Advances in Applied Mathematics 应用数学进展, 2025, 14(4), 100-110
Published Online April 2025 in Hans. https://www.hanspub.org/journal/aam
https://doi.org/10.12677/aam.2025.144143

Hans汉斯

一种自适应学习率的联邦学习算法

朱桁颍

广东工业大学数学与统计学院，广东 广州

收稿日期：2025年3月3日；录用日期：2025年3月26日；发布日期：2025年4月3日

摘 要

当前人们越来越重视个人信息的保护，联邦学习由于本地训练，不用上传数据而当作一种保护数据隐私的机器学习框架被提出。但是面对现实世界的数据异质性设备时，联邦学习出现全局模型性能下降，收敛速度降低等问题。针对这个现象，本文聚焦于联邦学习的聚合阶段，通过理论分析，结合联邦学习全局损失的收敛上界和更新过程提出了Fedalr算法。该算法通过使用动量方法估计全局梯度，自适应计算出局部梯度的学习率来优化聚合模型，目的是提高了联邦学习的收敛速度和全局模型性能。另外，我们还验证了算法收敛性。最后通过不同种类统计异质性数据的仿真实验，证明了该算法比起当前基准算法性能更优秀，性能最多优化提升了30.74%。

关键词

联邦学习，边缘计算，非独立同分布数据，机器学习

A Federated Learning Algorithm with Adaptive Learning Rate

Hengying Zhu

School of Mathematics and Statistics, Guangdong University of Technology, Guangzhou Guangdong

Received: Mar. 3rd, 2025; accepted: Mar. 26th, 2025; published: Apr. 3rd, 2025

Abstract

With growing concerns over personal information protection, federated learning has emerged as a privacy-preserving machine learning framework by enabling local training without data uploads. However, in real-world scenarios with heterogeneous data and devices, federated learning faces challenges such as degraded global model performance and slower convergence. To address this, we focus on the aggregation phase of federated learning and propose the Fedalr algorithm, based on theoretical analysis of the convergence upper bound and update process of the global loss. Fedalr

文章引用: 朱桁颍. 一种自适应学习率的联邦学习算法[J]. 应用数学进展, 2025, 14(4): 100-110.
DOI: 10.12677/aam.2025.144143

employs momentum methods to estimate global gradients and adaptively computes local gradient learning rates to optimize the aggregated model, aiming to improve convergence speed and global model performance. We also provide a theoretical proof of the algorithm's convergence. Experimental results on various statistically heterogeneous datasets demonstrate that Fedalr outperforms existing baseline algorithms, achieving performance improvements of up to 30.74%.

Keywords

Federated Learning, Edge Computing, Noniid Data, Machine Learning

1. 引言

随着硬件技术的发展，移动算力的提高，诸如智能手机、平板电脑、汽车设备等边缘设备的使用越来越频繁。在传统的数据中心式机器学习中，往往需要用户上传数据进行模型训练。在运输数据过程中容易出现数据泄露的问题，而上传云端的数据可能会受到攻击，这导致本地数据隐私问题频频出现。为了解决这方面的问题，谷歌公司于 2017 年提出了联邦学习(Federated Learning, FL)的算法框架[1]。该框架最大的特征是在保护用户数据隐私的前提下实现跨设备的协同训练。具体来说，该框架与传统集中式学习方法不同，联邦学习将训练模型的阶段下放到各个客户端，每个客户端仅需向中央服务器上传本地模型参数或者梯度，不需要上传数据。这种方式有效地保护本地数据的隐私，减少了隐私泄露的风险和通讯数据的开销，因此拥有广泛的应用场景以及研究意义。

然而在实际应用场景中，联邦学习面临着各种挑战。尽管已经被证实联邦学习在各个客户端 iid (独立同分布)数据下拥有良好的性能，以及他的收敛性已被证明[2]，但是在现实环境下，由于人们生活的地理环境、文化以及政策等因素的不同，通常在移动终端数据呈现 Noniid (非独立同分布)的特征。与此同时[3]也指出了在 Noniid 数据下，联邦学习框架聚合模型拥有收敛速度降低、全局模型性能下降以及对于本地数据不适应等问题。具体来说 Noniid 数据挑战的成因是由于参与聚合的局部客户端数据难以代表全体客户端的数据，因此产生了数据偏移，导致聚合模型产生偏差。因此，如何提高在 Noniid 数据下收敛速度以及全局模型性能是当前联邦学习的一个热门议题。

因此，本文首先在联邦学习的框架范式基础上进行理论分析，目标是找到一个全局模型以降低全局损失的上界，通过理论分析以及联邦学习的更新公式相结合，提出了 Fedalr 算法，该算法在联邦学习的聚合阶段进行优化，通过计算本地梯度与全局梯度来自适应地计算聚合时各梯度的学习率，能够有效抗击 Noniid 数据的冲击，提升了全局模型的收敛速度和性能。最后我们通过仿真实验模拟当前 Noniid 两种数据情况，发现 Fedalr 算法效果更优于当前其他基准算法。

2. 相关工作

Fedavg 开创了联邦学习优化算法[1]，显著地降低了通信成本。后续人们的工作均以 fedavg 为基础开展。面对 Noniid 数据冲击给经典联邦学习带来的挑战，许多研究人员提出了自己的见解。在理论层面，[3]通过推导 Fedavg 算法与集中式机器学习的区别来找到二者的差距；[4]等联邦学习进行了收敛性分析，提出了联邦学习的收敛上界；[5]提出了由于客户端本地模型的过拟合会导致全局模型出现性能震荡。

