Modeling and Simulation 建模与仿真, 2025, 14(4), 1200-1211
Published Online April 2025 in Hans. https://www.hanspub.org/journal/mos
https://doi.org/10.12677/mos.2025.144366

Hans汉斯

基于欧几里得距离的客户端选择联邦学习方案

邬淑敏，刘　亚*

上海理工大学光电信息与计算机工程学院，上海

收稿日期：2025年3月30日；录用日期：2025年4月23日；发布日期：2025年4月30日

摘　要

联邦学习(Federated Learning)作为一种先进的分布式机器学习框架，它允许多个参与者在各自数据不出本地的前提下，协同开展模型训练工作，保护数据隐私。然而，传统联邦学习在实际应用中存在数据异质性影响联邦学习系统的效率和收敛性，限制其广泛应用。针对上述问题，本文提出了ECS-FL (Euclidean-Based Client Selection in Federated Learning)框架。该框架融入基于欧几里得距离的客户端选择机制，通过依据客户端模型更新与全局模型的相似度来筛选客户端，有效应对了客户端差异与数据异质性带来的挑战，显著降低了模型训练过程中的偏差，大幅提升了模型的鲁棒性。在独立同分布和非独立同分布数据分布环境下开展的大量实验结果表明，ECS-FL框架能够切实有效地改善模型的收敛性、鲁棒性与准确性。

关键词

联邦学习，安全聚合，客户端选择，隐私保护机器学习

Euclidean-Based Client Selection in Federated Learning

Shumin Wu, Ya Liu*

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai

Received: Mar. 30th, 2025; accepted: Apr. 23rd, 2025; published: Apr. 30th, 2025

Abstract

Federated Learning, as an advanced distributed machine learning framework, allows multiple participants to collaboratively carry out model training work while keeping their respective data local, thus protecting data privacy. However, traditional federated learning has the problem in practical

*通讯作者。

文章引用: 邬淑敏, 刘亚. 基于欧几里得距离的客户端选择联邦学习方案[J]. 建模与仿真, 2025, 14(4): 1200-1211.
DOI: 10.12677/mos.2025.144366

applications that data heterogeneity affects the efficiency and convergence of the federated learning system, restricting its widespread application. In response to the above issues, this paper proposes the ECS-FL (Euclidean-Based Client Selection in Federated Learning) framework. This framework incorporates a client selection mechanism based on Euclidean distance. By screening clients according to the similarity between the client model updates and the global model, it effectively addresses the challenges posed by client differences and data heterogeneity, significantly reducing the bias during the model training process and greatly enhancing the robustness of the model. A large number of experimental results conducted in both independent and identically distributed and non-independent and identically distributed data distribution environments show that the ECS-FL framework can effectively improve the convergence, robustness, and accuracy of the model.

Keywords

Federated Learning, Secure Aggregation, Client Selection, Privacy-Preserving Machine Learning

1. 引言

在当前数据呈爆发式增长的时代，机器学习在数据挖掘中发挥着至关重要的作用，并广泛应用于众多领域[1]。然而，传统的集中式机器学习范式存在显著缺陷。它需要将大量原始数据集集中存储在单个中央服务器上，这无疑给用户数据隐私泄露带来了严重风险[2]。特别是在诸如医疗诊断等敏感领域，数据可能包含极其敏感的个人信息，这些信息严禁与第三方直接共享[3]。

为了应对这些挑战，联邦学习作为一种有前景的去中心化学习范式应运而生，并逐渐引起广泛关[4]注[5]。在联邦学习的众多算法中，FedAvg [6]是一种典型且被广泛使用的算法。在联邦学习框架中，多个地理位置分散的参与者可以使用自身的数据在本地训练机器学习模型，而无需将原始数据传输到中央服务器，从而有效降低数据隐私泄露的风险。具体而言，在每一轮训练中，聚合服务器将全局模型分发给选定的参与者。这些参与者随后基于本地数据训练本地模型，并将模型更新上传至聚合服务器。随后，服务器将这些更新融合，以获得全局更新后的模型[7] [8]。

通过确保敏感数据保留在本地设备上，联邦学习解决了隐私问题并符合数据主权法规，使其特别适用于医疗保健[9]和金融[10]等对隐私敏感的领域。尽管具有这些优势，联邦学习在边缘计算环境中面临关键挑战－数据异质性。在边缘计算中，客户端之间的数据往往是非独立同分布(非 IID)的，这进一步加剧了模型聚合的挑战。这种数据异质性会给全局模型带来显著偏差，使其难以实现快速稳定的收敛[11] [12]。在这种情况下，像 FedAvg 这样的传统联邦学习方法，由于其对客户端数据分布相似性的假设，可能表现不佳。FedAvg 通过对每个客户端的参数求平均值来聚合模型更新，在客户端数据为 IID 的场景中效果良好，但在客户端数据分布差异很大的非 IID 环境中表现不佳[13]。这种局限性导致了 FedProx [14]的引入，它在每个客户端的目标函数中添加了一个近邻项。该项有助于惩罚本地更新中的较大偏差，从而提高非 IID 环境中的收敛性和鲁棒性。

鉴于这些挑战，我们提出了 ECS-FL 框架，该框架引入了一种基于欧几里得距离的客户端选择机制，该机制根据客户端本地模型更新与全局模型的相似度来选择客户端，有效应对客户端数据异质性挑战。

