# 基于数据蒸馏的个性化联邦学习方法

陈根文[1)]

[1)](东南大学网络空间安全学院, 南京, 中国, 211189)

**摘 要** 现有的联邦学习（Federated Learning, FL）方法大多基于模型平均方案，其中中央服务器通过平均客户端共享的模型参数来获得最新的全局模型。但是这种传统方案在非独立同分布设置中的性能会因为局部训练中的更新偏差而严重下降。除此之外，传输模型参数会导致高通信开销和隐私问题。受数据集蒸馏（Dataset Distillation, DD）方法的启发，本文提出一种基于DD的新FL方案，FedRD。在FedRD中，整个模型由局部投影模型和全局任务模型组成。首先客户端使用个性化投影模型将本地数据集转换为低维表示，然后通过分布匹配学习数据表示的合成集。通过共享合成表示，客户端可以通过保护隐私和通信高效的方式将本地数据集的信息传输给中央服务器。在服务器端，全局任务模型是在合成表示的聚合上训练的，使得FedRD能够避免由数据异质性引起的模型平均方案中的更新偏差。最后在四个图像分类数据集上进行了实验，结果表明FedRD在模型准确性和通信开销方面显著优于比较方法。

**关键词** 联邦学习；数据蒸馏；通信效率；隐私保护

# Personalized Federated Learning Method Based on Dataset Distillation

CHEN Gen-Wen[1)]

[1)](School of Cyber Science and Engineering, Southeast University, City Nanjing, China)

**Abstract**

Existing Federated Learning (FL) methods are mostly based on the model-averaging scheme, in which the central server averages the model parameters shared by clients to obtain the latest global model. However, the performance of this traditional scheme will degrade seriously in non-i.i.d settings due to the update bias in local training. What's worse, transmitting model parameters leads to high communication overheads and privacy issues. Inspired by the dataset distillation (DD) methods, we propose a novel DD-based FL scheme called FedRD for addressing aforementioned problems. In FedRD, the whole model consists of the local projection model and the global task model. The client firstly uses the personalized projection model to transform local dataset into low-dimensional representations. Then, the synthetic set of data representations is learned via distribution matching. By sharing synthetic representations, clients can transmit the information of local dataset to the central server in a privacy-preserving and communication-efficient way. In the server side, the global task model is trained over the aggregation of synthetic representations. It enables FedRD to avoid the update bias in model-averaging schemes caused by data heterogeneity. Experiments are conducted over four image classification datasets, and the results demonstrate that FedRD outperforms compared methods significantly in terms of model accuracy and communication overhead.

**Keywords** Federated Learning; Data Privacy; Gradient inversion attack; Model poisoning; Fully Connected Neural Networks

# 1 介绍

联邦学习（Federated Learning, FL） [1] 被提出作为一种分布式学习范式，用于以隐私保护的方式协作训练全局模型。典型的FL系统通常由一个中心服务器和多个客户端组成。在每轮通信中，客户端基于本地私有数据集训练模型，并将模型参数或更新发送给服务器。中心服务器对客户端共享的消息进行平均，以获得最新的全局模型。

然而，基于模型平均的传统FL方法 [1, 2]仍面临三个主要挑战：1.数据异质性：在实际应用中，不同客户端之间的训练数据通常是不平衡且非独立同分布(non-i.i.d.)的。这种异质性会导致本地训练过程中模型更新的偏差，从而严重影响模型性能。一些改进方法 [2, 3]通过修改损失函数 [3]或采用归一化的平均方法来缓解目标不一致性问题 [4]，但这些基于模型平均的改进方法仅能部分减弱本地模型更新的偏差，而无法彻底避免。2. 通信开销：在实际设备通信带宽有限的情况下，多轮传输完整模型参数会造成巨大的通信开销。随着全局模型变得更加复杂（例如更深的神经网络），这个问题更加突出。3. 数据隐私问题：最近的研究发现，基于模型平均的方法易受隐私攻击 [5, 6]。攻击者可以通过推断客户端共享的模型参数窃取本地数据的隐私信息，甚至重建原始数据。因此，迫切需要提出一种新型的FL方案，以规避传统模型平均方法中的这些问题。

