Peer-to-Peer Learning with Iterative Parameter Alignment

Anonymous Author(s)
Affiliation
Address
email

Abstract

Learning from the collective knowledge of data dispersed across private sources can provide neural networks with enhanced generalization capabilities. Federated learning, a method for collaboratively training a machine learning model across remote devices, achieves this by aggregating client models via the orchestration of a central server. In this work, we reformulate the typical federated learning setup: rather than learning a single global model, we learn N peer models optimized for a common objective. To achieve this, we apply a weighted distance minimization to model parameters shared in a decentralized topology. The resulting framework, Iterative Parameter Alignment, applies naturally to the cross-silo setting, and has the key properties: (i) a unique solution for each participant, with the option to globally converge each model in the federation, and (ii) an early-stopping mechanism to elicit1 *fairness* among peers in collaborative learning settings. These characteristics jointly provide a flexible new framework for iteratively merging models trained on disparate datasets. We find that the technique achieves competitive results on heterogeneous data partitions compared to state-of-the-art approaches. Further, we show that the method is robust to divergent peer domains (i.e. disjoint classes across peers) where existing approaches struggle.

1 Introduction

Modern machine learning is driven heavily by its ability to learn from abundant data. However, concerns such as privacy, security, and access rights limit the capability to pool data sources. Federated Learning (FL) has emerged as a promising direction for addressing these issues, enabling wide-scale training of machine learning models across decentralized data [50, 79, 40].

Standard federated learning involves clients (e.g. mobile, edge devices) training a model locally with private data and communicating their model updates to a central server for aggregation with other client models. The server produces a global model to return to each client. This process repeats iteratively until a final global model is produced. In this scenario, the server is responsible for facilitating training, iteratively communicating with clients, and generating a global model for client use [50, 33, 19]. This client-server architecture is most effective in the *cross-device* setting where clients often consist of unreliable devices, such as those with limited computational capacity or slow network communication. A second setting, *cross-silo* FL, consists of reliable clients such as large organizations and companies with data silos (i.e. data centers) such as banks [1, 15] and hospitals [11, 54, 63, 17]. Clients with data-silos often have extensive computational resources as well as strong network communication [24]. Further, the setting often contains substantially fewer clients compared to cross-device FL. In cross-silo FL, clients may have as much capacity as the orchestrator; and the central server may in fact *inhibit* the speed of learning due to a communication bottleneck. A natural direction is to then replace the client-server architecture with peer-to-peer communication between individual silos [30, 69, 5, 9, 36, 89, 67, 64, 70], a topology that was studied in depth by Marfoq et al. [48]. In this work we concentrate on the cross-silo FL setting.

Submitted to 37th Conference on Neural Information Processing Systems (NeurIPS 2023). Do not distribute.

Essential to federated learning is the security of both client data *and* the client's final model. Localizing data on client devices served as a baseline for data protection in initial work [51]; differential privacy was proposed to address data security vulnerabilities identified in subsequent research [43, 20, 3, 29]. Protecting client models, however, is a more ambiguous task. Methods such as homomorphic encryption [85, 28] enable clients to encrypt their models so that the server can perform computation (i.e. aggregation) on the encryption. A single global model is aggregated and sent back to each client. In other words, each client ends up with the same global model during each round of the training process, and the final global model is identical for each client.

Aside from generic federeated learning, *personalized* FL offers the option for clients to produce individualized models unique to their data distribution, while also utilizing the full dataset of the federation. This can be achieved by simply fine-tuning the global model on one's individual dataset [30], or using more complex technqiues such as hypernetworks [62], shared feature extractors [42, 10], or encouraging interaction between related clients [8, 25]. Other techniques have reconciled the gap between personalized and generic FL, such as [7, 8] who both show that we can build accurate personal and global models simultaneously. However, in both cases the global model is the same across clients.

Motivation. In this work, we address a previously overlooked characteristic of existing federated learning research: *the global model is identical for each participant*. This property can lead to several important disadvantages for clients. First, the global model for each client is exposed to other participants in the federation. In the cross-silo setting this may leave your model unprotected against direct competitors, exposing obvious vulnerabilities such as white-box adversarial attacks [21]. Along these same lines, other clients in the federation may be able to reconstruct your private data [72, 80] in the case that it hasn't been trained with differential privacy techniques. Finally, the client-server architecture exposes the system to a single point a failure at the server, another limitation of generating a single global model. Aside from federated learning, emerging areas of interest include transfer learning and model merging...

Our Approach. Iterative Parameter Alignment is a decentralized framework practical in cross-silo settings. It produces a distinct model for each peer that is optimized for a common objective. The approach works by iteratively merging the parameters of peer models together during standard training until each peer learns a sufficient model. Figure 1 visualizes the approach. Our approach is robust to heterogeneous data sources, a known burden of generic FL [50]. We achieve competitive convergence results compared to other centralized approaches for heterogeneous data partitions [2, 19, 40, 33]. Further, the Iterative Parameter Alignment converges to baseline accuracy in scenarios with completely segregated labeling across peers (e.g. one peer has data of animals and another objects). Our method produces different models for each peer, which we analyze. The framework additionally contains a *built-in* incentive mechanism: model convergence in image classification tasks is a function of the amount of data the peer has provided as well as the data homogeneity. We additionally discuss several considerations for the decentralized setting.

Contributions. We present a novel decentralized framework that enables peers to learn their own unique model optimized for a global objective on image classification tasks. Specifically, we propose an algorithm that performs an iterative parameter alignment across peers during training. Our method is robust to heterogeneous data partitions. Additionally, we show that the framework contains a built-in incentive mechanism, which we analyze on image classification tasks.

2 Related Work

Federated Learning. The pioneering FL framework, Federated Averaging (`FedAvg`), aggregated a global model by averaging the weights of client models trained on private data [50]; heterogeneous data partitioning and inefficient communication across clients were identified as key challenges [35, 74, 37]. Subsequent work improved the convergence rate of heterogeneous client data through corrections to the gradients of local models [33], regularization of local models against the global model [40], feature alignment [82], dynamic regularization of local models [2], and correcting local model drift from the global model [19].