2. 相关工作

联邦学习作为一种分布式机器学习范式，在保护数据隐私的同时，实现了多客户端协同训练全局模

型的目标。为提高 FL 模型的性能和收敛速度，研究者提出了多种客户端选择策略，包括随机选择、基于性能的选择、基于数据代表性的选择。

随机选择是一种简单且应用广泛的客户端选择策略。在每轮训练中随机选取部分客户端参与，可保证参与的公平性，同时有效降低计算复杂度与通信开销。然而，这种策略的简单性也带来了潜在的低效问题，尤其是在异构环境中，随机选择可能引入数据不具代表性或质量较低的客户端，从而对全局模型的准确性和泛化能力产生负面影响[15] [16]。在诸如边缘计算等资源受限的场景中，这种低效利用计算资源的现象尤为明显[17]。

基于性能的选择旨在通过优先选择计算能力强、通信效率高或训练表现优异的客户端来优化训练过程[18] [19]。这种方法能够显著加快全局模型的收敛速度，但也存在过度偏向特定数据分布的问题，从而可能损害全局模型的公平性和泛化能力。基于数据代表性的选择通过优先选取局部数据分布最能代表全局分布的客户端，提高了全局模型的鲁棒性和准确性[20]。然而，该方法通常依赖于对数据分布的先验知识，这在实际场景中可能难以实现。此外，数据分析过程中产生的大量通信开销限制了该策略在大规模系统中的可扩展性。

尽管这些方法在不同场景中取得了一定进展，但也存在固有权衡。随机选择方法虽然简单且公平，但在异构环境中效率较低；基于性能的选择加速了收敛过程，却可能引入偏差。基于数据代表性的选择提高了模型的鲁棒性，但成本高昂。

3. 方案设计与实现

本节介绍了在边缘计算场景下 ECS-FL 方案的整体架构，基于欧几里得距离的客户端选择和模型聚合、客户端更新机制并详细介绍了其实现的算法步骤。

3.1. ESC-FL 方案架构概述

ECS-FL 框架中基于欧几里得距离的客户端选择策略，从本质上是对模型相似性的一种度量与筛选。当计算本地模型更新与全局模型之间的欧几里得距离时，较小的距离意味着本地模型更新与全局模型在参数空间中的差异较小，即本地模型的更新方向和幅度与全局模型的优化方向更为契合。这使得在模型聚合过程中，选择距离较小的客户端进行更新，能够最大程度地减少因客户端数据异质性带来的干扰，保证全局模型更新的稳定性和有效性。从数据分布的角度来看，在非独立同分布的数据环境下，不同客户端的数据分布存在显著差异。基于欧几里得距离的选择机制能够优先挑选出数据分布特征与全局模型更匹配的客户端。这些客户端的本地模型更新包含了对全局模型有价值的信息，有助于全局模型更好地捕捉数据的共性特征，增强模型对不同数据分布的适应能力，进而提高模型的鲁棒性和准确性。

本文所提出的 ESC-FL 框架由三个主要组件构成：中央服务器、边缘服务器和边缘设备(本地客户端)，在图 1 中有所展示。每个组件的主要功能概述如下：

中央服务器：中央服务器负责协调所有客户端的模型训练和聚合过程。它执行全局模型的更新和优化，确保全局模型的收敛。

边缘服务器：边缘服务器负责接收客户端更新，聚合模型更新，并计算欧几里得距离以选择最相关的客户端。然后，它将聚合后的模型发送到中央服务器。

边缘设备(本地客户端)：每个边缘设备在其各自的设备上执行本地训练，计算本地模型更新，并基于欧几里得距离的客户端选择机制参与全局模型的聚合。

图 2 展示了该框架内的交互情况。该框架基于本地模型和全局模型之间的相似性协作选择相关客户端。

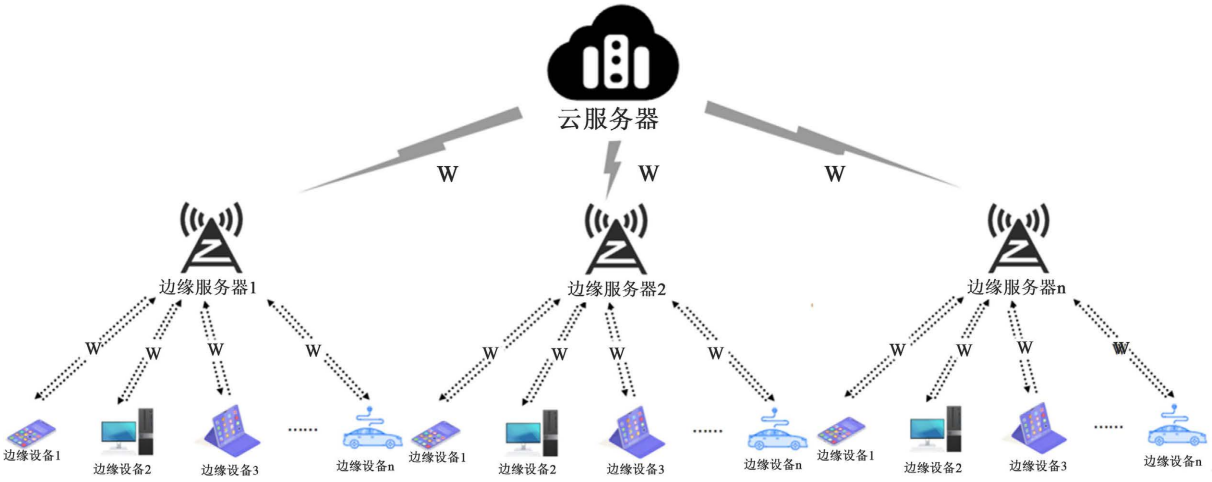


Figure 1. Federated learning training model of devices in the edge computing scenario
图 1. 边缘计算场景下设备的联邦学习训练模型