本文提出了一种新的FL方法:FedRD（个性化联邦学习的表示蒸馏方法）。受到数据蒸馏（Dataset Distillation, DD）方法 [7, 8]的启发，后者旨在学习原始数据集的浓缩版本。合成数据集虽然仅包含极少量的数据，但在此数据集上训练的模型可以达到与在完整数据集上训练的模型相近的性能。

FedRD 通过共享合成数据集而非模型参数来在客户端和服务器之间传递本地信息，从而避免了传统模型平均方法的不足。与FedRD最相关的工作是另一种基于DD的方法:FedDM [9]，它在本地设备上执行数据蒸馏，并将合成数据发送至服务器。然而，FedDM的性能显著受限于每类合成数据的数量（dpc）。增加dpc可以提高准确性，但也会带来更高的通信开销。

与FedDM直接压缩原始数据集不同，FedRD通过蒸馏特征表示来减少数据维度，这种方法被称为"示蒸馏"。在相同的通信开销下，FedRD可以通过传输更多的合成数据量实现更高的准确性。表示蒸馏让FedRD在模型准确性和通信开销之间达到了更好的平衡。

具体来说，FedRD 将模型划分为两部分：本地投影模型和全局任务模型。其中，本地投影模型是为每个客户端量身定制的，并在客户端侧进行训练。本地数据集首先通过投影模型被转化为低维特征表示，然后通过分布匹配方法 [10]生成浓缩的特征集合，并将其发送到服务器。全局任务模型则在服务器端基于所有客户端的合成特征表示进行训练。通过共享合成特征表示，FedRD 能够以隐私保护且高效的通信方式将本地数据集的信息传递到中心服务器。在全局范围内使用所有合成特征进行训练，有效避免了由于数据异质性引起的更新偏差。

本文主要主要贡献:

(1) 提出了一种新型的联邦学习方案FedRD：基于表示蒸馏的方法，专门用于解决数据异质性问题。通过共享合成的特征表示，FedRD 能够以隐私保护和通信高效的方式将客户端的本地数据信息传递到中心服务器。

(2) 提出对本地数据集的低维特征表示进行浓缩,通过减少特征的维度大小，FedRD 可以在不增加通信开销的前提下传输更多的合成数据，从而显著提升模型的准确性。

(3) 在四个标准数据集上开展了相关实验,结果表明，FedRD 在非独立同分布设置下，在模型准确性和通信开销方面均显著优于现有方法。

# 2 相关工作

## 2.1 联邦学习

联邦学习[21,16,34]是一种分布式学习范式，近年来受到了广泛关注。参与的客户通过共享梯度而不是私有局部数据的隐私保护方式合作训练全局模型。中央服务器通过迭代地聚合客户端的梯度来协调训练过程。在每一轮$k$中，客户端用局部训练集$(X_i, Y_i)$训练全局模型，并共享它们的梯度。然后，服务器计算平均梯度为：$\frac{1}{N}\sum_{i=1}^{N}\nabla_\theta \mathcal{L}_{\theta^k}(X_i, Y_i)$，其中$N$是参与方数量。最新的平均梯度被发送回各个客户端，他们可以根据等式(1)更新局部模型。这个过程被称为联邦SGD，将持续许多轮，直到全局模型收敛。

$$\theta^{k+1} = \theta^k - \tau\frac{1}{N}\sum_{i=1}^{N}\nabla_\theta \mathcal{L}_{\theta^k}(X_i, Y_i) \qquad (1)$$

## 2.2 数据蒸馏

数据集蒸馏（Dataset Distillation, DD）的概念最早由Wang等人 [7]提出。DD方法的目标是生成一小部分不存在于原始数据集中的合成数据点，这些数据点用于训练时，可以让模型表现出与使用完整数据集训练相近的效果。自此之后，许多研究致力于提升合成数据集的有效性。例如，获取软标