在框架层面，许多学者通过引用其他框架来解决统计异质性问题，例如多任务联合学习[6]、知识蒸馏[7]等等。同时基于联邦元学习(Federated Meta-Learning) [8]和联邦迁移学习(Federated Transfer Learning) [9]的方法展现出更强的适应性和性能提升。联邦元学习通过元优化策略，使模型能够快速适应不同分布的数据，从而有效缓解 Non-IID 数据带来的性能下降问题。例如，MAML 框架的联邦扩展(FedMeta) [10]通过共享元模型参数，提升了全局模型的泛化能力。此外，联邦迁移学习通过引入领域自适应技术，利用源域知识优化目标域模型，进一步提高了 Non-IID 场景下的学习效率[11]。例如 FedHealth [12]利用迁移学习在医疗数据联邦场景中实现了跨机构的知识共享，显著提升了模型在目标域的表现。而本文更加关注的是针对 Noniid 的全局模型性能和收敛速度。

在降低 noniid 数据对联邦学习的影响算法中，通常有两种优化角度：通过优化服务器端以及优化客户端，目的是为了通过优化全局聚合过程或者训练过程，是全局模型更加接近真实模型。

优化服务器端通常是优化客户端选择或者聚合过程优化。[13]就提出了对靠近全局数据的客户端进行重要性采样的观点。FedGroup [14]提出对相似数据的客户端进行聚类再进行聚合。Fedadp [15]根据上传的梯度来计算贡献度，通过贡献度来确定梯度的聚合权重。为改善联邦优化的稳定性，SCAFFOLD [5]通过控制变量减少客户端漂移，Fedmgda+ [16]通过与多目标优化结合，要求聚合方向不会降低任一客户端的性能以期望达到公平，q-FEL 和 q-Fedavg [17]算法引入了在非 iid 场景中向每个用户公平分配模型资源的问题，并引入了超参数 q 引入全局损失函数，使全局模型在每个局部数据集上都具有良好的性能。

优化客户端现在比较流行的是个性化联邦学习(Personalized Federated Learning, PFL)。现有的个性化联邦学习方法主要包括模型解耦、元学习[18]、正则化约束和聚类方法等。模型解耦方法(如 FedRep [19])通过分离共享层与个性化层，使全局模型保持泛化能力的同时允许客户端个性化调整。正则化方法(如 fedprox [20]、Moon [21])通过引入个性化正则项，在全局模型和本地模型之间寻求平衡，以提升个性化效果。

3. 算法介绍

在本节中主要介绍联邦学习框架以及我们提出的算法流程，最后通过理论推导，证明了我们提出的算法是收敛的。

3.1. 经典联邦学习框架

经典联邦学习具体框架包括 N 个客户端以及 1 个服务器。他们的目标是解决如下优化问题：

$$\min_{w^t} F\left(w^t\right)=\frac{1}{N}\sum_{i=1}^{N} f_i\left(w^t\right) \tag{1}$$

其中 w^t 是全局模型， f_i 是各客户端上的损失函数，F 是全局损失。局部目标函数一般被定义为 C 类分类问题，通常用交叉熵 $l\left(x,y,w\right)$ 来进行计算损失，例如

$$f_i\left(w\right)=\frac{1}{\left|D_i\right|}\sum\nolimits_{\left(x_i,y_i\right)\in D_i} l\left(x_i,y_i,w\right) \tag{2}$$

联邦学习的更新过程为：① 服务器广播全局模型 w^t 给各个客户端，② 各客户端接收到模型 w^t 后根据自己本地数据使用随机梯度下降算法(SGD)进行模型参数优化，直到本地训练轮次结束，最后上传本地模型 w_i^{t+1} 给服务器，③ 服务器对上传的本地梯度进行聚合更新，生成新的全局模型：

$$w^{t+1}=\sum_{i=1}^{N}\lambda_i w_i^{t+1}=w^t-\sum_{i=1}^{N}\lambda_i\eta_i\nabla f_i\left(w^t\right) \tag{3}$$

其中 λ_i 代表第 i 个本地模型的聚合权重， η_i 代表第 i 个客户端的学习率。(3)式为联邦学习更新公式。之

后返回第①步继续新一轮迭代直到通信轮次 t 结束或者达到预期目标。

3.2. Fedalr 算法

在本节中，我们将介绍我们提出的 Fedalr 算法。

首先，我们对联邦学习优化问题的目标函数进行如下假设[22]：

假设 1 (光滑性)：假设全局损失函数 $F(w)$ 和局部损失函数 $f_i(w), i \in (1,2,\cdots,N)$ 在 $\mathbb{R}^m$ 上可微且服从 L-smooth：

$$F\left(w^{t+1}\right) \le F\left(w^t\right) + \left(w^{t+1} - w^t\right)^T \nabla F\left(w^t\right) + \frac{L}{2}\left\|w^{t+1} - w^t\right\|^2$$

其中 $\nabla F\left(w^t\right)$ 代表全局梯度。由假设 1 结合上述联邦学习更新公式可得：

$$F\left(w^{t+1}\right) \le F\left(w^t\right) + \left\langle -\sum_{i=1}^{N} \lambda_i \eta_i \nabla f_i\left(w^t\right), \nabla F\left(w^t\right)\right\rangle + \frac{L}{2}\sum_{i=1}^{N} \eta_i^2 \left\|\lambda_i \nabla f_i\left(w^t\right)\right\|^2 \tag{4}$$

