EEG classification. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2023, 31: 1137-1148

[32] Lee H, Lu Jian-Feng, Tan Yi-Xin. Convergence of score-based generative modeling for general data distributions// Proceedings of the 34th International Conference on Algorithmic Learning Theory. Singapore, 2023(201): 946-985

[33] Schmid M., Welchowski T., N.Wright M., et al. Discrete-time survival forests with Hellinger distance decision trees. Data Mining and Knowledge Discovery, 2020, 34(3): 812-832

[34] Zeng Zi-Yue, Xiao Fu-Yuan. A generalized Hellinger distance for multisource information fusion and its application in pattern classification. Computational and Applied Mathematics, 2024, 43(1): 50

[35] Zhu Chao-Sheng, Xiao Fu-Yuan. A belief Hellinger distance for D-S evidence theory and its application in pattern recognition. Engineering Applications of Artificial Intelligence, 2021, 106: 104452

[36] Yin Zhu, Ma Xiao-Jian, Wang Hang. A new divergence based on the belief Bhattacharyya coefficient with an application in risk evaluation of aircraft turbine rotor blades. International Journal of Intelligent Systems, 2024, 1: 2140919

[37] Acar D. A. E., Zhao Yue, Matas R., et al. Federated learning based on dynamic regularization//Proceedings of the International Conference on Learning Representations. Vienna, Austria, 2021

[38] Zhang Michael, Sapra K., Fidler S., et al. Personalized federated learning with first order model optimization// Proceedings of the International Conference on Learning Representations. Addis Ababa, Ethiopia, 2020

[39] Simonyan K., Zisserman A. Very deep convolutional networks for large-scale image recognition//Proceedings of the International Conference on Learning Representations. San Diego, USA, 2015

[40] Zhou Huan, Li Ming-Ze, Sun Peng, et al. Accelerating federated learning via parameter selection and pre-synchronization in mobile edge-cloud networks. IEEE Transactions on Mobile Computing, 2024, 23(11): 10313-10328

附录Ⅰ LeNet5与VGG11模型架构

表16 LeNet5模型架构

组件	层设计
特征提取器	卷积层(输入通道数:3,输出通道数:6,卷积核大小=5,步幅:1,填充:0) 卷积层(输入通道数:3,输出通道数:16,卷积核大小=5,步幅:1,填充:0) 池化层(卷积核大小=2,步幅:2) 展平函数Flatten() 全连接层FC1(输出维度:120) 全连接层FC2(输出维度:84)
分类器	全连接层(输出维度:类别数)

表17 双全连接层LeNet5拓展模型架构

组件	层设计
特征提取器	卷积层(输入通道数:3,输出通道数:6,卷积核大小=5,步幅:1,填充:0) 卷积层(输入通道数:3,输出通道数:16,卷积核大小=5,步幅:1,填充:0) 池化层(卷积核大小=2,步幅:2) 展平函数Flatten() 全连接层FC1(输出维度:120)
分类器	全连接层(输出维度:类别数)

表18　VGG11模型架构

组件	层设计
特征提取器	卷积层(输入通道数:3,输出通道数:64,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:64,输出通道数:128,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:128,输出通道数:256,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:256,输出通道数:256,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:256,输出通道数:512,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:512,输出通道数:512,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:512,输出通道数:512,卷积核大小=3,步幅:1,填充:1)
	卷积层(输入通道数:512,输出通道数:512,卷积核大小=3,步幅:1,填充:1)
	池化层(卷积核大小=2,步幅:2)
	展平函数Flatten()
	全连接层FC1(输出维度:4096)
分类器	全连接层FC2(输出维度:4096)
	全连接层(输出维度:类别数)

附录Ⅱ　收敛性分析

(1) 不同方案在LeNet5上的收敛性比较

图8展示了FedCPMD与其他十三种算法在LeNet5架构下关于九个数据集的收敛性比较。每个子图表示$\alpha=0.1$时模型进行200轮通信的分类准确率变化情况,各算法的准确率曲线反映了相应的收敛趋势。图8显示,尽管FedCPMD在CIFAR10、CIFAR100和CINIC10等数据集上前期收敛速度较慢,但后期能迅速提升并稳定到较高的准确率水平。其原因在于,FedCPMD前期处于聚类准备阶段,一旦进入聚类联邦学习阶段,FedCPMD能准确识别并聚类相似异质性的边缘客户端到相同集群中,使得模型更新趋于一致,从而快速提升分类准确率。