签 [11]，匹配真实数据与合成数据的梯度 [12]或分布 [10]，引入数据增强 [13]，匹配长时间的训练轨迹 [14]，以及通过生成模型来合成训练数据 [8]等方法。此外，Wang等人 [15]还探讨了DD的隐私问题，并表明DD可以在实践中提供差分隐私(DP)的保障。

## 3 方法

### 3.1 问题描述

考虑一个典型的个性化联邦学习（FL）系统，该系统由一个中心服务器和$N$ 个客户端组成。客户端$k \in [N]$ 的学习模型$q_k(\cdot) = \phi_\theta(\varphi_{w_k}(\cdot))$包含两部分：由参数$w_k$ 表示的个性化的本地投影模型$\varphi_{w_k}$,以及由参数$\theta$ 表示的全局任务模型$\phi_\theta$。投影模型的参数$w_k$ 在不同客户端之间是独立的，而全局任务模型的参数$\theta$ 则由所有客户端共享。每个客户端拥有一个本地数据集$D_k = \{x, y\}$，且各客户端的数据集通常为非独立同分布。本文的研究聚焦于图像分类任务。中心服务器负责协调客户端，在不交换原始数据的情况下，基于全局数据集$D \triangleq \cup_{i\in[N]} D_k$ 开展协同学习。个性化联邦学习的通用形式可以表示为以下优化问题：

$$\min \frac{1}{N} \sum_{k=1}^{N} \mathbb{E}_{(x,y)\sim D_k}[l(q_k(x), y)] \tag{2}$$

其中，客户端$k$ 的学习模型$q_k : \mathbb{R}^d \to Y$ 将输入数据$x \in \mathbb{R}^d$ 映射到预测标签$q_k(x) \in Y$，损失函数$l$ 用于衡量$q_k(x)$ 与真实标签$y$ 之间的偏差。

### 3.2 FedRD算法

#### 3.2.1 FedRD概览

FedRD的总体框架如图1所示，具体算法见算法1。在每一轮学习中，总共包含以下五个步骤：

步骤1：服务器将最新的任务模型$\phi_\theta^r$ 分发给所有客户端。任务模型通常是用于图像任务的骨干模型（例如，ConvNet、VGG、ResNet）。

步骤2：客户端加载最新的任务模型$\phi_\theta^r$ 并在其本地数据集$D_k$ 上更新本地投影模型$\varphi_{w_k}$，通过最小化以下损失函数来实现:

$$L_\varphi = \mathbb{E}_{(x,y)\sim D_k}[l(\phi_\theta^r(\varphi_{w_k}(x)), y)] \tag{3}$$

其中，$l$ 是交叉熵损失函数。在训练过程中，任务模型的参数$\theta^r$ 在客户端是固定的。每一轮学习都基于最新的任务模型不断优化本地投影模型，目的是找到适合后续任务模型的表示集$Z_k = \{\varphi_{w_k}(x)|x \in D_k\}$。投影模型的输出维度为$d'$，即$\varphi_{w_k}(x) \in \mathbb{R}^{d'}$，其中$d' \ll d$。为方便起见，我们将投影表示$\varphi_{w_k}(x)$ 调整为大小为$1 \times \sqrt{d'} \times \sqrt{d'}$ 的图像格式，这样可以直接输入到后续任务模型中。

步骤3：客户端执行表示蒸馏，合成一个与真实表示$Z_k$ 分布相似的压缩表示集$S_k$。合成表示$s \in S_k$ 的维度与调整后的表示一致，即$1 \times \sqrt{d'} \times \sqrt{d'}$。每类数据的合成表示数量通常设置为10、50或100 [10]，因此$|S_k| \ll |Z_k|$。表示蒸馏的细节将在第III-B2部分中详细介绍。

