Federated Domain Generalization with Generalization Adjustment

Ruipeng Zhang[1,2], Qinwei Xu[1,2], Jiangchao Yao[1,2], Ya Zhang[1,2,✉], Qi Tian[3], Yanfeng Wang[1,2,✉]
[1]Cooperative Medianet Innovation Center, Shanghai Jiao Tong University
[2]Shanghai AI Laboratory [3]Huawei Cloud & AI
{zhangrp, qinweixu, Sunarker, ya_zhang, wangyanfeng}@sjtu.edu.cn, tian.qi1@huawei.com

Abstract

Federated Domain Generalization (FedDG) attempts to learn a global model in a privacy-preserving manner that generalizes well to new clients possibly with domain shift. Recent exploration mainly focuses on designing an unbiased training strategy within each individual domain. However, without the support of multi-domain data jointly in the mini-batch training, almost all methods cannot guarantee the generalization under domain shift. To overcome this problem, we propose a novel global objective incorporating a new variance reduction regularizer to encourage fairness. A novel FL-friendly method named Generalization Adjustment (GA) is proposed to optimize the above objective by dynamically calibrating the aggregation weights. The theoretical analysis of GA demonstrates the possibility to achieve a tighter generalization bound with an ***explicit*** *re-weighted aggregation, substituting the* ***implicit*** *multi-domain data sharing that is only applicable to the conventional DG settings. Besides, the proposed algorithm is generic and can be combined with any local client training-based methods. Extensive experiments on several benchmark datasets have shown the effectiveness of the proposed method, with consistent improvements over several FedDG algorithms when used in combination. The source code is released at* https://github.com/MediaBrain-SJTU/FedDG-GA

1. Introduction

Federated Learning (FL) has recently emerged as a prevalent privacy-preserving paradigm for collaborative learning on distributed data [32]. Existing studies mainly investigate the problem of how to improve the convergence and performance of the source clients' data distribution [18, 27, 44]. A more practical problem, how to make models trained on sites of heterogeneous distributions generalize to target clients of unknown distributions, *i.e. Federated Domain Generalization* (FedDG) [30], remains underexplored. While label distribution shift has been considered in traditional FL, FedDG focuses on the domain shift among clients and considers each client as an individual domain. The challenge lies in the domain shift [19] both among the training clients and from training to testing clients.

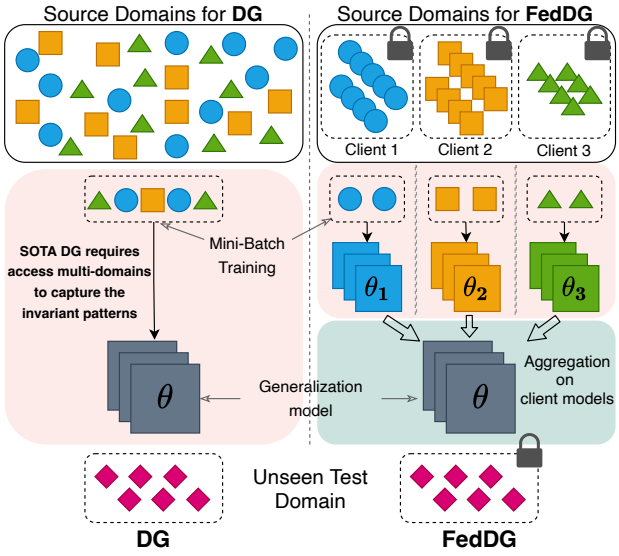


Figure 1. The difference between DG and FedDG is whether the domains are isolated in training. Specifically, previous SOTA DG methods that require access to multiple domains in the mini-batch training are inapplicable to FedDG.

While FedDG shares a similar goal as standard Domain Generalization (DG) [4, 12, 40], *i.e.,* generalizing from multi-source domains to unseen domains, it disallows direct data sharing among clients, as shown in Figure 1, which makes most existing DG methods hardly applicable. Current methods for FedDG focus on unbiased local training within each isolated domain. As the first attempt, Liu *et al.* [30] propose a meta-learning framework

with Fourier-based augmentation during the local training for better generalization. Jiang *et al*. [17] further propose constraining local models' flatness on top of a similar Fourier-based normalization method. However, only focusing on an improved local training strategy cannot guarantee that the global model is generalizable enough to unseen domains. Instead, a common practice for aggregating local models into a global model is by fixed weights as in FedAvg [32], assuming that each client constantly contributes to the global model. Even the subsequent improvements from the federated optimization perspective, *e.g.,* FedNova [44], are mainly designed for the statistical heterogeneity of the same domain, not for the setting of treating each client as an individual domain. Yuan *et al*. [50] have suggested that domains tend to contribute non-equally to the global model and ignoring their differences may significantly reduce the model's generalizability.

As one has no clue regarding to the distribution of unseen domains, it is reasonable to assume that a global model with fair performance among all clients may lead to better generalization performance. We thus introduce a new *fairness* objective measured by the variance of the generalization gaps among different source domains. The data privacy issue in the FL setting has prevented direct optimization of the proposed objective. We thus design a novel privacy-preserving method named Generalization Adjustment to optimize the objective. At the high level, GA leverages the domain flatness constraint, a surrogate of the intractable domain divergence constraint, to approximately explore the optimal domain weights. Technically, we use a momentum mechanism to dynamically compute a weight for each isolated domain by tracing the domain generalization gap, which is then involved in the aggregation of FedDG to enhance the generalization ability. Because the gap information does not contain any domain information of each client, GA will not cause additional risk of privacy leakage. Meanwhile, the theoretical analysis of our method shows that a tighter generalization bound is achieved by setting the aggregation weights inversely proportional to the generalization gaps, which leads to reduced variance in generalization gaps. The contribution of our paper is summarized as follows:

- We introduce a novel optimization objective for FedDG with a new variance reduction regularizer, which can constrain the fairness of the global model.
- We design an FL-friendly method named Generalization Adjustment to tackle the aforementioned novel objective. Our theoretical analysis has revealed that GA leads to a tighter generalization bound for FedDG.
- Extensive experiments on a range of benchmark datasets have shown consistent improvement when combining GA with different federated learning algorithms.

