FedLDA: Personalized Federated Learning Through Collaborative Linear Discriminant Analysis

Connor Mclaughlin
Northeastern University
mclaughlin.co@northeastern.edu

Lili Su
Northeastern University
l.su@northeastern.edu

Abstract

Data heterogeneity poses a significant challenge to federated learning. Observing the universality of neural networks in approximating the ground-truth, one emerging perspective is to train personalized models via learning a shared representation coupled with customized classifiers for each client. To the best of our knowledge, except for the concurrent work FedPAC [XTH23], individual classifiers in most existing works only utilize local datasets, which may result in poor generalization. In this work, we propose FedLDA which enables federation in training classifiers by performing collaborative Linear Discriminant Analysis (LDA) on top of the latent shared representation. Our algorithm design is motivated by the observation that upon network initialization the extracted features are highly Gaussian, and client LDA models may benefit from distributed estimation of the Gaussian parameters. To support the high-dimension, low-sample scenario often encountered in PFL, we utilize a momentum update of the Gaussian parameters and employ ℓ_1 regularization of local covariances. Our numerical results show that, surprisingly, in contrast to multiple state-of-the-art methods, our FedLDA is capable of maintaining the initial Gaussianity. More importantly, through empirical study, we demonstrate that our FedLDA method has improved generalization compared to state-of-the-art algorithms. Compared with FedPAC [XTH23] our method is communication-efficient and does not require the availability of a validation dataset.

1 Introduction

The empirical success of deep learning models is underpinned by the availability of large amounts of labeled data [KSH17]. Federated learning (FL) emerged as a privacy-preserving learning framework for training from decentralized datasets (clients) without collecting the raw data at a central location [MMR$^+$17, KMA$^+$21]. FedAvg [MMR$^+$17] is one of the most widely adopted federated learning algorithms under which a common model is trained but is used to serve possibly highly heterogeneous clients. Data heterogeneity poses a significant challenge to federated learning; highly skewed local data across clients easily leads to slow convergence and poor prediction prediction at individual clients [KMR20, LHY$^+$19, ZLL$^+$18].

Personalized federated learning (PFL) trains fully customized models for each of the participating clients [SCST17, TYCY22]. Neural networks are universal in approximating the ground-truth [MFSL19]. Observing this, one emerging perspective inspired by representation learning [BCV13] is to approach PFL as a problem of learning a shared representation coupled with a customized classifier for each client [LLZ$^+$20, CHMS21, OKY21, XTH23]. To the best of our knowledge, most existing methods restrict client collaboration for shared representation only, neglecting the potential of exploiting collaboration in training local classifiers. Consequently, the training of the local classifiers only utilizes the limited local datasets, which may result in poor generalization.

International Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS 2023 (FL@FM-NeurIPS'23). This workshop does not have official proceedings and this paper is non-archival.

One notable exception is the concurrent work [XTH23], in which collaboration in training local classifiers is done via a novel form of the soft clustering technique. Though elegant and insightful, this method requires each client to access the local classifiers of all other clients, which is vulnerable to privacy leakage. Moreover, it crucially relies on local validation data, which may be limiting in data-scarce real-world applications.

Contributions. We propose FedLDA which enables client federation in training personalized classifiers by performing collaborative Linear Discriminant Analysis (LDA) on top of the latent shared representation. Specifically, in addition to learning a set of shared neural network parameters, clients also collaboratively estimate the density of the extracted features. Based on initial observations that neural networks produce Gaussian representations at initialization, we model the density as a class-conditional Gaussian with common covariance matrix, enabling the use of LDA classifiers. Inspired by classical literature on sparse discriminant analysis, we mitigate local estimation noise through a momentum update reminiscent of shrinkage estimators, and a ℓ_1 regularization objective that encourages sparsity in the empirical covariance matrices.

In summary, the key contributions of our work include:

- Our numerical results show that, in contrast to multiple state-of-the-art methods, our FedLDA can maintain the level of Gaussianity encountered at initialization.
- Through rigourous empirical study, we demonstrate that our FedLDA method leads to faster convergence and improved generalization compared with commonly used and state of the art baselines. We consider both ResNet18 and WideResNet-16-2 neural network architectures, confirming the applicability of FedLDA to varying latent representation sizes.
- We show that our method outperforms the SOTA even in the challenging setting where the latent dimension exceeds the local data volume.