Collaborative Learning Important to federated learning is designing incentive mechanisms for peers to participate in a federation, sometimes referred to as collaborative learning. For example, cross-silo FL typically involves large organizations in a related industry who are often business competitors. As a result, a participant may have concerns with contributing their data for the benefit

of others. As a result, *fairness* schemes have been proposed using methods such as contract theory [31, 32], monetary payouts [83], and game-theoretic approaches [13, 6]. Lyu et al. [47] propose a credibility metric so that each participant receives a different version of the global model with performance comparable to its contribution. Similar to our work the authors use a decentralized framework, utilizing a differentially private GAN to share local samples for benchmarking. Xu et al. [78] propose a reward mechanism which sparsifies model updates at the server commensurate to a clients contributions. Other works utilize the Shapely value [45] and reputation lists [46] to evaluate client contributions.

Personalized Federated Learning. Personalized FL produces individualized models that are catered to a clients' data distribution while also leveraging the data of the federation [16]. Clients can create personalized models via local fine-tuning of the global model [30], or from more advanced techniques such as hypernetworks [62], pruning [68], encouraging interaction between related clients [66, 86, 8, 25], and learning client-level and shared feature extractors [42, 10]. Research also addresses *fairness* in personalized FL [52, 41], identifying performance disparity across clients as a key issue. Finally, other work has attempted to reconcile the gap in accuracy between global and personalized models [7, 8], showing that we can simultaneously create both. Unlike existing research in personalized FL, our work aims to learn an individualized model on a common objective for each peer in the network. Additionally, our work is the first to do so in a decentralized network topology.

Decentralized Learning. Kairoz et al. [30] identified the central server as a potential single point of failure in the generic FL framework.Fully decentralized algorithms have been proposed for personalized FL using gossip algorithms to smooth client models with similar objectives [69, 5] and similarity graphs between clients [84]. [38] propose peer-to-peer collaboration with neighbors to learn a (single) global model; [9] address decentralization through minimization of pairwise functions with gossip dual averaging. Other research in decentralized learning environments improves communication efficiency via compression [36], including in heterogeneous settings [89]. Finally, variants of SGD have been proposed to improve training in decentralized network topologies [67, 64, 70]. In contrast to existing research, Iterative Weight Alignment introduces a unique decentralized framework for training generic FL models across peers.

Cross-Silo Federated Learning Cross-silo FL involves training machine learning models across entities with large data-silos such as banks [15] and hospitals [63, 17, 54]. Distinct from cross-device FL, cross-silo FL involves training models across data silos that often have large quantities of valuable data as well as extensive computational resources. Peer-to-peer communication has been proposed as an effective alternative to centralized orchestration in federations with reliable participants. Marfoq et al. et al. [48] examine the effect of topology on the duration of communication rounds in cross-silo settings, and propose algorithms for measuring network characteristics to construct a high-throughput network topology. Guo et al. [22] use a hybrid device-to-device and device-to-server framework to improve communication in heterogeneous FL settings.

3 Methodology

We begin by reviewing the standard federated averaging objective, followed by describing the unique approach of Iterative Parameter Alignment (IPA).

Background In standard FL there are N clients in a federation, where each client i has a local dataset $\mathcal{D}_i$. The goal is to solve a common objective over universal dataset $\mathcal{D} = \cup_{i \in [N]}$ by aggregating each local model into a global model. The system iterates between local training on each client and global aggregation at the server. `FedAvg`, the original FL algorithm [50], involves a weighted averaging of client parameters at the server:

$$\textbf{Local :} \quad \theta_i = \arg\min_{\theta \in \mathbb{R}} \mathcal{L}_i(\mathcal{D}_i; \theta), \text{ initialized with } \theta; \quad \textbf{Global:} \quad \theta = \sum_{i=1}^{N} \frac{|\mathcal{D}_i|}{|\mathcal{D}|} \theta_i \tag{1}$$

where θ_i is the local model parameters, θ is the global models parameters, $\mathcal{L}_i(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}_i}\left[\ell_i(f(x), y; \theta)\right]$ is the local empirical loss of model i on dataset $\mathcal{D}_i$, and x and y are the samples and labels in $\mathcal{D}_i$. Next we describe the

Iterative Parameter Alignment To begin, we consider a set of N peers (rather than clients) where peer i has access to local dataset $\mathcal{D}_i$. Our goal is to solve an objective over universal dataset $\mathcal{D} = \cup_{i \in [N]}$ for each peer model $f(\theta_i)$. To do this, each client solves both an empirical learning objective, denoted $\mathcal{L}_i$, as well as an alignment objective $\mathcal{A}_i$, which together minimize the set of peer parameters for each client:

$$\theta^* = \arg\min_{\theta^* \in \mathbb{R}} \big[\mathcal{L}_i(\mathcal{D}_i; \theta_i) + \mathcal{A}_i(\theta^*)\big] \tag{2}$$

where $\mathcal{L}_i(\theta_i) = \mathbb{E}_{(x,y)\sim\mathcal{D}_i}\big[\ell_i(f(x), y; \theta_i)\big]$. For experiments in this work we set ℓ to be cross entropy loss for image classification problems. Importantly, $\mathcal{D}_i$ is only seen by peer model $f(\theta_i)$ for which the empirical loss is calculated. Moreover, peers are not able to share data with each other, only model parameters. This is similar to parameter sharing among the client and server in standard FL [50].

Key to global convergence of a peer model is the alignment of parameters during training. Specifically, model i holds parameters θ^* locally, and during each minibatch updates θ^* by minimizing the distance between θ_i and each θ_n:

$$\mathcal{R}_i(\theta^*) = \sum_{n=1}^{N} ||\theta_i - \theta_n||_p, \text{ where } i \neq n \tag{3}$$

where p is the L_1 or L_2 distance. Generalizing parameter alignment across all weights and biases of each layer $1, l, ...L$ of a neural network we achieve our alignment objective for model i:

$$\mathcal{A}_i(\theta^*) = \lambda \sum_{l=1}^{L} \mathcal{R}_i(\theta^*) \tag{4}$$

where λ is a global scale factor on the weight alignment objective. We set λ to 1 in this work, although we experimented with different values. We note that a result of this objective function updates each θ^*, essentially updating other peer's parameters while simultaneously updating the local model in order to align θ_i with θ_n.