2. Related Work

2.1. Domain Generalization

Domain generalization aims to train a model from multiple source domains that can generalize well on unseen domains. Most DG studies follow the domain alignment idea of minimizing the domain discrepancy across source domains [8, 24, 29, 33, 37, 39, 40, 45] or using a meta-learning strategy to simulate the domain shift [2, 8, 22]. These methods typically require both the shared multi-source domains and their domain labels. Other DG methods that relax the constraint of the domain labels still require the multi-domain data in a mini-batch to achieve cross-domain generalization, such as data augmentations [15, 42, 47, 52–54], self-supervised training [5, 45] and heuristics training [16, 23]. Therefore, most of these methods become inapplicable for privacy reasons in FedDG, and the remaining methods [5, 16, 35, 47] will perform poorly due to the restricted distribution of training data.

2.2. Federated Learning

Federated Learning has been an attractive paradigm for multi-site data collaboration in communication efficiency and privacy preservation [9, 32]. Existing explorations mainly focus on solving the heterogeneity issue in FL from the optimization perspective [14, 18, 27, 36, 44]. The discussed situation is almost the class imbalance among clients [7, 25, 38, 48, 49] instead of the domain shift problem. Some works [26, 31, 44] focus on the re-weighting design but with a different purpose. FedNova [44] aims to correct the objective inconsistency caused by the different local steps and still focuses on convergence. FedCSA [31] reallocates weights of the corresponding parameters of the classifier based on the percentage of each class on the client, respectively. Furthermore, ARFL [26] minimizes the weights on poorly performed clients assuming that clients with high risks have more corrupted data. Recently, Yuan *et al*. [50] propose that there are two gaps in FL. Most existing methods only consider the out-of-sample gap for unseen client data with known distribution without discussing the critical participation gap for unseen client distributions.

2.3. Federated Domain Generalization

Federated Domain Generalization is one emerging research area that considers the generalization ability of the unknown target client with domain shift. To our best knowledge, there are only a few explorations in this direction. Liu *et al*. [30] use the amplitude spectrum on the frequency domain as the data distribution information and exchange them among clients. However, the exchange operations can introduce additional costs and risks of data privacy leakage. Jiang *et al*. [17] further propose a flatness-aware optimization method for better generalization on local updates. An-

other method, like [46], trains the personalized models on each client and selects the most similar personalized local model for the unseen domains. Unlike these methods improved in the local training, we focus on the global aggregation. And we argue that the generalization ability needs to be considered in the global optimization of FL.

3. Method

3.1. Preliminaries

Denote the set of all domains as $\mathcal{D} = \{D_1, D_2, \dots\}$ and the sampled counterpart for training as $\widehat{D} = \{\widehat{D}_1, \widehat{D}_2, \cdots, \widehat{D}_M\}$ where M is the number of training domains (or clients). Let (x, y) denote the sample pair from one domain, and $\mathcal{L}$ denote the loss function measuring the distance between the model prediction $f(x;\theta)$ (parameterized by θ) and the label y. Then, given a domain $D_i \in \mathcal{D}$, we define the expected risk as $\mathcal{E}_{D_i}(\theta) = \mathbb{E}_{(x,y)\in D_i}[\mathcal{L}(f(x;\theta), y)]$, and given a sampled counterpart $\widehat{D}_i = \{x_j^i, y_j^i\}_{j=1}^{N_i}$, we define the empirical risk as $\widehat{\mathcal{E}}_{\widehat{D}_i}(\theta) = \frac{1}{N_i}\sum_{j=1}^{N_i}\mathcal{L}\left(f(x_j^i;\theta), y_j^i\right)$.

The ideal objective of the FedDG is to minimize the overall loss function on $\mathcal{D}$. In practice, we usually have the sampled domains $\widehat{D}$ and the corresponding sampled data points $\{x_j^i, y_j^i\}_{j=1}^{N_i}$ in each domain $\widehat{D}_i$. Thus, instead of the unknown expected risk, we optimize the following empirical risk objective:

$$\min_{\theta} \mathcal{E}_{\mathcal{D}}(\theta) \approx \sum_{i=1}^{M} p_i \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta) = \sum_{i=1}^{M} p_i \sum_{j=1}^{N_i} \mathcal{L}\left(f(x_j^i;\theta), y_j^i\right) \quad s.t.\ p_i = \frac{N_i}{\sum_{i'=1}^{M} N_{i'}}. \tag{1}$$

We shall note two distinct points in FedDG compared to the cross-device federated learning [32]. First, FedDG follows the cross-silo federated learning, and the client/domain number M is small, while the client number in cross-device federated learning is large and the client sampling is usually performed before the global aggregation. Second, although the local data of each client in cross-device federated learning is heterogeneous, they are all from the same overall distribution. In contrast, one client corresponds to one domain in FedDG and the data is not only heterogeneous but from different domains. These make FedDG more challenging than ordinary federated learning. Therefore, in the FedDG scenarios, the global objective in Eq. (1) can easily incur conflicts among local models, and the local training process tends to overfit the local data distribution of each client domain, both of which reduce the generalization performance of the global model.

3.2. Motivation

In centralized learning, generalizable optimization technique, such as invariant risks [1], robust optimization [20], fairness and flatness [6, 10], has been well studied,. However, all of them require data from multiple domains involved in a mini-batch, which is not applicable in FedDG due to its privacy-preserving nature that each domain is isolated into each client. Fortunately, we notice that the flatness of each domain can be reflected by the generalization gaps between the global model and the local model, which is defined as

$$G_{\widehat{D}_i}(\theta) = G_{\widehat{D}_i}(\textstyle\sum_j a_j\theta_j^*) = \widehat{\mathcal{E}}_{\widehat{D}_i}(\textstyle\sum_j a_j\theta_j^*) - \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta_i^*),$$

where θ_i^* means the local optimal on domain $\widehat{D}_i$ and $\theta = \sum_j a_j\theta_j^*$. Based on the above generalization gaps and the design inspiration from [20], we propose a new global objective for FedDG that considers the variance of generalization gaps among local clients to guarantee the flatness of the optimal global model on all domains. The global objective of our method is shown in the following.

$$\min_{\theta_1,\dots,\theta_M,\mathbf{a}} \widehat{\mathcal{E}}_{\widehat{D}}(\theta) = \sum_{i=1}^{M} a_i \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta) + \beta \mathrm{Var}(\{G_{\widehat{D}_i}(\theta)\}_{i=1}^{M}) \quad s.t. \sum_{i=1}^{M} a_i = 1,\ \theta = \sum_{i=1}^{M} a_i \cdot \theta_i,\ \text{and } \forall i,\ a_i \geq 0. \tag{2}$$