2 Related Work

Non-IID Federated Learning. The prototypical work in Federated Learning FedAvg [MMR+17] aims to learn a single global model that minimizes the average error across clients. However, this method suffers from slow convergence and even divergence when the client data heterogeneity is high [LSZ+20, KKM+20, OKY21]. Particularly, [LHY+19] showed that a decaying learning rate is required for convergence, resulting in slower training.

To facilitate convergence of FedAvg under non-IID data, several algorithmic solutions have been proposed including local model regularization [LSZ+20] and client variance reduction [KKM+20]. Other techniques such as loss balancing [HQB20, WXWZ21, CC21], knowledge distillation [LKSJ20, ZHZ21], and prototype learning [TLL+22] have also been successfully applied to non-IID FL. However, these methods implicitly call for well-controlled data heterogeneity [KMA+21].

Personalized Federated Learning. PFL has emerged as a field to handle data heterogeneity via exploiting the underlying connections with the local learning tasks. Popular techniques include meta-learning a shared global model which is amenable to client fine-tuning [JKRK19, FMO20]; multi-task learning with model similarity regularization [TDTN20, LHBS21]; cross-client model collaboration [ZSF+21, ZHW+22]; and decoupled representation and classifier learning [CHMS21, LLZ+20, OKY21, XTH23]. Our work shares the greatest similarity with the latter group.

FedRep [CHMS21] shares parameters of a neural network up to the classification layer across all clients, and achieves personalization through client-specific training of classifiers. In each local step, clients first learn their own optimal linear classifier with respect to the current global (fixed) feature extractor. Then, the shared feature extractor is locally tuned using the optimal client classifier. FedBABU [OKY21] adopts a similar setup, but fixes client classifiers on a common initialization until the base model has converged, in order to enforce a common criterion for representation learning. FedPAC [XTH23] also follows the format of FedRep, but with additional regularization to push client feature distributions towards global feature distributions. Additionally, FedPAC enables further client collaboration by learning a convex combination of personal classifiers. Our work incorporates aspects of each of these methods. We share a common training procedure to FedRep, but we restrict personalization to solely the bias term of the local classifier, resembling FedBABU.

Similar to FedPAC, our clients collaborate on classifier learning, however we do not require access to an additional validation set.

Linear Discriminant Analysis. Linear Discriminant Analysis (LDA) is a classical classification method which has had much popularity due to its simplicity and performance [Han06]. In this paper we view LDA as multivariate-Gaussian modelling, which considers samples $x \in \mathbb{R}^d$ follow a Gaussian distribution with a class-specific mean μ^c for $c \in \mathcal{C}$ and a common covariance matrix Σ for all classes. A well-adopted LDA classification rule can be derived by applying maximum-likelihood estimation, Fishers' linear discriminant [Fis36], as well as the optimal scoring problem [HTB94].

While the LDA classification rule is Bayes' optimal and has been shown to be more efficient than softmax regression [Efr75], it can not be applied in the settings where $n \ll d$, i.e., low data volume yet high dimensional feature, because that the corresponding covariance matrix Σ is singular. The problem of supporting LDA in high-dimensions is well-studied, with common solutions introducing sparsity constraints [CHWE11, SWDW11, CL11] or regularizing the estimate of Σ, e.g. through shrinkage methods [PVN82, Fri89]. These alterations of the LDA have been successful in high-dimensional settings, to the extent of exhibiting similar optimally as in the low-dimensionality setting under certain conditions [CHWE11].

Several works have explored the use of LDA in conjunction with a neural networks. Stuhlsatz [SLZ12] used Fishers' LDA criterion as a way of fine-tuning a pretrained stack of restricted Boltzman machines. Dorfer et al. [DKW15] proposed to train a neural network from end-to-end using the generalized eigenvalue formulation [GKC19] of Fishers' LDA. Multivariate-Gaussian LDA has been applied on top of trained networks for applications of lifelong-learning [HK20] and biomedical imaging [DIS20]. Pang et al. [PDZ18] utilized LDA with a fixed mean and covariance in order to train adversarially robust neural networks. Departing from these works, we use the multivariate-LDA to enable collaborative training personalized classifiers on top of a shared neural network base.