IPA leads to a minimization of the global loss in individual models who have never seen the global dataset. In other words, when solving for the alignment objective in Equation 3, we show that a peer model with access to the full parameter set θ^* iteratively converges to an objective solved over the global dataset $\mathcal{D}$: $\arg\min_{\theta_i} \mathcal{L}_i(\mathcal{D}_i; \theta_i) \rightarrow \arg\min_{\theta} \mathcal{L}(\mathcal{D}; \theta)$. Compared to standard FL, the IPA algorithm only updates parameters on peer devices in a decentralized and synchronous architecture. Further, the method relies on independent (i.e. never aggregated) peer models. In the next section we highlight the benefits of the approach in various settings.

Algorithm 1 Parameter Alignment, One Iteration

Input:
N peers
Peer$_i$ has: $\mathcal{D}_i$, f_i with weights θ_i
θ^* is all peer parameters.

Output:
Models $f_1(\theta_1), f_2(\theta_2), ...f_N(\theta_N)$

Initialize θ^*, send to `peer`$_1$
for each `peer`$_i \in N$ **do**
  for each batch $b \in \mathcal{D}_i$ **do**
    $\mathcal{L}_i = \ell(f_i(b; \theta_i)) +$ **ParamAlign**(f_i, θ^*)
    $\theta_i \leftarrow \theta_i - \nabla\mathcal{L}$
  `Transfer` θ^* `to peer`$_{i+1}$

ParamAlign(f_i, θ^*):
  $\mathcal{R}_i \leftarrow 0$
  for each layer $\in f_i$ **do**
    for each $\theta_j \in \theta^*, j \neq i$ **do**
      $\mathcal{R}_i \leftarrow \mathcal{R}_i + |\theta_i - \theta_j|_p$
  return $\mathcal{R}_i$

4 Experiments

We begin by evaluating Iterative Parameter Alignment against existing methods in federated learning, including experiments merging peer models trained on isolated classes. Next, we show how our method naturally produces fair models (at epoch t), converging thereafter to globally optimized solutions.

Next, we quantify the difference between peer models, showing that each peer produces a distinct model in both parameter space and during inference.

4.1 Comparison to Existing Approaches

Our first empirical study compares the convergence rate of Iterative Parameter Alignment against existing FL algorithms. McMahan et al. [50] noted the slow convergence of their initial algorithm, `FedAvg`, when clients had heterogeneous data partitions. Since the initial research, much effort has been put into improving this convergence rate,which is measured by the *number of communication rounds* between the clients and the server until the global model converges to some target accuracy on the test set. We test our approach in a similar fashion.

Experimental Setup We construct our FL experiments from a set of realistic scenarios of up to twenty peers with homogeneous and heterogeneous data partitions. In heterogeneous settings our label ratios follow the Dirichlet distribution with $\alpha = 0.3$ and $\alpha = 0.6$, similar to previous works. Lower α indicates a higher data heterogeneity.

We compare Iterative Parameter Alignment to the standard FL algorithm `FedAvg` [50] as well as state-of-the-art approaches `FedProx` [40], `Scaffold` [33], `FedDyn` [2], and `FedDC` [19]. The original hyperparameters are used for each algorithm. We compare each algorithm using the MNIST, FashionMNIST, CIFAR-10, and CIFAR-100 datasets. We use the same architecture as previous works for the MNIST and FashionMNIST datasets; for the CIFAR-10 and CIFAR-100 datasets we use a larger CNN model which includes four convolutional layers followed by three linear layers. We consider one round of communication as each client training the model and sending it back to the server for aggregation (100% client participation). For IPA, one round of communication round is equal to every peer training their model in a ring topology. We consider other topologies in the next sections, finding similar convergence patterns. For IPA, we report the number of communication rounds it takes for the first peer to reach a target accuracy.

Unique to Iterative Parameter Alignment, we report the convergence rates of peer models with *different initializations*, i.e. each peer model is generated from a different random seed. In the original FL work, the authors highlighted the success of naive parameter averaging when models contained the same initial weights. However, averaging did not perform as well when the models were initialized differently. This phenomena was also reported in model merging literature [49], where the authors required models trained from the same initial weights. Research has suggested permutation invariance of neural networks as a driving force for this observation, i.e. a neural network has many variants which differ only in the ordering of its parameters [70].

Results Table 1 highlights the results of IPA against five state-of-the-art methods. Unsurprisingly, under IID settings IPA converges quickly towards the target accuracy on all four datasets. While the algorithm only feeds dataset $\mathcal{D}_i$ to $f(\theta_i)$, it otherwise has $20 \times \bar{\theta}$ parameters, optimizing $19 \times \bar{\theta}$ of the parameters using alignment and the final θ_i using alignment plus empirical loss. As a result, the balanced, overparameterized networks converge quickly despite only having access to a fraction of the training samples.

Under increasingly heterogeneous settings (from top to bottom) we observe a longer convergence rate for IPA compared to other algorithms. IPA remains competitive for MNIST and FashionMNIST, however has a slightly longer convergence rate for CIFAR-10 at Dirichlet($a = 0.3$) as well as CIFAR-100. We argue that convergence rate is less of a concern in cross-silo settings since large companies likely have adequate compute. Further, data heterogeneity creates additional questions in collaborative learning which we discuss in Section **??**.

A key parameter in the convergence of IPA is the p distance used for alignment. When $p = 1$ and we have an absolute error distance, heterogeneous and segregated label models converge faster. However, $p = 2$ (squared error distance) works well for homogeneous and balanced data partitioning. We examine this, as well as the effects of different initializations in the Appendix.

4.2 Peer Class Disjointedness

Unique to this work we experiment merging peer models who have completely isolated classes. For example, Peer_1 may only have images of dogs while Peer_2 only has images of cats. Such scenarios are important in the real-world such as those involving GDPR where an entire demographic segment is isolated, or cross-industry learning where the domains of individual peers are disjoint.

Class segregation also highlights the distinction between IPA and personalized FL. Personalized FL creates models for each client unique to their data distribution. In the example above, personalized

Dataset	Target Acc. (%)	FedAvg	FedProx	Scaffold	FedDyn	FedDC	IPA
IID, 20 Peers, $p = 2$							
MNIST	98	49	46	50	20	33	3
FashionMNIST	89	148	151	165	35	100	14
CIFAR-10	85	42	46	31	20	20	15
CIFAR-100	50	82	84	43	60	43	30
Dirichlet ($\alpha = 0.6$), 20 Peers, $p = 1$							
MNIST	98	147	140	52	20	35	28
FashionMNIST	87	60	67	62	15	40	60
CIFAR-10	85	64	65	44	22	24	45
CIFAR-100	50	105	105	56	61	55	97
Dirichlet ($\alpha = 0.3$), 20 Peers, $p = 1$							
MNIST	98	139	199	57	45	39	70
FashionMNIST	87	98	93	92	25	50	93
CIFAR-10	85	133	144	58	28	29	95
CIFAR-100	50	111	110	64	74	55	146

Table 1: **Communication rounds required to achieve target accuracy:** We compare the number of communication rounds required for IPA and other state-of-the-art FL algorithms to reach a target accuracy. IPA converges quickly on IID data partitions, with competitive results on heterogeneous splits.