Here we denote the learnable client/domain weights as $\mathbf{a} = (a_1, a_2, \cdots, a_M)$, and $\beta \in [0, \infty)$ controls the balance between reducing global risk and enforcing the fairness among generalization gaps, with $\beta = 0$ recovering the FedAvg algorithm, and $\beta \to \infty$ only making the generalization gaps equal. Different from the V-REx objective in [20], we use the generalization gaps other than the risks, and the V-REx can be seen as a particular case of our method when the optimal local risk $\mathcal{E}_{D_i}^*$ is zero. Another method, FedSAM [36], has a similar motivation that constrains the flatness during local training. However, the flatness on the local objectives cannot guarantee the flatness of the overall global

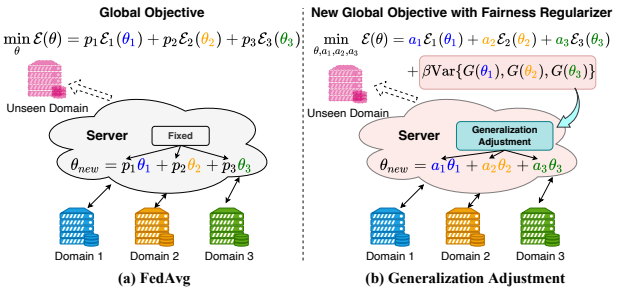


Figure 2. The overall structure of (a) FedAvg and (b) Generalization Adjustment. The *colored* models θ_i and weights a_i are learnable and the weights p_i are fixed during training ($i = 1, 2, 3$). The global model θ_{new} will broadcast to each domain in the next round. GA has a new global objective with a fairness regularizer which will be optimized by dynamically calibrating the weights.

objective. Unlike the FedSAM, our objective can improve the flatness of the global model.

3.3. Generalization Adjustment for FedDG

In Eq. (2), $\mathbf{a}$ and θ_i cannot be simultaneously optimized under the federated learning framework. We observe that there is a relationship between the aggregation weight a_i and the corresponding generalization gap $G_{\widehat{D}_i}(\theta)$ and changing the weights $\mathbf{a}$ can influence the variance of generalization gaps. So we divide the optimization operations into two phases, the local training phase optimizes the model parameters θ_i by gradient descent, and the aggregation phase optimizes the weights a_i by our generalization adjustment algorithm. The main idea of our proposed method is shown in Figure 2 that adjusting the weights to minimize the variance of generalization gaps can obtain a generalizable global model.

The mechanism of the interaction between a_i and $G_{\widehat{D}_i}(\theta)$ is shown below. Formally, given a specific domain k, the relationship between the global parameter θ and the local model parameter θ_k can be expressed as follows,

$$\theta = \theta_k + \Delta\theta, \text{ where } \Delta\theta = (1 - a_k)\theta_k + \textstyle\sum_{i \neq k} a_i \theta_i.$$

Let us consider $\Delta\theta$ as a perturbation on θ_k. For $\widehat{\mathcal{E}}_{\widehat{D}_k}(\theta)$, if we increase a_k, it will correspondingly reduce the percentage of $\{a_i\}_{i \neq k}$ and the degree of perturbation $\Delta\theta$ will decline. In this case, θ is closer to the local optimal θ_k, making the loss $\widehat{\mathcal{E}}_{\widehat{D}_k}(\theta)$ in domain $\widehat{D}_k$ decrease. And in the next round, the new global model θ is used as the initial weight for all clients, which will result in closer proximity between θ_k and $\Delta\theta$, making the mean value of gaps slowly decrease as well. Vice versa if we decrease a_k. And the $\widehat{\mathcal{E}}_{\widehat{D}_k}(\theta_k^*)$ can be considered as a constant, so $G_{\widehat{D}_k}(\theta)$ is different under different a_k.

Following the previous notations, assume we have M clients (*i.e.,* training domains) and the corresponding training sets is $\widehat{D} = \{\widehat{D}_i\}_{i=1}^M$. Denote the total communication round by R, the local epoch during each local training step by E, and the local training algorithm as $\mathcal{A}lg(\theta_i^r, \widehat{D}_i, E)$ that indicates the model θ_i^r is trained on $\widehat{D}_i$ dataset for E epochs on the client i in communication round r. In the client at the communication round r, before the local training, GA requires an extra evaluation for the global model θ^r first. Formally, the following generalization gap will be computed.

$$G_{\widehat{D}_i}(\theta^r) = \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta^r) - \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta_i^{r-1\prime}),\ i = 1, 2, \ldots, M.$$

Then, after the local training by $\mathcal{A}lg(\theta_i^r, \widehat{D}_i, E)$, we send both $G_{\widehat{D}_i}(\theta^r)$ and the updated local model parameter $\theta_i^{r\prime}$ to the global server. On the server side, we design a momentum update to compute the generalization weights $\mathbf{a}^r$. Specifically, given the generalization gaps on all domains $\{G_{\widehat{D}_i}(\theta^r)\}_{i=1}^M$ and the previous weights $\mathbf{a}^{r-1}$, we compute $\mathbf{a}^r$ by the following equations.

$$a_i^{r\prime} = \frac{(G_{\widehat{D}_i}(\theta^r) - \mu) * d^r}{\max_j (G_{\widehat{D}_j}(\theta^r) - \mu)} + a_i^{r-1},\ a_i^r = \frac{a_i^{r\prime}}{\sum_{i=1}^M a_i^{r\prime}}, \quad (3)$$

where $\mu = \frac{1}{M}\sum_{i=1}^M G_{\widehat{D}_i}(\theta^r)$ and $d^r = (1 - r/R) * d$. $d \in (0, 1)$ is a hyperparameter to control the magnitude of each modification, which can be seen as a substitute for β in Eq. (2). A linear-decay strategy is applied to stabilize the training because we empirically find that a fixed step size d could cause instability of the weights in the later phase of training due to the progressive reduction among gaps and also leads to slower convergence at the beginning of training. The maximum magnitude of each adjustment process is limited to d^r and after the adjustment, we clip the $a_i^{r\prime}$ greater than 0 and restrain the sum of $a_i^{r\prime}$ to 1. Then, the global model θ^{r+1} will be aggregated with the generalization weights $\mathbf{a}^r$. At the end of round r, the global model θ^{r+1} will be broadcast to every client as the initialized parameters in the next round. Note that, if the weight at client i is zero, other domains represent this domain well. The overall algorithm of GA is presented in Algorithm 1, which follows the standard implementation of FedAvg [32] and the pink parks are the improvements of our GA. The pseudo-code is presented in the supplementary.

