FedMix: A Simple and Communication-Efficient Alternative to Local Methods in Federated Learning

Elnur Gasanov
KAUST, Thuwal
Saudi Arabia

Ahmed Khaled
Cairo University
Egypt

Samuel Horváth
KAUST, Thuwal
Saudi Arabia

Peter Richtárik
KAUST, Thuwal
Saudi Arabia

Abstract

Federated Learning (FL) is an increasingly popular machine learning paradigm in which multiple nodes try to collaboratively learn under privacy, communication and multiple heterogeneity constraints. A persistent problem in federated learning is that it is not clear what the optimization objective should be: the standard average risk minimization of supervised learning is inadequate in handling several major constraints specific to federated learning, such as communication adaptivity and personalization control. We identify several key desiderata in frameworks for federated learning and introduce a new framework, FedMix, that takes into account the unique challenges brought by federated learning. FedMix has a standard finite-sum form, which enables practitioners to tap into the immense wealth of existing (potentially non-local) methods for distributed optimization. Through a smart initialization that does not require any communication, FedMix does not require the use of local steps but is still provably capable of performing dissimilarity regularization on par with local methods. We give several algorithms for solving the FedMix formulation efficiently under communication constraints. Finally, we corroborate our theoretical results with extensive experimentation.

1 Introduction

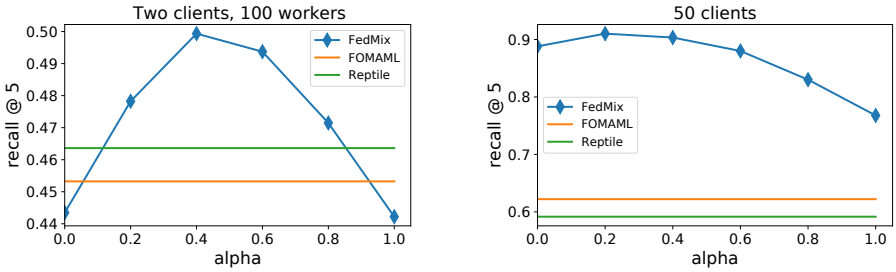
Federated Learning (FL) aims to enable machine learning in the decentralized setting while respecting data privacy. Application domains of federated learning include healthcare, learning language models for virtual keyboards, and speech recognition (Kairouz et al., 2019). The promise of federated learning is that by participating in a distributed training process, clients can learn better machine learning models than they can using only their own data. The main cost in using federated learning over local training lies in the network bandwidth used for the distributed training process. Hence, federated learning must be flexible enough to provide a benefit to users without a prohibitive communication cost. The standard formulation of FL is to cast it as an optimization problem of the form

$$\min_{x\in\mathbb{R}^d}\left[f(x) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} f_i(x)\right], \tag{ERM}$$

where f_i is the loss function on client i. Thus, the goal of classical FL is for the n clients to collaboratively learn a single model, $x^* = \arg\min f$, to be deployed on all clients. Recent development shows that using a single model for all clients can be severely detrimental to individual performance on many clients (Yu et al., 2020), defeating the purpose of joining distributed training. Furthermore, (ERM) offers no clear tunable knobs that can accommodate constraints on the network bandwidth.

* The framework developed in this paper will be renamed from FedMix to FLIX in subsequent versions to distinguish this work from concurrent publications.

5th Workshop on Meta-Learning at NeurIPS 2021, held virtually.



(a) 100 workers created out of two clients' data (b) 50 workers with distinct data distributions

Figure 1: Test accuracy of FedMix model for different personalization parameter values, FOMAML and Reptile. $\alpha_i = \alpha$ is set to the value indicated on horizontal axis. FOMAML and Reptile are independent from the personalization parameter α. Plots correspond to different data splittings.

Can we find a formulation for federated learning that is flexible enough to accommodate the needs of federated learning, yet also solvable using standard methods?