FL would aid Peer_1 to better generalize to its own data (dogs) by utilizing Peer_2 data. However, Peer_1 may not gain much value from Peer_2's information about cats. Some methods create high performing personalized *and* global models: `FedRoD` [7] utilizes an additional local layer on a global model to create a high performing personalized model, while `FedHKD` [8] uses local "hyper knowledge" to aggregate the global model. However, these approaches create identical global models across clients. Further, the methods centrally aggregate the global model. IPA instead creates an *individualized* model for each peer which can perform well on a global task, while merging peer knowledge from completely independent domains.

We compose our experiments with simple class splits, such as a two peer class split where one peer has all training data labeled 0 to 4 and the second peer has training data with labeled 5 to 9 (in a dataset with 10 classes). We also consider imbalanced splits such as peers with an unequal number of classes.

Results Figure **??** highlights the convergence of peer models trained using the IPA algorithm on disjoint classes. We find that, compared to `FedAvg` and other existing approaches, parameter alignment has more stable training.

We note that for `FedDyn` and `FedDC` we apply smoothing to the test accuracy as a result of the instability of the global model.

We hypothesize that existing federated learning algorithms are unstable in the segregated class scenario because the gradient updates of local models move in disparate directions as a result of the extreme domain discrepancy. Existing work has shown that clients with heterogeneous data partitions have inconsistent optimization directions [35, 33], which cause drifts in the local models away from a global solution.

4.3 Peer Model Comparison

In this section we look at the quantitative differences between peer models across a variety of metrics to assess whether we create *sufficiently unique* models. Existing literature has found that neural networks are known to be sensitive to small changes in their parameters [73], causing drastic changes in model inference and generalization. There is a rich area of a research examining this phenomena for injecting adversarial attacks [90, 44, 73], evaluating the generalization gap of model minima [53, 34], and assessing the effects of model quantization [26], among other things. As a result, even the smallest differences in the weights of peer models can create vastly unique results.

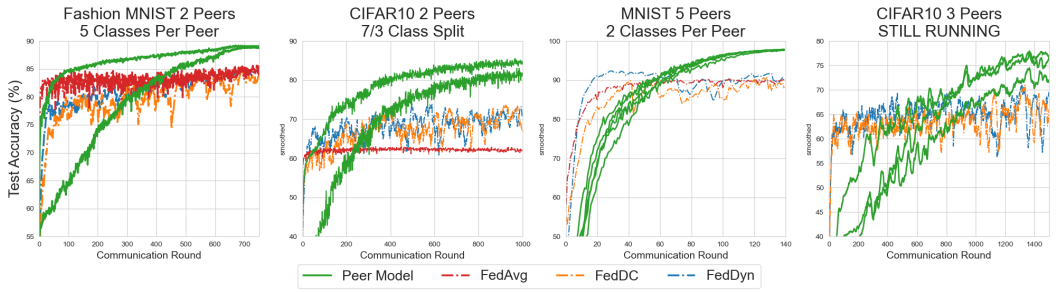


Figure 1: **Aligning Peer Models Trained on Disjoint Classes:** We find that existing federated learning approaches such as `FedAvg` struggle when trying to merge divergent (rather than heterogeneous) data partitions. Peer with disjoint classes serve as a For `FedDyn` and `FedDC` we apply smoothing to the test accuracy as a result of the instability of the global model.

Experiments. To quantify the difference between two neural networks, we compare both the networks parameters as well as their predictions. We measure the distance between two models' parameters as $||\theta_i - \theta_j||_p$, where $p = \{1, 2\}$. To measure the difference between model predictions, we compute the Hamming distance between two models' predictions on the test set, which we denote $\mathcal{H}(f_i, f_j$. We also present a count of when both models predictions are correct (denoted $f_1 \wedge f_2$), as well as both incorrect ($\overline{f_i} \wedge \overline{f_j}$).

We test heterogeneous (Dirichlet with $\alpha = 0.6$) and homogeneous scenarios with both the FashionMNIST and CIFAR10 datasets. All experiments use ten peers. We choose a lower number of peers compared to previous experiments in order to magnify potential similarities between models. Heterogeneous experiments are trained for 200 epochs and homogeneous experiments are trained for 50 epochs. The FashionMNIST experiment on homogeneous (IID) data had a test accuracy of $88.9\%\pm0.24$, while the heterogeneous scenario had a test accuracy of $82.5\%\pm3.42$. The CIFAR10 experiment on homogeneous (IID) data had a mean test accuracy of $86.4\%\pm0.44$, while the heterogeneous scenario had a mean test accuracy of $79.5\%\pm4.12$. We run each experiment three times and present the average of the model differences.

Results. Figure 2 highlights the differences between peer models across four experiments. The first two rows indicate a dissimilarity between peer model parameters across L_1 distance, with a smaller discrepancy when measured with L_2 distance. We hypothesized that IID data experiments would have closer parameters, however, the heterogeneous experiments yielded smaller values. We attribute this to training heterogeneous data for 200 epochs compared to just 50 epochs for IID data.

The bottom three rows measure the difference in test inference between peer models, with both datasets having a test set size of 10k. The smallest Hamming distance was between IID models, with 650 and 1,043 respectively. We argue that these values indicate a significant difference from each other since IID models achieve 88.9% and 86.4% respectively. Finally, we note that the standard error was negligible across all experiments.

4.4 Fairness with Early Stopping

In cross-silo settings organizations may be competing against each other, hence the contribution of participants becomes a critical measure. Designing proper incentive mechanisms and rewards for participation can encourage peers to join a federation. Previous work has proposed fairness schemes such as those described in Section 2; many of these methods produce different versions of the global model, i.e. models for each client whose performance is commensurate to its data contribution.