In FedDG, the larger the generalization gap of a client, the worse the global model generalizes in this client, indi-

Algorithm 1: Generalization Adjustment (GA)

Input: Global model $\theta = \theta^0$, M clients $\widehat{D} = \{\widehat{D}_1, \widehat{D}_2, \cdots, \widehat{D}_M\}$, the initial weights $\mathbf{a}^0 = (\frac{1}{M}, \frac{1}{M}, \cdots, \frac{1}{M})$. (Hyperparameters: local epoch E, total communication round R and step size d for GA.)

Output: Global model θ^R.

1: **Server:** initialize the local models θ_i^0 by the global model: $\theta_i^0 = \theta^0$.
2: **for all** r in $0 \cdots R-1$ **do**
3: **Client:**
Compute $G_{\widehat{D}_i}(\theta^r)$ for θ^r on each client.
Training the local model θ_i^r on domain $\widehat{D}_i$:
$\theta_i^{r\prime} = \mathcal{A}lg(\theta_i^r, \widehat{D}_i, E)$.
Get the empirical loss on local model $\widehat{\mathcal{E}}_{\widehat{D}_i}(\theta_i^{r\prime})$.
4: **Server:**
Update $\mathbf{a}^r$ by $\mathbf{a}^{r-1}$ and $\{G_{\widehat{D}_i}(\theta^r)\}_{i=1}^M$:
$\mathbf{a}^r = \mathbf{GA}(\mathbf{a}^{r-1}, \{G_{\widehat{D}_i}(\theta^r)\}_{i=1}^M, d^r)$.
Aggregate θ_i^{r+1} with $\mathbf{a}^r$ to get a new global model:
$\theta^{r+1} = \sum_{i=1}^M a_i^r \cdot \theta_i^{r\prime}$.
5: Broadcast θ^{r+1} to all clients $\theta_i^{r+1} = \theta^{r+1}$.
6: **end for**

cating that the global model has more domain bias toward other clients. If we increase the weights on the large gap's client, the corresponding gap will be reduced. Although the other domains' gaps will increase, we observed that for well-generalized clients, the weights reduction has less impact on the corresponding gaps. Besides, our GA method can improve the out-of-domain generalization ability on top of agnostic local training methods with little additional cost and no additional privacy risk. The GA method dynamically adjusts the weights only through the generalization gaps during training, which can better protect the privacy of each client and prevent the model from biasing to clients with a larger sampling size. Previous methods [17,36] mainly constrain the local models to be stable with the global model or align the features across clients during local training. In contrast, our GA pays attention to the aggregation phase in the server. Actually, GA is orthogonal and compatible with most existing methods since we do not place a restriction on the local training phase.

3.4. Theoretical analysis

We first prove that the generalization gap on an unseen domain T by the optimal solution on $\widehat{\mathcal{E}}_{\widehat{D}}(\theta)$ is upper bounded by the generalization gap in source domains in Theorem 1 (see proof in the supplementary). From Theorem 1, the domain generalization gap on unseen domain T is bounded by the domain flatness $G_{\widehat{D}_i}(\theta)$ and the domain divergence $d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T)$. Conventional DG can optimize the domain divergence $d_{\mathcal{H}\Delta\mathcal{H}}$ by implicit multi-domain alignment to tighten the bound, while FedDG cannot directly implement this due to only one domain available in each client. In this case, FedDG can actually make the bound tighter through intervention on $\mathbf{a}$.

Theorem 1. *Let θ denote the global model after R round federated learning, θ_i^* and θ_T^* mean the local optimal for each source domain and the unseen target domain, respectively. We use the generalization weights $\mathbf{a}$ from Eq.(5). For any $\delta \in (0, 1)$, the domain generalization gap for the unseen domain T can be bounded by the following equation with a probability of at least $1 - \delta$.*

$$\mathcal{E}_T(\theta) - \mathcal{E}_T(\theta_T^*) \leq \sum_{i=1}^{M} a_i \left(G_{\widehat{D}_i}(\theta) + d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T) + \frac{\sqrt{\log \frac{d}{\delta}} + \sqrt{\log \frac{Md}{\delta}}}{\sqrt{2N_i}} \right) + \lambda \tag{4}$$

However, the accurate estimation of $\mathbf{a}$ requires knowledge of the domain divergence $d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T)$ and the partial derivatives $\partial\widehat{\mathcal{E}}_{\widehat{D}}(\theta)/\partial a_i$ on each domain, which is intractable without the information about T in practice. To overcome this dilemma, we explore the surrogate way to approximately compute $\mathbf{a}$. According to [6], the global model can be approximated as the solution of robust risk minimization on client i, *i.e.,* $\widehat{\mathcal{E}}_{\widehat{D}_i}(\theta) = \max_{||\Delta\theta|| \leq \gamma} \widehat{\mathcal{E}}_{\widehat{D}_i}(\theta_i + \Delta\theta)$. Since the optimal estimation requires the flatness of all domains should be similar, we can optimize $\mathbf{a}$ to pursue the variance minimization on the flatness of all domains. Then, the optimal generalization weights $\mathbf{a}$ on the domain divergence term can be approximately obtained by solving the following variance minimization problem regarding the domain flatness.

$$\begin{aligned} &\arg\min_{\mathbf{a}} \sum_{i=1}^{M} a_i d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T) \\ &\quad \approx \arg\min_{\mathbf{a}} \mathrm{Var}\left(\left\{ G_{\widehat{D}_1}(\theta), \ldots, G_{\widehat{D}_M}(\theta) \right\} \right) \end{aligned} \tag{5}$$

As Eq. (5) targets to have a zero variance, we will approximately achieve a constant flatness C for all domains in the optimal, namely $\forall i,\ G_{\widehat{D}_i}(\theta) = \mathrm{C}$. In this case, we have an equality $\sum_{i=1}^{M} a_i G_{\widehat{D}_i}(\theta) = \sum_{i=1}^{M} p_i G_{\widehat{D}_i}(\theta)$ satisfied. Besides, considering $\mathbf{a}$ in Eq. (5) is an optimal, we have another inequality $\forall \mathbf{p}, \min \sum_{i=1}^{M} a_i d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T) \leq \min \sum_{i=1}^{M} p_i d_{\mathcal{H}\Delta\mathcal{H}}(\widehat{D}_i, T)$. Putting them together, we can conclude that the bound with $\mathbf{a}$ in Theorem 1 is tighter than the bound where we substitute $\mathbf{a}$ with the constant $\mathbf{p} = \{p_1, p_2, \ldots, p_M\}$ as in FedAvg. It indicates that the proposed global objective with GA can achieve a better generalization on unseen target domains.

