Personalized Federated Learning via Low-Rank Matrix Factorization

Ali Dadras ALI.DADRAS@UMU.SE
Umeå University, Sweden

Sebastian U. Stich STICH@CISPA.DE
CISPA Helmholtz Center, Germany

Alp Yurtsever ALP.YURTSEVER@UMU.SE
Umeå University, Sweden

Abstract

Personalized Federated Learning (pFL) has gained attention for building a suite of models tailored to different clients. In pFL, the challenge lies in balancing the reliance on local datasets, which may lack representativeness, against the diversity of other clients' models, whose quality and relevance are uncertain. Focusing on the clustered FL scenario, where devices are grouped based on similarities in their data distributions without prior knowledge of cluster memberships, we develop a mathematical model for pFL using low-rank matrix optimization. Building on this formulation, we propose a pFL approach leveraging the Burer-Monteiro factorization technique. We examine the convergence guarantees of the proposed method, and present numerical experiments on training deep neural networks, demonstrating the empirical performance of the proposed method in scenarios where personalization is crucial.

1. Introduction

Federated Learning (FL) holds a great promise for training machine learning models over a large network with restricted data sharing. It is most suitable when clients require collaboration—often due to the absence of a large, representative dataset available locally—but in an environment where sharing datasets with collaborators is prohibited—often driven by concerns and regulations surrounding data sharing and storage. Consequently, FL research has been focused on designing algorithms that can solve optimization and learning problems on a network without sharing essential data. However, limitations on data sharing hinder effective control over the quality and relevance of the client data—a major concern that led to the rise of personalized federated learning models (pFL).

The goal in pFL is to find a right balance between the reliability of local datasets which may lack representativeness, and the diversity of collaborators' models whose quality and relevance are uncertain. Thus, pFL lacks a clear definition and direction without specified data distributions, and appears to lack a universally accepted metric for evaluating personalization success. Adding to this concern, many existing pFL methods are tested in settings that are inherently unsuited for pFL, where either Federated Averaging (FedAvg) or local training produces the best accuracies.

Motivated by these observations, our first step is to formulate a mathematical problem that highlights the role and necessity of a pFL approach. Suppose there are n clients collaborating on a FL system, indexed by $i = 1, \dots, n$, and assume that the data for each client comes from a specific data distribution, denoted by $\mathcal{D}_i$. We define the true objective function for each client as follows:

$$f_i^\natural(\mathbf{x}) := \mathbb{E}_{\xi \sim \mathcal{D}_i} \, \ell(\mathbf{x}, \xi), \tag{1}$$

where $\ell : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$ is a loss function. When data distributions are known, a solution can be found by minimizing $f_i^\natural(\mathbf{x}_i)$ locally: $\min_{\mathbf{x}_i}\ f_i^\natural(\mathbf{x}_i)$. However, the true distribution $\mathcal{D}_i$ is unknown in practice. Instead, clients have access to an empirical sample $\mathcal{S}_i$ with the corresponding objective:

$$f_i(\mathbf{x}) := \frac{1}{|\mathcal{S}_i|} \sum_{\xi \in \mathcal{S}_i} \ell(\mathbf{x}, \xi). \tag{2}$$

We operate under the assumption that the dataset $\mathcal{S}_i$ is not large enough for clients to accurately approximate a solution to their local problem on their own. Otherwise, FL would not be required.

An effective solution to this problem is possible only if the distributions $\mathcal{D}_i$ exhibit some correlation. At one extreme, when all distributions are the same, the standard template can be used:

$$\min_{\mathbf{x}_1,\dots,\mathbf{x}_n} \ \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}_i) \quad \text{s.t.} \quad \mathbf{x}_1 = \cdots = \mathbf{x}_n, \qquad \text{or equivalently as} \qquad \min_{\mathbf{x}} \ \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}). \tag{3}$$

A significant portion of existing pFL methods are designed by relaxing the equality constraint; examples include Moreau envelope smoothing and quadratic penalty regularization [22, 37]. However, these approaches that penalize model dissimilarity using a specific norm have limitations, as they rely on the assumption that similarity in distributions $\mathcal{D}_i$ translates to the proximity of client models in a given norm. The following simple examples demonstrate these limitations:

Example 1 (Label noise in classification) Consider a linear binary classification problem with two groups of clients that differ in their sign conventions. These groups label the positive and negative classes in opposite ways due to a misalignment. The data distributions of these two groups differ only by one bit. However, this difference results in the optimal models for the two groups having opposite signs, leading to solutions $\mathbf{x}^\star_{\text{group1}} = -\mathbf{x}^\star_{\text{group2}}$, which are distant in all norms.

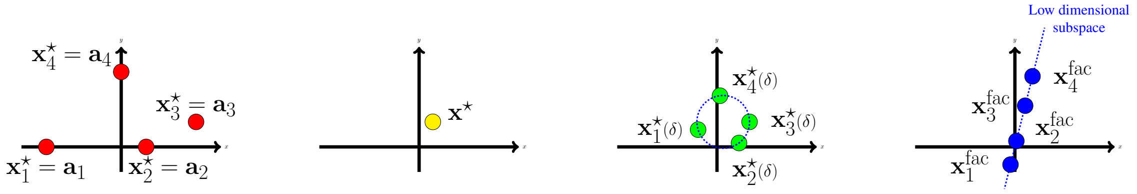
Example 2 (Clustered FL) Suppose each client draws data from one of r distinct distributions, forming r clusters of clients. We assume that cluster memberships are unknown, and the challenge is to establish effective collaboration without knowing in advance which clients share similar data.

Example 3 (Collaborative filtering) Consider the classical problem of recommendation systems. Suppose there are n clients and p items. Let $\mathbf{x}_i \in \mathbb{R}^p$ represent the relevance scores of client i for the items. The data consists of the actual scores rated by the clients, where each client rates only a subset of the items, denoted by $S_i \subseteq \{1, \dots, p\}$. We denote these scores by $x^\ast_{ij}$ for $j \in S_i$. The goal is predicting unknown scores that are not in S_i, based on hidden patterns among different clients.[1]

Models based on Euclidean distance regularizations fail in accurately predicting recommentation systems; instead, low-rank matrix factorization is among the most successful methods to collaborative filtering [20]. This is typically explained as user preferences being well-parameterized by a few meaningful factors; a more nuanced argument generalizes this by noting that low-rank matrices naturally arise in latent variable models (LVMs). While this is standard for LVMs with linear parameterizations, [39] demonstrate that low-rank models are effective for a broad class of (possibly high-dimensional) LVMs parameterized by a piecewise analytic function.

Inspired by these examples, we explore how to formulate pFL without relying on a specific distance metric. This leads us to investigate low-dimensional subspace formulations, where personalized models are related by their membership to a low-dimensional subspace, rather than their proximity in

[1] The decision variable in matrix completion reveals the data, limiting FL's privacy benefits. Nevertheless, the problem highlights the challenge of distributed learning with personalized models.


$\mathbf{x}_4^\star = \mathbf{a}_4$
$\mathbf{x}_3^\star = \mathbf{a}_3$
$\mathbf{x}_1^\star = \mathbf{a}_1$
$\mathbf{x}_2^\star = \mathbf{a}_2$
$\mathbf{x}^\star$
$\mathbf{x}_4^\star(\delta)$
$\mathbf{x}_1^\star(\delta)$
$\mathbf{x}_3^\star(\delta)$
$\mathbf{x}_2^\star(\delta)$
Low dimensional subspace
$\mathbf{x}_4^{\text{fac}}$
$\mathbf{x}_3^{\text{fac}}$
$\mathbf{x}_2^{\text{fac}}$
$\mathbf{x}_1^{\text{fac}}$

(a) Local (b) FedAvg (c) Distance based (d) Low-rank (ours)

Figure 1: Solutions to $\frac{1}{n}\sum_{i=1}^{n} \|\mathbf{x}_i - \mathbf{a}_i\|^2$, where $\mathbf{x}_i, \mathbf{a}_i \in \mathbb{R}^2$. Red points denote individual minimizers $\mathbf{x}_i^\star = \mathbf{a}_i$. FEDAVG solution, $\mathbf{x}^\star = \frac{1}{n}\sum_{i=1}^{n} \mathbf{a}_i$, is in yellow. The green points satisfy $\|\mathbf{x}_i - \mathbf{x}_j\| \le \delta$, and blue points ($r = 1$) lie in a low-dimensional subspace.

a distance metric. This approach allows us to conceptualize pFL by focusing on the inherent structure of the model relationships rather than their spatial closeness, as illustrated in Figure 1. Drawing parallels to collaborative filtering, we specifically focus on low-rank formulations.