图9展示了$\alpha=0.5$时各算法关于9个数据集的收敛情况。随着数据异质性水平降低,大部分算法的收敛性有所改善。对于MNIST,相较于$\alpha=0.1$,CFL、FedBN和pFedSim等算法的收敛曲线波动程度明显降低。在FMNIST、MNIST等数据集上,几乎所有算法都能在前75轮通信内达到较为稳定的高准确率。对于复杂数据集,如CIFAR100、Tiny ImageNet,各算法的收敛速度普遍较慢,且准确率曲线波动较大。这表明数据集的复杂性会影响模型训练过程,增加模型在学习过程中的不确定性。

图10则显示了$\alpha=1.0$时不同方案在九个数据集上的收敛情况,此时数据异质性降到最低,各客户端数据均匀分布。分析图10可知,FedCPMD在CIFAR100、CINIC10和Tiny ImageNet等复杂数据集上表现出色。FedCMD虽收敛速度较快,但最终结果劣于FedCPMD,这表明基于个性化层的聚类机制在处理复杂数据集时具有优势。分析图8~图10,随着数据异质性逐渐降低,各算法在简单数据集上的收敛性明显改善,表现为收敛曲线的波动显著下降;然而,对于复杂数据集,尽管部分算法的表现有所提升,但整体收敛性仍不理想,这说明数据集的复杂程度对模型的稳定训练有显著影响。

(2) 不同方案在VGG11上的收敛性比较

图11展示了在VGG11架构下,$\alpha=0.1$时FedCPMD与其他十一种联邦方案关于9个不同数据集的收敛情况,此时客户端的数据异质性最大。每个子图的横轴表示联邦学习的通信轮次,纵轴表示每轮学习所取得的准确率。观察图11可知,对于简单数据集,如MNIST,大多数算法收敛较快,通常在前25轮内即可达到接近95%的准确率。然而,在CIFAR100等复杂数据集上,多数算法的收敛速度明显较慢,收敛曲线波动较大,且不同算法之间的最终准确率差异显著。这表明,复杂数据集的高度异构性加剧了联邦学习模型的收敛难度,导致模型难以在较少的通信轮次内实现稳定收敛。

图12展示了$\alpha=0.5$时各算法在九个不同数据集上的收敛表现。分析图12可知,FedCPMD和FedCMD在复杂数据集上表现出色,最终准确率往往优于其他算法。FedPer和pFedSim在中等复杂数据集,如CIFAR10上收敛速度较快且最终准确率较高,但表现仍次于FedCPMD。值得注意的是,尽管FedCPMD前期收敛速度较慢,但后期收敛曲线更加平稳,且准确率高于其他算法,具有明显的后期优势。这表明,FedCPMD能够更好地适应不同客户端的数据分布,通过个性化聚类机制,能够在后