4. Experimental Results

4.1. Dataset and implementation details

We evaluate our proposed method on four widely used DG benchmarks. Namely, **PACS** [21] (9,991 images, four domains), **OfficeHome** [41] (15,588 images, four domains), **TerraInc** [3] (24,788 images, four domains) and **DomainNet** [34] (569,010 images, six domains). We carry out leave-one-domain-out evaluations for all benchmarks, which means, by turn, we select one domain as the unseen client, and all the left domains are used as the source clients for training. The split of train and validation set within each source domain is kept the same as that in [13, 16, 47] for PACS, OfficeHome and TerraInc and [28] for DomainNet, and the whole target domain is used for testing.

As for the local training, we follow the protocols used in [47] and [13]. Specifically, we use the ImageNet pretrained ResNet18 for PACS and OfficeHome, ResNet50 for TerraInc and AlexNet for DomainNet (the same model structure as in [28]). For fairness, the batch size and learning rate are set to 16 and 0.001 during local training in all the experiments. To guarantee that the local model converges within the local training phase of each round, we set the number of local epochs E to 5, and the number of total communication rounds R to 40. The step size of our GA

Table 1. **Results on three benchmarks (PACS, OfficeHome, TerraInc) under the FedDG setting**. The results on each dataset are the average of four leave-one-domain-out cases. "+*GA*" represents aggregation with our generalization weight a_i.

Method	PACS					OfficeHome					TerraInc					Avg.
	P	A	C	S	Avg.	P	A	C	R	Avg.	L38	L100	L43	L46	Avg.	
ARFL	92.10	76.25	75.79	80.47	81.15	73.89	56.98	53.18	73.16	64.30	56.83	40.04	41.58	30.81	42.32	62.59
FedAvg	92.77	77.29	77.97	81.03	82.26	72.72	57.60	52.28	73.88	64.12	52.66	40.56	41.56	36.91	42.92	63.10
+*GA*	93.97	81.28	76.73	82.57	**83.64**	73.39	58.57	54.39	74.73	**65.27**	54.36	41.66	48.68	40.43	**46.28**	**65.06**
FedCSA	91.88	77.00	76.79	80.84	81.63	72.96	56.08	52.51	72.79	63.58	54.33	41.08	41.52	33.51	42.61	62.61
+*GA*	94.12	79.30	77.69	81.62	**83.18**	72.96	57.58	53.99	73.98	**64.63**	54.91	44.74	46.90	38.53	**46.27**	**64.69**
FedNova	94.03	79.93	76.39	79.26	82.40	73.72	58.81	49.89	73.33	63.94	56.80	38.96	42.49	31.99	42.56	62.97
+*GA*	94.13	81.30	77.73	80.30	**83.37**	72.58	57.89	54.25	73.86	**64.65**	55.15	41.55	47.05	35.25	**44.75**	**64.26**
FedProx	93.15	77.72	77.73	80.77	82.34	73.37	58.76	52.67	73.88	64.67	54.00	39.84	43.90	38.31	44.01	63.67
+*GA*	94.91	80.24	77.20	81.48	**83.46**	73.81	58.28	54.03	74.80	**65.23**	54.03	40.93	49.28	38.84	**45.77**	**64.82**
FedSAM	91.20	74.45	77.77	83.35	81.69	73.58	55.34	54.75	73.74	64.35	57.21	38.24	40.21	31.24	41.73	62.59
+*GA*	92.87	77.76	77.86	85.16	**83.41**	73.29	55.21	56.82	74.49	**64.95**	60.04	38.95	48.39	37.43	**46.20**	**64.85**
HarmoFL	90.99	74.51	77.43	81.73	81.16	73.89	57.44	53.42	74.95	64.93	60.04	38.57	39.21	33.87	42.92	63.01
+*GA*	93.83	77.39	77.07	82.51	**82.70**	73.76	58.14	54.44	75.74	**65.53**	61.81	38.53	46.65	37.96	**46.24**	**64.82**
Scaffold	92.50	78.09	77.23	80.67	82.12	72.16	59.00	52.78	73.22	64.29	54.10	37.28	45.09	38.38	43.71	63.37
+*GA*	94.79	80.14	76.91	82.12	**83.49**	73.45	57.93	54.42	74.62	**65.10**	55.40	39.74	50.08	39.68	**46.22**	**64.94**
AM	93.29	80.86	77.62	81.05	83.20	73.24	58.76	51.87	73.84	64.42	57.36	37.43	45.00	33.60	43.35	63.66
+*GA*	94.03	83.19	76.85	82.93	**84.25**	73.67	58.80	54.28	74.72	**65.37**	56.30	40.55	49.42	38.08	**46.08**	**65.23**
RSC	92.67	77.98	77.80	82.90	82.91	73.26	57.44	50.31	73.42	63.61	54.25	41.61	43.94	35.55	43.84	63.45
+*GA*	93.79	81.69	77.23	82.75	**83.87**	72.35	58.55	51.42	75.01	**64.33**	54.87	43.93	50.08	39.04	**46.98**	**65.06**

method is set to 0.05 by default. All the reported results are averaged over three runs.