步骤4：每个客户端将其合成表示$S_k$ 发送到中央服务器。

步骤5：服务器收集所有客户端的合成表示，并在聚合数据集$\{S_k\}_{k=1}^N$ 上训练全局任务模型$\phi_\theta^r$。

**Input:** $\phi_\theta$: 全局任务模型, $\phi_\theta$ 用$\theta$参数化; $\eta$: 学习率

**Server executes:**
**for** *each round* $r = 1, \cdots, R$ **do**
  **for** *each client* $k = 1, \cdots, K$ **do**
    $S_k \leftarrow$ `ClientUpdate`$(k, \theta_r)$;
  **end**
  Update the global task model $\phi_\theta$, on aggregated data $\{S_k\}_{k=1}^K$ by SGD with the learning rate $\eta_g$;
**end**

**ClientUpdate**($k$, $\theta_r$):
Update the local projection model $\varphi_{w_k}$ on local data $D_k$ by minimizing the loss $\mathcal{L}_\phi$ in Eq. (2);
Initialize $S_k$ from random noise or real examples;
**for** *each round* $t = 1, \cdots, T$ **do**
  Sample $\vartheta \sim P_\vartheta$;
  Sample mini-batch pairs $B_{D_k}^c \sim D_k$ and $B_{S_k}^c \sim S_k$ for every class $c$;
  Compute the loss $\mathcal{L}_k$ in Eq. (5);
  Update $S_k \leftarrow S_k - \eta_d \nabla_{S_k} \mathcal{L}_k$;
**end**

**Algorithm 1:** FedRD算法实现

#### 3.2.2 表示蒸馏

受数据集蒸馏方法 [7, 10, 12]的启发，我们提出了表示蒸馏，用于在联邦学习中生成本地私有数据集的压缩表示。通过共享合成表示，我们可以以隐私保护且通信高效的方式将本地数据集的信息传递到中央服务器。

首先，我们利用本地投影模型$\varphi_{w_k}$ 将原始数据$x \in \mathbb{R}^d$ 转换为数据表示$z = \varphi_{w_k}(x) \in \mathbb{R}^{d'}$，其中$d' \ll d$。类似于方法DM[13]，我们通过分布匹配生成压缩表示$S$。最大均值差异（MMD） [16]