We can now summarize our main contributions: We introduce a new formulation for pFL based on low-rank matrix optimization. Utilizing a nonconvex matrix factorization method applied to this formulation, we propose a new method called Personalized Federated Learning via Matrix Factorization ($\texttt{pFL}^{\texttt{MF}}$). We investigate the convergence guarantees of the proposed method. For the smooth nonconvex minimization problem, we show that the proposed method converges to a first-order stationary point at a rate of $\mathcal{O}(1/T)$; with the stochastic gradients, the rate becomes $\mathcal{O}(1/\sqrt{T})$. Finally, we present numerical experiments on training various types of neural networks.

2. Algorithm

We propose a novel formulation for pFL based on low-rank matrix optimization:

$$\min_{\mathbf{X}\in\mathbb{R}^{d\times n}} \; F(\mathbf{X}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}_i) \quad \text{s.t.} \quad \operatorname{rank}(\mathbf{X}) \le r. \tag{4}$$

Here, $\mathbf{X} \in \mathbb{R}^{d\times n}$ denotes the system-level decision variable obtained by concatenating clients' decision variables as $\mathbf{X} := [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$, and r is problem specific tuning parameter. Note that this formulation suits well for the examples we discussed in the introduction.

There exists a rich literature on rank-constrained matrix optimization problems, see [4, 5, 7, 11, 16, 21, 29, 33, 36] and the references therein. We adopt the nonconvex matrix factorization technique, *aka* Burer-Monteiro (BM) factorization, which replaces the system-level decision variable $\mathbf{X} \in \mathbb{R}^{d\times n}$ with a factorized form of $\mathbf{X} = \mathbf{U}\mathbf{V}^\top$. This leads to the following problem:

$$\min_{\mathbf{U}\in\mathbb{R}^{d\times r},\, \mathbf{V}\in\mathbb{R}^{n\times r}} \psi(\mathbf{U},\mathbf{V}), \qquad \text{where} \qquad \psi(\mathbf{U},\mathbf{V}) := F(\mathbf{U}\mathbf{V}^\top) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{U}\mathbf{v}_i). \tag{P}$$

We denote by $\mathbf{V}^\top := [\mathbf{v}_1, \cdots, \mathbf{v}_n] \in \mathbb{R}^{r\times n}$. In this notation, personalized model parameters can be computed as $\mathbf{x}_i = \mathbf{U}\mathbf{v}_i \in \mathbb{R}^d$.

While various optimization techniques can address problem (P), we simply use block-coordinate gradient updates. We can compute the gradient of ψ with respect to $\mathbf{U}$ and $\mathbf{v}_i$ as follows:

$$\nabla_{\mathbf{U}}\psi(\mathbf{U},\mathbf{V}) = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\mathbf{U}\mathbf{v}_i)\, \mathbf{v}_i^\top, \quad \text{and} \quad \nabla_{\mathbf{v}_i}\psi(\mathbf{U},\mathbf{V}) = \frac{1}{n}\mathbf{U}^\top \nabla f_i(\mathbf{U}\mathbf{v}_i). \tag{5}$$

Algorithm 1 Personalized Federated Learning via Matrix Factorization (`pFL`$^{\texttt{MF}}$)

```
set $\mathbf{U}^0 \in \mathbb{R}^{m\times r}, \mathbf{v}_i^0 \in \mathbb{R}^r \ \forall i \in [n]$.
for round $t = 0, 1, \ldots, T-1$ do
    — Client-level local training ————————-
    for client $i \in \mathcal{S}_t$ do
        set $\mathbf{v}_i^{t,1} = \mathbf{v}_i^t$.
        for  $k = 0, \ldots, K-1$ do
            $\mathbf{v}_i^{t,k+1} = \mathbf{v}_i^{t,k} - \eta \frac{1}{n} \mathbf{U}^{t^\top} \ \tilde{\nabla} f_i(\mathbf{U}^t \mathbf{v}_i^{t,k})$
        end for
        $\mathbf{v}_i^{t+1} = \mathbf{v}_i^{t,K}$
        $\mathbf{G}_i^t = \tilde{\nabla} f_i(\mathbf{U}^t \mathbf{v}_i^t) \ \mathbf{v}_i^{t^\top}$
        Client communicates $\mathbf{G}_i^t$ to the server.
    end for
    — Server-level aggregation ————————
    $\mathbf{U}^{t+1} = \mathbf{U}^t - \eta \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \ \mathbf{G}_i^t$
    Server communicates $\mathbf{U}^{t+1}$ to the clients.
end for
```

It is crucial that ψ is separable with respect to $\mathbf{v}_i$, enabling clients to compute $\nabla_{\mathbf{v}_i} \psi(\mathbf{U}, \mathbf{V})$ in parallel without requiring access to data or model parameters from other clients, given the features $\mathbf{U}$. Consequently, for a given step-size $\eta_i > 0$, local training steps can be independently formulated and performed by each participating client as:

$$\mathbf{v}_i^{t+1} = \mathbf{v}_i^t - \eta_i \frac{1}{n} \mathbf{U}^{t^\top} \ \nabla f_i(\mathbf{U}^t \mathbf{v}_i^t). \tag{6}$$

On the other hand, ψ is not separable with respect to the rows or columns of $\mathbf{U}$, necessitating collaboration among clients for computing $\nabla_{\mathbf{U}} \psi(\mathbf{U}, \mathbf{V})$. Consequently, the gradient step in $\mathbf{U}$ requires communication and will be performed at the server, forming our aggregation step:

$$\mathbf{U}^{t+1} = \mathbf{U}^t - \frac{1}{n} \sum_{i=1}^{n} \ \eta_i \big(\nabla f_i(\mathbf{U}^t \mathbf{v}_i^t)\big) \ \mathbf{v}_i^{t^\top}. \tag{7}$$

Algorithm 1 depicts the pseudo-code of our algorithm. Here, K is the number of local passes each client performs, and the output of the algorithm is a set of personalized parameters $\mathbf{x}_i = \mathbf{U}\mathbf{v}_i$ that each client can compute locally using its feature extractors $\mathbf{v}_i$ and the shared feature representation $\mathbf{U}$.

Convergence Guarantees. Several works have studied the convergence for the problem (P) under different assumptions; we refer to [8] and references therein. [4, 30] proved linear/sub-linear rates for smooth functions and smooth and strongly convex functions, respectively. Due to the nonconvex nature of BM factorization, even in cases where $F(.)$ is convex in $\mathbf{X}$, it is not possible to prove a convergence theorem to the global minimum. For more specialized cases (e.g., matrix sensing problems under some technical assumptions called restricted isometry property, convergence to a global solution can be characterized with careful initialization procedures [17, 28, 30, 43]. Since our focus is primarily on neural network applications, where objectives are already nonconvex in $\mathbf{X}$, we derive convergence guarantees to a stationary point, both with full and stochastic gradient settings.

Assumption 1 (Directional smoothness) *We assume that $F(\mathbf{U}\mathbf{V}^\top)$ is smooth with respect to $\mathbf{U}$ and $\mathbf{V}$, i.e., there exist constants $L_U, L_V \geq 0$ such that for all $\mathbf{U}_1, \mathbf{U}_2 \in \mathbb{R}^{d\times r}$ and $\mathbf{V}_1, \mathbf{V}_2 \in \mathbb{R}^{n\times r}$:*

$$\|\nabla_{\mathbf{U}} F(\mathbf{U}_1\mathbf{V}_1^\top) - \nabla_{\mathbf{U}} F(\mathbf{U}_2\mathbf{V}_2^\top)\|_F \leq L_U\Big(\|\mathbf{U}_1 - \mathbf{U}_2\|_F + \|\mathbf{V}_1 - \mathbf{V}_2\|_F\Big)$$
$$\|\nabla_{\mathbf{V}} F(\mathbf{U}_1\mathbf{V}_1^\top) - \nabla_{\mathbf{V}} F(\mathbf{U}_2\mathbf{V}_2^\top)\|_F \leq L_V\Big(\|\mathbf{U}_1 - \mathbf{U}_2\|_F + \|\mathbf{V}_1 - \mathbf{V}_2\|_F\Big)$$