由于目标是 $\min_{w^{t+1}} F\left(w^{t+1}\right)$，而在现实环境中全局损失 F 是很难求得的，因此我们可以运用(4)式来替换目标。又根据联邦学习的迭代公式可知，新的全局模型 w^{t+1} 实际上是由聚合梯度 $\sum_{i=1}^{N} \lambda_i \eta_i \nabla f_i\left(w^t\right)$ 控制，而通常我们无法控制本地数据情况，因此通过本地数据生成的梯度 $\nabla f_i\left(w^t\right)$ 是无法改变的；同时，对于聚合权重 λ_i 一般默认为 $\frac{1}{N}$，因此我们的目标可以转变为：

$$\min_{\eta_i} F\left(w^t\right) - \frac{1}{N}\left\langle \sum_{i=1}^{N} \eta_i \nabla f_i\left(w^t\right), \nabla F\left(w^t\right)\right\rangle + \frac{L}{2N^2}\sum_{i=1}^{N} \eta_i^2 \left\|\nabla f_i\left(w^t\right)\right\|^2 \tag{5}$$

令 $L = N$ 并对 η_i 求偏导可以求得：

$$\eta_i = \frac{\left\langle \nabla f_i\left(w^t\right), \nabla F\left(w^t\right)\right\rangle}{\left\|\nabla f_i\left(w^t\right)\right\|^2} \tag{6}$$

由此我们可以通过公式(6)对不同客户端的学习率进行优化。这种优化结果有如下性质：

公式(6)近似于求局部梯度向量 $\nabla f_i\left(w^t\right)$ 和全局梯度 $\nabla F\left(w^t\right)$ 的余弦，由此可以推断出该方法求得的学习率大小与 $\nabla f_i\left(w^t\right)$ 和 $\nabla F\left(w^t\right)$ 的夹角有关，夹角越小则学习率越大，从而控制聚合梯度 $\sum_{i=1}^{N} \lambda_i \eta_i \nabla f_i\left(w^t\right)$ 往全局方向前进，从而加快模型的速率速度。

为了防止不同客户端设置不同的训练参数，某些客户端训练的步长或者轮次不一致，即参数异质性，这样会到导致上传的梯度长度不一致，进而导致聚合梯度出现偏置，影响全局模型性能。因此我们在计算学习率前对局部梯度 $\nabla f_i\left(w^t\right)$ 进行归一化处理：$\nabla f_i\left(w^t\right) = \frac{\nabla f_i\left(w^t\right)}{\left\|\nabla f_i\left(w^t\right)\right\|}$ 来统一梯度长度，同时减少模型震荡。

由于 $\left\langle \nabla f_i\left(w^t\right), \nabla F\left(w^t\right)\right\rangle$ 计算会出现负数形式，为了保持全局模型的学习效率，我们设计了一个函数 $\eta_i = e^{\eta_i - 1}$，目的是将 η_i 的映射范围缩小为 $(0,1]$。

在我们提出的算法中，全局梯度 $\nabla F\left(w^t\right)$ 是一个很重要的量，因为这会影响不同梯度的学习率进而模型的学习速率。在实际场景中，由于联邦学习的客户端数据保密性，全局模型 $\nabla F\left(w^t\right)$ 是很难通过计算、观测等方式得到，因此，目前通常使用 $\nabla F\left(w^t\right) = \frac{1}{N}\sum_{i=1}^{N} \nabla f_i\left(w^t\right)$ 来替代全局梯度。为了减少由于随机选择

客户端带来的梯度随机性，我们在前人计算全局梯度 $\nabla F\left(w^t\right)$ 的基础上加入动量(Momentum)机制，因此 $\nabla F\left(w^t\right)$ 的计算过程如下：

$$\nabla F\left(w^t\right)=\begin{cases}\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t\right) & t=1\\ \frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t\right)*\frac{1}{t}+\nabla F\left(w^t\right)*\frac{t-1}{t} & t>1\end{cases} \tag{7}$$

由此通过不同局部梯度 $\nabla f_i\left(w^t\right)$ 计算出的 η_i，从而对联邦学习的学习率参数进行优化。综上，与上文的联邦学习框架相比，我们优化的是第③步，其余步骤不发生改变，具体优化流程如下：服务器接收各客户端上传的本地梯度后，先对本地梯度进行归一化处理，然后按照公式(7)估算出当前的全局梯度，再按照公式(6)计算出不同本地梯度的自适应学习率，并将其映射到 $[0,1]$ 范围内，之后根据联邦学习迭代公式(3)更新全局模型。

3.3. 收敛性证明

本节中，我们对提出的 Fedalr 算法进行了收敛性分析，确保算法是可收敛的。在假设 1 的基础上，我们对全局梯度进行了如下假设：

假设 2：假设 ξ_i^t 从第 i 个客户端的本地数据中随机均匀采样。每个设备中随机梯度方差有界：

$$\mathbb{E}\left\|\nabla f_i\left(w^t,\xi_i^t\right)-\nabla f_i\left(w^t\right)\right\|^2\le\sigma_i^2$$

这里 $i\in[N]$。假如令全局梯度等于局部全体梯度的平均值，即 $\nabla F\left(w^t\right)=\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t\right)$，同时令随机梯度下降(SGD)的估计全局均值定义为局部梯度的平均值：$\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t,\xi_i^t\right)$，同时满足

$$\nabla F\left(w^t\right)=\mathbb{E}\left(\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t,\xi_i^t\right)\right)$$

因此全局梯度方差有界：

$$\mathbb{E}\left\|\nabla F\left(w^t\right)-\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t,\xi_i^t\right)\right\|^2\le\frac{1}{N}\sum_{i=1}^{N}\delta_i^2.$$