3 Problem Setup

Learning Goal. A parameter server and M clients (each indexed $i \in \{1, \cdots, M\} := [M]$) collaboratively train machine learning models without having the clients disclose their private data. Each client i has a local dataset $\mathcal{D}_i = \{(x_i^j, y_i^j)\}_{j=1}^{n_i}$, where n_i is the number of data points at client i, and $x_i^j \in \mathbb{R}^I$ and $y_i^j \in \{1, \cdots, C\} := \mathcal{C}$ are the covariate and label of the j-th local data tuple.

Our goal is to solve a classification task under this setting, using a neural network with parameters θ. In the personalized federated learning, we are interested in finding a unique set of parameters for each client $\{\theta_i\}_{i=1}^M$ that minimize the expected loss for each of the M local datasets:

$$\min_{\theta_1,\ldots,\theta_M \in \mathcal{Q}_M} \left(f(\theta_1, ..., \theta_M) := \frac{1}{M} \sum_{i=1}^{M} F_i(\theta_i) \right), \tag{1}$$

where $\mathcal{Q}_M$ is the space of feasible sets of M models, $F_i(\theta_i) = \mathbb{E}_{(x,y)\sim\mathcal{D}_i}[L(x, y; \theta_i)]$ is the empirical risk of dataset $\mathcal{D}_i$, and L is the loss function that penalizes the difference between y and the prediction provided by θ_i given input x.

Following [CHMS21, OKY21, XTH23], we consider the setting wherein there exists a common feature extractor $\phi : \mathbb{R}^I \to \mathbb{R}^d$ and client-specific heads $h_i : \mathbb{R}^d \to \mathcal{C}$ such that θ_i can be written as the composition of h_i and ϕ, i.e., $\theta_i(x) = (h_i \circ \phi)(x)$. For example, in an l-layer perceptron, ϕ consists of the first $l - 1$ layers of the neural network, and h_i is the final linear classification layer. With such decomposition, the objective in Eq.(1) can be rewritten as

$$\min_{\phi \in \Phi} \frac{1}{M} \sum_{i=1}^{M} \min_{h_i \in \mathcal{H}} F_i(h_i \circ \phi). \tag{2}$$

Data Heterogeneity. Observing that the local data distribution $\mathcal{P}_i$ is a joint distribution on $X \times Y$, which can be written as $\mathcal{P}_i(x, y) = \mathcal{P}_i(y)\mathcal{P}_i(x \mid y)$, data non-IID arises in both prior probability shift (i.e., $\mathcal{P}_i(y) \neq \mathcal{P}_{i'}(y)$) and concept drift (i.e., $\mathcal{P}_i(x|y) \neq \mathcal{P}_{i'}(x|y)$) [KMA+21, LDCH22]. Furthermore, the local dataset $\mathcal{D}_i$ may be unbalanced, i.e., $n_i \neq n_{i'}$. In this work, we focus on the prior probability shift scenario, as do most recent works.

4 FedLDA Algorithm

Our algorithm works by alternating between client update and server update routines, which are formally described in Algorithm 1. These routines make use of three key sub-routines described below. The variables that are iteratively refined are the shared representation ϕ, the estimates of the per-class mean of extracted features $u^c \in \mathbb{R}^d$ for $c \in \mathcal{C}$, and the estimate of common covariance $\Sigma \in \mathbb{R}^{d\times d}$. Let $\pi_i \in \Delta^C$ be the local empirical distribution of classes for client i, i.e., $\pi_i^c = n_i^c/n_i$ where n_i^c is the number of samples of class c at client i's local dataset $\mathcal{D}_i$.