IPA takes a different approach: in Section 4 and Figure **??** we show that peer models converge to a global solution if given enough training time. However, the figures also highlight the variable convergence rates of peer models with heterogeneous data partitions. In other words, we find that the convergence of a peer model trained with IPA is a function of the peers standalone model performance.

We induce fairness in the IPA algorithm by implementing *early stopping* of the system at some iteration $t < T$, where T is the number of iterations it takes for *all* peer models to converge to some target accuracy. Typically, early stopping is induced in neural network training to avoid overfitting.

	FashionMNIST		CIFAR-10	
Distance	Dir(0.3)	IID	Dir(0.3)	IID
$\|\|\theta_i - \theta_j\|\|_1$	196.5	1352.6	1.8×10^3	$3.3{\times}10^4$
$\|\|\theta_i - \theta_j\|\|_2$	0.7	4.9	2.0	35.9
$\mathcal{H}(f_i, f_j)$	1,990	650	2,504	1,043
$f_i \wedge f_j$	7,358	8,603	6,871	8,214
$\overline{f_i} \wedge \overline{f_j}$	947	837	1,094	947

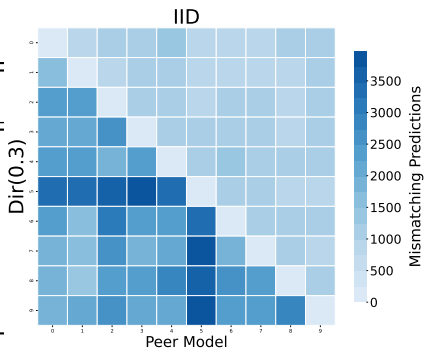
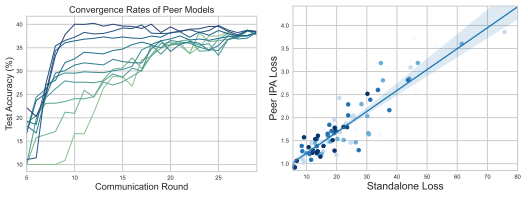


Figure 2: **Comparing Peer Models:** We measure the distance between peer models across a variety of metrics. Each experiment contains ten peers and is aggregated across three runs, with the mean and standard error presented for each. **Left:** Measuring the distance between models across parameters (first two rows) and model predictions (the last three rows). 1 The last three rows denote the Hamming distance between predictions, mutual correct predictions, and mutual incorrect predictions on the test set. Test set size for both datasets is size 10k. **Right:** A similarity matrix of Hamming distances between peer model predictions for: 1) heterogeneous data partition (bottom triangle) and 2) homogeneous (IID) data partition (top triangle). The distances represent the number of mismatching predictions in the test set for each model. For reference, the lowest (averaged) Hamming distance between models in the IID setting is 880, with a test set size of 10k.

	MNIST		CIFAR-10	
Distance	Dir(0.3)	IID	Dir(0.3)	IID
$\|\|\theta_i - \theta_j\|\|_1$	196.5	1352.6	1.8×10^3	$3.3{\times}10^4$
$\|\|\theta_i - \theta_j\|\|_1$	196.5	1352.6	1.8×10^3	$3.3{\times}10^4$
$\|\|\theta_i - \theta_j\|\|_1$	196.5	1352.6	1.8×10^3	$3.3{\times}10^4$
$\|\|\theta_i - \theta_j\|\|_1$	196.5	1352.6	1.8×10^3	$3.3{\times}10^4$



We instead use it to provide fairness to the federation so that each member receives a model similar whose performance is similar to their contribution.

To test our approach, we conduct several experiments designed from benchmarks in previous works. First, we measure fairness using Pearson's coefficient $\rho(\varphi, \xi) \in [-1, 1]$ [47, 78, 46]. Specifically, we measure the correlation between the test set accuracy on the standalone model compared to the test set accuracy of the IPA model. We compare our method with the benchmarks of Xu et al. [78] since their approach provides theoretically guaranteed fairness metrics.

Results.

In Section 4 we found that more heterogeneous data partitions take longer to converge compared to uniform data splits. As a result, we find that heterogeneous partitions *stay within the fair regime* for a longer number of iterations. This is a potentially desirable property to enable fairness among peers, and provides flexibility in choosing the early-stopping hyperparameter t.

5 Discussion

Additional Considerations for Decentralization Key to our approach is sharing model parameters across peers during the IPA training process. While we claim enhanced security as a result of independent global models, the reality is that each peer has access to each others models during training. To counter this security flaw, techniques such as homomorphic encryption [56] and garbled ciruits [39] enable peers to encrypt their models for enhanced protection; such techniques have been applied to FL systems [85, 28, 77]. Differential privacy is an additional protocol which provides formalized privacy guarantees [14]. It is commonly applied to training data, however, it can also be applied to model training [27]. It has been applied to the FL pipeline [51, 20, 29, 3] including in the cross-silo setting [23] where additional considerations need to be made such as securing the privacy of sample-level (rather than client level) data [43].

Applications Beyond Federated Learning Iterative Weight Alignment may be of interest to other fields of machine learning such as ensembling [75, 87, 55], domain adaptation and transfer learning [57, 81, 76, 12, 60, 59], model merging and patching [49, 65, 4], and other contexts with variable data distributions. For example, transfer learning enables fine-tuning a pretrained model to enhance performance on some target domain, however, fine-tuning was found to cause reduced robustness on source domain distribution shift benchmarks [61, 58]. Wortsmann et al. [76] proposed ensembling the pre-trained and fine-tuned models for increased performance on source domain robustness. Insights such as this could potentially be gleaned from IPA, where iteratively merging the parameters of segregated domains provides enhanced performance. Domain divergence is also an active area of research in negative transfer learning [71, 88]. We leave this analysis to future work.

ADD: A FEW LINES ABOUT MODEL MERGING: ONLY TAKES A FEW ITERATIONS TO ALIGN MODELS, EVEN MODELS WITH DIFFERENT INITIALIZATIONS. MODEL FUSION:

Limitations The IPA algorithm is feasible in cross-silo settings with a limited number of clients or smaller models. It does not scale well to many peers as a result of requiring $N \times \bar{\theta}$ parameters during training. However, we note that advances in network sparsity may enable the method to scale in the future [18]. Additionally, we note that our method converges slower under more heterogeneous environments as depicted in Table 1. While this may be undesirable in many scenarios, we argue that this is less of a constraint in cross-silo settings. Other configurations are examined in the Appendix. Finally, there are additional settings we have not considered in this paper such as tasks other than image classification and vertically aligned FL [30].