Assumption 2 (Stochastic gradients) *We assume access to an unbiased stochastic gradient estimator with bounded variance, i.e.,, there exists $\sigma < +\infty$ such that for all $\mathbf{U} \in \mathbb{R}^{d\times r}$ and $\mathbf{V} \in \mathbb{R}^{n\times r}$:*

$$\mathbb{E}\left[\tilde{\nabla} F(\mathbf{U}\mathbf{V}^\top)\right] = \nabla F(\mathbf{U}\mathbf{V}^\top) \qquad \textit{and} \qquad \begin{aligned} \mathbb{E}\left[\left\|\tilde{\nabla}_{\mathbf{U}} F(\mathbf{U}\mathbf{V}^\top) - \nabla_{\mathbf{U}} F(\mathbf{U}\mathbf{V}^\top)\right\|^2\right] &\leq \sigma^2 \\ \mathbb{E}\left[\left\|\tilde{\nabla}_{\mathbf{V}} F(\mathbf{U}\mathbf{V}^\top) - \nabla_{\mathbf{V}} F(\mathbf{U}\mathbf{V}^\top)\right\|^2\right] &\leq \sigma^2. \end{aligned}$$

Theorem 1 *Consider problem (P) with smooth loss functions $f_i(.)$ in the sense that Assumption 1 holds. Assume access to a stochastic gradient estimator such that Assumption 2 holds. Furthermore, assume that every client participates in each round with probability p and performs K local steps per iteration. Then, the sequence $\mathbf{U}^t, \mathbf{V}^t$ generated by $\texttt{pFL}^{\texttt{MF}}$ with step-sizes $\eta_v = \frac{p\eta_u}{K}$ and $\eta_u < \frac{1}{2L}$, where $L := \max\{L_U, L_V\}$, satisfies the following bound:*

$$\frac{1}{T}\sum_{t=0}^{T-1}\left(\mathbb{E}\left[\|\nabla_{\mathbf{U}} F(\mathbf{U}^t\mathbf{V}^{t\top})\|^2\right] + \mathbb{E}\left[\frac{1}{K}\sum_{k=0}^{K-1}\|\nabla_{\mathbf{V}} F(\mathbf{U}^t\mathbf{V}^{t,k\top})\|^2\right]\right)$$
$$\leq \frac{2\Big(F(\mathbf{U}^0\mathbf{V}^{0\top}) - F^\star\Big)}{\eta T\Big(1 - 2\eta L\Big)} + \frac{2\eta L\sigma^2}{1 - 2\eta L}.$$

Corollary 2 *Choosing $\eta = \frac{1}{2L\sqrt{T}}$ in Theorem 1 yields a rate of $\mathcal{O}(1/\sqrt{T})$ in the stochastic setting. If full gradients are available ($\sigma = 0$), then $\eta = \frac{1}{4L}$ results in a convergence rate of $\mathcal{O}(1/T)$.*

3. Numerical Experiments

We compare the performance of $\texttt{pFL}^{\texttt{MF}}$ against several baselines, including LOCAL training, FEDAVG [25], FEDPER [3], FEDREP [9], APFL [10], and CFL [35] by implementing $\texttt{pFL}^{\texttt{MF}}$ in the *FL-Bench* benchmark [38].

We used a three-layer neural network, consisting of three linear layers, on the MNIST and FEMNIST datasets and a four-layer convolutional neural network, consisting of two convolutional layers followed by two linear layers, on the CIFAR10 and CIFAR100 datasets. For FEDPER and FEDREP, we treated the last layer as the classifier, while in $\texttt{pFL}^{\texttt{MF}}$, we factorized the entire model.

We conducted experiments in four different setups (see Appendix C for more details and data visualization): **Setup (1)** For the MNIST, CIFAR10, and CIFAR100 datasets, we split the data according to the Dirichlet distribution $\mathrm{Dir}(0.5)$ and $\mathrm{Dir}(1)$ across 100 clients. **Setup (2)** For the CIFAR-100 dataset, we partitioned the 100 classes into 20 groups, each containing 5 distinct labels. Data was then distributed among 500 clients, with each client exclusively assigned data from a single group, resulting in highly heterogeneous data. **Setup (3)** For the MNIST, we follow the experimental setup in [34] and consider 1000 clients divided into 10 groups, and labels in each group

	MNIST		CIFAR10		CIFAR100	
	Dir(0.5)	**Dir(1)**	**Dir(0.5)**	**Dir(1)**	**Dir(0.5)**	**Dir(1)**
LOCAL	92.12% (±0.59)	89.15% (±1.12)	59.14% (±3.74)	48.53% (±2.53)	16.09% (±1.52)	10.66% (±0.98)
FEDAVG	96.92% (±0.65)	97.01% (±0.54)	65.21% (±2.11)	65.44% (±1.68)	28.30% (±1.82)	28.36% (±1.32)
FEDPER	96.30% (±0.26)	95.16% (±0.58)	66.86% (±3.19)	58.25% (±2.22)	19.98% (±1.6)	14.22% (±1.01)
FEDREP	95.04% (±0.40)	93.33% (±0.94)	65.16% (±3.44)	55.4% (±2.06)	17.49% (±1.12)	12.14% (±1.05)
APFL	97.93% (±0.51)	97.64% (±0.39)	65.99% (±2.06)	65.14% (±1.54)	27.07% (±1.57)	27.07% (±1.36)
CFL	96.92% (±0.72)	97.04% (±0.5)	64.97% (±2.68)	65.98% (±1.70)	27.02% (±1.48)	24.84% (±0.91)
pFL$^{\text{MF}}$						
$r = 1$	96.75% (±0.61)	96.53% (±0.59)	43.89% (±3.49)	64.03% (±1.66)	34.32% (±1.96)	35.24% (±1.77)
$r = 5$	96.78% (±0.51)	96.55% (±0.60)	60.73% (±2.86)	65.89% (±1.88)	35.64% (±2.09)	35.75% (±1.24)
$r = 10$	96.98% (±0.70)	96.84% (±0.56)	65.10% (±2.30)	**67.68**% (±1.56)	35.28% (±1.73)	**36.84**% (±1.50)
$r = 15$	**98.24**% (±0.26)	**97.93**% (±0.22)	**68.13**% (±2.43)	65.88% (±1.62)	**35.70**% (±1.77)	36.12% (±1.46)

Table 1: Performance of the algorithms for **Setup (1)**. The best accuracy is shown in boldface, and the second best is underlined.

	MNIST (permuted labels)	CIFAR100 (super groups)	FEMNIST	
	1000 clients	500 clients	1091 clients	
	1 local epoch	1 local epoch	1 local epoch	5 local epochs
LOCAL	25.36% (±0.013)	10.49% (±0.95)	50.77% (±0.053)	65.69%(0.012)
FEDAVG	12.02% (±0.022)	36.40% (±1.31)	65.40% (±0.017)	**77.19**%(0.013)
FEDPER	19.86% (±0.141)	14.80% (±0.81)	66.05% (±0.010)	67.72%(0.009)
FEDREP	21.30% (±0.148)	12.18% (±0.80)	66.10% (±0.013)	66.29%(±0.010)
pFL$^{\text{MF}}$				
$r = 1$	14.70% (±0.083)	42.94% (±1.22)	67.82% (±0.134)	71.42%(0.044)
$r = 5$	23.75% (±0.027)	44.70% (±1.91)	69.99% (±0.123)	72.09%(±0.208)
$r = 10$	34.23% (±0.090)	**45.57**% (±1.97)	72.56% (±0.023)	72.47%(±0.010)
$r = 15$	**39.31**% (±0.042)	45.43% (±1.23)	**73.59**% (0.092)	76.41%(±0.006)

Table 2: Performance of the algorithms for **Setup (2)**, **Setup (3)**, and **Setup (4)**. The best accuracy is shown in boldface, and the second best is underlined.

are re-mapped (permuted) according to a random permutation map. In other words, clients in group one would have the same numbers $\{0, \cdots, 9\}$ but labeled differently. **Setup (4)** We sampled 30% of the total clients from FEMNIST dataset without changing the underlying data distribution, then we removed clients with less than 10 data points. The remaining set has 1091 clients. We ran the experiments for 1 and 5 numbers of local epochs.

Observations. In the heterogeneous experiments (**Setup (1)**), $\texttt{pFL}^{\text{MF}}$ outperforms the other pFL methods in most of the cases, although algorithms perform very closely on the MNIST dataset. $\texttt{pFL}^{\text{MF}}$ improves the average test accuracy significantly when different groups of clients have similar data distributions but their data distributions are different from other groups' distributions, see Table 2. It is worth mentioning that the low test accuracy in **Setup (2)** is due to the simplicity of the neural network model rather than the algorithms used. Another important observation is the convergence behavior of $\texttt{pFL}^{\text{MF}}$ when clients perform multiple local updates. Although our analysis assumes a single local update per communication round, our experiments indicate that $\texttt{pFL}^{\text{MF}}$ can also achieve convergence with multiple local updates—a direction we plan to investigate further.