Collaborative-Moment-Computation(CMC): Rather than only using local data to compute the new per-class means and common covariance of the updated features, a client takes a momentum weighted average of the local moments and the most recent global moments, i.e.,

$$\widehat{\mu}_i^c = \frac{1}{n_i^c}\sum_{(x_i^j,y_i^j)\in\mathcal{D}_i} \phi_i(x_i^j)\mathbf{1}_{\{y_i^j=c\}}, \qquad \mu_i^c = \beta\mu^c + (1-\beta)\widehat{\mu}_i^c$$

$$\widehat{\Sigma}_i = \frac{1}{n_i-1}\sum_{(x_i^j,y_i^j)\in\mathcal{D}_i} \mathbf{1}_{\{y_i=c\}}(\phi_i(x_i^j)-\mu_i^c)^T(\phi_i(x_i^j)-\mu_i^c), \qquad \Sigma_i = \beta\Sigma + (1-\beta)\widehat{\Sigma}_i,$$

where parameter β controls the update speed. This approach mitigates local noise that arises from data unbalance and may be viewed as a shrinkage estimator towards the global model.

LDA Prediction: The classifier for client i with parameters h_i consists of a w_i and bias b_i, which models the posterior of latent representations z as follows:

$$\widehat{p}_i(y=c|z) = \frac{\exp(z^T w_i^c + b_i^c)}{\sum_{c'\in\mathcal{C}}\exp(z^T w_i^{c'} + b_i^{c'})} \tag{3}$$

in our LDA model, the corresponding weight w_i and bias b_i is derived from the server estimate of μ, Σ, and the local data prior π_i, resulting in Eq (4). A full derivation is given in appendix 6.1.

$$\widehat{p}_i(y=c|z) = \frac{\exp(z^T\Sigma^{-1}\mu^c - \frac{1}{2}(\mu^c)^T\Sigma^{-1}\mu^c + \log\pi_i^c)}{\sum_{c'\in\mathcal{C}}\exp(z^T\Sigma^{-1}\mu^{c'} - \frac{1}{2}(\mu^{c'})^T\Sigma^{-1}\mu^{c'} + \log\pi_i^c)} \tag{4}$$

Local Loss Function: Using this classifier h_i, the client updates the backbone parameters ϕ through E epochs of gradient descent on the local dataset. The loss function for a mini-batch (X_b, Y_b) with n_b samples is the following regularized cross-entropy loss function:

$$F_i(X_b, Y_b) = -\frac{1}{n_b}\sum_{j=1}^{n_b}\sum_{c=1}^{\mathcal{C}} y_j^c \log\widehat{p}_i(y=c|\phi(x_j)) + L_{reg}(\widehat{\Sigma}), \tag{5}$$

where $L_{reg}(\widehat{\Sigma}) = \frac{\lambda}{d}\sum_{n\neq m}^{d}\left|\widehat{\Sigma}_{m,n}\right|$, and $\widehat{\Sigma}_{m,n}$ is the mth column and nth row of the batch covariance matrix. We choose this particular regularization because that the local estimation of $\widehat{\Sigma}$ is challenging as the empirical covariance may be singular ($n_i \ll d$). Historically this issue has been alleviated through sparse estimation or by discarding off-diagonal entries, however these methods may concede discriminative information if the underlying covariance matrix does not meet these conditions. Thus, we introduce this regularization loss to encourage the network to produce features with a sparse covariance matrix, enabling us to use empirical covariance estimates while maintaining the optimality of high-dimension LDA.

Server Update. The neural network representation ϕ is initialized using the popular Kaiming method [HZRS15]. Each coordinate of u^c is randomly and independently initialized according to $\mathcal{U}(-0.1, 0.1)$. Σ is initialized to be the identity matrix I_d. In each round, the server randomly selects $S(t)$ clients to perform local updates. After the client updates are complete, the server computes a weighted average of $(\{\mu_i^c\}_{c\in\mathcal{C}}, \Sigma_i, \phi_i)$ for $i \in S(t)$ as is done in FedAvg [MMR+17].

Client Update. If client i is selected to participate in a global round, it first receives the server estimates for $(\{\mu^c\}_{c\in\mathcal{C}}, \Sigma, \phi)$. After computing the local LDA classifier, the client fine-tunes representation ϕ using the local loss function described above. Finally, the client updates its estimate of the Gaussian parameters according to the CMC method and broadcasts all updated parameters to the server.

Discussion. The primary advantage of our FedLDA method comes from our Bayes-optimal classifiers based on collaboratively estimated statistics. Other PFL methods, such as FedRep [CHMS21], produce client classifiers through iterations of gradient descent on local datasets. Thus the resulting classifiers may generalize poorly due to incomplete convergence or limited training data. Even FedPAC [XTH23], which combines classifiers across clients based on validation set performance, is restricted by the original training scheme of each client classifier.