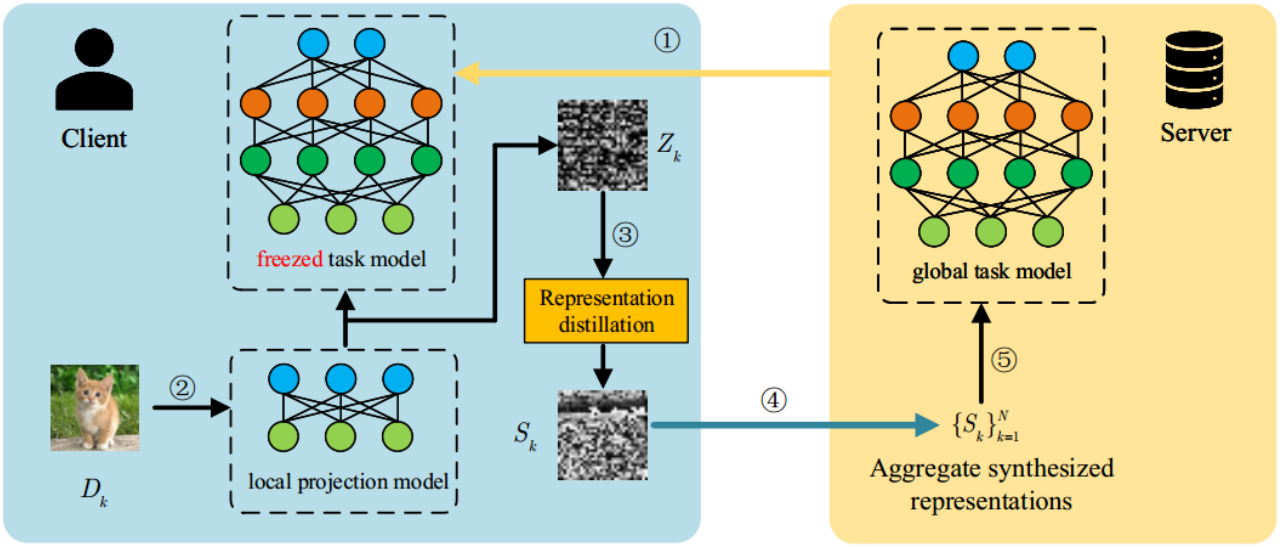



**Fig. 1** FedRD概览

被广泛用于在低维潜在空间中估计真实数据分布：

$$\sup_{\|\psi_\vartheta\|_H \le 1} \left(\mathbb{E}[\psi_\vartheta(Z)] - \mathbb{E}[\psi_\vartheta(S)]\right) \tag{4}$$

其中，$H$ 是重现核Hilbert space，$\psi_\vartheta$ 是将输入映射到潜在空间的嵌入函数。由于我们无法获取真实数据分布的真实值，因此采用MMD的经验估计：

$$\mathbb{E}_{\vartheta\sim P_\vartheta}\left\| \frac{1}{|D_k|}\sum_{(x,y)\in D_k}\psi_\vartheta(\varphi_{w_k}(x)) - \frac{1}{|S_k|}\sum_{(s,y)\in S_k}\psi_\vartheta(s)\right\|_2 \tag{5}$$

其中，嵌入函数$\psi_\vartheta$ 可以是参数化为$\vartheta$ 的神经网络，从分布$P_\vartheta$ 中采样得到。为了提高计算效率，我们在每次合成表示的学习迭代中，随机从类别$c \in [C]$ 中抽取小批量数据$B^c_{D_k} \sim D_k$和$B^c_{S_k} \sim S_k$。根据 [13]中的工作，我们在训练中对真实数据应用可微的Siamese数据增强$A(\cdot)$。然后，本地投影模型$\varphi_{w_k}$ 以$A(B^c_{D_k})$ 作为输入，生成真实表示的小批量。通过在不同嵌入空间中采样$\vartheta$，我们计算合成表示和真实表示的分布差异。通过最小化以下损失函数$L_k$ 学习合成表示$S_k$：

$$L_k = \sum_{c=0}^{C-1}\mathbb{E}_{\vartheta\sim P_\vartheta}\left\| \frac{1}{|B^c_{D_k}|}\sum_{(x,y)\in B^c_{D_k}}\psi_\vartheta(\varphi_{w_k}(A(x))) - \frac{1}{|B^c_{S_k}|}\sum_{(s,y)\in B^c_{S_k}}\psi_\vartheta(s)\right\|_2 \tag{6}$$

## 4 实验

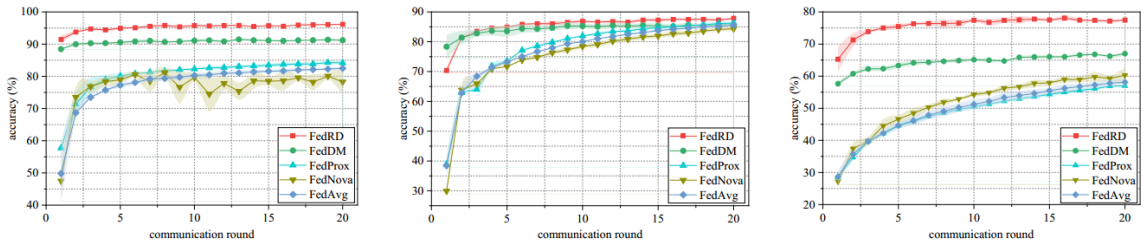
### 4.1 实验设置

(1) 数据集:实验在四个标准图像数据集上进行，包括Fashion MNIST、SVHN、CIFAR10和CIFAR100。通过调整Dirichlet分布参数α来模拟非独立同分布设置。较小的α值表示数据异质性较高。默认情况下，将整个数据集按照α=0.1划分为10个客户端。每个客户端随机将本地子集按8:2的比例划分为训练集和测试集。所有客户端在其本地测试集上的平均准确率。