4. Conclusions

We introduced a new pFL formulation based on low-rank matrix optimization and developed a novel pFL algorithm utilizing Burer-Monteiro factorization. We further established convergence guarantees for the proposed method: for minimizing a smooth non-convex objective, the algorithm converges to a stationary point at a rate of $\mathcal{O}(1/T)$ with full gradients; and $\mathcal{O}(1/\sqrt{T})$ for the stochastic setting. Evaluations across four experimental setups highlight the practical significance of the proposed method, especially in scenarios where personalization is essential, and standard approaches are unable to adequately capture the complexity of the underlying data distributions.

We conclude by listing some limitations and future directions. Our numerical experiments demonstrate improved performance of $\texttt{pFL}^{\texttt{MF}}$ with multiple local steps; however, this enhancement is not reflected in our theoretical convergence guarantees. Establishing stronger guarantees that reflect this behavior is a valuable direction for future research. Another notable limitation is that our formulation currently factorizes the entire model (decision variable), which can be computationally intensive in some cases, particularly in large-scale neural network applications. A more efficient approach might be to apply the BM factorization selectively, targeting only a subset of the parameters, which could reduce overhead while maintaining its benefits. Exploring such partial factorizations is a promising direction for future research

Acknowledgments

This work was partially conducted during a visit to CISPA. We sincerely appreciate their support and the stimulating research environment they provided. AD and AY were supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation. The computations were enabled by the Berzelius resource provided by the Knut and Alice Wallenberg Foundation at the National Supercomputer Centre. AY further acknowledges support from the Swedish Research Council, under registration number 2023-05476.

References

[1] Muhammad Ammad-Ud-Din, Elena Ivannikova, Suleiman A Khan, Were Oyomno, Qiang Fu, Kuan Eeik Tan, and Adrian Flanagan. Federated collaborative filtering for privacy-preserving personalized recommendation system. *arXiv preprint arXiv:1901.09888*, 2019.

[2] Vito Walter Anelli, Yashar Deldjoo, Tommaso Di Noia, Antonio Ferrara, and Fedelucio Narducci. User-controlled federated matrix factorization for recommender systems. *Journal of Intelligent Information Systems*, 58(2):287–309, 2022.

[3] Manoj Ghuhan Arivazhagan, Vinay Aggarwal, Aaditya Kumar Singh, and Sunav Choudhary. Federated learning with personalization layers. *arXiv preprint arXiv:1912.00818*, 2019.

[4] Srinadh Bhojanapalli, Anastasios Kyrillidis, and Sujay Sanghavi. Dropping convexity for faster semi-definite optimization. In *Conference on Learning Theory*, pages 530–582. PMLR, 2016.

[5] Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical programming*, 95(2):329–357, 2003.

[6] Chenghao Cai, Dengfeng Ke, Yanyan Xu, and Kaile Su. Fast learning of deep neural networks via singular value decomposition. In *PRICAI 2014: Trends in Artificial Intelligence: 13th Pacific Rim International Conference on Artificial Intelligence, Gold Coast, QLD, Australia, December 1-5, 2014. Proceedings 13*, pages 820–826. Springer, 2014.

[7] Emmanuel Candes and Benjamin Recht. Exact matrix completion via convex optimization. *Communications of the ACM*, 55(6):111–119, 2012.

[8] Yuejie Chi, Yue M Lu, and Yuxin Chen. Nonconvex optimization meets low-rank matrix factorization: An overview. *IEEE Transactions on Signal Processing*, 67(20):5239–5269, 2019.

[9] 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.

[10] Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

[11] Donald Goldfarb and Shiqian Ma. Convergence of fixed-point continuation algorithms for matrix rank minimization. *Foundations of Computational Mathematics*, 11(2):183–210, 2011.

[12] Mary Gooneratne, Khe Chai Sim, Petr Zadrazil, Andreas Kabel, Françoise Beaufays, and Giovanni Motta. Low-rank gradient approximation for memory-efficient on-device training of deep neural network. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3017–3021. IEEE, 2020.

[13] Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020.

[14] Weituo Hao, Nikhil Mehta, Kevin J Liang, Pengyu Cheng, Mostafa El-Khamy, and Lawrence Carin. Waffle: Weight anonymized factorization for federated learning. *IEEE Access*, 10: 49207–49218, 2022.

[15] Jiwei Huang, Zeyu Tong, and Zihan Feng. Geographical poi recommendation for internet of things: A federated learning approach using matrix factorization. *International Journal of Communication Systems*, page e5161, 2022.

[16] Prateek Jain, Raghu Meka, and Inderjit Dhillon. Guaranteed rank minimization via singular value projection. *Advances in Neural Information Processing Systems*, 23, 2010.

[17] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

[18] Wonyong Jeong and Sung Ju Hwang. Factorized-fl: Personalized federated learning with parameter factorization & similarity matching. *Advances in Neural Information Processing Systems*, 35:35684–35695, 2022.

[19] Shiva Prasad Kasiviswanathan. Sgd with low-dimensional gradients with applications to private and distributed learning. In *Uncertainty in Artificial Intelligence*, pages 1905–1915. PMLR, 2021.

[20] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[21] Anastasios Kyrillidis and Volkan Cevher. Matrix recipes for hard thresholding methods. *Journal of mathematical imaging and vision*, 48:235–265, 2014.

[22] 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.

[23] 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.

[24] Othmane Marfoq, Giovanni Neglia, Aurélien Bellet, Laetitia Kameni, and Richard Vidal. Federated multi-task learning under a mixture of distributions. *Advances in Neural Information Processing Systems*, 34:15434–15447, 2021.

[25] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. *Artificial intelligence and statistics, PMLR*, pages 1273–1282, 2017.

[26] Konstantin Mishchenko, Rustem Islamov, Eduard Gorbunov, and Samuel Horváth. Partially personalized federated learning: Breaking the curse of data heterogeneity. *arXiv preprint arXiv:2305.18285*, 2023.

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

[28] Dohyung Park, Anastasios Kyrillidis, Srinadh Bhojanapalli, Constantine Caramanis, and Sujay Sanghavi. Provable burer-monteiro factorization for a class of norm-constrained matrix problems. *arXiv preprint arXiv:1606.01316*, 2016.

[29] Dohyung Park, Anastasios Kyrillidis, Constantine Carmanis, and Sujay Sanghavi. Non-square matrix sensing without spurious local minima via the burer-monteiro approach. In *Artificial Intelligence and Statistics*, pages 65–74. PMLR, 2017.

[30] Dohyung Park, Anastasios Kyrillidis, Constantine Caramanis, and Sujay Sanghavi. Finding low-rank solutions via nonconvex matrix factorization, efficiently and provably. *SIAM Journal on Imaging Sciences*, 11(4):2165–2204, 2018.

[31] Krishna Pillutla, Kshitiz Malik, Abdel-Rahman Mohamed, Mike Rabbat, Maziar Sanjabi, and Lin Xiao. Federated learning with partial model personalization. In *International Conference on Machine Learning*, pages 17716–17758. PMLR, 2022.

[32] Francesco Pinto, Yaxi Hu, Fanny Yang, and Amartya Sanyal. Pillar: How to make semi-private learning more effective. *arXiv preprint arXiv:2306.03962*, 2023.

[33] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[34] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning. In *Proceedings of the NeurIPS'19 Workshop on Federated Learning for Data Privacy and Confidentiality*, pages 1–5, 2019.

[35] Felix Sattler, Klaus-Robert Müller, and Wojciech Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE transactions on neural networks and learning systems*, 32(8):3710–3722, 2020.

[36] Ruoyu Sun and Zhi-Quan Luo. Guaranteed matrix completion via non-convex factorization. *IEEE Transactions on Information Theory*, 62(11):6535–6579, 2016.

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

[38] Jiahao Tan, Yipeng Zhou, Gang Liu, Jessie Hui Wang, and Shui Yu. pfedsim: Similarity-aware model aggregation towards personalized federated learning. *arXiv preprint arXiv:2305.15706*, 2023.

[39] Madeleine Udell and Alex Townsend. Why are big data matrices approximately low rank? *SIAM Journal on Mathematics of Data Science*, 1(1):144–160, 2019.

[40] Dezhong Yao, Wanning Pan, Michael J O'Neill, Yutong Dai, Yao Wan, Hai Jin, and Lichao Sun. Fedhm: Efficient federated learning for heterogeneous models via low-rank factorization. *arXiv preprint arXiv:2111.14655*, 2021.