Algorithm 1 Personalized LDA
(Inputs: learning rate α, momentum parameter β, number of local epochs E)

```
Initialize neural network with Kaiming method [HZRS15];
Initialize per-class means: $\mu^c(0) \sim \mathcal{U}(0.1, -0.1)_d$ for each $c \in \mathcal{C}$;
Initialize tied covariance: $\Sigma(0) \leftarrow I_d$;
for $t = 1, 2, \cdots$ do
    Randomly sample $S(t)$ clients to participate in round $t$;
    for each client $i \in S(t)$ do
        $(\phi_i(t), \{\mu_i^c(t)\}_{c\in\mathcal{C}}, \Sigma_i(t)) \leftarrow$ ClientUpdate($\phi(t)$, $\{\mu^c(t)\}_{c\in\mathcal{C}}$, $\Sigma(t)$);
    end for
    $m(t) \leftarrow \sum_{i\in S(t)} n_i$;                    ▷ weighted average of client updates
    $\phi(t+1) \leftarrow \sum_{i\in S(t)} \frac{n_i}{m(t)} \phi_i(t)$;
    $\mu^c(t+1) \leftarrow \sum_{i\in S(t)} \frac{n_i}{m(t)} \mu_i^c(t)$ for each $c \in \mathcal{C}$;
    $\Sigma(t+1) \leftarrow \sum_{i\in S(t)} \frac{n_i}{m(t)} \Sigma_i(t)$;
end for

ClientUpdate($\phi$, $\{\mu^c\}_{c\in\mathcal{C}}$, $\Sigma$):                    ▷ local update of participating client $i$
$\phi_i \leftarrow \phi$;
$w_i^c \leftarrow \Sigma^{-1}\mu^c$,    $b_i^c \leftarrow -\frac{1}{2}(\mu^c)^\top \Sigma^{-1} \mu^c + \log \pi_i^c$    for all $c \in \mathcal{C}$;
$h_i \leftarrow \{w_i^c,\ b_i^c\}_{c\in\mathcal{C}}$;
for $r = 1, \cdots, E$ do
    Split $\mathcal{D}_i$ into $B$ mini-batches $\mathcal{B} = \{(X_b, Y_b)\}_{b=1}^B$;
    for Each batch $(X_b, Y_b) \in \mathcal{B}$ do
        $\phi_i \leftarrow \phi_i - \alpha \nabla_{\phi_i} F_i(X_b, Y_b)$
    end for
end for
$(\{\mu_i^c\}_{c\in\mathcal{C}}, \Sigma_i) \leftarrow \mathrm{CMC}(X_i, \phi_i)$        ▷ Momentum update of statistics using tuned feature extractor $\phi_i$
return $\phi_i, \{\mu_i^c\}_{c\in\mathcal{C}}, \Sigma_i$
```

5 Experiments

5.1 Experimental Setup

Datasets and Models. We evaluate our method on three popular image classification datasets: Fashion-MNIST with 10 categories of clothing, and CIFAR10/CIFAR100 with 10 and 100 classes of natural images.

We use two different model architectures for each experiment, to illustrate the robustness of our method to the latent dimension d. These architectures are ResNet18 [HZRS16] with $d = 512$ and WideResNet-16-2 [ZK16] with $d = 128$.

Non-IID Partition. We consider the commonly used realistic non-IID setting where the client label distributions are drawn from the Dirichlet distribution, as in [LKSJ20]. We fix the Dirichlet parameter $\alpha = 0.1$ for all experiments. The data volume on each client follows a 80-20% split between training and validation. Unless stated otherwise, we fix the total volume size to 500 for CIFAR-10 and FMNIST, and 1000 for CIFAR-100.

Model Training. We train all models with SGD for 200 global rounds with $E = 5$ local epochs per round, and a local batch size of 50 for all experiments. Hyperparameters were tuned for all methods using the CIFAR10 dataset with ResNet18 and $n = 2000$ local samples per client. Notably, we use $\beta = 0.5$ and $\lambda = 0.1$ for FedLDA. We report the average test accuracy across clients.