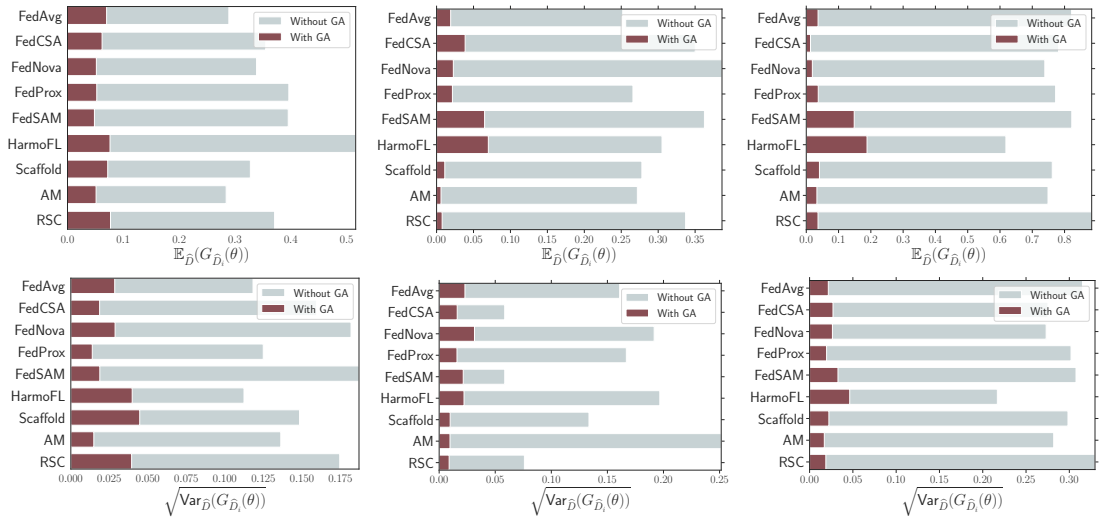
Table 2. **Results on the DomainNet benchmarks under the FedDG setting.** "+*GA*" represents using weight a_i by our GA.

Method	C	I	P	Q	R	S	Avg.
ARFL	69.93	31.32	60.68	57.45	67.76	60.62	57.96
FedAvg	67.92	32.77	60.27	52.90	68.72	61.15	57.29
+*GA*	71.86	34.40	63.25	57.50	67.26	67.15	**60.24**
FedCSA	68.94	33.67	61.66	58.25	67.25	61.15	58.49
+*GA*	70.34	33.71	64.22	56.85	66.06	67.33	**59.72**
FedNova	68.45	32.95	61.70	59.05	67.21	61.57	58.49
+*GA*	73.29	34.09	64.38	57.05	68.08	65.25	**60.36**
FedProx	68.55	32.12	60.79	55.63	68.17	61.20	57.75
+*GA*	69.39	33.26	63.25	57.15	67.50	65.79	**59.39**
FedSAM	66.47	34.97	56.90	51.11	66.28	55.82	55.26
+*GA*	72.62	36.30	64.62	57.75	69.35	65.43	**61.01**
HarmoFL	71.39	34.70	61.23	57.50	67.01	56.77	58.10
+*GA*	72.81	35.77	63.73	59.30	68.08	64.35	**60.67**
Scaffold	68.04	33.20	60.60	54.39	67.32	60.88	57.41
+*GA*	71.67	34.93	62.20	57.70	67.46	66.97	**60.16**
AM	71.91	32.54	63.70	56.87	67.80	69.42	60.37
+*GA*	74.33	35.31	64.54	58.55	68.61	72.02	**62.23**
RSC	70.96	34.25	60.31	55.20	66.91	63.84	58.58
+*GA*	71.96	35.62	62.52	56.95	67.13	64.98	**59.86**

4.2. Main results

Compared methods We select several representative methods in the area of DG and FL for comparison. The baseline method is FedAvg [32], which acts as a strong baseline under the FedDG paradigm. As for the existing methods from the domain generalization perspective, we choose two methods. A regularization-based DG method called RSC [16] that can be migrated to FedDG and a powerful Fourier-based augmentation method named Amplitude Mix (AM). AM is widely used in many DG methods [43, 47, 51] and the main component of FedDG-ELCFS proposed by [30] (We remove the loss designed for the segmentation task). As for the federated learning methods designed for data heterogeneity, we select several approaches. Four methods, FedProx [27], HarmoFL [17], Scaffold [18] and FedSAM [36] which are all initially proposed to solve the heterogeneous issue across clients, and three re-weight methods: ARFL [26] is also a dynamic re-weighting method that focuses on the convergence under corrupted data sources and is unable to combine with our GA; FedCSA [31] aims to calibrate the class-imbalance problem among clients; FedNova [44] tries to correct the objective inconsistency caused by the different local update steps for better convergence.

Comparison with existing methods We report the overall performance on four FedDG benchmarks in Table 1 and Table 2. According to the comparison, our GA achieves significant and consistent improvements when combined with different algorithms on different datasets. We observe that when equipped with our GA method, the domain generalization performances could be largely improved in most cases, regardless of the type of algorithms on four benchmarks in Table 1 and Table 2. These results demonstrate the superiority and generalizability of our GA method, com-



(a) PACS (b) OfficeHome (c) TerraInc

Figure 3. **Comparison of the generalization gap on each source domain.** We show the mean (the first row) and standard derivation (the second row) of the source domains' generalization gaps at the end of the training process. The results with our GA are shown in the light pink part of each bar, which is significantly smaller than the base method.

pared to the conventional FedAvg that might overfit in the local training phase. DG algorithms like AM and RSC enjoy a relatively more significant performance improvement when equipped with GA than the FL algorithms. It indicates that our GA can better complement the unbiased local training strategy involved in classic DG solutions. Those FL methods like FedProx and Scaffold improve the convergence consistency and speed of the global model while reducing the generalization during the local training. FedNova can improve convergence under challenging tasks in the DomainNet benchmark, but there is no significant help for the generalization ability under domain shift. Especially the results of FedSAM and HarmoFL are also poor, which means the operation to improve the flatness during the local training process cannot guarantee flatness and generalization under domain shifts. However, our GA can effectively unlock their potential with significant improvements on unseen domains. Moreover, the re-weight based methods FedCSA and ARFL cannot handle the domain shift problem in the FedDG because of the different weight adjustment objectives. Notably, as an FL-friendly model aggregating mechanism, GA could be easily applied to any existing DG or FL algorithms at a relatively low cost, which makes our method a general technique for FedDG.