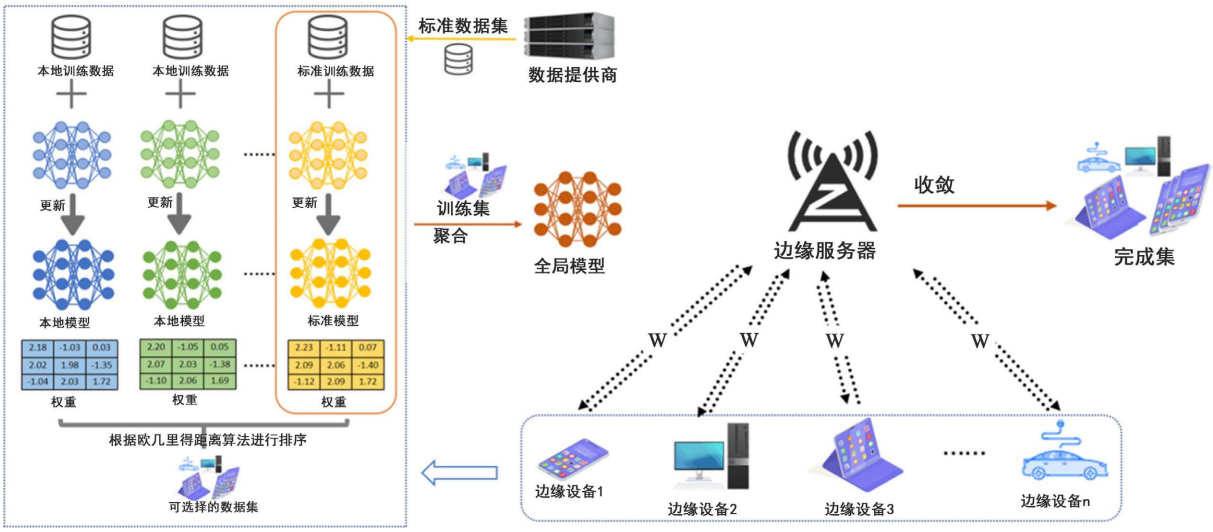


Figure 2. Overview of the ECS-FL framework
图 2. ECS-FL 框架概述

3.2. 基于欧几里得距离的客户端选择算法

欧几里得距离被广泛用于度量数据点之间的相似性，并且它为评估本地模型更新与全局模型的一致性提供了一种有效的方式。在联邦学习中，欧几里得距离用于量化本地模型和全局模型之间的差异，这对于模型聚合过程至关重要。在一个 n 维空间中，两个向量 $\boldsymbol{x}$ 和 $\boldsymbol{y}$ 之间的欧几里得距离定义为：

$$E(\boldsymbol{x},\boldsymbol{y})=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{1}$$

这里，$\boldsymbol{x}=(x_1,x_2,\cdots,x_n)$ 和 $\boldsymbol{y}=(y_1,y_2,\cdots,y_n)$ 表示 n 维空间中的两个点。该度量计算两点之间的直线距离，并且被广泛用于度量模型或数据点之间的相似性。在联邦学习中，此距离用于确定本地模型与全局模型的相似程度。较小的欧几里得距离意味着本地模型更新与全局模型更加一致，因此在聚合过程中应具有更高的权重。相反，与全局模型距离较远的本地模型对模型更新的贡献较小。

基于欧几里得距离的算法 1 可以使用以下伪代码实现。关键步骤包括展平模型、计算欧几里得距离以及对本地模型更新进行加权聚合。

算法 1. 基于客户端选择的欧几里得距离计算

输入：全局模型集合 *WSet*
输出：经过排序后的客户端集合 *T*

展平标准模型 $M_n^0 = \left[x_n^1, x_n^2, \cdots, x_n^t\right]$
  For 每个权重 $(i = 1, 2, 3, \cdots)$ 来自 ***WSet***
    展平本地模型 $M_n^k = \left[x_n^{k1}, x_n^{k2}, \cdots, x_n^{kt}\right]$
    计算欧几里得距离 $d_t = \left\|M_n^0 - M_n^k\right\|$
    对 $\boldsymbol{ED}_{i,j}$ 按 d_t 排序
加入 $T \leftarrow$ 按照排序选择 $\boldsymbol{ED}_{i,j}$

1) 模型初始化和通信过程

在联邦学习中，训练过程涉及参与者和中央服务器之间的多轮通信。在每一轮中，全局模型在所有 k 个参与者之间共享，参与者使用当前全局模型作为初始化，在他们的私有数据上训练本地模型。一旦训练完成，本地模型被发送回中央服务器，在那里全局模型基于聚合过程进行更新。令 M_k^n 表示第 n 轮中第 k 个参与者的本地模型，并且 M_n^0 表示全局模型。