1.1 Key properties of the FedMixture framework

Our main contribution is *FedMixture*, a novel and flexible formulation for federated learning: define $\alpha_i > 0$ to be the *personalization parameter for node* i, and let $x_i \overset{\text{def}}{=} \min_{x \in \mathbb{R}^d} f_i(x)$ be the local solution to the i-th objective– note that x_i can be found by solely running a local optimizer, and hence computing it requires no communication at all. The FedMixture problem is

$$\min_{x \in \mathbb{R}^d} \tilde{f}(x) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(\alpha_i x + (1 - \alpha_i)\, x_i). \tag{FedMix}$$

Once we find a solution x_* of sufficient quality for (FedMix), we deploy $T_i(x_*) = \alpha_i x_* + (1 - \alpha_i)\, x_i$ on node i as its final personalized model. We now enumerate some of the key properties of (FedMix):

- **Efficiently solvable as a finite-sum problem.** (FedMix) preserves the standard finite-sum formulation of empirical risk minimization. Moreover, it preserves problem structure: we show (in Section 3) that when the f_i are smooth (and/or convex), $\tilde{f}$ is also smooth (resp. convex).
- **Adaptive to communication constraints.** Communication efficiency is an important concern in federated learning, as often bandwidth is valuable and limited (Kairouz et al., 2019; Konečný et al., 2016; Li et al., 2019). Computing x_i, a precondition to solving (FedMix), requires no communication at all, and can be done purely locally on node i. If $\alpha_i = 0$, then no communication at all is needed to compute the personalized model $T_i(x_*)$. By varying α_i between 0 and 1, we can control the amount of communication needed to compute $T_i(x_*)$. We show (in Section 3) that given a communication budget of R steps, we can find parameters α_i that allows us to solve (FedMix) in no more than R communication steps.
- **Adaptivity to personalization.** Our end-goal in federated learning is to generalize well *on each client*: this means that the solution deployed on node i should be tailored to its local data distribution, which may differ from the data distributions on other nodes. In FedMix, varying α_i enables us to amplify or reduce the effect of other objectives on the solution deployed on node i. In situations where the data on all of the nodes is sufficiently heterogeneous, we set α_i to be small and the effect of other data on node i will be neglibile. On the other hand, when the data on the different nodes is related we may set α_i to be closer to 1. We observe a benefit to varying α in this manner in practice: Figure 1 shows the effect of varying the α_i on real data (see Section 4 for the details and for other experiments).

FedMix fills a gap that is unsatisfied by existing methods. To the best of our knowledge, there is no other method for federated learning that is efficiently solvable via standard algorithms and also adaptive to communication and personalization constraints, and indeed both constraints are important in practice (Li et al., 2020). We believe the key properties we enumerate can also serve as natural desiderata in the development of new formulations and methods for federated learning.

1.2 Related work

Personalization has garnered significant recent interest in federated learning as personalized models often perform well in practice compared to non-personalized models (Jiang et al., 2019; Yu et al., 2020). FedMixture is a *model mixture* method: the personalized solution is a mixture of a global model and a local model. In recent work, Deng et al. (2020) and Mansour et al. (2020) propose model mixture methods and prove their statistical benefits, while Zec et al. (2021) introduce a similar formulation based on the mixture of experts framework. Unfortunately, we show in the supplementary material that, from the perspective of optimization and without additional data, the formulations in all three works are trivially minimized at the local minimizers $x_1, \ldots, x_n$. An alternative to model mixing is mixing in function space, where we optimize a mixture of objectives rather than a model mixture. This mixture is often constructed to control model variance: examples of this approach can be found in (Dinh et al., 2021a;b; Hanzely and Richtárik, 2020; Huang et al., 2021). In FedMix, we take the model mixture approach as it allows us to use pretraining to better solve the problem while still regularizing model variance (see Section 2.1). A parallel line of work applies meta-learning methods like MAML to federated learning (Fallah et al., 2020a; Jiang et al., 2019): in Section 2.2 we motivate FedMix by taking MAML as our starting point. Chen et al. (2021) discuss the statistical limits of personalization and show that either solving empirical risk minimization or local training is optimal, depending on certain problem parameters; However, as of yet there is no single optimal adaptive algorithm (from the statistical perspective). There are several other techniques in federated learning that can be combined with our approach for better results, such as clustering (Sattler et al., 2020) or robust optimization (Reisizadeh et al., 2020).