(2) Baseline模型:将FedRD与三种基于模型聚合的方法进行比较：FedAvg [1], FedProx [3], FedNova [4]和一种基于数据蒸馏的方法FedDM [9]。

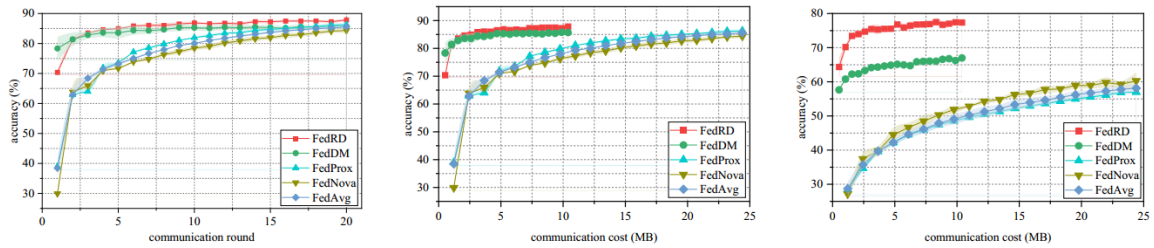
(3) 模型:全局任务模型是一个ConvNet，包含三个卷积层和一个全连接层。三个卷积层均有128个滤波器，卷积核大小为3，之后分别接一个BatchNorm层、一个ReLU激活层和一个卷积核大小为2的MaxPooling层。局部投影模型仅包含两个卷积层，第一层有8个滤波器，第二层有16个滤波器。局部投影模型的输出维度为$d' = w \times h$，其中$w$ 和$h$ 分别是输入数据的宽度和高度。用于计算MMD的映射函数采用与任务模型相同的结构，但去掉了最后的全连接层。

(4) 超参数和模型:对于所有比较方法，总通信轮数$R = 20$。对于基于数据蒸馏的方法（FedRD和FedDM），客户端在每轮通信中进行$T = 1000$ 次迭代以学习合成表示，学习率为$\eta_s = 1$。真实数据和合成数据的批量大小均为256。FedDM在所有四个数据集上的每类




(a) FashionMNIST (b) SVHN (c) CIFAR10

**Fig. 2** 方法准确性随着通信轮次对比




(a) FashionMNIST (b) SVHN (c) CIFAR10

**Fig. 3** 方法准确性随着通信开销对比

合成数据量均为10；FedRD在FashionMNIST上的每类合成表示数量为10，在其他三个数据集上为30。该设置确保FedRD和FedDM在发送合成数据时的通信成本相同。FedRD的局部投影模型在客户端通过Adam优化器以学习率$\eta_l = 0.01$，批量大小为256，训练10个周期。在服务器端，全局任务模型以批量大小256，通过Adam优化器以学习率$\eta_l = 0.01$训练500个周期。对于基于模型聚合的方法（FedAvg、FedProx和FedNova），超参数选取以获得最佳性能。

### 4.2 模型精度与通信开销

本节从模型精度和通信开销两个角度评估了FedRD方法。通信开销指的是每个客户端共享的消息总大小。表1列出了不同方法在模型精度上的均值和方差。结果显示，FedRD在四个数据集上的五种方法中表现最佳。

分析原因在于，在聚合的合成数据集上训练全局模型能够捕获所有客户端数据集的全局信息。与模型平均方法相比，全局模型仅在客户端进行训练，并只能看到局部数据，这导致客户端共享的模型更新存在严重偏差，特别是在非独立同分布设置中。

与基于DD的方法FedDM相比，FedRD的额外精度提升得益于客户端共享的大量合成表示。原始图像的尺寸为$d = c \times h \times w$，其中$c$、$h$和$w$分别代表图像的通道数、宽度和高度，而合成表示的尺寸为$d' = 1 \times h \times w$。在SVHN、CIFAR10和CIFAR100数据集上，图像的通道数$c = 3$，因此合成表示的维度大小为$d = 3d'$。在通信开销相同的情况下，FedRD中合成数据集的数据量是FedDM的三倍。合成数据集中的大量数据点能够更好地表征原始数据集的分布，这也是提高在合成数据集上训练模型性能的关键，已有研究 [7,9,12]对此进行了讨论。