具体证明可见附录。

假设 3：令 $\delta^2=\frac{1}{N}\sum_{i=1}^{N}\delta_i^2$，则平均随机梯度的二阶原点矩与全局梯度之间的差有界：

$$\mathbb{E}\left\|\frac{1}{N}\sum_{i=1}^{N}\nabla f_i\left(w^t,\xi_i^t\right)\right\|^2\le\left\|\nabla F\left(w^t\right)\right\|^2+\delta^2$$

定理一：通过上述假设 1、2、3，我们假设 $L\ge1$ 可以得到，Fedalr 算法收敛，其中收敛率如下所示，具体收敛过程请查阅附录：

$$\min_{t=1,2,\cdots,T}\left\|\nabla F\left(w^t\right)\right\|^2\le\frac{\sum_{t=1}^{T}\left\|\nabla F\left(w^t\right)\right\|^2}{T}\le\delta^2+2\frac{\mathbb{E}\left(F\left(w^0\right)-F\left(w^{t+1}\right)\right)}{T}$$

从收敛性分析结果可以看出，该算法收敛性主要受到全局梯度 $\nabla F\left(w^t\right)$ 与平均随机梯度的差距影响，而平均随机梯度与本地客户端的数据分布有关。当参与训练聚合的局部客户端们的全体数据分布与整体

客户端数据分布不一致，则容易影响该算法的收敛速度。

4. 实验评估

本环节将介绍仿真实验设置、实验结果。通过在数据集 Cifar10 与 Cifar100 上进行的仿真实验来证明本文提出的算法与当前基准算法相比具有优越性。

4.1. 实验设置

4.1.1. 数据集介绍

Cifar10 数据集是一个广泛使用的图像分类基准数据集，包含 10 个不同类别的彩色图像，每个类别有图像分辨率为 32 × 32 像素的 6000 张图像，共计 60,000 张，其中 50,000 张将被划分为训练集，10,000 张将被划分为测试集。

CIFAR100 数据集也是一个用于图像分类的基准数据集，包含 100 个不同的类别，每个类别有 600 张图像，共计 60,000 张图像。与 Cifar10 数据集类似，Cifar100 中的图像是彩色的，分辨率为 32 × 32 像素。该数据集分为 50,000 张训练图像和 10,000 张测试图像。

4.1.2. 数据划分

在我们的实验中，为了营造客户端数据 Noniid 的情况，我们打算对客户端数据进行如下两类形式划分：(1) 固定种类划分：我们计划将训练集数据先按照种类进行排序，并将每类数据平均分为 20 份总共 200 个分片，每个客户端随机分配 2 个分片，确保每个客户端拥有的分片有且只有 2 个且种类各不相同，我们将这种划分称为 c = 2；(2) 固定分布划分：通过 Dirichlet 分布生成抽样矩阵，确定每个客户端拥有的图像数据种类和数量。我们使用的是参数 d = 0.1 的 Dirichlet 分布进行仿真实验[23]。以 Cifar10 为例具体数据分布情况如图 1 所示：横坐标代表客户端数据数量，列坐标代表客户端编号，每一行都代表了客户端里拥有的数据，不同颜色代表不同数据类型。

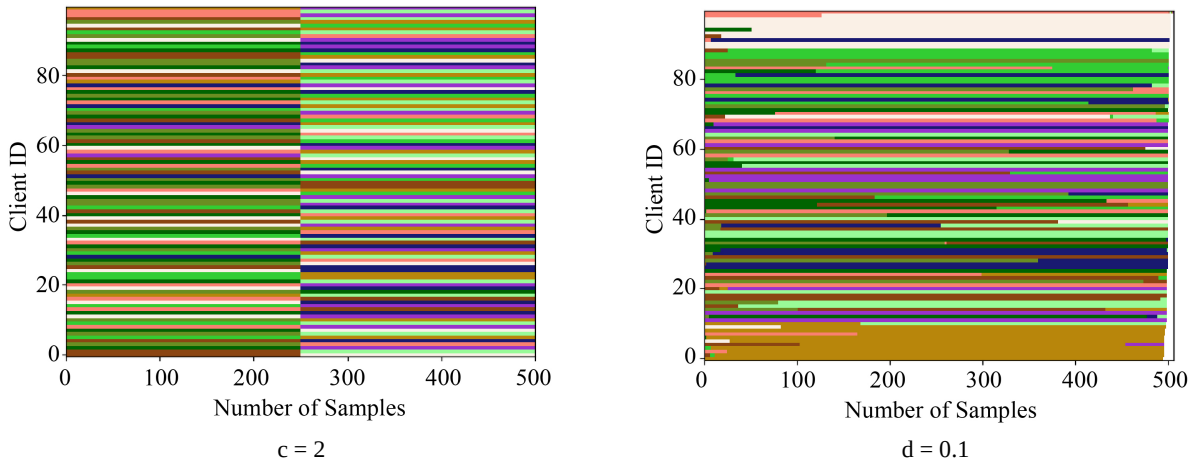


Figure 1. Data distribution map of different data partitions
图 1. 不同数据划分的数据分布图

4.1.3. 基准算法

为了证明我们提出的算法优越性，我们将比较如下几种算法，其中包括 Fedavg，一种经典的联邦学习算法以及 Fedprox、Scaffold、Moon、q-Fedavg 以及 Fedmgda+这几种为了对抗 Noniid 数据冲击导致全