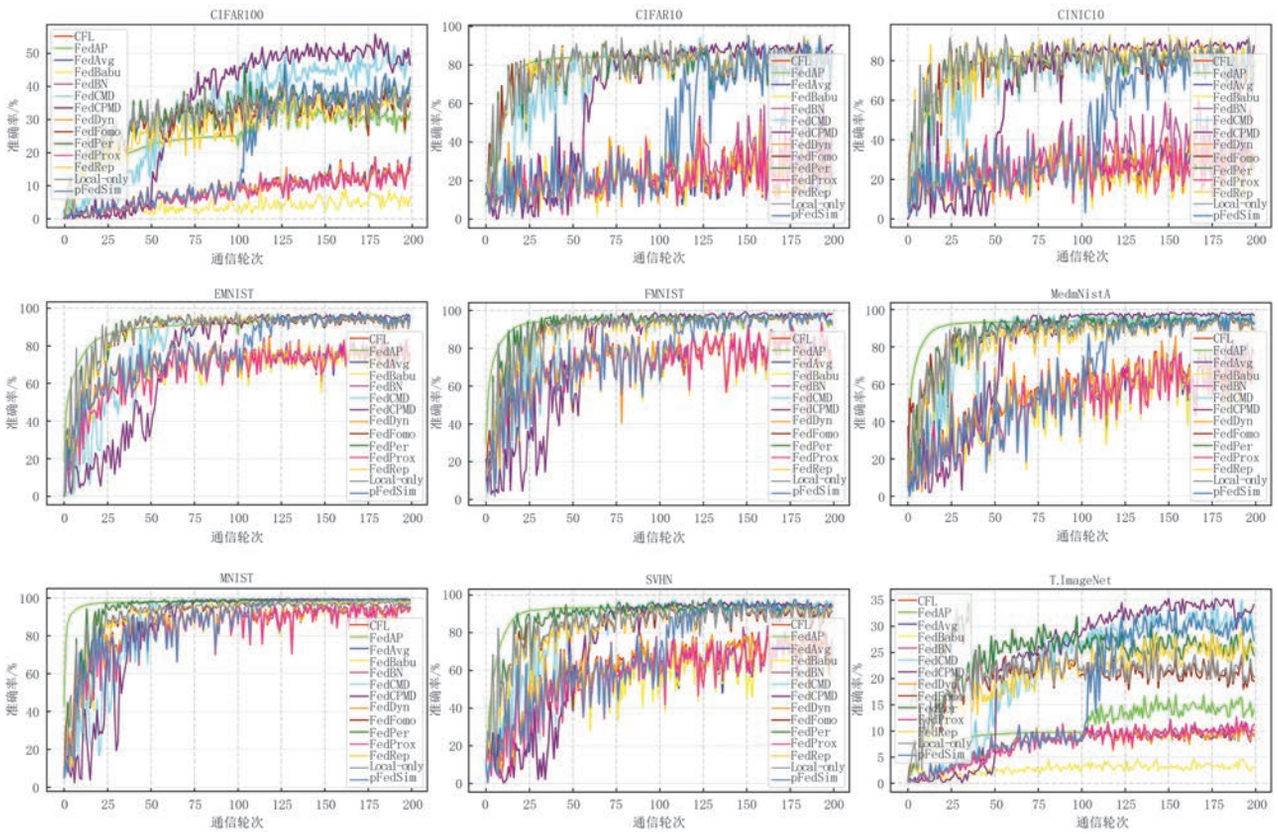

图8 $\alpha=0.1$时，FedCPMD在LeNet5架构上与其他13种方案的收敛性比较

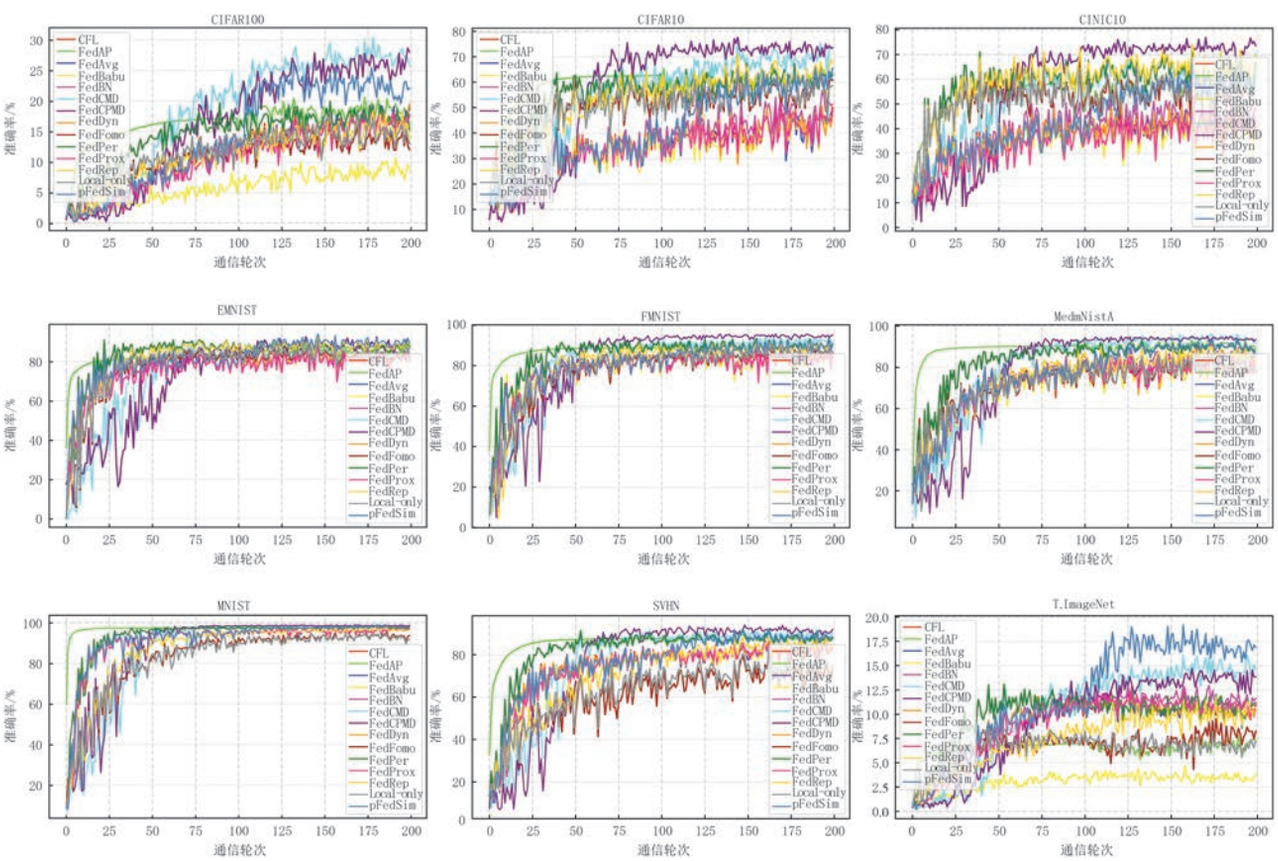

图9 $\alpha=0.5$时，FedCPMD在LeNet5架构上与其他13种方案的收敛性比较

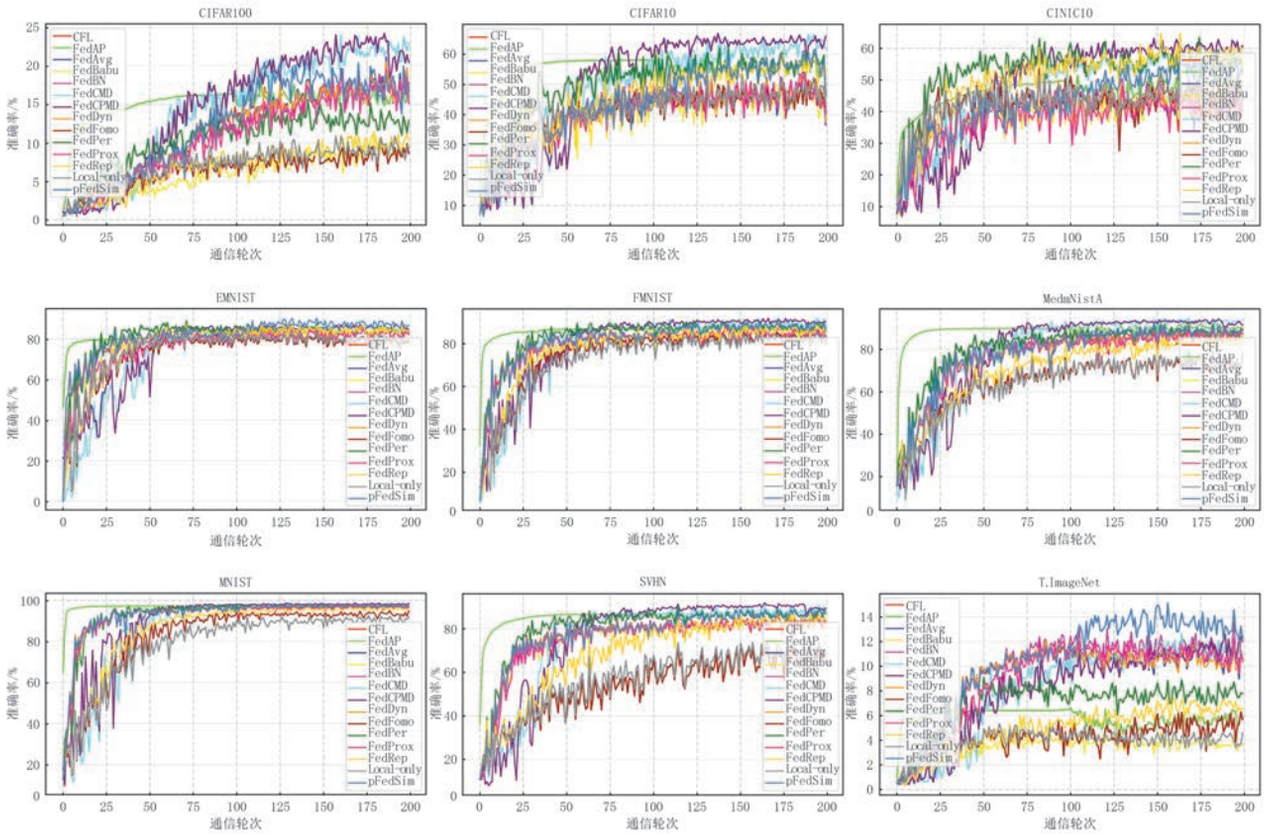

图10　$\alpha=1.0$时，FedCPMD在LeNet5架构上与其他13种方案的收敛性比较

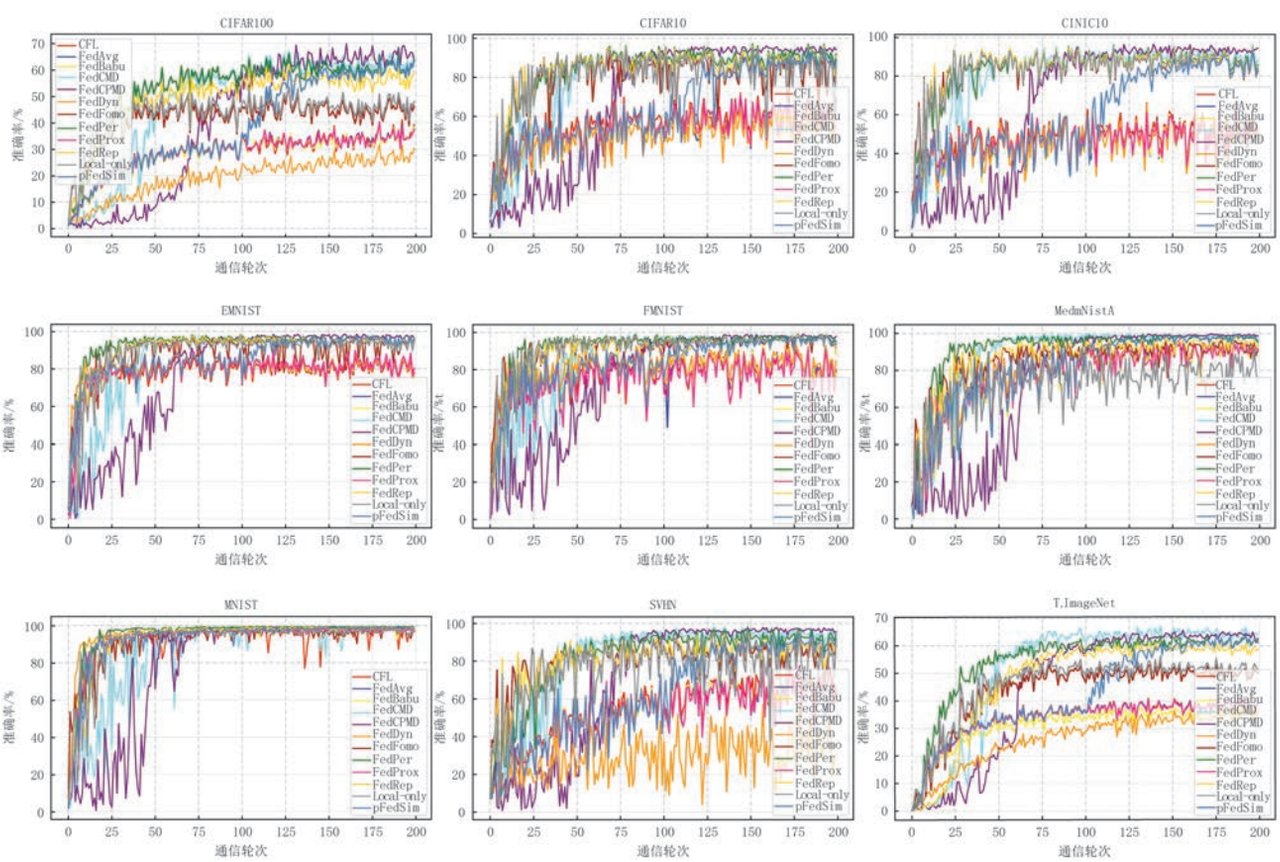

图11　$\alpha=0.1$时，FedCPMD在VGG11架构上与其他11种方案的收敛性比较

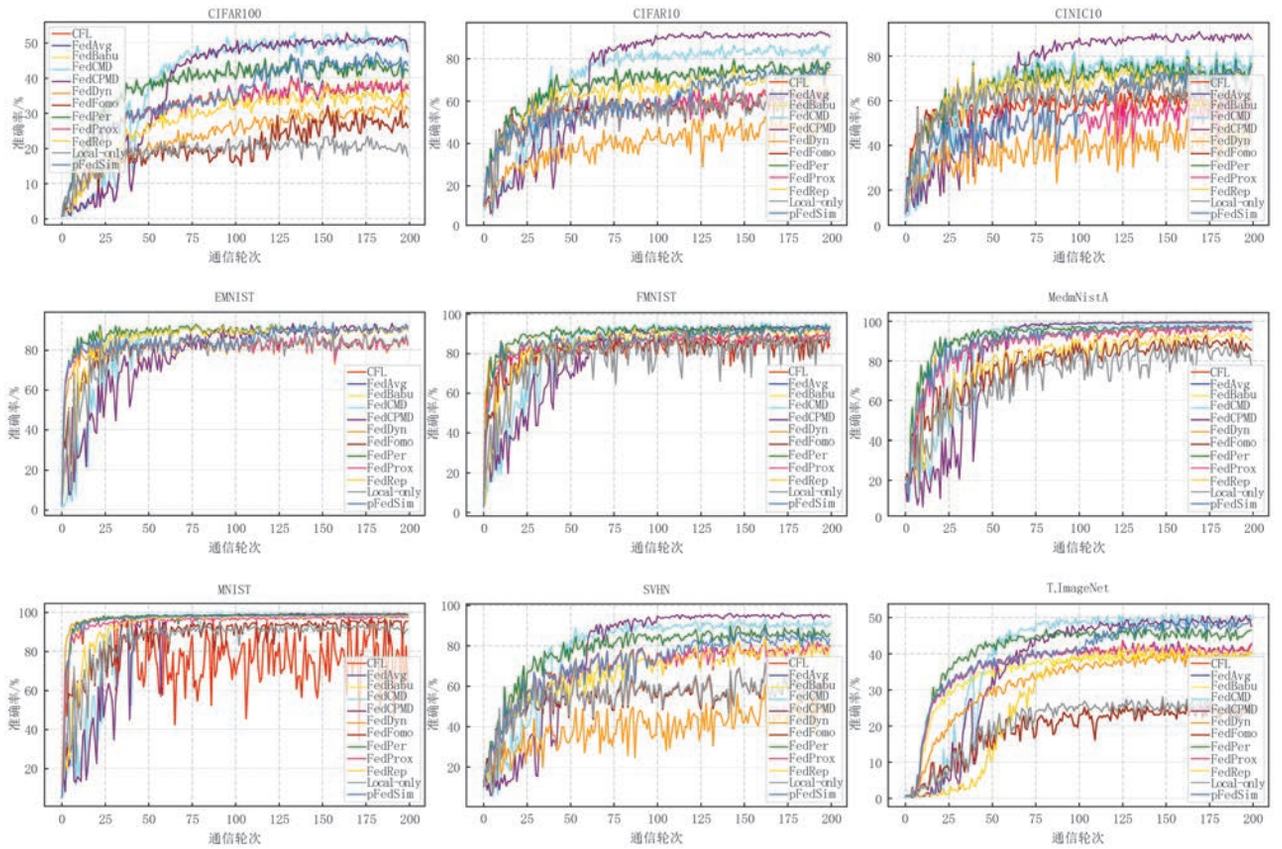

图12 $\alpha=0.5$时，FedCPMD在VGG11架构上与其他11种方案的收敛性比较

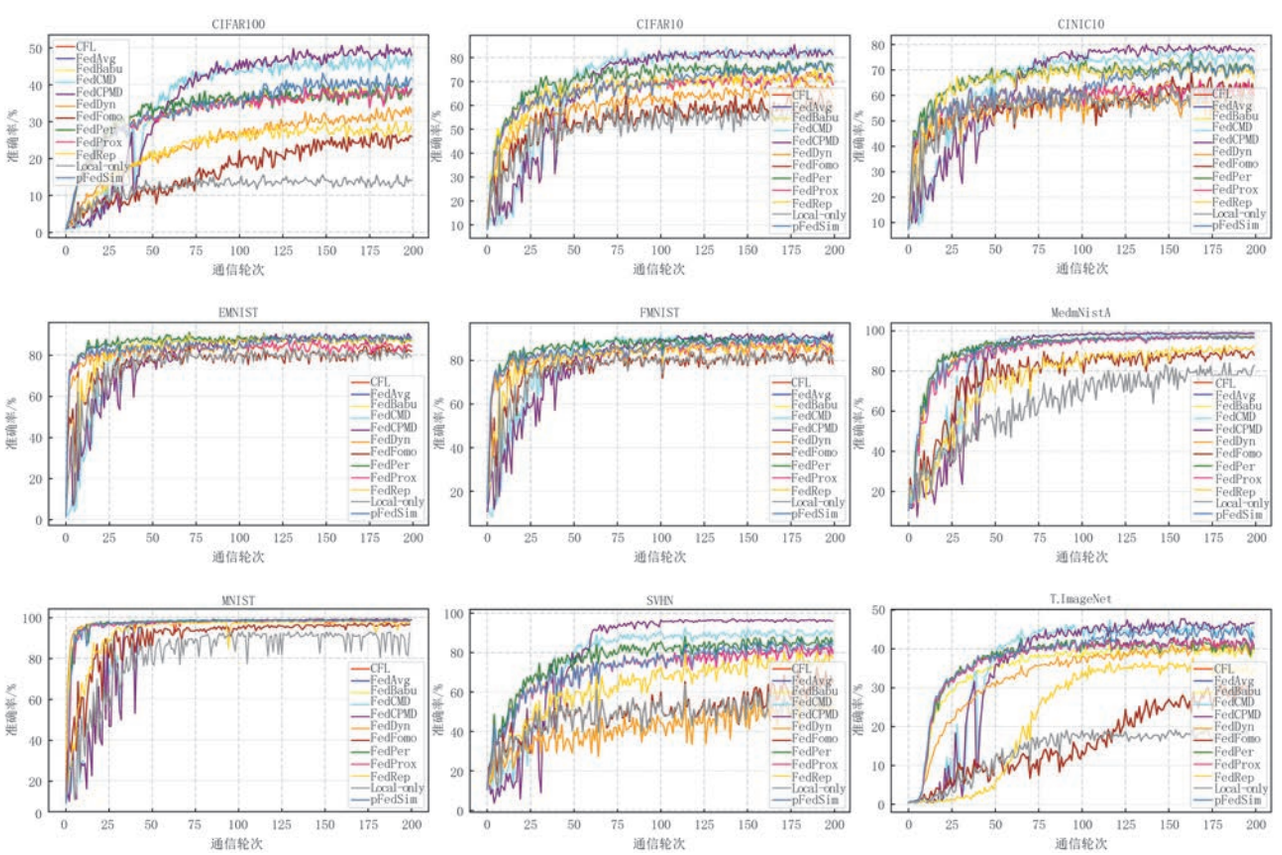

图13 $\alpha=1.0$时，FedCPMD在VGG11架构上与其他11种方案的收敛性比较

期有效聚类相似异质性的客户端，以提升模型准确率。

图13则显示了$\alpha=1.0$时12种方案在9个不同数据集上的收敛情况。对于简单数据集，如MNIST，大多数算法能够快速收敛，并在较少的通信轮次内达到稳定的表现。对于中等复杂数据集，如CIFAR10和CINIC10，基于模型解耦的FedPer、pFedSim和FedCMD等算法也能够较快收敛，但最终准确率低于FedCPMD。这表明，模型解耦有助于处理较为复杂的数据集，但将其结合聚类机制的效果更佳。