[41] Da Yu, Huishuai Zhang, Wei Chen, and Tie-Yan Liu. Do not let privacy overbill utility: Gradient embedding perturbation for private learning. *arXiv preprint arXiv:2102.12677*, 2021.

[42] Yong Zhao, Jinyu Li, and Yifan Gong. Low-rank plus diagonal adaptation for deep neural networks. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5005–5009. IEEE, 2016.

[43] Qinqing Zheng and John Lafferty. Convergence analysis for rectangular matrix completion using burer-monteiro factorization and gradient descent. *arXiv preprint arXiv:1605.07051*, 2016.

Appendix A. Proof of Theorem 1

A.1. Compact Notation

We assume that each client participates in the learning process independently with probability p. To model this, we define a partial participation matrix D_t for each time step t as a diagonal matrix, where each diagonal entry $[D_t]_{i,i}$ represents the participation status of client i at time t. Specifically,

$$[D_t]_{i,i} := \begin{cases} 1, & \text{with probability } p, \\ 0, & \text{with probability } 1-p, \end{cases}$$

where each $[D_t]_{i,i}$ is an independent Bernoulli random variable with parameter p. This implies that for each client i, $[D_t]_{i,i} = 1$ if the client participates in the training process at time t, and $[D_t]_{i,i} = 0$ otherwise. We can write our algorithm in the compact form as follows:

$$\begin{aligned}
&\mathbf{V}^{t,0} = \mathbf{V}^t \\
&\text{for } k = 0, \ldots, K-1, \text{ do} \\
&\qquad \mathbf{V}^{t,k+1} = \mathbf{V}^{t,k} - \eta_v \mathbf{D}_t \tilde{\nabla}_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top}) \\
&\text{end for} \\
&\mathbf{U}^{t+1} = \mathbf{U}^t - \eta_u \tilde{\nabla}_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top}) \\
&\mathbf{V}^{t+1} = \mathbf{V}^{t,K}\,.
\end{aligned}$$

We define expectations with respect to gradient noise as $\mathbb{E}^{\mathbf{V}}_{\text{noise}}[\cdot]$ and $\mathbb{E}^{\mathbf{U}}_{\text{noise}}[\cdot]$, and expectation with respect to participation randomness as $\mathbb{E}_{\mathcal{S}_t}[\cdot]$. For participation randomness, we assume that $\mathbb{E}_{\mathcal{S}_t}[\mathbf{D}_t] = p\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and p is the probability of client participation under independent sampling. We define the conditional expectation given all randomness before iteration t and local step k as

$$\mathbb{E}_{t,k}[\cdot] := \mathbb{E}^{\mathbf{V}}_{\text{noise}}\Big[\cdot \mid \text{randomness before } (t,k), \mathbf{D}_t\Big],$$

where the randomness includes all prior gradient noise and participation randomness up to local step k of iteration t. Additionally, we define the conditional expectation given all randomness in the algorithm before iteration t and the final local step K as

$$\mathbb{E}_t[\cdot] := \mathbb{E}^{\mathbf{U}}_{\text{noise}}\Big[\cdot \mid \text{randomness before } (t,K), \mathbf{D}_t\Big].$$

Finally, we use $\mathbb{E}[\cdot]$ to denote the total expectation over all sources of randomness in the algorithm, including gradient noise and client participation.

A.2. Convergence Analysis

We start with proving some useful bounds. For any t,

$$\begin{aligned}
\mathbb{E}_t\left[\|\tilde{\nabla}_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\|_F^2\right] &= \mathbb{E}_t\Big[\|\nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\|_F^2 + \|\tilde{\nabla}_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\|_F^2 \\
&\qquad + 2\langle \nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top}), \tilde{\nabla}_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\rangle\Big] \\
&\leq \|\nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\|_F^2 + \sigma^2.
\end{aligned} \tag{8}$$

Similar to above, we can write, for any t and k,

$$\begin{aligned}
\mathbb{E}_{t,k}\left[\|\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\right] &= \mathbb{E}_{t,k}\Bigg[\|\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \|\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 \\
&\qquad + 2\langle\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \underbrace{\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})}_{\mathbb{E}_{t,k}[\,.\,|\mathbf{D}_t]=0}\rangle\Bigg] \\
&= \|\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \|\mathbf{D}_t\|_2^2\cdot\mathbb{E}_{t,k}\left[\|\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\right] \\
&= \|\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \|\mathbf{D}_t\|_2^2\cdot\sigma^2
\end{aligned}$$

where in the third line, we used the submultiplicative property of the Frobenius norm. Now we take the expectation with respect to the participation probability

$$\begin{aligned}
\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[\|\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\right]\right] &= \mathbb{E}_{\mathcal{S}_t}\left[\|\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \|\mathbf{D}_t\|_2^2\cdot\sigma^2\right] \\
&=\leq p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2 + \sigma^2\,, \qquad (9)
\end{aligned}$$

where we used Theorem 5, and Theorem 7.
(A) First, we use the smoothness of $F(\mathbf{U}\mathbf{V}^{t^\top})$ with respect to $\mathbf{V}$ and write

$$\begin{aligned}
F(\mathbf{U}^t\mathbf{V}^{t,k+1^\top}) &\leq F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) + \langle\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \mathbf{V}^{t,k+1} - \mathbf{V}^{t,k}\rangle + \frac{L_V}{2}\|\mathbf{V}^{t,k+1} - \mathbf{V}^{t,k}\|_F^2 \\
&= F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_v\langle\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle + \eta_v^2\frac{L_V}{2}\|\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2.
\end{aligned}$$

Taking conditional expectation, we get

$$\begin{aligned}
\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[F(\mathbf{U}^t\mathbf{V}^{t,k+1^\top})\right]\right] &\leq F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_v p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \eta_v^2\frac{pL_V}{2}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \eta_v^2\frac{L_V}{2}\sigma^2 \\
&= F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_v p\left(1 - \eta_v\frac{L_V}{2}\right)\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \eta_v^2\frac{L_V\sigma^2}{2}
\end{aligned}$$

where we used (9) in the second line. We rearrange the inequality above and average over k to obtain

$$\begin{aligned}
&\eta_v p\left(1 - \eta_v\frac{L_V}{2}\right)\frac{1}{K}\sum_{k=0}^{K-1}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 \\
&\qquad\leq \frac{1}{K}\left(F(\mathbf{U}^t\mathbf{V}^{t,0^\top}) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^t\mathbf{V}^{t,K^\top})\right]\right]\right) + \eta_v^2\frac{L_V\sigma^2}{2} \\
&\qquad= \frac{1}{K}\left(F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^t\mathbf{V}^{t+1^\top})\right]\right]\right) + \eta_v^2\frac{L_V\sigma^2}{2} \qquad (10)
\end{aligned}$$

where, in the second line, we used $\mathbf{V}^{t,0} = \mathbf{V}^t$ and $\mathbf{V}^{t,K} = \mathbf{V}^{t+1}$.
(B) Now, we will use the smoothness again, but this time with respect to $\mathbf{U}$:

$$\begin{aligned}
F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top}) &\leq F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) + \langle\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \mathbf{U}^{t+1} - \mathbf{U}^t\rangle + \frac{L_U}{2}\|\mathbf{U}^{t+1} - \mathbf{U}^t\|_F^2 \\
&\leq F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_u\langle\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \tilde{\nabla}_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\rangle + \eta_u^2\frac{L_U}{2}\|\tilde{\nabla}_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2
\end{aligned}$$

Similar to the previous case, if we take expectation with respect to the randomness in $\mathbf{U}$ update at iteration t, we obtain the following bound by using (8):

$$\mathbb{E}_t\big[F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})\big] \le F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_u\langle \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\rangle + \eta_u^2\frac{L_U}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2 + \eta_u^2\frac{L_U\sigma^2}{2}.$$

If we split the inner product term as

$$\begin{aligned}\langle \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\rangle &= \langle \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) + \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle\\ &= \langle \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}), \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle + \|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\ge -\frac{\eta_u L_U}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 - \frac{1}{2\eta_u L_U}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\quad + \|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2,\end{aligned}$$

where the last line follows from Young's inequality (17) with $\alpha = \eta_u L_U$. Moreover, by the smoothness assumption, we have

$$\frac{1}{2L_U\eta_u}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|^2 \le \frac{L_U}{2\eta_u}\|\mathbf{V}^t - \mathbf{V}^{t,k}\|^2 \le \frac{\eta_v^2}{\eta_u}\frac{L_U}{2}\|\sum_{i=0}^{k-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}_i^{t^\top})\|_F^2.$$

Combining all these bounds, we get

$$\begin{aligned}\mathbb{E}_t\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})\right] \le F(\mathbf{U}^t\mathbf{V}^{t,k^\top}) - \eta_u\left(1-\eta_u\frac{L_U}{2}\right)\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \eta_u^2\frac{L_U}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2\\ + \eta_v^2\frac{L_U}{2}\|\sum_{i=0}^{k-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}_i^{t^\top})\|_F^2 + \eta_u^2\frac{L_U\sigma^2}{2}. \quad (11)\end{aligned}$$

Now we consider two cases $k = 0$ and $k = K$ in (11).