局模型下降的算法。

4.1.4. 参数设置

我们将客户端设置为 100 个，每轮随机抽取 20 个客户端参与训练和聚合(Participation rate = 0.2)，总共 2000 次通讯论数。其中本地训练的 batchsize 为 64，本地学习率(Learning rate)为 0.01，epoch = 1。为了防止实验结果出现随机性，所有算法的实验结果将在四个随机种子下实验平均后得到。

4.1.5. 模型设置

我们的模型是当前流行的双层卷积层的卷积神经网络(CNN)模型，每一层卷积核大小均为 5 * 5，后面均有一个尺寸为 2 * 2 的池化层。模型后面还有三层全连接层，确保输出的结果为 10 个类别的预测分数。

4.1.6. 实验环境

本文实验均基于开发语言 python3.8.0，实验集成环境包 flgo 0.4.3 上进行。使用 GPU 分别为两个 NVIDIA GeForce RTX 3090 以及一个 NVIDIA GeForce RTX 4090。

4.2. 实验结果

4.2.1. 不同数据划分下的性能比较

如图 2 所示，在 Cifar10 数据集中，Fedalr (黄色折线)表现出更快的收敛速度以及全局模型表现出更高的性能，具体来看，如表 1 所示(最优性能已用粗体标出)：在 c = 2 的数据分布下，Fedalr 第 2000 轮的全局模型性能达到 72.15%，比起当前最优算法 Fedavg 性能 63.91%提升了 12.89%；而在 d = 0.1 的数据分布下，Fedalr 算法的全局模型性能高达 68.68%，比起当前最优算法提升了 30.74%。

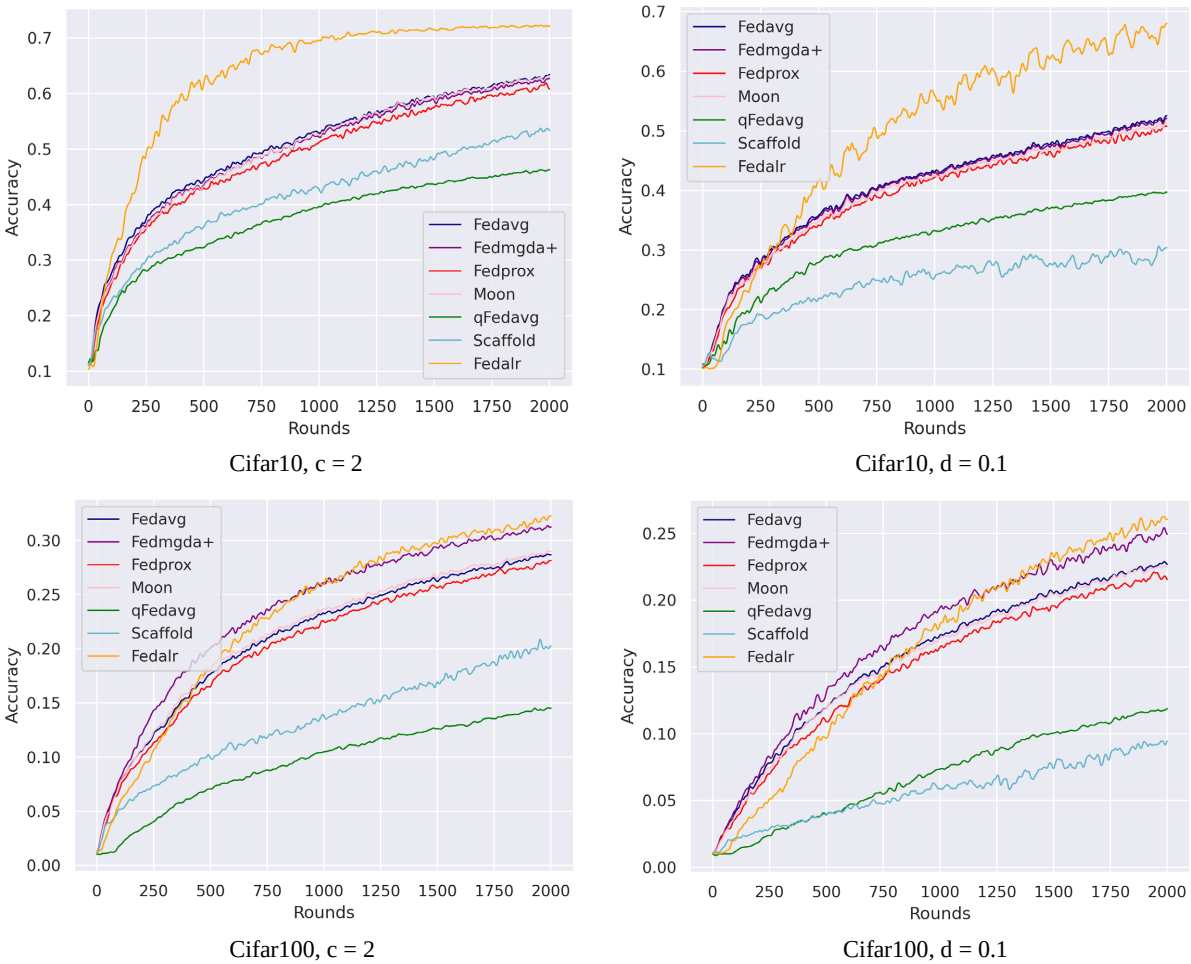
Table 1. Round 2000 federated learning algorithm performance table
表 1. 第 2000 轮联邦学习算法性能表