Conclusion We propose a new method of iteratively aligning parameters across peer machines. IPA is favorable in highly segregated class settings in image classification tasks, and achieves competitive convergence under heterogeneity. We assess our approach across novel and existing benchmarks, and further show that the method generates unique peer models that converge as a function of their dataset uniformity.

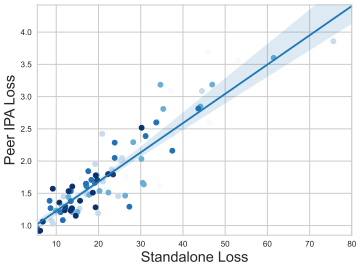


Figure 3: Correlation between the test loss of models trained on standalone data (x-axis) and collaboratively using IPA (y-axis). We run five experiments using 20 peers with a Dirichlet data split with $a = 0.25$, averaging test losses across 100 iterations.

References

[1] FedSyn: Federated learning meets Blockchain. URL https://www.jpmorgan.com/technology/federated-learning-meets-blockchain/.

[2] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. Federated learning based on dynamic regularization. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=B7v4QMR6Z9w.

[3] Naman Agarwal, Peter Kairouz, and Ziyu Liu. The skellam mechanism for differentially private federated learning. *Advances in Neural Information Processing Systems*, 34:5052–5064, 2021.

[4] Samuel K Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. *arXiv preprint arXiv:2209.04836*, 2022.

[5] Aurélien Bellet, Rachid Guerraoui, Mahsa Taziki, and Marc Tommasi. Personalized and private peer-to-peer

machine learning. In Amos Storkey and Fernando Perez-Cruz, editors, *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pages 473–481. PMLR, 09–11 Apr 2018. URL https://proceedings.mlr.press/v84/bellet18a.html.

[6] Avrim Blum, Nika Haghtalab, Richard Lanas Phillips, and Han Shao. One for one, or all for all: Equilibria and optimality of collaboration in federated learning. In *International Conference on Machine Learning*, pages 1005–1014. PMLR, 2021.

[7] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. *arXiv preprint arXiv:2107.00778*, 2021.

[8] Huancheng Chen, Chaining Wang, and Haris Vikalo. The best of both worlds: Accurate global and personalized models through federated learning with data-free hyper-knowledge distillation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=29V3AWjVAFi.

[9] Igor Colin, Aurelien Bellet, Joseph Salmon, and Stéphan Clémençon. Gossip dual averaging for decentralized optimization of pairwise functions. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1388–1396, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/colin16.html.

[10] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. In *International Conference on Machine Learning*, pages 2089–2099. PMLR, 2021.

[11] Ittai Dayan, Holger R Roth, Aoxiao Zhong, Ahmed Harouni, Amilcare Gentili, Anas Z Abidin, Andrew Liu, Anthony Beardsworth Costa, Bradford J Wood, Chien-Sung Tsai, et al. Federated learning for predicting clinical outcomes in patients with covid-19. *Nature medicine*, 27 (10):1735–1743, 2021.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[13] Kate Donahue and Jon Kleinberg. Optimality and stability in federated learning: A game-theoretic approach. *Advances in Neural Information Processing Systems*, 34: 1287–1298, 2021.

[14] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.

[15] editor2fedai. WeBank and Swiss Re signed Cooperation MoU. URL https://www.fedai.org/news/webank-and-swiss-re-signed-cooperation-mou/.

[16] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020.

[17] Mona Flores. NVIDIA Blogs: NVIDIA Blogs: AI models for Mammogram Assessment, April 2020. URL https://blogs.nvidia.com/blog/2020/04/15/federated-learning-mammogram-assessment/.

[18] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. *arXiv preprint arXiv:1803.03635*, 2018.

[19] Liang Gao, Huazhu Fu, Li Li, Yingwen Chen, Ming Xu, and Cheng-Zhong Xu. Feddc: Federated learning with non-iid data via local drift decoupling and correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10112–10121, 2022.

[20] Robin C Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

[21] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[22] Yuanxiong Guo, Ying Sun, Rui Hu, and Yanmin Gong. Hybrid local sgd for federated learning with heterogeneous communications. In *International Conference on Learning Representations*, 2021.

[23] Mikko A Heikkilä, Antti Koskela, Kana Shimizu, Samuel Kaski, and Antti Honkela. Differentially private cross-silo federated learning. *arXiv preprint arXiv:2007.05553*, 2020.

[24] Chao Huang, Jianwei Huang, and Xin Liu. Cross-silo federated learning: Challenges and opportunities. *arXiv preprint arXiv:2206.12949*, 2022.

[25] Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 7865–7873, 2021.

[26] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Quantized neural networks: Training neural networks with low precision weights and activations. *The Journal of Machine Learning Research*, 18(1):6869–6898, 2017.

[27] Matthew Jagielski, Jonathan Ullman, and Alina Oprea. Auditing differentially private machine learning: How private is private sgd? *Advances in Neural Information Processing Systems*, 33:22205–22216, 2020.

[28] Zhifeng Jiang, Wei Wang, and Yang Liu. Flashe: Additively symmetric homomorphic encryption for cross-silo federated learning. *arXiv preprint arXiv:2109.00675*, 2021.

[29] Peter Kairouz, Ziyu Liu, and Thomas Steinke. The distributed discrete gaussian mechanism for federated learning with secure aggregation. In *International Conference on Machine Learning*, pages 5201–5212. PMLR, 2021.

[30] Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021.

[31] Jiawen Kang, Zehui Xiong, Dusit Niyato, Shengli Xie, and Junshan Zhang. Incentive mechanism for reliable federated learning: A joint optimization approach to combining reputation and contract theory. *IEEE Internet of Things Journal*, 6(6):10700–10714, 2019. doi: 10.1109/JIOT.2019.2940820.

[32] Jiawen Kang, Zehui Xiong, Dusit Niyato, Han Yu, Ying-Chang Liang, and Dong In Kim. Incentive design for efficient federated learning in mobile networks: A contract theory approach. In *2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)*, pages 1–5. IEEE, 2019.

[33] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR, 2020.