5.2 Numerical Results

Performance Comparison. We conduct two main experiments to illustrate the robustness of our method to limited data volume sizes and consistency across multiple datasets. As presented in Table 1, our method performs favorably in both small and large-sample settings, and and additionally is a top performer for multiple benchmark datasets as shown in Table 2. The benefit of our method is even larger for the WideResNet architecture due to the smaller latent dimension.

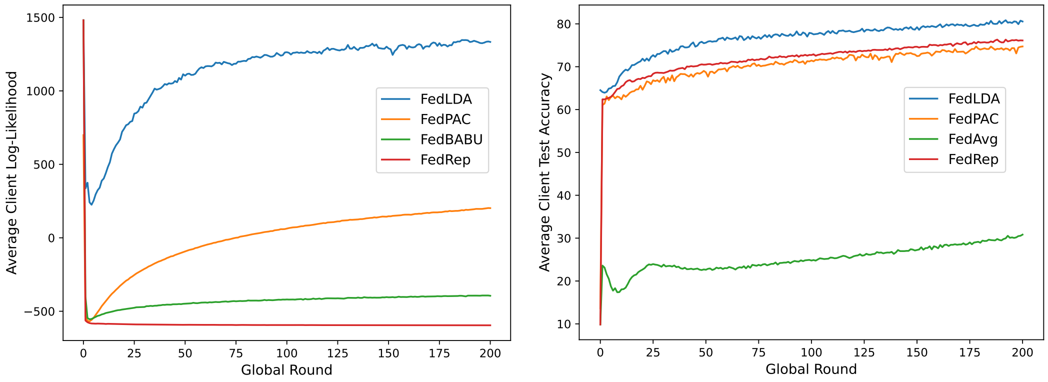
Method	ResNet18					WideResNet-16-2				
	500	1000	1500	2000	Dir	500	1000	1500	2000	Dir
FedAvg	34.40	42.60	48.95	52.40	43.90	25.05	30.83	34.42	36.81	27.79
Ditto	76.30	78.03	80.07	80.14	87.52	73.50	77.48	80.42	81.30	88.00
LG-FedAvg	74.25	76.20	77.65	78.15	86.94	71.75	76.18	79.02	79.14	86.24
FedRep	76.30	78.30	79.68	79.40	86.85	75.45	76.12	78.73	79.40	87.15
FedBABU	75.70	80.10	81.87	**82.25**	87.59	71.5	74.25	75.28	77.32	85.80
FedPAC	72.45	75.48	75.20	76.75	83.30	72.75	74.72	76.88	77.42	84.92
FedLDA	**77.75**	**81.25**	**81.96**	82.11	**87.86**	**78.85**	**80.18**	**82.17**	**84.50**	**90.38**

Table 1: Results on CIFAR-10 for varying local data volume size. In column 'Dir', we use the entirety of CIFAR-10 with local volume sizes distributed according to the Dirichlet distribution.

Method	ResNet18			WideResNet-16-2		
	CIFAR10	CIFAR100	FMNIST	CIFAR10	CIFAR100	FMNIST
FedAvg	34.40	18.02	76.60	25.05	11.75	28.90
Ditto	76.30	38.10	93.15	73.50	39.55	85.95
LG-FedAvg	74.25	39.48	93.05	71.75	43.88	90.35
FedRep	76.30	41.45	**93.20**	75.45	40.12	83.05
FedBABU	75.70	38.30	**93.20**	71.50	30.35	83.80
FedPAC	72.45	38.02	90.50	72.75	33.32	65.50
FedLDA (ours)	**77.75**	**45.68**	92.75	**78.85**	**48.20**	**93.10**

Table 2: Average client accuracy on heterogeneous CIFAR10, CIFAR100, and FMNIST datasets.

Gaussianity and Convergence. We additionally show the mean negative log likelihood of the class-conditional Gaussian across clients using ground-truth statistics for μ, Σ in Figure 1a. This demonstrates that our method successfully maintains the Gaussianity of latent representations, enabling the use of LDA for client classifiers. We additionally compare the testing accuracy at each round in Figure 1b, showing that FedLDA quickly converges to a better model.



(a) Gaussianity on CIFAR-10 with WideResNet.