1. For $\mathbf{k} = \mathbf{0}$, we have

$$\mathbb{E}_t\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t^\top})\right] \le F(\mathbf{U}^t\mathbf{V}^{t^\top}) - \eta_u\left(1-\eta_u L_U\right)\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2 + \eta_u^2\frac{L_U\sigma^2}{2}. \quad (12)$$

   where we used $\mathbf{V}^{t,0} = \mathbf{V}^t$.

2. For $\mathbf{k} = \mathbf{K}$, we have

$$\begin{aligned}\mathbb{E}_t\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t+1})\right] &\le F(\mathbf{U}^t\mathbf{V}^{t+1^\top}) - \eta_u\left(1-\eta_u\frac{L_U}{2}\right)\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t+1^\top})\|_F^2 + \eta_u^2\frac{L_U}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2\\ &\quad + \eta_v^2\frac{L_U}{2}\|\sum_{k=0}^{K-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2 + \eta_u^2\frac{L_U\sigma^2}{2}\\ &\le F(\mathbf{U}^t\mathbf{V}^{t+1^\top}) + \eta_u^2\frac{L_U}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2 + \eta_v^2\frac{L_U}{2}\|\sum_{k=0}^{K-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\quad + \eta_u^2\frac{L_U\sigma^2}{2}.\end{aligned}$$

Rearranging the terms we can write

$$-\frac{L_U}{2}\left(\eta_u^2\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2+\eta_v^2\|\sum_{k=0}^{K-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\right) \tag{13}$$

$$\leq F(\mathbf{U}^t\mathbf{V}^{t+1^\top})-\mathbb{E}_t\big[F(\mathbf{U}^{t+1}\mathbf{V}^{t+1^\top})\big]+\eta_u^2\frac{L_U\sigma^2}{2}\,. \tag{14}$$

(C) We once again use smoothness with respect to $\mathbf{V}$:

$$\begin{aligned}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k+1^\top}) &\leq F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})+\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top}),\mathbf{V}^{t,k+1}-\mathbf{V}^{t,k}\rangle+\frac{L_V}{2}\|\mathbf{V}^{t,k+1}-\mathbf{V}^{t,k}\|_F^2\\ &=F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\eta_v\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top}),\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle+\eta_v^2\frac{L_V}{2}\|\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\,.\end{aligned}$$

We take the conditional expectation

$$\begin{aligned}\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t,k+1^\top})\right]\right]&\leq F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\eta_v\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top}),\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle\right]\right]\\ &\quad+\eta_v^2\frac{pL_V}{2}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2+\eta_v^2\frac{L_V\sigma^2}{2}\,,\end{aligned}$$

where we used (9). Focusing again on the inner product term, we obtain

$$\begin{aligned}&\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top}),\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle\right]\right]\\ &=\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})\pm\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}),\mathbf{D}_t\tilde{\nabla}_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle\right]\right]\\ &=\mathbb{E}_{\mathcal{S}_t}\left[\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}),\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle+\langle\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}),\mathbf{D}_t\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle\right]\\ &=p\langle\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top}),\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\rangle+p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\geq-\frac{p}{2\eta_vKL_V}\|\nabla_{\mathbf{V}}F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2-\frac{\eta_vpKL_V}{2}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2+p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\geq-\frac{pL_V}{2\eta_vK}\|\mathbf{U}^{t+1}-\mathbf{U}^t\|_F^2-\frac{\eta_vpKL_V}{2}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2+p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &=-\frac{pL_V}{2\eta_vK}\eta_u^2\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2-\frac{\eta_vpKL_V}{2}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2+p\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\,,\end{aligned}$$

where we used Young's inequality (17) in the fourth line with $\alpha=\eta_vKL_V$. Substituting back, we get

$$\begin{aligned}\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,k}\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t,k+1^\top})\right]\right]&\leq F(\mathbf{U}^{t+1}\mathbf{V}^{t,k^\top})-\eta_vp\left(1-\eta_v\frac{(K+1)L_V}{2}\right)\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2\\ &\quad+\eta_u^2\frac{pL_V}{2K}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|_F^2+\eta_v^2\frac{L_V\sigma^2}{2}\end{aligned}$$

By summing over k and appropriately rearranging the terms in the inequality, we obtain the following:

$$\begin{aligned}&\eta_vp\left(1-\eta_v\frac{(K+1)L_V}{2}\right)\sum_{k=0}^{K-1}\|\nabla_{\mathbf{V}}F(\mathbf{U}^t\mathbf{V}^{t,k^\top})\|_F^2-\eta_u^2\frac{pL_V}{2}\|\nabla_{\mathbf{U}}F(\mathbf{U}^t\mathbf{V}^{t^\top})\|^2\\ &\qquad\leq F(\mathbf{U}^{t+1}\mathbf{V}^{t^\top})-\mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^{t+1}\mathbf{V}^{t+1^\top})\right]\right]+\eta_v^2\frac{KL_V\sigma^2}{2}\,.\end{aligned}$$

where we used $\mathbf{V}^{t,K} = \mathbf{V}^{t+1}$ and $\mathbf{V}^{t,0} = \mathbf{V}^t$. We define

$$
\begin{aligned}
\Delta_U &:= \mathbb{E}_t\left[\|\nabla_{\mathbf{U}} F(\mathbf{U}^t {\mathbf{V}^t}^\top)\|_F^2\right] \\
\Delta_V &:= \mathbb{E}_{\mathcal{S}_t}\left[\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}_{t,k}\left[\|\nabla_{\mathbf{V}} F(\mathbf{U}^t {\mathbf{V}^{t,k}}^\top)\|_F^2\right]\right] .
\end{aligned} \tag{15}
$$

We summarize the resulting inequalities in part **(A)** to **(C)** as

$$
\begin{aligned}
&\eta_v p\left(1-\eta_v\frac{L_V}{2}\right)\Delta_V \le \frac{1}{K}\left(F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^t{\mathbf{V}^{t+1}}^\top)\right]\right]\right) + \eta_v^2\frac{L_V\sigma^2}{2} \\
&\eta_u\big(1-\eta_u L_U\big)\Delta_U \le F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_t\big[F(\mathbf{U}^{t+1}{\mathbf{V}^t}^\top)\big] + \eta_u^2\frac{L_U\sigma^2}{2} \\
&-\frac{L_U}{2}\left(\eta_u^2\Delta_U + \eta_v^2\Big\|\sum_{k=0}^{K-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t{\mathbf{V}^{t,k}}^\top)\Big\|^2\right) \le F(\mathbf{U}^t{\mathbf{V}^{t+1}}^\top) - \mathbb{E}_t\big[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\big] + \eta_u^2\frac{L_U\sigma^2}{2} \\
&\eta_v p\left(1-\eta_v\frac{(K+1)L_V}{2}\right)K\Delta_V - \eta_u^2\frac{pL_V}{2}\Delta_U \le F(\mathbf{U}^{t+1}{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\right]\right] + \eta_v^2\frac{KL_V\sigma^2}{2} .
\end{aligned}
$$

where we used the definitions (15). We rewrite four inequalities above using $\eta_v = \frac{p\eta_u}{K} := \frac{p\eta}{K}$ and defining $L := \max\{L_U, L_V\}$ as

$$
\begin{aligned}
&\eta p^2\big(1-\eta\frac{pL}{2K}\big)\Delta_V \le F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^t{\mathbf{V}^{t+1}}^\top)\right]\right] + \eta^2\frac{L\sigma^2}{2K} \\
&\eta\big(1-\eta L\big)\Delta_U \le F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_t\big[F(\mathbf{U}^{t+1}{\mathbf{V}^t}^\top)\big] + \eta^2\frac{L\sigma^2}{2} \\
&-\frac{L}{2}\left(\eta^2\mathbb{E}_t[\Delta_U] + \frac{\eta^2p^2}{K^2}\Big\|\sum_{k=0}^{K-1}\nabla_{\mathbf{V}}F(\mathbf{U}^t{\mathbf{V}^{t,k}}^\top)\Big\|^2\right) \le F(\mathbf{U}^t{\mathbf{V}^{t+1}}^\top) - \mathbb{E}_t\big[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\big] + \eta^2\frac{L\sigma^2}{2} \\
&\frac{\eta p^2}{K}\big(1-\eta p\frac{(K+1)L}{2K}\big)K\Delta_V - \eta^2\frac{pL}{2}\Delta_U \le F(\mathbf{U}^{t+1}{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\right]\right] + \eta^2\frac{L\sigma^2}{2K} .
\end{aligned}
$$