2) 展平本地和全局模型

为了便于聚合过程，本地和全局模型都被展平为一维向量。此展平步骤允许直接比较本地和全局模型的权重。令第\n\轮中展平后的全局模型 M_n^0 和第 k 个参与者的本地模型 M_k^n 表示为：

$$M_n^0 = \left[x_n^1, x_n^2, \cdots, x_n^t\right] \tag{2}$$

$$M_k^n = \left[x_k^1, x_k^2, \cdots, x_k^t\right] \tag{3}$$

其中 t 是每个模型中的权重数量。展平之后，每个模型的权重都表示为一维数组，从而允许直接计算本地和全局模型之间的欧几里得距离。

3) 欧几里得距离计算

在中央服务器处，计算每个本地模型和全局模型之间的欧几里得距离以评估它们的相似性。对于每个本地模型 M_k^n 和全局模型 M_n^0，欧几里得距离计算如下：

$$d_t = \left\|M_n^0 - M_k^n\right\| \tag{4}$$

这里，d_t 表示第 k 个参与者的本地模型和全局模型之间的欧几里得距离。较小的距离表明本地模型更接近全局模型，意味着本地模型的更新与全局模型更加一致。

4) 本地模型更新的加权平均

一旦为每个本地模型计算了欧几里得距离，模型更新将根据它们与全局模型的距离进行加权。距离较小的本地模型被赋予更高的权重，因为它们与全局模型更相似。本地模型更新的加权方式如下：

$$M_k^n = \frac{1}{d_t} M_k^n \tag{5}$$

此公式表明本地模型更新与欧几里得距离成反比。更接近全局模型(即距离较小)的本地模型对全局模型更新具有更大的影响。

5) 全局模型聚合

聚合过程的最后一步涉及通过组合所有本地模型更新(根据它们各自的欧几里得距离加权)来更新全局模型。更新后的全局模型计算如下：

$$M_n^0 = \frac{\sum_{k=1}^{K} \frac{1}{d_t} M_k^n}{\sum_{k=1}^{K} \frac{1}{d_t}} \tag{6}$$

在该等式中，M_n^0 表示聚合后的更新后的全局模型。本地模型根据它们的欧几里得距离进行加权，以便更相似的模型对全局模型更新贡献更大更准确。

3.3. 客户端更新机制

算法 2 概述了每一轮训练中联邦学习的执行过程。以下部分将对算法的每个部分进行逐步解释。

步骤 1：算法的第一步涉及边缘服务器初始化全局模型 w_0 并将其广播给所有参与的客户端。全局模型 w_0 表示模型的初始状态，它可以随机初始化或使用预训练的模型。广播 w_0 确保所有客户端从相同的模型开始，从而在所有客户端的训练过程中保持一致性。

步骤 2：每个客户端 k 在其私有数据集上并行进行本地训练，使用全局模型 $W(t-1)$ 作为初始化。客户端通过对其本地数据应用梯度下降来更新其本地模型以最小化损失函数。本地模型的更新规则如下：

$$W_k(t) = W(t-1) - \eta \nabla l\left(W_k(t-1); b\right) \tag{7}$$

在该等式中，$W_k(t)$ 表示第 t 轮中客户端 k 的更新后的本地模型，$W(t-1)$ 表示上一轮的全局模型。学习率用 η 表示，而 $\nabla l\left(W_k(t-1); b\right)$ 是关于在小批量 b 上计算的模型参数 $W_k(t-1)$ 的损失函数 l 的梯度。梯度下降步骤允许每个客户端根据其本地数据更新其模型，并且该过程会重复进行。每个客户端通过将其本地模型与全局模型进行比较来检查其本地模型是否已收敛。如果本地模型 $W_k(t-1)$ 和全局模型 $W(t-1)$ 之间的差异小于预定义的阈值 ε，客户端将终止进一步的训练。收敛条件如下：

$$\left\| W_k(t-1) - W(t-1) \right\| < \varepsilon \tag{8}$$

如果本地和全局模型之间的差异小于 ε，客户端认为模型已充分收敛并停止本地训练过程。否则，客户端将继续其训练直至达到收敛。

步骤 3：在本地模型更新之后，边缘服务器计算每个客户端的本地模型和全局模型 $W(t-1)$ 之间的欧几里得距离。欧几里得距离的计算如下：

$$d_k = \left\| W_k(t) - W(t-1) \right\| \tag{9}$$

在该公式中，d_k 表示客户端 k 的本地模型 $W_k(t)$ 和全局模型 $W(t-1)$ 之间的欧几里得距离，它反映了两个模型之间的差异。$\left\| W_k(t) - W(t-1) \right\|$ 表示本地和全局模型之间差异的 $L2$ 范数，提供了它们一致性的度量。d_k 的较小值表明本地模型与全局模型更加一致。边缘服务器根据这些距离对客户端进行排序，并选择那些模型与全局模型最相似的客户端用于下一个聚合步骤。

步骤 4：在根据欧几里得距离选择相关客户端之后，边缘服务器通过对所选模型进行加权平均来聚合本地模型。更新后的全局模型计算为所选客户端的本地模型的平均值：

$$W(t) = \frac{1}{|T|} \sum_{k \in T} W_k(t) \tag{10}$$

这里，w^t 表示第 t 轮之后更新的全局模型，T 表示根据其与全局模型的相似性所选择的客户端集合。

$|T|$是所选客户端的数量，$W_k(t)$是第 t 轮中客户端 k 的本地模型。此聚合过程确保使用最相关的本地更新来更新全局模型，从而提高训练效率并减少异常值的影响。

步骤 5：一旦本地训练完成，每个客户端将其更新后的模型$W_k(t)$返回给边缘服务器。然后，边缘服务器聚合本地模型以生成更新后的全局模型$W(t)$，并将其广播回所有客户端用于下一轮训练。此过程对于预定义数量的全局迭代重复进行，或直到达到收敛，使全局模型逐步改进并收敛到最优状态。

算法 2. ECS-FL 算法

输入：初始聚合模型 $\boldsymbol{w}_0$，最大迭代次数 E，选择批次的大小 B，学习率 η，全局迭代次数 t，损失函数 $\eta\nabla l\left(W_k(t-1);b\right)$。