此外，FedRD的收敛速度也明显优于其他方法，如图2所示。在前5轮通信中，FedRD能够迅速收敛，并实现最高的模型精度。在CIFAR10数据集上，FedRD在第一轮通信后就能达到65%的准确率，而模型平均方法的准确率则低于30%。考虑到通信开销，FedRD的优势更加突出。模型精度和通信开销的对比见图3。在进行20轮通信后，模型平均方法中每个客户端的通信开销为25MB，几乎是FedRD和FedDM的2.5倍。对于FedRD来说，模型精度可达到75%，且通信开销仅为2.5MB。

## 5 参考文献

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, “Communication-efficient learning of deep networks from decentralized data,” in *Artificial intelligence and statistics*.

**Table 1** 不同数据集下准确率对比

| | F-MNIST | SVHN | CIFAR10 | CIFAR100 |
|---|---|---|---|---|
| FedAvg | 82.25±0.53 | 85.56±0.52 | 58.13±1.01 | 25.67±0.21 |
| FedProx | 84.27±0.58 | 86.26±0.59 | 56.94±0.48 | 25.64±0.32 |
| FedNova | 81.15±1.38 | 84.34±0.73 | 60.28±2.47 | 24.49±0.91 |
| FedDM | 91.35±0.51 | 85.55±0.92 | 66.98±0.55 | 34.93±0.24 |
| FedRD | 96.16±0.13 | 87.86±0.35 | 78.07±0.82 | 40.46±0.51 |

PMLR, 2017, pp. 1273–1282.

[2] L. Collins, H. Hassani, A. Mokhtari, and S. Shakkottai, "Exploiting shared representations for personalized federated learning," in *International conference on machine learning*. PMLR, 2021, pp. 2089–2099.

[3] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine learning and systems*, vol. 2, pp. 429–450, 2020.

[4] J. Wang, Q. Liu, H. Liang, G. Joshi, and H. V. Poor, "Tackling the objective inconsistency problem in heterogeneous federated optimization," *Advances in neural information processing systems*, vol. 33, pp. 7611–7623, 2020.

[5] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *2019 IEEE symposium on security and privacy (SP)*. IEEE, 2019, pp. 691–706.

[6] H. Yin, A. Mallya, A. Vahdat, J. M. Alvarez, J. Kautz, and P. Molchanov, "See through gradients: Image batch recovery via gradinversion," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 16 337–16 346.

[7] T. Wang, J.-Y. Zhu, A. Torralba, and A. A. Efros, "Dataset distillation," *arXiv preprint arXiv:1811.10959*, 2018.

[8] G. Cazenavette, T. Wang, A. Torralba, A. A. Efros, and J.-Y. Zhu, "Generalizing dataset distillation via deep generative prior," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 3739–3748.

[9] Y. Xiong, R. Wang, M. Cheng, F. Yu, and C.-J. Hsieh, "Feddm: Iterative distribution matching for communication-efficient federated learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16 323–16 332.

[10] B. Zhao and H. Bilen, "Dataset condensation with distribution matching," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 6514–6523.

[11] I. Sucholutsky and M. Schonlau, "Soft-label dataset distillation and text dataset distillation," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[12] B. Zhao, K. R. Mopuri, and H. Bilen, "Dataset condensation with gradient matching," *arXiv preprint arXiv:2006.05929*, 2020.

[13] B. Zhao and H. Bilen, "Dataset condensation with differentiable siamese augmentation," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 674–12 685.

[14] G. Cazenavette, T. Wang, A. Torralba, A. A. Efros, and J.-Y. Zhu, "Dataset distillation by matching training trajectories," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 4750–4759.

[15] T. Dong, B. Zhao, and L. Lyu, "Privacy for free: How does dataset condensation help privacy?" in *International Conference on Machine Learning*. PMLR, 2022, pp. 5378–5396.

[16] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.