Summing up the inequalities above, we get

$$
\begin{aligned}
\eta\Big(1-2\eta L\Big)\left(\Delta_U+\Delta_V\right) &\le \eta\Big(1-2\eta L\Big)\Delta_U + \eta p\big(1-\eta\frac{pL}{2K}\big)\Delta_V \\
&\le 2\left(F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\right]\right]\right) + \eta^2\big(1+\frac{1}{K}\big)L\sigma^2 \\
&\le 2\left(F(\mathbf{U}^t{\mathbf{V}^t}^\top) - \mathbb{E}_{\mathcal{S}_t}\left[\mathbb{E}_{t,K}\left[F(\mathbf{U}^{t+1}{\mathbf{V}^{t+1}}^\top)\right]\right]\right) + 2\eta^2L\sigma^2 ,
\end{aligned} \tag{16}
$$

where we used the following inequality

$$
\begin{aligned}
&\eta^2 \frac{p^2 L}{2K^2} \Big\| \sum_{k=0}^{K-1} \nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top}) \Big\|^2 - \frac{\eta p^2}{K}\left(1 - \eta p \frac{(K+1)L}{2K}\right) \sum_{k=0}^{K-1} \|\nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top})\|^2 \\
&\quad \le \eta^2 \frac{p^2 L}{2K^2} K \sum_{k=0}^{K-1} \Big\|\nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top})\Big\|^2 - \frac{\eta p^2}{K}\left(1 - \eta p \frac{(K+1)L}{2K}\right) \sum_{k=0}^{K-1} \|\nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top})\|^2 \\
&\quad \le \frac{\eta p^2}{K}\Big(\eta \frac{L}{2}(\frac{(1+p)K+1}{K}) - 1\Big) \sum_{k=0}^{K-1} \|\nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top})\|^2 \\
&\quad \le 0 \quad (\eta \le \frac{1}{2L}),
\end{aligned}
$$

where in the second line, we used Jensen's inequality, see (18). Next, we take the expectation over all sources of randomness in the algorithm. Then, we average both sides of (16) over the iterations t, followed by dividing both sides by $\eta\,(1 - 2\eta L)$, yielding the following expression:

$$
\begin{aligned}
&\frac{1}{T} \sum_{t=0}^{T-1} \left(\mathbb{E}\Big[\|\nabla_{\mathbf{U}} F(\mathbf{U}^t \mathbf{V}^{t^\top})\|^2\Big] + \mathbb{E}\left[\frac{1}{K} \sum_{k=0}^{K-1} \|\nabla_{\mathbf{V}} F(\mathbf{U}^t \mathbf{V}^{t,k^\top})\|^2\right]\right) \\
&\qquad \le \frac{2\Big(F(\mathbf{U}^0 \mathbf{V}^{0^\top}) - F(\mathbf{U}^T \mathbf{V}^{T^\top})\Big)}{\eta T\Big(1 - 2\eta L\Big)} + \frac{2\eta L \sigma^2}{1 - 2\eta L} \\
&\qquad \le \frac{2\Big(F(\mathbf{U}^0 \mathbf{V}^{0^\top}) - F^\star\Big)}{\eta T\Big(1 - 2\eta L\Big)} + \frac{2\eta L \sigma^2}{1 - 2\eta L},
\end{aligned}
$$

provided that $L = \max\{L_U, L_V\}$, $\eta_v = \frac{p\eta_u}{K} = \frac{p\eta}{K}$, and $\eta \le \frac{1}{2L}$. This completes the proof.

□

A.3. Useful Inequalities

Lemma 3 (Young's inequality) *Let $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^{d_1 \times d_2}$ and $\alpha > 0$. Then, the following inequality holds:*

$$\langle \mathbf{X}, \mathbf{Y} \rangle \leq \frac{\alpha}{2} \|\mathbf{X}\|_F^2 + \frac{1}{2\alpha} \|\mathbf{Y}\|_F^2 \,. \tag{17}$$

Lemma 4 *Let $\mathbf{X}_i \in \mathbb{R}^{d_1 \times d_2}$ for $i \in 0, \ldots, K-1$. Then, the following bound holds:*

$$\| \sum_{i=0}^{K-1} \mathbf{X}_i \|_F^2 \leq K \sum_{j=0}^{K-1} \|\mathbf{X}_i\|_F^2 \,. \tag{18}$$

Proof. *This inequality follows directly from Jensen's inequality applied to the Frobenius norm.* □

Lemma 5 *Let $\mathbf{D}$ be a diagonal matrix with diagonal entries that are 1 with probability p and 0 with probability $1-p$, and let $\mathbf{A}$ be an arbitrary matrix. Then the expectation of the squared Frobenius norm of the product $\mathbf{DA}$ is given by*

$$\mathbb{E}_{\mathbf{D}} \left[\|\mathbf{DA}\|_F^2 \right] = p \|\mathbf{A}\|_F^2.$$

proof. *The Frobenius norm squared of $\mathbf{DA}$ is defined as:*

$$\|\mathbf{DA}\|_F^2 = \sum_{i,j} (\mathbf{DA})_{ij}^2.$$

Since $\mathbf{D}$ is diagonal, the product $\mathbf{DA}$ will zero out all rows of $\mathbf{A}$ where the corresponding diagonal entry in $\mathbf{D}$ is 0. Let d_i represent the i-th diagonal entry of $\mathbf{D}$, where each d_i is a Bernoulli random variable with $\mathbb{E}_{\mathbf{D}}[d_i] = p$.

Thus, we can express $\|\mathbf{DA}\|_F^2$ as:

$$\|\mathbf{DA}\|_F^2 = \sum_{i=1}^n d_i^2 \sum_{j=1}^m A_{ij}^2 = \sum_{i=1}^n d_i \|\mathbf{A}_{i,\cdot}\|_2^2,$$

where $\|\mathbf{A}_{i,\cdot}\|_2^2 = \sum_{j=1}^m A_{ij}^2$ is the squared norm of the i-th row of $\mathbf{A}$.

Now, taking the expectation, we have:

$$\mathbb{E}_{\mathbf{D}} \left[\|\mathbf{DA}\|_F^2 \right] = \sum_{i=1}^n \mathbb{E}_{\mathbf{D}}[d_i] \|\mathbf{A}_{i,\cdot}\|_2^2 = \sum_{i=1}^n p \|\mathbf{A}_{i,\cdot}\|_2^2.$$

Simplifying, we get:

$$\mathbb{E}_{\mathbf{D}} \left[\|\mathbf{DA}\|_F^2 \right] = p \sum_{i=1}^n \|\mathbf{A}_{i,\cdot}\|_2^2 = p \|\mathbf{A}\|_F^2.$$

Therefore, the expectation of $\|\mathbf{DA}\|_F^2$ is:

$$\mathbb{E}_{\mathbf{D}} \left[\|\mathbf{DA}\|_F^2 \right] = p \|\mathbf{A}\|_F^2.$$

This completes the proof. □

Lemma 6 *Let $\mathbf{D}$ be a diagonal $n \times n$ matrix where each diagonal entry is independently 1 with probability p and 0 with probability $1 - p$. Then the expected value of the spectral norm $\|\mathbf{D}\|_2$ is given by*

$$\mathbb{E}(\|\mathbf{D}\|_2) = 1 - (1 - p)^n \leq 1.$$

Proof. *Since $\mathbf{D}$ is diagonal, its spectral norm $\|\mathbf{D}\|_2$ is the largest absolute value among its diagonal entries. Therefore, $\|\mathbf{D}\|_2 = 1$ if at least one diagonal entry is 1, and $\|\mathbf{D}\|_2 = 0$ only if all diagonal entries are 0.*

Define X as the event that all diagonal entries are 0. The probability of this event, $\Pr(X)$, is:

$$\Pr(X) = (1 - p)^n,$$

since each diagonal entry is 0 independently with probability $1 - p$.