2 The FedMix formulation

In this section we reintroduce and motivate the FedMix formulation in detail. We define the FedMix objective as

$$\tilde{f}(x; \alpha_1, \ldots, \alpha_n, x_1, \ldots, x_n) \overset{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(\alpha_i x + (1 - \alpha_i)\, x_i), \tag{1}$$

where $\alpha_i \in (0, 1)$ is the personalization coefficient for node i and x_i is the minimizer of f_i, for all $i = 1, 2, \ldots, n$. We will use $\tilde{f}(x)$ to refer to the objective in (1) when the α_i and x_i are clear from the context. The FedMixture problem is then

$$\min_{x \in \mathbb{R}^d} \left[\tilde{f}(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(\alpha_i x + (1 - \alpha_i)\, x_i) \right]. \tag{2}$$

Let $\alpha = [\alpha_1, \ldots, \alpha_n]$ be the vector of the personalization coefficients. If $x_* = x_*(\alpha)$ is a solution of (2), we call $T_i(x; \alpha_i, x_i) = \alpha_i x_* + (1 - \alpha_i)\, x_i$ the *deployed solution on node* i. Like with $\tilde{f}$, we will refer to the deployed solution on node i as $T_i(x)$ when x_i and α_i are clear from the context.

2.1 Motivation 1: from Local GD to FedMix

The most popular algorithm for solving federated learning problems is the Federated Averaging algorithm (Kairouz et al., 2019), also known as Local (Stochastic) Gradient Descent (Local GD/SGD). Local GD alternates steps of local computation on each node with steps of communication and aggregation. More concretely, the Local GD update is:

$$x_{t+1}^i = \begin{cases} x_t^i - \gamma \nabla f_i(x_t^i) & \text{if } t \mod H \neq 0 \\ \frac{1}{n} \sum_{i=1}^{n} \left[x_t^i - \gamma \nabla f_i(x_t^i) \right] & \text{if } t \mod H = 0 \end{cases}, \tag{3}$$

where H is the number of local steps. Early papers on federated learning (such as e.g. (Konečný et al., 2016)) motivated local methods as communication-efficient ways of solving (ERM), but subsequent theoretical development reveals that local methods are, in fact, quite bad solvers for (ERM) whenever there is significant statistical heterogeneity among the clients (Woodworth et al., 2020). Moreover, Pathak and Wainwright (2020) show that for the linear least-squares problem, Local GD converges to a different point than the minimizer of (ERM). More generally, the fixed points of Algorithm (3) can be very different from the minimizer of (ERM) whenever $H > 1$ (Malinovskiy et al., 2020). Hanzely

and Richtárik (2020) show that a mild variant of Local GD can be interpreted as SGD applied on the nd-dimensional regularized objective f_λ defined by

$$f_\lambda(y_1, y_2, \ldots, y_n) \stackrel{\text{def}}{=} \left[\frac{1}{n}\sum_{i=1}^{n} f_i(y_i) + \frac{\lambda}{2n}\sum_{i=1}^{n}\|y_i - \bar{y}\|^2\right], \tag{4}$$

where $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$ is the counterpart of x in (ERM), and where λ is a regularization parameter determined according to the number of local steps. Objective f_λ is the summation of two terms: the first asks that each node i finds a solution y_i that minimizes its local objective well, while the regularizer $\psi(y_1, \ldots, y_n) = \frac{1}{2n}\sum_{i=1}^{n}\|y_i - \bar{y}\|^2$ forces the solutions $y_1, y_2, \ldots, y_n$ to be close to their average $\bar{y}$. Hence, Local GD incentivizes finding *personalized* solutions $y_1, y_2, \ldots, y_n$ that have small population variance. Hanzely and Richtárik (2020) note that as the λ parameter varies between 0 and ∞, the solutions found by Local GD interpolate between the pure local optimal models (i.e. $x_i = \operatorname{argmin}_x f_i(x)$) and the solution of the global problem x_* (the minimizer of (ERM)). We observe that the solutions $y_1, \ldots, y_n$ found by Local GD are an *implicit mixture* of the local minimizers $x_1, \ldots, x_n$ and the global empirical risk minimizer x_*. Rather than seeking an implicit mixture of the local and global optimal models, we instead propose to find an *explicit* mixture of the local optimal models and a global model: given any global model x (not necessarily the empirical risk minimizer), we choose coefficients $\alpha_1, \alpha_2, \ldots, \alpha_n$ (all between 0 and 1) and then deploy on node i the mixture

$$T_i(x) = \alpha_i x + (1 - \alpha_i)\, x_i, \tag{5}$$

we may then choose x as *the best* such global model by explicitly solving the optimization problem $\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f_i\left(\alpha_i x + (1-\alpha_i)\, x_i\right)$, and this is exactly the FedMix formulation. Observe that coefficients $\alpha_1, \alpha_2, \ldots, \alpha_n$ should regularize the population variance of the deployed solutions $T_1(x), T_2(x), \ldots, T_n(x)$, as in local methods. We show this rigorously for equal α_i in Proposition 3. Our development thus leads us to a natural framework that captures the strength of local methods while also satisfying the desiderata specified in Section 1.1.