	Cifar10		Cifar100	
	c = 2	d = 0.1	c = 2	d = 0.1
Fedavg	63.91	52.53	28.45	22.90
Fedmgda+	61.81	52.19	31.10	24.48
Fedprox	62.11	51.12	28.27	21.69
Moon	62.47	51.02	28.84	22.44
qFedavg	46.51	39.87	14.49	11.90
Scaffold	52.95	30.01	20.46	9.81
Fedalr	**72.15**	**68.68**	**32.21**	**26.21**
提升比例	12.89%	30.74%	3.57%	7.07%

在 Cifar100 数据集中，虽然 Fedalr 前期没有表现出较快的收敛性，这可能是因为 cifar100 任务下难以找到较为合适的全局梯度导致收敛速度下降，但是最后面的全局模型性能仍然超过各个算法。根据表 1 可以看到，c = 2 和 d = 0.1 的情况下，Fedalr 算法性能均达到 32.2%1 和 26.21%，比起当前最优算法 Fedmgda+性能分别提升了 3.57%和 7.07%。

在不同的数据划分下，我们提出的算法仍然保持稳健。具体来说，相比于第一种数据划分(c = 2)方式，第二种数据划分(d = 0.1)对联邦学习框架的冲击更大。从图 2 可以看出，同种数据任务下，所有算法在 c

= 2 的表现(如模型性能、收敛速度等)比 d = 0.1 要更好这是因为 c = 2 的划分情况与 d = 0.1 的划分情况相比，局部数据分布更容易接近全局数据分布。这和前文提到的局部数据分布与整体数据不一致会影响该算法的收敛速度相吻合的。但是无论是哪种数据划分方式，我们提出的 Fedalr 算法性能以及收敛速度等方面均优于其他基准算法。



Cifar10, c = 2　　Cifar10, d = 0.1

Cifar100, c = 2　　Cifar100, d = 0.1

Figure 2. Performance graph of different federated learning algorithms
图 2. 不同联邦学习算法性能图

4.2.2. 实验参数对算法的影响

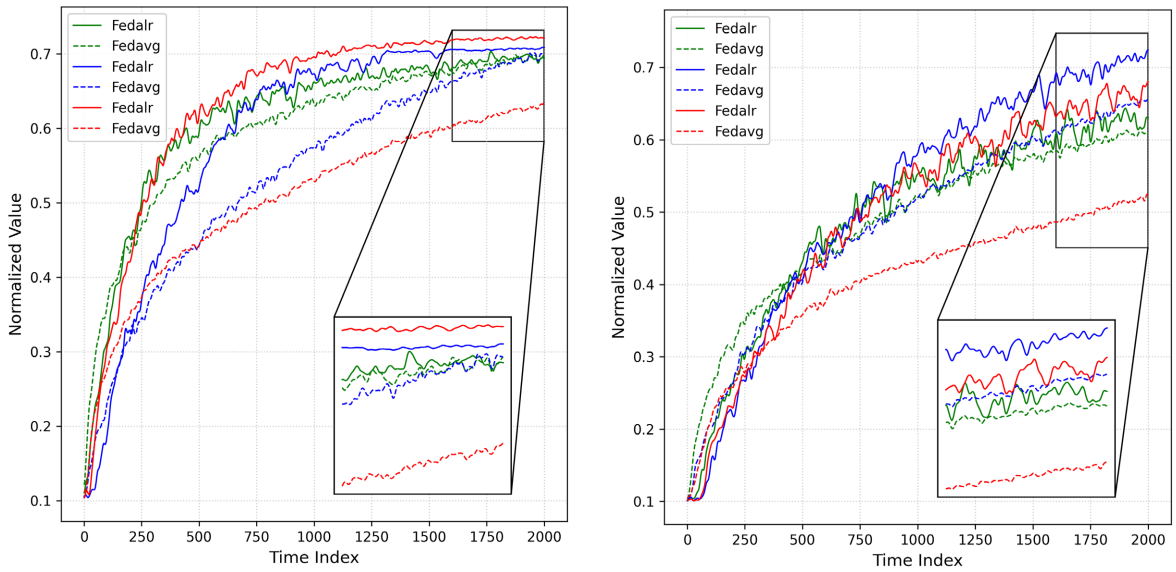
为了考察不同实验参数对算法的影响，我们在 Cifar10 数据集任务 c = 2 和 d = 0.1 的数据划分下设置了不同参数进行实验，具体参数设置如表 2 所示。

由图 3 可以观察到，相比于基础算法 Fedavg (虚线)，我们的 Fedalr 算法(实线)无论参数如何改变，收敛速度以及模型性能均优于 Fedavg。从收敛速度角度来说，无论哪种数据划分，不同参数设置下 Fedalr 算法的收敛速度都十分相近。从模型性能角度来说，当数据划分为 c = 2 时，不同参数下的 2000 轮训练后模型性能差别不大；当数据划分为 d = 0.1 时，不同参数下的 2000 轮训练后模型性能略有差别，最好的参数设置情况是设置 2，2000 轮模型性能为 72.18%，最差的参数设置是设置 1，性能为 62.03%，性能差为 10.15%；而在同种数据划分下的 fedavg 算法性能最好是设置 2，为 66.23%，最差的设置是设置 3，性能为 52.53%，性能差为 13.70%。因此可以认为 Fedalr 算法对于实验参数的敏感性不高，属于鲁棒性较

高的算法。

Table 2. Experimental parameter setting table
表 2. 实验参数设置表

	Learning rate	Batch size	Participation rate
设置 1 (绿色折线)	0.01	10	0.1
设置 2 (蓝色折线)	0.1	400	0.1
设置 3 (红色折线)	0.01	64	0.2



c = 2　　d = 0.1

Figure 3. Algorithm performance line graph with different experimental parameters
图 3. 不同实验参数算法性能折线图