(b) Client Accuracy on CIFAR-10 with WideResNet.

References

[BCV13] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[CC21] Hong-You Chen and Wei-Lun Chao. On bridging generic and personalized federated learning for image classification. *arXiv preprint arXiv:2107.00778*, 2021.

[CHMS21] 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.

[CHWE11] Line Clemmensen, Trevor Hastie, Daniela Witten, and Bjarne Ersbøll. Sparse discriminant analysis. *Technometrics*, 53(4):406–413, 2011.

[CL11] Tony Cai and Weidong Liu. A direct estimation approach to sparse linear discriminant analysis. *Journal of the American statistical association*, 106(496):1566–1577, 2011.

[DIS20] Fatih Demir, Aras Masood Ismael, and Abdulkadir Sengur. Classification of lung sounds with cnn model using parallel pooling structure. *IEEE Access*, 8:105376–105383, 2020.

[DKW15] Matthias Dorfer, Rainer Kelz, and Gerhard Widmer. Deep linear discriminant analysis. *arXiv preprint arXiv:1511.04707*, 2015.

[Efr75] Bradley Efron. The efficiency of logistic regression compared to normal discriminant analysis. *Journal of the American Statistical Association*, 70(352):892–898, 1975.

[Fis36] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[FMO20] Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning with theoretical guarantees: A model-agnostic meta-learning approach. *Advances in Neural Information Processing Systems*, 33:3557–3568, 2020.

[Fri89] Jerome H Friedman. Regularized discriminant analysis. *Journal of the American statistical association*, 84(405):165–175, 1989.

[GKC19] Benyamin Ghojogh, Fakhri Karray, and Mark Crowley. Eigenvalue and generalized eigenvalue problems: Tutorial. *arXiv preprint arXiv:1903.11240*, 2019.

[Han06] David J Hand. Classifier technology and the illusion of progress. 2006.

[HK20] Tyler L Hayes and Christopher Kanan. Lifelong machine learning with deep streaming linear discriminant analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 220–221, 2020.

[HQB20] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. Federated visual classification with real-world data distribution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 76–92. Springer, 2020.

[HTB94] Trevor Hastie, Robert Tibshirani, and Andreas Buja. Flexible discriminant analysis by optimal scoring. *Journal of the American statistical association*, 89(428):1255–1270, 1994.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification, 2015.

[HZRS16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[JKRK19] Yihan Jiang, Jakub Konečnỳ, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019.

[KKM$^+$20] 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.

[KMA$^+$21] 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.

[KMR20] 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.

[KSH17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[LDCH22] Qinbin Li, Yiqun Diao, Quan Chen, and Bingsheng He. Federated learning on non-iid data silos: An experimental study. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 965–978. IEEE, 2022.

[LHBS21] 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.

[LHY$^+$19] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. On the convergence of fedavg on non-iid data. *arXiv preprint arXiv:1907.02189*, 2019.

[LKSJ20] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems*, 33:2351–2363, 2020.

[LLZ$^+$20] 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.

[LSZ$^+$20] 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.

[MFSL19] Haggai Maron, Ethan Fetaya, Nimrod Segol, and Yaron Lipman. On the universality of invariant networks. In *International conference on machine learning*, pages 4363–4371. PMLR, 2019.

[MMR$^+$17] 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.

[OKY21] Jaehoon Oh, Sangmook Kim, and Se-Young Yun. Fedbabu: Towards enhanced representation for federated image classification. *arXiv preprint arXiv:2106.06042*, 2021.

[PDZ18] Tianyu Pang, Chao Du, and Jun Zhu. Max-mahalanobis linear discriminant analysis networks. In *International Conference on Machine Learning*, pages 4016–4025. PMLR, 2018.

[PVN82] Roger Peck and John Van Ness. The use of shrinkage estimators in linear discriminant analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (5):530–537, 1982.

[SCST17] Virginia Smith, Chao-Kai Chiang, Maziar Sanjabi, and Ameet S Talwalkar. Federated multi-task learning. *Advances in neural information processing systems*, 30, 2017.

[SLZ12] Andre Stuhlsatz, Jens Lippel, and Thomas Zielke. Feature extraction with deep neural networks by a generalized discriminant analysis. *IEEE transactions on neural networks and learning systems*, 23(4):596–608, 2012.