然而，对于复杂数据集，如CIFAR100、CINIC10和Tiny ImageNet，所有算法的收敛速度和准确率均有所下降，这反映出数据集的复杂性对模型训练有负面影响。结合图11和图12分析可知，随着数据异质性降低，各算法的收敛速度普遍加快，准确率波动显著下降。这进一步表明，数据异质性对联邦学习的收敛性具有显著影响。数据异质性越高，算法实现模型稳定所需的通信轮次越多。

DU Tian，Ph. D. candidate. Her main research interests include heterogeneous federated learning and time series clustering.

CHEN Xing-Yan，Ph. D.，associate professor. His research interests include computing-network convergence and distributed learning.

KOU Gang，Ph. D.，Changjiang scholar distinguished professor. His research interests include big data and financial intelligence，data science and intelligent decision-making，business intelligence，information systems.

ZHAO Yu，Ph. D.，professor. His main research interests include data mining，natural language processing，graph learning and machine learning.

XU Chang-Qiao，Ph. D.，professor. His research interests include future internet technology，multimedia communications，network security and artificial intelligence.

Background

Federated Learning（FL）seeks to enable the collaborative training of a global model across multiple edge clients while preserving data privacy. However，the heterogeneous nature of data distribution across edge clients introduces two significant challenges：suboptimal model training efficiency and heightened communication costs. Existing strategies to address these challenges typically involve techniques such as knowledge distillation，the incorporation of additional loss functions，similarity aggregation，and model decoupling. However，these approaches are not without limitations. For example，knowledge distillation often depends on auxiliary public datasets，the design of