Thus, the probability that $\|\mathbf{D}\|_2 = 1$ (i.e., the event X does not occur) is:

$$1 - \Pr(X) = 1 - (1 - p)^n.$$

Therefore, the expected value of $\|\mathbf{D}\|_2$ is:

$$\mathbb{E}(\|\mathbf{D}\|_2) = 1 \cdot (1 - (1 - p)^n) + 0 \cdot (1 - p)^n = 1 - (1 - p)^n \leq 1.$$

This completes the proof. □

Lemma 7 *For any matrices $\mathbf{A} \in \mathbb{R}^{d_1 \times d_2}$ and $\mathbf{B} \in \mathbb{R}^{d_2 \times d_3}$, the Frobenius norm of the product $\mathbf{AB}$ satisfies*

$$\|\mathbf{AB}\|_F \leq \|\mathbf{A}\|_2 \|\mathbf{B}\|_F.$$

where $\|.\|_2$ is the spectral norm.

Appendix B. Related Work

Many pFL algorithms impose the closeness of the learning models; the main assumption is that personal models are close with respect to some measures [37]. Learning a mixture of the global model and the local models is proposed in [13], where these personalized models are encouraged to stay relatively close to their average by incorporating a quadratic penalty.

Although giving degrees of freedom to personal models may improve the generalization behavior of the models, it contradicts the following fact. It is shown that model similarities between different neural networks, especially classifier layers, are highly correlated with the similarity of the training data distributions [38], meaning that assuming the closeness of models is equivalent to assuming the similar data distributions over different clients. This leads us to ask the question: *How to train personalized models without assuming similarity between clients' data distributions?*

Recently, model decoupling methods have been proposed [3, 26, 27] showing a better performance than distance-based pFL methods. The main idea is to decouple each local model into two blocks: a feature extractor block followed by a classifier block. The feature extractor block is communicated and aggregated over clients and the classifier block is trained locally by each client. Arivazhagan et al. [3] introduced a personalization of some specific layers of the neural network that all user devices share a set of base layers with the same weights and have distinct personalization layers that can potentially adapt to individual data. The base layers are shared with the server while the personalization layers are kept private by each device. In [27], the entire network is decomposed into the body (extractor), which is related to universality, and the head (classifier), which is related to personalization. This reduces the update and aggregation parts from the entire model to the body of the model during federated training.

Anelli et al. [2] investigate federated pair-wise learning for factorization models in a recommendation scenario. Huang et al. [15] propose an FL framework for solving the POI (Point-of-Interest) recommendation problem. Ammad-Ud-Din et al. [1] introduces a federated implementation of collaborative filtering that is limited to recommendation systems. Liang et al. [23] introduce LG-FEDAVG combines local representation learning with global model learning in an end-to-end manner. Each local device learns to extract higher-level representations from raw data before a global model operates on the representations (rather than raw data) from all devices. Tan et al. [38] propose a decoupling algorithm that also personalizes feature extractors by adjusting aggregation weights based on classifier similarity. Deng et al. [10] introduce APFL algorithm which. aims to learn a personalized model for each user that is a convex combination of local and global models, and coefficients of these linear combinations are adaptively learned during the training. Hao et al. [14] assume factorized weights for neural networks and, instead of learning a unique global model, aims at learning a dictionary of rank-1 weight factor matrices. Each client then uses this dictionary to construct a model customized to its unique data distribution. Jeong and Hwang [18] consider factorization of the model parameters and allows clients to perform a selective aggregation scheme to utilize only the knowledge from the relevant participants for each client.

Perhaps the most relevant works to ours are [9, 24]. In [9], server tries to learn the common low-dimensional features of the data, and each client learns local features suited to its requirements. This method, Federated Representation Learning (FedRep), leverages all of the data stored across clients to learn a global low-dimensional representation using gradient-based updates. Further, it enables each client to compute a personalized, low-dimensional classifier that accounts for the unique labeling of each client's local data. The main difference between this method and $\mathtt{pFL}^{\mathtt{MF}}$ is that we consider that the concatenation of the parameters of the clients lie on a low dimensional space while this paper assumes that each client has a low-rank parameter. In other words, their feature extractor extracts the features from a single shared global model while $\texttt{pFL}^{\texttt{MF}}$ trains a set of models and each client uses a combination of these models as its personalized model. In another word, it is assumed that $\mathbf{x}_i$ are low rank not $\mathbf{X} := [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]$. Also, this work focuses on the linear representation setting with quadratic loss. Recently, an expectation-maximization is proposed in [24], viewing pFL as solving a mixture model.

Another line of research is partitioning the variables, Mishchenko et al. [26], Pillutla et al. [31] partition the model parameters into two groups: the shared parameters and the personal parameters. Clients do simultaneous or alternating updates and only share shared parameters.

It is worth mentioning that our work is fundamentally different from the following set of works. (1) Low-rank structure for the network assumption, such as [32], proposed algorithm projects the private dataset onto a low-dimensional space spanned by the top principal components estimated with the public unlabeled dataset and then applies gradient-based private algorithms (e.g., Noisy-SGD) to learn a linear classifier on top of the projected features, or [6, 42], one of the layers in the neural net is assumed to be low rank. And (2) Low-rank structure for the gradient assumption such as [12, 19, 41]. Yao et al. [40] introduced FEDHM, that low-rank factorized neural networks with a specified size are trained, and the server translates this to the full rank global model using model shape alignment method.

Appendix C. Additional Details on Numerical Experiments

Details on the four problem setups. We consider the following experimental setups, including standard settings that mimic the heterogeneity of the system, as well as more realistic scenarios where neither FEDAVG nor local training produces the best accuracies.

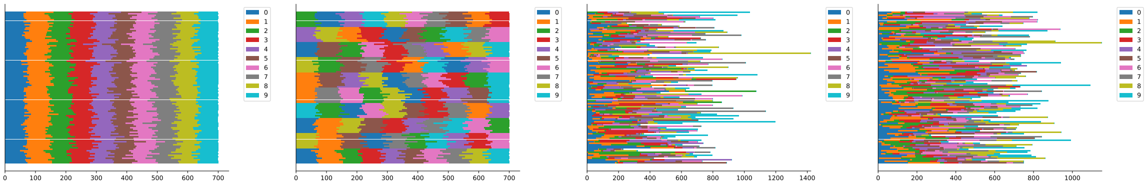
Setup (1) For the MNIST, CIFAR10, and CIFAR100 datasets, we split the data according to the Dirichlet distribution $\mathrm{Dir}(0.5)$ and $\mathrm{Dir}(1)$ across 100 clients. The labels' distribution is shown in Figures 2(*c*) and 2(*d*). Performance of the algorithms for 2000 global iterations, one local epoch for all algorithms, learning rate equal to 10^{-4} is shown in Table 1.

Setup (2) For the CIFAR100, we partitioned the data based on labels into 20 groups with distinct labels, and then the data in each group was distributed according to uniform distribution across 500 clients. We ran the experiments for 2000 global iterations with a fixed step size equal to 10^{-4}. Results are shown in Table 2.

Setup (3) For the MNIST, we follow the experimental setup in [34] and consider 1000 clients divided into 10 groups, and labels in each group are re-mapped (permuted) according to a random permutation map. In other words, clients in group one would have the same numbers $\{0, \cdots, 9\}$ but labeled differently; group one may consider 0 with label 0, and group two may consider 0 with label 8. Figures 2(*a*) and 2(*b*) show the distribution of the labels before and after re-labeling, respectively. We ran the experiments for 4000 global iterations with a fixed step size equal to 10^{-4}. Results are shown in Table 2.

Setup (4) We sampled a subset of clients, 30% of the total clients, from FEMNIST dataset without changing the underlying data distribution, then we removed clients with less than 10 data points. The remaining set has 1091 clients. We ran the experiments for 1 and 5 numbers of local epochs. We ran the experiments for 2000 global iterations with a fixed step size equal to 0.01. Results are shown in Table 2.

Hyper-parameters. We consider partial participation with probability equal to 0.1. We set the batch size equal to 256 for all algorithms. The rank in the problem (P) is set to $r \in \{1, 5, 10, 15\}$, meaning that we consider personalized parameters, concatenated weights of neural networks, to belong to a subspace with rank $r \in \{1, 5, 10, 15\}$. All experiments have 75% train and 25% test data splits on each client's data. We chose the best step size for each algorithm from the set $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}\}$.



(a) **Uniform** (b) **Permuted** (c) $\mathbf{Dir(0.5)}$ (d) $\mathbf{Dir(1)}$

Figure 2: Distribution of the labels for MNIST dataset across 100 clients. The vertical and horizontal axes show clients and the size of each client's data, respectively.