4.3. Ablation studies

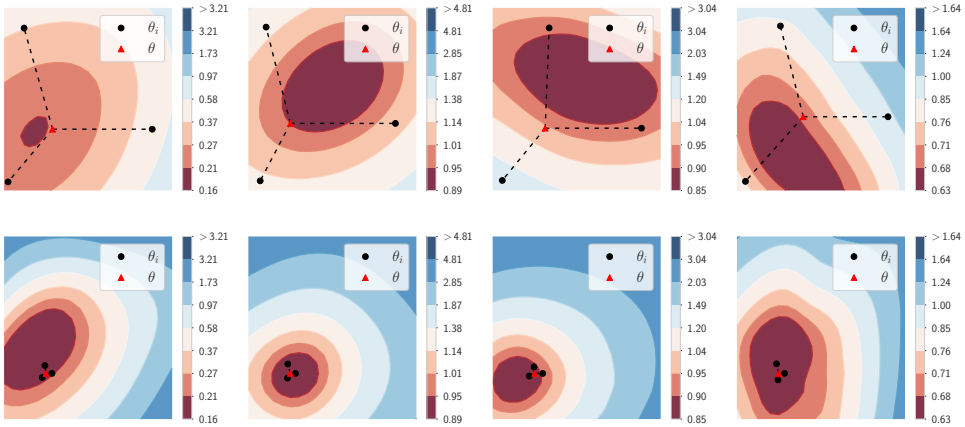
Ablation studies of the step size and linear decay strategy. We provide the ablation studies on different step sizes and the linear-decay strategy in Table 3. According to the results, the aggregated global model shows a more favorable performance when equipped with the linear-decay strategy.

Table 3. **Ablation study on the step sizes and the linear-decay strategy in GA.** (*with (w.)* or *without (wo.)* decay). Note that step size 0 makes the decay no effect; in this case, the weights degenerated to the uniform on each domain. "Fix" means the aggregation with the original weight p_i that follows the sample ratio in each domain and is not uniform. The experimental backbone algorithm is FedAvg.

Step	PACS	OfficeHome	TerraInc	Avg.
	w. / wo.	*w. / wo.*	*w. / wo.*	*w. / wo.*
0.2	82.83/83.19	65.22/65.23	46.19/45.24	64.75/64.55
0.1	**83.81**/83.07	64.88/64.71	46.40/45.60	65.03/64.46
0.05	83.64/83.35	**65.27**/64.88	46.28/46.39	**65.06**/64.87
0.01	83.64/83.23	64.54/65.17	45.78/**46.78**	64.65/65.05
0	82.44	64.56	44.38	63.79
Fix	82.26	64.12	42.92	63.10

Besides, we observe that the improvement of GA is not sensitive to the specific choices of step size and is consistent with or without the linear decay strategy, which shows the stability of our method. More surprisingly, we observe that with step size 0, the domain weights in GA will degenerate to $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$, which can still achieve better performance than fixed p_i which is the sample number percentage. It demonstrates that the original weights in federated learning are not proper for FedDG. GA utilizes the approximately optimal weights to affect the aggregation, which makes the global model achieves a better generalization.

Comparison of the domain generalization gaps. We illustrate the mean and the standard deviation of the generalization gap on each source domain with or without GA in Figure 3. The short dark red parts represent the gaps by the generalization weights, and the much longer bars represent



(a) Photo (b) Art (c) Cartoon (d) Sketch

Figure 4. **Loss surfaces w.r.t. model parameters on the PACS dataset for each target domain.** We plot three local models θ_i and the corresponding global model θ on the target domain's loss surface. The first and second rows represent the loss surfaces of FedAvg with Fixed and FedAvg with GA weights. GA significantly reduces the gaps between θ_i and θ and induces a flatter area on the unseen test domain. (Best viewed in color)

the gaps with originally fixed weights p_i from baselines. It can be seen that our method has a significant effect on the gap reduction. Both the mean and the standard deviation are decreased after training with our GA, which corroborates our analysis and empirically validates the gap-aware generalization adjustment is an effective way to avoid the domain bias during the aggregation in FedDG, as shown in Eq.(4). It also corresponds to the intuition that smaller gaps can increase the generalization ability on the unseen domain. Especially in Figure 3 (c), we find that the gap reduction on the TerraInc is very significant. Correspondingly the performance improvement in the generalization ability is the largest among the three benchmarks.

Loss surface visualization. We visualize the loss surfaces on the test domain for the PACS benchmark dataset in Figure 4, which takes the global model θ as the anchor and locates the local models $(\theta_1, \theta_2, \theta_3)$. The first row of Figure 4 represents the vanilla FedAvg method, and the second row is our FedAvg with GA. We use the same visualization technique in [11].

We observe that in all cases, our GA method induces a more generalizable solution on the target domain, as both the global and local models converge in the flatter area of the loss surfaces. It confirms our theoretical motivation that the weights minimizing the variance of the domain gaps make us achieve a tighter generalization bound. Note that the loss surfaces are drawn in the unseen target domains, which indicates better convergence, consistency and generalization of global models with our GA method. Moreover, we can see that length of the dashed lines, *i.e.,* the gaps between global and local models, are much smaller for FedAvg with GA. It shows the advantage of GA in maintaining a more consistent optimization objective among different clients, which is critical for heterogeneous data under diverse domains.

5. Conclusion and Discussion

In this paper, we propose a new method to solve the FedDG problem that adjusts the aggregation weights to pursue better out-of-domain generalization. Specifically, we propose a new global optimization objective that encourages flatness and fairness by reducing the variance of generalization gaps for the FedDG and then design an FL-friendly method named Generalization Adjustment to dynamically calibrate the weights for achieving the proposed objective. Our theoretical analysis shows that our new objective and GA method can achieve a tighter generalization bound on top of other methods. Extensive experiments on several FedDG benchmarks prove the generic of our GA method and also show the effectiveness of minimizing the source domains' generalization gap. However, it is still an open problem to evaluate the optimal generalization weights by optimization directly, and our GA method that approximately estimates the generalization weights might not be the best. GA may be over-confident in the domains that are easy to fit and have inconsistent gradients. We hope this study can promote more practical progress for FedDG and the broader out-of-domain generalization tasks under the privacy-preserving context.

Acknowledgement: This work is supported by the National Key R&D Program of China (2022ZD0160702), STCSM (22511106101, 18DZ2270700, 21DZ1100100), 111 plan (BP0719010), and State Key Laboratory of UHD Video and Audio Production and Presentation. Ruipeng Zhang is partially supported by Wu Wen Jun Honorary Doctoral Scholarship, AI Institute, Shanghai Jiao Tong University.