5. 总结

在本文中，我们针对联邦学习受到 Noniid 数据的挑战导致收敛速度降低，全局模型性能下降等问题，提出了 Fedalr 算法，该算法通过计算近似不同客户端上传的本地梯度 $\nabla f_i\left(w^t\right)$ 以及全局梯度 $\nabla F\left(w^t\right)$ 的余弦来自适应地分配不同本地梯度 $\nabla f_i\left(w^t\right)$ 的聚合学习率，从而提高全局模型的性能以及收敛速度，从而减少在 Noniid 环境下联邦学习的通信轮次。为了更准确地计算出全局模型，我们在前人不同本地梯度的基础上采用了动量策略，以消除随机性。最后我们通过实验证明了我们的算法比起其他基线算法更加优秀，性能提升比例高达 30.64%。

参考文献

[1] McMahan, B., Moore, E., Ramage, D., Hampson, S. and Agüera y Arcas, B. (2017) Communication-Efficient Learning of Deep Networks from Decentralized Data. *Proceedings of the* 20*th International Conference on Artificial Intelligence and Statistics*, Fort Lauderdale, 20-22 April 2017, 1273-1282.

[2] Wang, H., Kaplan, Z., Niu, D. and Li, B. (2020) Optimizing Federated Learning on Non-IID Data with Reinforcement Learning. *IEEE INFOCOM* 2020—*IEEE Conference on Computer Communications*, Toronto, 6-9 July 2020, 1698-1707.

https://doi.org/10.1109/infocom41043.2020.9155494

[3] Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D. and Chandra, V. (2018) Federated Learning with Non-IID Data.

[4] Li, X., Huang, K., Yang, W., Wang, S. and Zhang, Z. (2020) On the Convergence of FedAvg on Non-IID Data. 8*th International Conference on Learning Representations, ICLR* 2020, Addis Ababa, 26-30 April 2020, 26 p.

[5] Karimireddy, S.P., Kale, S., Mohri, M., Reddi, S.J., Stich, S.U. and Suresh, A.T. (2019) SCAFFOLD: Stochastic Controlled Averaging for On-Device Federated Learning.

[6] Marfoq, O., Neglia, G., Bellet, A., Kameni, L. and Vidal, R. (2021) Federated Multi-Task Learning under a Mixture of Distributions. *Advances in Neural Information Processing Systems* 34: *Annual Conference on Neural Information Processing Systems* 2021, *NeurIPS* 2021, 6-14 December 2021, 15434-15447.

[7] Zhu, Z., Hong, J. and Zhou, J. (2021) Data-Free Knowledge Distillation for Heterogeneous Federated Learning. *Proceedings of the* 38*th International Conference on Machine Learning*, Vol. 139, 12878-12889.

[8] Zheng, W., Yan, L., Gou, C. and Wang, F. (2020) Federated Meta-Learning for Fraudulent Credit Card Detection. *Proceedings of the* 29*th International Joint Conference on Artificial Intelligence*, Yokohama, 11-17 July 2020, 4654-4660. https://doi.org/10.24963/ijcai.2020/642

[9] Liu, Y., Kang, Y., Xing, C., Chen, T. and Yang, Q. (2020) A Secure Federated Transfer Learning Framework. *IEEE Intelligent Systems*, **35**, 70-82. https://doi.org/10.1109/mis.2020.2988525

[10] Chen, F., Luo, M., Dong, Z., *et al*. (2018) Federated Meta-Learning with Fast Convergence and Efficient Communication.

[11] Liu, Y., Kang, Y., Xing, C., Chen, T. and Yang, Q. (2020) Federated Transfer Learning with Adversarial Adaptation for Non-IID Data. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**, 4696-4703.

[12] Chen, Y., Qin, X., Wang, J., Yu, C. and Gao, W. (2020) Fedhealth: A Federated Transfer Learning Framework for Wearable Healthcare. *IEEE Intelligent Systems*, **35**, 83-93. https://doi.org/10.1109/mis.2020.2988604

[13] Rizk, E., Vlaski, S. and Sayed, A.H. (2022) Federated Learning under Importance Sampling. *IEEE Transactions on Signal Processing*, **70**, 5381-5396. https://doi.org/10.1109/tsp.2022.3210365

[14] Duan, M., Liu, D., Ji, X., Liu, R., Liang, L., Chen, X., *et al*. (2021) Fedgroup: Efficient Federated Learning via Decomposed Similarity-Based Clustering. 2021 *IEEE International Conference on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking* (*ISPA*/*BDCloud*/*SocialCom*/SustainCom), New York, 30 September-3 October 2021, 228-237. https://doi.org/10.1109/ispa-bdcloud-socialcom-sustaincom52081.2021.00042

[15] Wu, H. and Wang, P. (2021) Fast-Convergent Federated Learning with Adaptive Weighting. *IEEE Transactions on Cognitive Communications and Networking*, **7**, 1078-1088. https://doi.org/10.1109/tccn.2021.3084406

[16] Hu, Z., Shaloudegi, K., Zhang, G. and Yu, Y. (2022) Federated Learning Meets Multi-Objective Optimization. *IEEE Transactions on Network Science and Engineering*, **9**, 2039-2051. https://doi.org/10.1109/tnse.2022.3169117

[17] Li, T., Sanjabi, M., Beirami, A. and Smith, V. (2020) Fair Resource Allocation in Federated Learning. 8*th International Conference on Learning Representations, ICLR* 2020, Addis Ababa, 26-30 April 2020, 27 p.