[SWDW11] Jun Shao, Yazhen Wang, Xinwei Deng, and Sijian Wang. Sparse linear discriminant analysis by thresholding for high dimensional data. *The Annals of Statistics*, 39(2), apr 2011.

[TDTN20] Canh T Dinh, Nguyen Tran, and Josh Nguyen. Personalized federated learning with moreau envelopes. *Advances in Neural Information Processing Systems*, 33:21394–21405, 2020.

[TLL$^+$22] Yue Tan, Guodong Long, Lu Liu, Tianyi Zhou, Qinghua Lu, Jing Jiang, and Chengqi Zhang. Fedproto: Federated prototype learning across heterogeneous clients. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8432–8440, 2022.

[TYCY22] Alysa Ziying Tan, Han Yu, Lizhen Cui, and Qiang Yang. Towards personalized federated learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2022.

[WXWZ21] Lixu Wang, Shichao Xu, Xiao Wang, and Qi Zhu. Addressing class imbalance in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 10165–10173, 2021.

[XTH23] Jian Xu, Xinyi Tong, and Shao-Lun Huang. Personalized federated learning with feature alignment and classifier collaboration. In *The Eleventh International Conference on Learning Representations*, 2023.

[ZHW$^+$22] Jianqing Zhang, Yang Hua, Hao Wang, Tao Song, Zhengui Xue, Ruhui Ma, and Haibing Guan. Fedala: Adaptive local aggregation for personalized federated learning. *arXiv preprint arXiv:2212.01197*, 2022.

[ZHZ21] Zhuangdi Zhu, Junyuan Hong, and Jiayu Zhou. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pages 12878–12889. PMLR, 2021.

[ZK16] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[ZLL$^+$18] Yue Zhao, Meng Li, Liangzhen Lai, Naveen Suda, Damon Civin, and Vikas Chandra. Federated learning with non-iid data. 2018.

[ZSF$^+$21] Michael Zhang, Karan Sapra, Sanja Fidler, Serena Yeung, and Jose M. Alvarez. Personalized federated learning with first order model optimization, 2021.

6 Appendix

6.1 Appendix A: From Softmax to LDA.

We first review the form of the softmax classifier. On client i, the local parameters h_i consist of a weight w_i and bias b_i, which model the posterior of latent representations z as follows:

$$p_i(y=c|z) = \frac{\exp(z^T w_i^c + b_i^c)}{\sum_{c'} \exp(z^T w_i^{(c')} + b_i^{(c')})} \tag{6}$$

Directly learning $p_i(y|z)$ in this manner can be challenging due to the shifts in client priors $p_i(y)$. Instead, we tackle a decomposed view of the posterior:

$$p_i(y=c|z) = \frac{p_i(y=c)p(z|y=c)}{\sum_{c'} p_i(y=c')p(z|y=c')} \tag{7}$$

Noting that the class-conditional distribution $p(z|y)$ is constant across clients, we propose to estimate this distribution in parallel to model training in order to efficiently obtain personalized solutions with the simple inclusion of client priors.

For ease of computation and for direct comparison with softmax classifiers, we consider the use of a class-conditional Gaussian with a tied covariance matrix to approximate $p(z|y)$. This model as Linear Discriminant Analysis, as the resulting decision boundary is linear (10).This is equivalent to a softmax classifier with weight $w^c = \Sigma^{-1}\mu^c$ and bias $b^c = -\frac{1}{2}(\mu^c)^T\Sigma^{-1}\mu^c + \log p_i(y=c)$.

$$\begin{aligned} p_i(y=c|z) &= \frac{p_i(y=c)N(z|\mu^c,\Sigma)}{\sum_{c'} p_i(y=c')N(z|\mu^c,\Sigma)} && (8) \\ \log p_i(y=c|z) &\propto \log p_i(y=c) - \frac{1}{2}(z-\mu^c)^T\Sigma^{-1}(z-\mu^c) && (9) \\ &= \log p_i(y=c) + z^T\Sigma^{-1}\mu^c - \frac{1}{2}(\mu^c)^T\Sigma^{-1}\mu^c && (10) \end{aligned}$$