References

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 3

[2] Yogesh Balaji et al. Metareg: Towards domain generalization using meta-regularization. In *NeurIPS*, pages 998–1008, 2018. 2

[3] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *ECCV*, pages 456–473, 2018. 5

[4] Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *NeurIPS*, 24, 2011. 1

[5] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *CVPR*, pages 2229–2238, 2019. 2

[6] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. *NeurIPS*, 34, 2021. 3, 5

[7] Jiahua Dong, Lixu Wang, Zhen Fang, Gan Sun, Shichao Xu, Xiao Wang, and Qi Zhu. Federated class-incremental learning. In *CVPR*, pages 10164–10173, 2022. 2

[8] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, pages 6450–6461, 2019. 2

[9] Ziqing Fan, Yanfeng Wang, Jiangchao Yao, Lingjuan Lyu, Ya Zhang, and Qi Tian. Fedskip: Combatting statistical heterogeneity with federated skip aggregation. In *ICDM*, pages 131–140, 2022. 2

[10] Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *ICLR*, 2021. 3

[11] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In *NeurIPS*, 2018. 8

[12] Muhammad Ghifary, David Balduzzi, W Bastiaan Kleijn, and Mengjie Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1414–1430, 2016. 1

[13] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *ICLR*, 2021. 5

[14] Yuanxiong Guo, Ying Sun, Rui Hu, and Yanmin Gong. Hybrid local SGD for federated learning with heterogeneous communications. In *ICLR*, 2022. 2

[15] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Fsdr: Frequency space domain randomization for domain generalization. In *CVPR*, pages 6891–6902, 2021. 2

[16] Zeyi Huang, Haohan Wang, E. Xing, and D. Huang. Self-challenging improves cross-domain generalization. In *ECCV*, 2020. 2, 5, 6

[17] Meirui Jiang, Zirui Wang, and Qi Dou. Harmofl: Harmonizing local and global drifts in federated learning on heterogeneous medical images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1087–1095, 2022. 2, 5, 6

[18] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *ICML*, pages 5132–5143. PMLR, 2020. 1, 2, 6

[19] Aditya Khosla, Tinghui Zhou, Tomasz Malisiewicz, Alexei A Efros, and Antonio Torralba. Undoing the damage of dataset bias. In *ECCV*, pages 158–171. Springer, 2012. 1

[20] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). In *ICML*, pages 5815–5826. PMLR, 2021. 3

[21] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5542–5550, 2017. 5

[22] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018. 2

[23] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *ICCV*, pages 1446–1455, 2019. 2

[24] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *CVPR*, pages 5400–5409, 2018. 2

[25] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. *CoRR*, abs/2102.02079, 2021. 2

[26] Shenghui Li, Edith Ngai, Fanghua Ye, and Thiemo Voigt. Auto-weighted robust federated learning with corrupted data sources. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2022. 2, 6

[27] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems*, 2:429–450, 2020. 1, 2, 6

[28] Xiaoxiao Li, Meirui Jiang, Xiaofei Zhang, Michael Kamp, and Qi Dou. Fed{bn}: Federated learning on non-{iid} features via local batch normalization. In *ICLR*, 2021. 5

[29] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, pages 624–639, 2018. 2

[30] Quande Liu, Cheng Chen, Jing Qin, Qi Dou, and Pheng-Ann Heng. Feddg: Federated domain generalization on medical image segmentation via episodic learning in continuous frequency space. *CVPR*, 2021. 1, 2, 6

[31] Zezhong Ma, Mengying Zhao, Xiaojun Cai, and Zhiping Jia. Fast-convergent federated learning with class-weighted aggregation. *Journal of Systems Architecture*, 117:102125, 2021. 2, 6

[32] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data.

In *Artificial intelligence and statistics*, pages 1273–1282. PMLR, 2017. 1, 2, 3, 4, 6

[33] Krikamol Muandet, David Balduzzi, Bernhard Schölkopf, et al. Domain generalization via invariant feature representation. In *ICML*, pages 10–18, 2013. 2

[34] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019. 5

[35] Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *CVPR*, pages 12556–12565, 2020. 2

[36] Zhe Qu, Xingyu Li, Rui Duan, Yao Liu, Bo Tang, and Zhuo Lu. Generalized federated learning via sharpness aware minimization. In *ICML*, 2022. 2, 3, 5, 6

[37] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *CVPR*, pages 10023–10031, 2019. 2

[38] Zebang Shen, Juan Cervino, Hamed Hassani, and Alejandro Ribeiro. An agnostic approach to federated learning with class imbalance. In *ICLR*, 2022. 2

[39] Yuge Shi, Jeffrey Seely, Philip Torr, Siddharth N, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. Gradient matching for domain generalization. In *ICLR*, 2022. 2

[40] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, pages 443–450. Springer, 2016. 1, 2

[41] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017. 5

[42] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, pages 5334–5344, 2018. 2

[43] Jingye Wang, Ruoyi Du, Dongliang Chang, and Zhanyu Ma. Domain generalization via frequency-based feature disentanglement and interaction. *CoRR*, abs/2201.08029, 2022. 6

[44] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H Vincent Poor. Tackling the objective inconsistency problem in heterogeneous federated optimization. *NeurIPS*, 33:7611–7623, 2020. 1, 2, 6

[45] Shujun Wang, Lequan Yu, Caizi Li, Chi-Wing Fu, and Pheng-Ann Heng. Learning from extrinsic and intrinsic supervisions for domain generalization. In *ECCV*, 2020. 2

[46] An Xu, Wenqi Li, Pengfei Guo, Dong Yang, Holger Roth, Ali Hatamizadeh, Can Zhao, Daguang Xu, Heng Huang, and Ziyue Xu. Closing the generalization gap of cross-silo federated medical image segmentation. *arXiv preprint arXiv:2203.10144*, 2022. 3

[47] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *CVPR*, pages 14383–14392, 2021. 2, 5, 6

[48] Rui Ye, Zhenyang Ni, Chenxin Xu, Jianyu Wang, Siheng Chen, and Yonina C Eldar. Fedfm: Anchor-based feature matching for data heterogeneity in federated learning. *arXiv preprint arXiv:2210.07615*, 2022. 2

[49] Jaehong Yoon, Wonyong Jeong, Giwoong Lee, Eunho Yang, and Sung Ju Hwang. Federated continual learning with weighted inter-client transfer. In *ICML*, pages 12073–12086. PMLR, 2021. 2

[50] Honglin Yuan, Warren Richard Morningstar, Lin Ning, and Karan Singhal. What do we mean by generalization in federated learning? In *ICLR*, 2022. 2

[51] Ruipeng Zhang, Qinwei Xu, Chaoqin Huang, Ya Zhang, and Yanfeng Wang. Semi-supervised domain generalization for medical image analysis. In *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, pages 1–5. IEEE, 2022. 6

[52] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, 2020. 2

[53] Kaiyang Zhou, Yongxin Yang, Timothy M Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020. 2

[54] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 2