输出：最终的模型 $W(t)$。

边缘服务器：

初始化并广播 $\boldsymbol{w}_0$

for 每个轮次 $t=1,2,3,\cdots$

  for 每个用户 k 并行执行

    $\boldsymbol{W}(t)\leftarrow$ **DeviceUpdate**$\left(W(t-1)\right)$

    将 $\boldsymbol{W}(t)$ 加入 $\boldsymbol{WSet}$

  加入至客户端集合 $T\leftarrow$ 根据 $\boldsymbol{WSet}$ 计算欧几里得距离

  计算最终的全局模型 $W(t)=\frac{1}{|T|}\sum_{k\in T}W_k(t)$

边缘设备：

DeviceUpdate$\left(W(t-1)\right)$：

  If $\left\|W_k(t-1)-W(t-1)\right\|<\varepsilon$

    收敛，停止模型训练。

  else

    $\boldsymbol{W}_k(t-1)=\boldsymbol{W}(t-1)$

  for 每个本地迭代 i 在 E 中

    for 每个批次 β 在 B 中

      $W_k(t)=W(t-1)-\eta\nabla l\left(W_k(t-1);b\right)$

3.4. ESC-FL 算法的收敛性界限

假设各客户端的损失函数 l 满足 L-光滑性，即对于任意两个模型参数 W_1 和 W_2，有 $\left|\nabla l(W_1)-\nabla l(W_2)\right|\le L\left|W_1-W_2\right|$。这一条件保证了损失函数的梯度变化是相对平滑的，不会出现剧烈波动。同时，假设损失函数满足 μ-强凸性，即对于任意的W_1和W_2，有

$$l(W_1)-l(W_2)-\nabla l(W_2)^{\mathrm{T}}(W_1-W_2)\ge\frac{\mu}{2}|W_1-W_2|^2 \tag{11}$$

强凸性确保了损失函数具有良好的优化性质，存在唯一的全局最优解。此外，假设客户端数据的方差有界，这一假设能够控制数据的波动对模型更新的影响，避免因数据噪声过大导致模型更新不稳定。

在上述假设基础上，定义全局模型在第 n 轮的更新公式为

$$M_n^0=\frac{\sum_{k=1}^{K}\frac{1}{d_t}M_k^n}{\sum_{k=1}^{K}\frac{1}{d_t}} \tag{12}$$

其中 d_t 表示第 k 个客户端本地模型与全局模型的欧几里得距离。在每一轮更新中，计算全局模型的变化

量 $\Delta M_n^0 = M_n^0 - M_{n-1}^0$。利用损失函数的光滑性和强凸性条件，结合欧几里得距离的计算方式，对 ΔM_n^0 进行放缩处理。

通过逐步推导和迭代，可以得到关于模型收敛性的不等式。经过 T 轮迭代后，模型与最优解的距离 $\left|M_T^0 - M^*\right|$ 满足 $\left|M_T^0 - M^*\right| \le \frac{1}{\sqrt{T}} \cdot C$，其中 C 是一个与模型参数、数据分布、光滑性和强凸性参数相关的常数。这一结果表明，随着迭代轮数的不断增加，ECS-FL 算法中的模型会以 $\frac{1}{\sqrt{T}}$ 的速率收敛到最优解附近，从而从理论上证明了 ECS-FL 算法的收敛性。

4. 实验设置及结果分析

本小节通过将所提出方案与 FedAVG 和 FedProx 算法进行比较来对其进行评估。所有方案均使用 Python 编程语言和 PyTorch 框架实现。为保证一致性，所有参与的客户端都使用相同的模型参数。

4.1. 数据集划分及相关配置

为了全面评估 ECS-FL 方案在各种数据分布策略和模型架构下的性能，使用两个图像分类数据集 Fashion MNIST 和 CIFAR-10 进行了实验。将 ECS-FL 的性能与两种基准联邦学习算法(FedAvg 和 FedProx)进行比较。详细的实验参数设置如表 1 所示。

Table 1. Overview of the experimental settings
表 1. 实验设置的概述

参数	详情
CPU	AMD Ryzen5-5600 @ 3.50 GHz
内存	16.0 GB 内存
操作系统	Windows 10
学习率	0.01
用户比例	0.1
批次大小	64
优化器	Adam

1) 数据集和模型架构

本实验使用的数据集是图像分类领域中广泛认可的基准，旨在类似于边缘计算环境的实际条件下测试各种模型架构。在这种环境中，数据通常分布在本地设备上，每个设备处理一部分数据。选择 FashionMNIST 和 CIFAR-10 数据集是为了评估 ECS-FL 在不同数据复杂度水平下的性能。在模型架构方面，使用了一个设计均衡的卷积神经网络(CNN)，它包含卷积层和全连接层。具体而言，第一和第二卷积层各自包含 32 个大小为 3 × 3 的滤波器，而第三和第四卷积层各自包含 64 个相同大小的滤波器。全连接层分别具有 384 × 192 和 192 × 10 的参数。

Fashion MNIST 数据集包含 10 个类别，每个类别有 7000 张 28 × 28 的灰度图像。训练集包含 60,000 张图像，测试集包含 10,000 张图像。与传统的 MNIST 数据集相比，Fashion MNIST 具有更高的视觉复杂性，使其成为评估复杂视觉任务模型的更合适的基准。在该数据集上使用卷积神经网络(CNN)进行训练。