refined loss functions can increase computational complexity，and certain similarity aggregation methods based on client features can heighten the risk of privacy leakage. This has prompted us to develop an efficient clustering-based federated learning framework，FedCPMD，which leverages the feature representations of each layer of client models to identify data heterogeneity and achieve personalized layer selection. Based on the layer selection results，the framework further clusters clients to mitigate the inefficiency in model training caused by data heterogeneity，providing a novel solution for heterogeneous federated learning.

In this paper，we first propose a heterogeneity layer identification metric based on feature distribution transfer distance. This metric is applicable to common distribution distance functions，allowing for precise matching of the most heterogeneous neural network layers，which are then selected as personalized layers for clustering preparation. Next，clients with the same personalized layers are grouped into clusters，and model decoupling-based federated learning is conducted for each cluster. This effectively avoids the inefficiency in model training and increased communication rounds caused by data heterogeneity. Extensive tests on nine real-world datasets show that，compared with ten existing state-of-the-art methods，FedCPMDachieves an average accuracy improvement of 2.450%（$\alpha=0.1$）on the LeNet5 architecture and 3.963%（$\alpha=0.1$）on the VGG11 architecture on complex datasets such as CIFAR100，CINIC10，SVHN and Tiny ImageNet.

This research was supported by the National Natural Science Foundation of China Youth Program（62302400），National

Natural Science Foundation of China General Program (62376227), National Science Fund for Distinguished Young Scholars (62225105); Sichuan Provincial Natural Science Foundation Youth Program (2023NSFSC0114), Sichuan Provincial Natural Science Foundation Key Program (2023 NSFSC5094); Fundamental Research Funds for the Central Universities (JBK2406080). These projects aim to provide better performance for computer networks and information systems, with this paper offering theoretical support. Our research group has been dedicated to the study of distributed computer networks and has published numerous excellent papers in prominent international conferences and journals such as IEEE INFOCOM and IEEE Transactions on Mobile Computing.