[34] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[35] Ahmed Khaled, Konstantin Mishchenko, and Peter Richtárik. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pages 4519–4529. PMLR, 2020.

[36] Anastasia Koloskova, Tao Lin, Sebastian U Stich, and Martin Jaggi. Decentralized deep learning with arbitrary communication compression. *arXiv preprint arXiv:1907.09356*, 2019.

[37] Jakub Konečnỳ, H Brendan McMahan, Felix X Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

[38] Anusha Lalitha, Osman Cihan Kilinc, Tara Javidi, and Farinaz Koushanfar. Peer-to-peer federated learning on graphs. *arXiv preprint arXiv:1901.11173*, 2019.

[39] Ibrahim Lazrig, Toan C. Ong, Indrajit Ray, Indrakshi Ray, Xiaoqian Jiang, and Jaideep Vaidya. Privacy preserving probabilistic record linkage without trusted third party.

In Robert H. Deng, Stephen Marsh, Jason Nurse, Rongxing Lu, Sakir Sezer, Paul Miller, Liqun Chen, Kieran McLaughlin, and Ali Ghorbani, editors, *2018 16th Annual Conference on Privacy, Security and Trust, PST 2018*, 2018 16th Annual Conference on Privacy, Security and Trust, PST 2018, United States, October 2018. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/PST.2018.8514192. Funding Information: This work was supported by grants from UC Anschutz Medical Center, NSF under award no. CNS 1650573, AFRL, CableLabs, Furuno Electric Company, and SecureNok. Publisher Copyright: © 2018 IEEE.; 16th Annual Conference on Privacy, Security and Trust, PST 2018 ; Conference date: 28-08-2018 Through 30-08-2018.

[40] 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.

[41] Tian Li, Shengyuan Hu, Ahmad Beirami, and Virginia Smith. Ditto: Fair and robust federated learning through personalization. In *International Conference on Machine Learning*, pages 6357–6368. PMLR, 2021.

[42] Paul Pu Liang, Terrance Liu, Liu Ziyin, Nicholas B Allen, Randy P Auerbach, David Brent, Ruslan Salakhutdinov, and Louis-Philippe Morency. Think locally, act globally: Federated learning with local and global representations. *arXiv preprint arXiv:2001.01523*, 2020.

[43] Ken Liu, Shengyuan Hu, Steven Wu, and Virginia Smith. On privacy and personalization in cross-silo federated learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=Oq2bdIQQOIZ.

[44] Yannan Liu, Lingxiao Wei, Bo Luo, and Qiang Xu. Fault injection attack on deep neural network. In *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 131–138, 2017. doi: 10.1109/ICCAD.2017.8203770.

[45] Zelei Liu, Yuanyuan Chen, Han Yu, Yang Liu, and Lizhen Cui. Gtg-shapley: Efficient and accurate participant contribution evaluation in federated learning. *ACM Trans. Intell. Syst. Technol.*, 13(4), may 2022. ISSN 2157-6904. doi: 10.1145/3501811. URL https://doi.org/10.1145/3501811.

[46] Lingjuan Lyu, Xinyi Xu, Qian Wang, and Han Yu. Collaborative fairness in federated learning. *Federated Learning: Privacy and Incentive*, pages 189–204, 2020.

[47] Lingjuan Lyu, Jiangshan Yu, Karthik Nandakumar, Yitong Li, Xingjun Ma, Jiong Jin, Han Yu, and Kee Siong Ng. Towards fair and privacy-preserving federated deep models. *IEEE Transactions on Parallel and Distributed Systems*, 31(11):2524–2541, 2020.

[48] Othmane Marfoq, Chuan Xu, Giovanni Neglia, and Richard Vidal. Throughput-optimal topology design for cross-silo federated learning. *Advances in Neural Information Processing Systems*, 33:19478–19487, 2020.

[49] Michael S Matena and Colin A Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022.

[50] 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.

[51] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963*, 2017.

[52] Mehryar Mohri, Gary Sivek, and Ananda Theertha Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR, 2019.

[53] Behnam Neyshabur, Srinadh Bhojanapalli, David McAllester, and Nati Srebro. Exploring generalization in deep learning. *Advances in neural information processing systems*, 30, 2017.

[54] Jean Ogier du Terrail, Armand Leopold, Clément Joly, Constance Béguier, Mathieu Andreux, Charles Maussion, Benoît Schmauch, Eric W. Tramel, Etienne Bendjebbar, Mikhail Zaslavskiy, Gilles Wainrib, Maud Milder, Julie Gervasoni, Julien Guerin, Thierry Durand, Alain Livartowski, Kelvin Moutet, Clément Gautier, Inal Djafar, Anne-Laure Moisson, Camille Marini, Mathieu Galtier, Félix Balazard, Rémy Dubois, Jeverson Moreira, Antoine Simon, Damien Drubay, Magali Lacroix-Triki, Camille Franchet, Guillaume Bataillon, and Pierre-Etienne Heudel. Federated learning for predicting histological response to neoadjuvant chemotherapy in triple-negative breast cancer. *Nature Medicine*, 29(1):135–146, January 2023. ISSN 1546-170X. doi: 10.1038/s41591-022-02155-w. URL https://www.nature.com/articles/s41591-022-02155-w. Number: 1 Publisher: Nature Publishing Group.

[55] Matthew Olson, Abraham Wyner, and Richard Berk. Modern neural networks generalize on small data sets. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/fface8385abbf94b4593a0ed53a0c70f-Paper.pdf.

[56] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *Advances in Cryptology—EUROCRYPT'99: International Conference on the Theory and Application of Cryptographic Techniques Prague, Czech Republic, May 2–6, 1999 Proceedings 18*, pages 223–238. Springer, 1999.

[57] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[58] Hieu Pham, Zihang Dai, Golnaz Ghiasi, Hanxiao Liu, Adams Wei Yu, Minh-Thang Luong, Mingxing Tan, and Quoc V Le. Combined scaling for zero-shot transfer learning. *arXiv preprint arXiv:2111.10050*, 2021.

[59] Jason Phang, Thibault Févry, and Samuel R Bowman. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088*, 2018.

[60] Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. Intermediate-task transfer learning with pretrained models for natural language understanding: When and why does it work?, 2020.

[61] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[62] Aviv Shamsian, Aviv Navon, Ethan Fetaya, and Gal Chechik. Personalized federated learning using hypernetworks. In *International Conference on Machine Learning*, pages 9489–9502. PMLR, 2021.