CIFAR-10 数据集包含 10 个类别，每个类别有 6000 张 32 × 32 的彩色图像。训练集包括 50,000 张图

像，测试集包括 10,000 张图像。CIFAR-10 具有中等复杂度，被广泛用于评估各种图像分类算法。在该数据集上也使用卷积神经网络(CNN)进行训练。

2) 数据分布策略

在两种不同的数据分布策略下评估 ECS-FL 的性能，这反映了边缘设备在现实场景中通常会遇到的数据异质性。这些分布策略——独立同分布(IID)和非独立同分布(Non-IID)能够评估所提出的方法在理想化和现实设置中的鲁棒性和适应性。

在 IID 设置中，数据从整个数据集中随机采样，并均匀分布在所有客户端(即边缘设备)上，假设数据是独立同分布的。每个客户端接收相同的数据类别集，确保数据分布均衡。

在 Non-IID 设置中，数据在客户端之间不均匀分布，反映了边缘计算环境中经常遇到的数据异质性。例如，一些客户端可能仅拥有某些类别的数据，导致数据分布出现显著差异。

4.2. 结果评估

1) IID 数据的准确性比较

表 2 展示了 FedAvg、FedProx 和 ECS-FL 在使用 IID 数据划分的 FashionMNIST 和 CIFAR-10 数据集上的性能比较。评估指标包括最终准确性、最佳准确性和损失。结果表明，ECS-FL 在这两个数据集上的性能均优于 FedAvg 和 FedProx，凸显了其在联邦学习场景中的卓越性能，尤其是在数据分布于异构客户端的边缘计算环境中。

Table 2. Performance comparison of FedAvg, FedProx and ECS-FL on the Fashion MNIST and CIFAR-10 datasets
表 2. FedAvg、FedProx 和 ECS-FL 在 Fashion MNIST 和 CIFAR-10 数据集上的性能比较

	Fashion MNIST (独立同分布)			CIFAR-10 (独立同分布)			损失
方案	FedAvg	FedProx	ECS-FL	FedAvg	FedProx	ECS-FL	
最终准确率(%)	84.50	86.51	87.12	64.80	65.67	67.22	0.038
最佳准确率(%)	86.04	87.81	88.13	65.21	66.33	68.07	0.049

对于 Fashion MNIST 数据集，ECS-FL 实现了 87.12%的最终准确性和 88.13%的最佳准确性，显著优于 FedAvg (84.50%和 86.04%)和 FedProx (86.51%和 87.81%)。这些结果表明，ECS-FL 不仅收敛更高效，而且在更简单的数据集上实现了更好的整体性能。在更具挑战性的 CIFAR-10 数据集上，ECS-FL 实现了 67.22%的最终准确性和 68.07%的最佳准确性，超过了 FedAvg (64.80%和 65.21%)和 FedProx (65.67%和 66.33%)。这证明了 ECS-FL 在处理更复杂任务时的鲁棒性和优化能力，特别是在资源受限的边缘计算环境中。

此外，损失值的比较表明，ECS-FL 实现的损失低于 FedAvg 和 FedProx，显示出在训练期间更稳定的收敛。

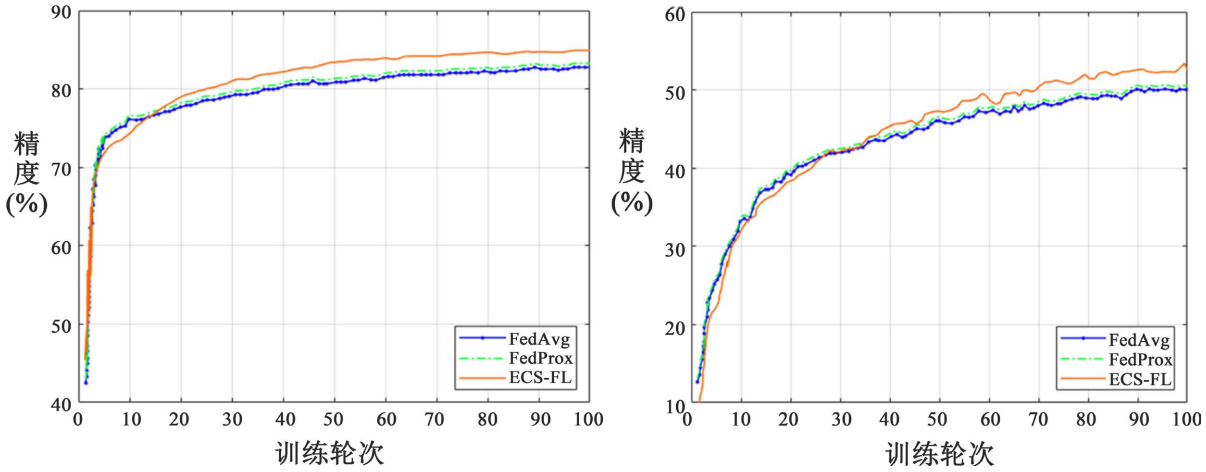
2) Non-IID 数据的准确性比较

图 3 展示了 FedAvg、FedProx 和 ECS-FL 在使用 Non-IID 数据分布的 Fashion MNIST 和 CIFAR-10 数据集上经过 100 轮通信后的准确性表现。结果表明，ECS-FL 在模型准确性方面始终优于其他两种方法，尤其是在初始训练轮次之后，展现出其对现实世界数据异质性的适应性，这是边缘联邦学习中的一个常见挑战。