[18] Fallah, A., Mokhtari, A. and Ozdaglar, A.E. (2020) Personalized Federated Learning with Theoretical Guarantees: A Model-Agnostic Meta-Learning Approach. *Advances in Neural Information Processing Systems* 33: *Annual Conference on Neural Information Processing Systems* 2020, *NeurIPS* 2020, 6-12 December 2020, 12 p.

[19] Collins, L., Hassani, H., Mokhtari, A. and Shakkottai, S. (2021) Exploiting Shared Representations for Personalized Federated Learning. *Proceedings of the* 38*th International Conference on Machine Learning*, Vol. 139, 2089-2099.

[20] Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A. and Smith, V. (2020) Federated Optimization in Heterogeneous Networks. *Proceedings of the* 3*rd Conference on Machine Learning and Systems, MLSys* 2020, Austin, 2-4 March 2020, 22 p.

[21] Li, Q., He, B. and Song, D. (2021) Model-Contrastive Federated Learning. 2021 *IEEE*/*CVF Conference on Computer Vision and Pattern Recognition* (*CVPR*), Nashville, 19-25 June 2021, 10713-10722. https://doi.org/10.1109/cvpr46437.2021.01057

[22] Dinh, C.T., Tran, N.H. and Nguyen, T.D. (2020) Personalized Federated Learning with Moreau Envelopes. *Advances in Neural Information Processing Systems* 33: *Annual Conference on Neural Information Processing Systems* 2020, *NeurIPS* 2020, 6-12 December 2020, 12 p.

[23] Li, Q., Diao, Y., Chen, Q. and He, B. (2022) Federated Learning on Non-IID Data Silos: An Experimental Study. 2022 *IEEE* 38*th International Conference on Data Engineering* (*ICDE*), Kuala Lumpur, 9-12 May 2022, 965-978. https://doi.org/10.1109/icde53745.2022.00077

附　录

本节重要展示 Fedalr 的收敛过程。首先我们考察了通过 Fedalr 算法所计算的学习率 η_i 的性质，进而将性质与收敛性分析方法进行结合得到相应结论。下文中我们采用以下符号以区分全局梯度($\nabla F\left(w^t\right)$)与模拟的全局梯度($\widetilde{\nabla F\left(w^t\right)}$)。

性质 1：对于 $\forall i \in [N]$，$\eta_i \le 1$。

性质 1 证明：

由于 $\eta_i = \dfrac{\left\langle \nabla f_i\left(w^t\right), \widetilde{\nabla F\left(w^t\right)} \right\rangle}{\left\|\nabla f_i\left(w^t\right)\right\|^2}$，且 $\widetilde{\nabla F\left(w^t\right)} = \dfrac{1}{N}\sum_{i=1}^{N} \nabla f_i\left(w^t\right)$，因此 $\eta_i = \dfrac{\left\langle \nabla f_i\left(w^t\right), \sum_{j=1}^{N} \nabla f_j\left(w^t\right) \right\rangle}{N\left\|\nabla f_i\left(w^t\right)\right\|^2}$。

又因为 $\forall i \in [N]$，$\nabla f_i\left(w^t\right) = \dfrac{\nabla f_i\left(w^t\right)}{\left\|\nabla f_i\left(w^t\right)\right\|}$，所以 $\left\|\nabla f_i\left(w^t\right)\right\|^2 = 1$。

所以 $\eta_i = \sum_{j=1}^{N} \dfrac{\left\langle \nabla f_i\left(w^t\right), \nabla f_j\left(w^t\right) \right\rangle}{N\left\|\nabla f_i\left(w^t\right)\right\|^2} * \dfrac{\left\|\nabla f_j\left(w^t\right)\right\|}{\left\|\nabla f_j\left(w^t\right)\right\|} = \dfrac{1}{N}\sum_{j=1}^{N} \cos\theta_{ij} \le 1$。

收敛性证明：

由假设一同时 $L \ge 1$，可得：

$$F\left(w^{t+1}\right) - F\left(w^t\right) \le + \left\langle -\sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right), \nabla F\left(w^t\right) \right\rangle + \frac{1}{2}\left\| \sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right) \right\|^2$$

$$\left\langle -\sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right), \nabla F\left(w^t\right) \right\rangle \le \frac{1}{2}\left(\left\| \nabla F\left(w^t\right) - \sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right) \right\|^2 - \left\| \nabla F\left(w^t\right) \right\|^2 - \left\| \sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right) \right\|^2 \right)$$

代入可得：

$$F\left(w^{t+1}\right) - F\left(w^t\right) \le \frac{1}{2}\left(\left\| \nabla F\left(w^t\right) - \sum_{i=1}^{N} \frac{1}{N}\eta_i \nabla f_i\left(w^t\right) \right\|^2 - \left\| \nabla F\left(w^t\right) \right\|^2 \right)$$

两边取期望，再结合性质 1 以及假设 2，可得：

$$\mathbb{E}\left\| \nabla F\left(w^t\right) \right\|^2 \le \delta^2 + 2\mathbb{E}\left(F\left(w^t\right) - F\left(w^{t+1}\right) \right)$$

所以 $\min_{t=1,2,\cdots,T} \left\| \nabla F\left(w^t\right) \right\|^2 \le \dfrac{\sum_{t=1}^{T} \left\| \nabla F\left(w^t\right) \right\|^2}{T} \le \delta^2 + 2\dfrac{\mathbb{E}\left(F\left(w^0\right) - F\left(w^{t+1}\right) \right)}{T}$，即可得证。