[63] Santiago Silva, Boris A. Gutman, Eduardo Romero, Paul M. Thompson, Andre Altmann, and Marco Lorenzi. Federated Learning in Distributed Medical Databases: Meta-Analysis of Large-Scale Subcortical Brain Data. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, pages 270–274, April 2019. doi: 10.1109/ISBI.2019.8759317. ISSN: 1945-8452.

[64] Navjot Singh, Deepesh Data, Jemin George, and Suhas Diggavi. Squarm-sgd: Communication-efficient momentum sgd for decentralized optimization. In *2021 IEEE International Symposium on Information Theory (ISIT)*, pages 1212–1217, 2021. doi: 10.1109/ISIT45174.2021.9517986.

[65] Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. *Advances in Neural Information Processing Systems*, 33:22045–22055, 2020.

[66] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.

[67] Hanlin Tang, Xiangru Lian, Ming Yan, Ce Zhang, and Ji Liu. D^2: Decentralized training over decentralized data. In *International Conference on Machine Learning*, pages 4848–4856. PMLR, 2018.

[68] Saeed Vahidian, Mahdi Morafah, and Bill Lin. Personalized federated learning by structured and unstructured pruning under data heterogeneity. In *2021 IEEE 41st International Conference on Distributed Computing Systems Workshops (ICDCSW)*, pages 27–34, 2021. doi: 10.1109/ICDCSW53096.2021.00012.

[69] Paul Vanhaesebrouck, Aurélien Bellet, and Marc Tommasi. Decentralized Collaborative Learning of Personalized Models over Networks. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 509–517. PMLR, 20–22 Apr 2017. URL https://proceedings.mlr.press/v54/vanhaesebrouck17a.html.

[70] Jianyu Wang, Anit Kumar Sahu, Zhouyi Yang, Gauri Joshi, and Soummya Kar. Matcha: Speeding up decentralized sgd via matching decomposition sampling. In *2019 Sixth Indian Control Conference (ICC)*, pages 299–300. IEEE, 2019.

[71] Zirui Wang, Zihang Dai, Barnabás Póczos, and Jaime Carbonell. Characterizing and avoiding negative transfer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11293–11302, 2019.

[72] Yuxin Wen, Jonas A. Geiping, Liam Fowl, Micah Goldblum, and Tom Goldstein. Fishing for user data in large-batch federated learning via gradient magnification. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 23668–23684. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/wen22a.html.

[73] Tsui-Wei Weng, Pu Zhao, Sijia Liu, Pin-Yu Chen, Xue Lin, and Luca Daniel. Towards certificated model robustness against weight perturbations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6356–6363, 2020.

[74] Blake E Woodworth, Jialei Wang, Adam Smith, Brendan McMahan, and Nati Srebro. Graph oracle models, lower bounds, and gaps for parallel stochastic optimization. *Advances in neural information processing systems*, 31, 2018.

[75] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning*, pages 23965–23998. PMLR, 2022.

[76] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi,

Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7959–7971, 2022.

[77] Guowen Xu, Hongwei Li, Yun Zhang, Shengmin Xu, Jianting Ning, and Robert H. Deng. Privacy-preserving federated deep learning with irregular users. *IEEE Transactions on Dependable and Secure Computing*, 19(2): 1364–1381, 2022. doi: 10.1109/TDSC.2020.3005909.

[78] Xinyi Xu, Lingjuan Lyu, Xingjun Ma, Chenglin Miao, Chuan Sheng Foo, and Bryan Kian Hsiang Low. Gradient driven rewards to guarantee fairness in collaborative machine learning. *Advances in Neural Information Processing Systems*, 34:16104–16117, 2021.

[79] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–19, 2019.

[80] Hongxu Yin, Arun Mallya, Arash Vahdat, Jose M Alvarez, Jan Kautz, and Pavlo Molchanov. See through gradients: Image batch recovery via gradinversion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16337–16346, 2021.

[81] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in neural information processing systems*, 27, 2014.

[82] Fuxun Yu, Weishan Zhang, Zhuwei Qin, Zirui Xu, Di Wang, Chenchen Liu, Zhi Tian, and Xiang Chen. Fed2: Feature-aligned federated learning. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery Data Mining*, KDD '21, page 2066–2074, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383325. doi: 10.1145/3447548.3467309. URL https://doi.org/10.1145/3447548.3467309.

[83] Han Yu, Zelei Liu, Yang Liu, Tianjian Chen, Mingshu Cong, Xi Weng, Dusit Niyato, and Qiang Yang. A sustainable incentive scheme for federated learning. *IEEE Intelligent Systems*, 35(4):58–69, 2020.

[84] Valentina Zantedeschi, Aurélien Bellet, and Marc Tommasi. Fully decentralized joint learning of personalized models and collaboration graphs. In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 864–874. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/zantedeschi20a.html.

[85] Chengliang Zhang, Suyi Li, Junzhe Xia, Wei Wang, Feng Yan, and Yang Liu. Batchcrypt: Efficient homomorphic encryption for cross-silo federated learning. In *Proceedings of the 2020 USENIX Annual Technical Conference (USENIX ATC 2020)*, 2020.

[86] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M Alvarez. Personalized federated learning with first order model optimization. *arXiv preprint arXiv:2012.08565*, 2020.

[87] Shaofeng Zhang, Meng Liu, and Junchi Yan. The diversified ensemble neural network. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 16001–16011. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/b86e8d03fe992d1b0e19656875ee557c-Paper.pdf.

[88] Wen Zhang, Lingfei Deng, Lei Zhang, and Dongrui Wu. A survey on negative transfer. *IEEE/CAA Journal of Automatica Sinica*, 2022.

[89] Haoyu Zhao, Boyue Li, Zhize Li, Peter Richtárik, and Yuejie Chi. BEER: Fast $o(1/t)$ rate for decentralized nonconvex optimization with communication compression. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=I47eFCKa1f3.

[90] Pu Zhao, Siyue Wang, Cheng Gongye, Yanzhi Wang, Yunsi Fei, and Xue Lin. Fault sneaking attack: A stealthy framework for misleading deep neural networks. In *Proceedings of the 56th Annual Design Automation Conference 2019*, DAC '19, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367257. doi: 10.1145/3316781.3317825. URL https://doi.org/10.1145/3316781.3317825.