在 FashionMNIST 数据集上，如图 3(a)所示，ECS-FL 从第 10 轮开始就表现出对 FedAvg 和 FedProx

的显著优势。这种优势在整个训练过程中持续存在，到 100 轮结束时，准确性比其他两种方法高出多达 2%。相比之下，FedAvg 和 FedProx 表现出相当的性能，它们之间没有观察到显著的准确性差距。同样，在 CIFAR-10 数据集上，如图 3(b)所示，ECS-FL 从第 33 轮开始超过其他方法。随着训练的进行，这种改进持续扩大，到第 100 轮时最终实现了比 FedAvg 和 FedProx 高出多达 2.5%的准确性提升。

这些结果凸显了 ECS-FL 在处理 Non-IID 数据方面的卓越能力，表明其与 FedAvg 和 FedProx 相比具有更快的收敛速度和更高的准确性。



(a) Fashion MNIST MLP non-IID　　(b) CIFAR-10 CNN non-IID

Figure 3. Comparison of accuracies on non-independent and identically distributed (Non-IID) data
图 3. 非独立同分布(Non-IID)数据上的准确率比较

3) 客户端数量对模型准确性的影响

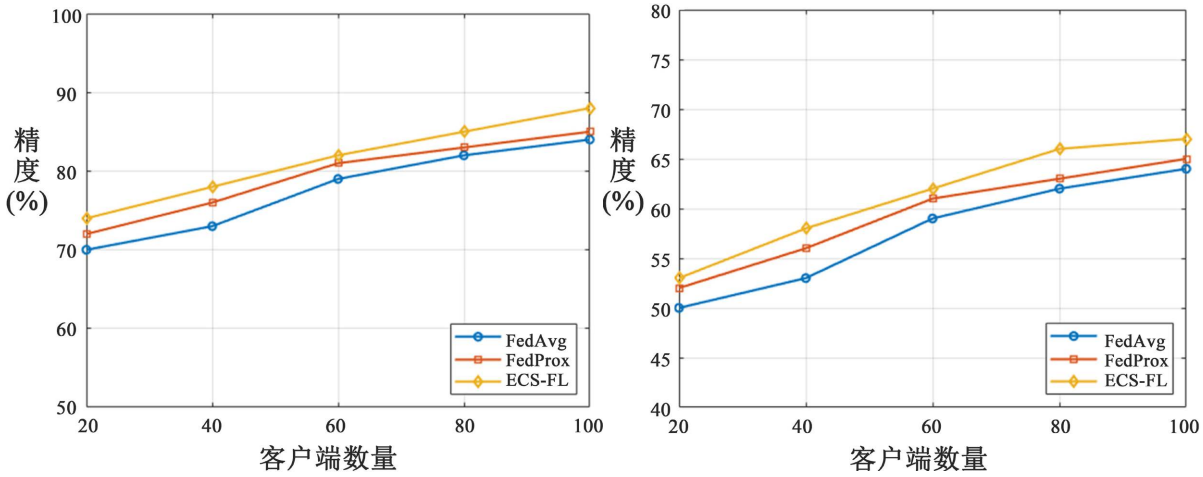
图 4 表明，ECS-FL 在 CIFAR-10 和 Fashion MNIST 数据集上的性能始终优于 FedAvg 和 FedProx。具体而言，ECS-FL 在所有客户端数量下都实现了更高的准确性，特别是当客户端数量增加时。例如，在 CIFAR-10 数据集上，ECS-FL 在有 100 个客户端时达到了 67%的准确性，而 FedAvg 和 FedProx 分别稳定在 64%和 65%。同样，在 Fashion MNIST 数据集上，ECS-FL 实现了 88%的准确性，而 FedAvg 和 FedProx 分别为 84%和 85%。ECS-FL 性能的提升可归因于其增强的聚合策略，该策略促进了不同客户端数据分布下更好的模型收敛。随着客户端数量的增加，ECS-FL 从数据多样性的增加中受益更多，从而实现更快的收敛和更高的整体准确性。

总之，实验结果表明，ECS-FL 不仅实现了更高的准确性，而且在具有大量客户端的环境中展现出更好的可扩展性和鲁棒性。

5. 总结

本文的 ECS-FL 核心在于深入考虑了边缘设备间的数据异质性特征，通过巧妙的算法设计与参数调整，使得系统能够更好地适应这种数据分布的差异。相较于传统的聚合方式，ECS-FL 能够更加精准地整合各客户端的本地模型更新，从而加速全局模型的收敛速度，并提高整体的预测准确性。

实验结果在分类准确率这一关键指标上，ECS-FL 超越了诸如 FedAvg 和 FedProx 等传统联邦学习方法。值得强调的是，当客户端数量增加时，这种优势愈发明显。随着客户端数量的增多，数据的多样性也随之增加，ECS-FL 能够敏锐地捕捉并充分利用这种数据多样性。通过挖掘不同客户端数据中的独特信息，方案能够更全面地学习数据特征，进而加速模型的收敛过程，并进一步提高分类准确率。



(a) Fashion MNIST CNN IID　　(b) CIFAR-10 CNN IID

Figure 4. The influence of the number of clients on the model accuracy
图 4. 客户端数量对模型准确率的影响

在边缘计算的大背景下，边缘设备数量众多、数据分布动态多变且设备间数据异质性显著，传统联邦学习方法在这样的环境中面临诸多性能瓶颈。针对此，后续将继续研究一种极具效力的优化途径，可在不同客户端条件以及动态数据分布情形下，显著提升联邦学习系统的性能。

基金项目

香港狮子山网络空间安全实验室(LRL24017)。

参考文献

[1] Sheller, M.J., Edwards, B., Reina, G.A., Martin, J., Pati, S., Kotrotsou, A., *et al*. (2020) Federated Learning in Medicine: Facilitating Multi-Institutional Collaborations without Sharing Patient Data. *Scientific Reports*, **10**, Article No. 12598. https://doi.org/10.1038/s41598-020-69250-1

[2] Yang, Q., Liu, Y., Chen, T. and Tong, Y. (2019) Federated Machine Learning. *ACM Transactions on Intelligent Systems and Technology*, **10**, 1-19. https://doi.org/10.1145/3298981

[3] Bonawitz, K., Eichner, H., Grieskamp, W., *et al*. (2019) Towards Federated Learning at Scale: System Design. *Proceedings of the Conference on Machine Learning and Systems*, Stanford, 31 March-2 April 2019. https://arxiv.org/abs/1902.01046

[4] Kairouz, P., McMahan, H.B., *et al*. (2019) Advances and Open Problems in Federated Learning.

[5] Zhao, Y., Li, M., Lai, L., *et al*. (2018) Federated Learning with Non-IID Data.

[6] McMahan, H.B., Moore, E., Ramage, D., *et al*. (2017) Communication-Efficient Learning of Deep Networks from Decentralized Data. *Artificial Intelligence and Statistics*, **2017**, 1273-1282.

[7] Miao, Y., Liu, Z., Li, H., Choo, K.R. and Deng, R.H. (2022) Privacy-Preserving Byzantine-Robust Federated Learning via Blockchain Systems. *IEEE Transactions on Information Forensics and Security*, **17**, 2848-2861. https://doi.org/10.1109/tifs.2022.3196274

[8] Sattler, F., Muller, K., Wiegand, T. and Samek, W. (2020) On the Byzantine Robustness of Clustered Federated Learning. 2020 *IEEE International Conference on Acoustics, Speech and Signal Processing* (*ICASSP*), Barcelona, 4-8 May 2020, 8861-8865. https://doi.org/10.1109/icassp40776.2020.9054676

[9] Fu, L., Zhang, H., Gao, G., Zhang, M. and Liu, X. (2023) Client Selection in Federated Learning: Principles, Challenges, and Opportunities. *IEEE Internet of Things Journal*, **10**, 21811-21819. https://doi.org/10.1109/jiot.2023.3299573

[10] Li, T., Sahu, A.K., Talwalkar, A. and Smith, V. (2020) Federated Learning: Challenges, Methods, and Future Directions. *IEEE Signal Processing Magazine*, **37**, 50-60. https://doi.org/10.1109/msp.2020.2975749

[11] Chen, W., Horvath, S. and Richtarik, P. (2022) Optimal Client Sampling for Federated Learning. *Transactions on*

Machine Learning Research. https://arxiv.org/abs/2010.13723

[12] Briggs, C., Fan, Z. and Andras, P. (2020). Federated Learning with Hierarchical Clustering of Local Updates to Improve Training on Non-IID Data. 2020 *International Joint Conference on Neural Networks* (*IJCNN*), Glasgow, 19-24 July 2020, 1-9. https://doi.org/10.1109/ijcnn48605.2020.9207469

[13] Zhang, J., Guo, S., Qu, Z., Zeng, D., Zhan, Y., Liu, Q., *et al*. (2022) Adaptive Federated Learning on Non-IID Data with Resource Constraint. *IEEE Transactions on Computers*, **71**, 1655-1667. https://doi.org/10.1109/tc.2021.3099723

[14] Sahu, A.K., Li, T., Sanjabi, M., Zaheer, M., *et al*. (2020) Federated Optimization in Heterogeneous Networks. *Proceedings of Machine Learning and Systems*, **2**, 429-450.

[15] Ji, S., Jiang, W., Walid, A. and Li, X. (2022) Dynamic Sampling and Selective Masking for Communication-Efficient Federated Learning. *IEEE Intelligent Systems*, **37**, 27-34. https://doi.org/10.1109/mis.2021.3114610

[16] Wolfrath, J., Sreekumar, N., Kumar, D., Wang, Y. and Chandra, A. (2022) HACCS: Heterogeneity-Aware Clustered Client Selection for Accelerated Federated Learning. 2022 *IEEE International Parallel and Distributed Processing Symposium* (*IPDPS*), Lyon, 30 May-3 June 2022, 985-995. https://doi.org/10.1109/ipdps53621.2022.00100

[17] Mao, Y., Shen, L., Wu, J., Ping, P. and Wu, J. (2024) Federated Dynamic Client Selection for Fairness Guarantee in Heterogeneous Edge Computing. *Journal of Computer Science and Technology*, **39**, 139-158. https://doi.org/10.1007/s11390-023-2972-9

[18] Seo, S., Lee, J., Ko, H. and Pack, S. (2022) Performance-Aware Client and Quantization Level Selection Algorithm for Fast Federated Learning. 2022 *IEEE Wireless Communications and Networking Conference* (*WCNC*), Austin, 10-13 April 2022, 1892-1897. https://doi.org/10.1109/wcnc51071.2022.9771600

[19] Damaskinos, G., Guerraoui, R., Kermarrec, A., Nitu, V., Patra, R. and Taiani, F. (2022) Fleet: Online Federated Learning via Staleness Awareness and Performance Prediction. *ACM Transactions on Intelligent Systems and Technology*, **13**, 1-30. https://doi.org/10.1145/3527621

[20] Cai, H., Wang, J., Gao, L. and Li, F. (2024) FLMAAcBD: Defending against Backdoors in Federated Learning via Model Anomalous Activation Behavior Detection. *Knowledge-Based Systems*, **289**, Article 111511. https://doi.org/10.1016/j.knosys.2024.111511