2.2 Motivation 2: from model-agnostic meta-learning to FedMix

We now motivate FedMix differently by starting with *personalization via fine-tuning*. The ordinary formulation of the federated learning problem (ERM) asks for a single global model to be used on all clients. If the clients are sufficiently heterogeneous, a single model may perform badly on many of them (Jiang et al., 2019). Personalizing a global model to each of the users' custom data is often beneficial in practice; For example, Wang et al. (2019) study the benefits of personalizing language models for a virtual keyboard application used by tens of millions of users. They observe that a sizeable fraction of the users benefit from personalization. Personalization is often done in two steps:

Step I: initial model training. Find a "good" global model x_{global}.
Step II: fine-tuning. Personalize the global model x_{global} on each client to get the *personalized* local models x_i.

Methods that fit this framework are known as *finite-tuning approaches*: they include the model-agnostic meta-learning (MAML) family of methods (Finn et al., 2017). In addition to its practical popularity, recent theoretical investigations reveal that fine-tuning approaches, such as MAML, are also benefical from a statistical perspective (Chua et al., 2021; Fallah et al., 2021). In MAML, we find x_{global} by optimizing for the loss after a single step of gradient descent, i.e. the MAML objective is

$$\text{Find} \quad x_{\text{global}} \in \operatorname*{argmin}_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f_i(x - \gamma\nabla f_i(x)), \tag{6}$$

where γ is a given stepsize. Once x_{global} is found, we may then fine-tune it by running gradient descent for a number of steps on each node i locally using its own objective f_i (Finn et al., 2017). To gain further insight into what fine-tuning is doing, we now consider the case when each f_i is a quadratic function. Because this problem is amenable to analysis, several authors have used it to study the theoretical properties of MAML (Charles and Konečný, 2021; Collins et al., 2020; Gao and Sener, 2020), and we follow in their footsteps. Assume that each f_i is a quadratic function, suppose that we have some initial global model x^0, and we fine-tune it by running gradient descent for H steps on node i: the next proposition shows the final iterate is a matrix-weighted average of the initial solution and the optimal local solution:

Proposition 1. *Suppose that we run gradient descent for H steps on the quadratic objective $f_i = \frac{1}{2}x^T A_i x - b_i^T x + c$ starting from x^0 with stepsize $\gamma > 0$. Suppose that the stepsize satisfies $\gamma \leq \frac{1}{L_i}$, where $L_i = \lambda_{\max}(A_i)$. Then the final iterate x_i^H can be written as*

$$x_i^H = \left(I - J_i^H\right) x_i + J_i^H x^0,$$

where x_i minimizes f_i and $J_i \in \mathbb{R}^{d\times d}$ is a matrix with maximum eigenvalue smaller than 1, i.e. $\lambda_{\max}(J) < 1$.

The proof of Proposition 1 and all subsequent proofs are relegated to the supplementary. Plugging the result of Proposition 1 into Equation (6), observe that in MAML we find the initial model x^0 by solving the problem $\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f_i((I - J_i)x_i + J_i x)$. Hence, MAML is optimizing *for a specific weighted average* of the initial model x^0 and the local solutions $x_1, x_2, \ldots, x_n$. We thus propose to dispense with the specific matrix J_i and instead optimize an average weighted with an arbitrary constant α_i: $\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n} f_i(\alpha_i x + (1-\alpha_i)\, x_i)$, and this is exactly the FedMix formulation. Observe that by properly normalizing or whitening the data and tuning α_i we may accomplish a similar effect to multiplying by J_i^H for any H. This gives FedMix a new interpretation as an approximate generalized MAML, where we optimize the global model for performance after potentially many gradient descent steps rather than just a single step.

3 Theory and algorithms

In this section we aim to develop algorithms to solve (FedMix) in a communication-efficient manner. Before discussing concrete algorithms, we study a few algorithm-independent properties of (FedMix) that will come in handy for understanding the formulation and proving convergence bounds. The following proposition shows that the formulation preserves smoothness and convexity. This is in contrast, for example, to MAML, where the objective may be nonsmooth (Fallah et al., 2020b).

Proposition 2. *Suppose that each objective f_i is L_i-smooth. That is, for any $x, y \in \mathbb{R}^d$ we have $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_i \|x - y\|$. Then the FedMix objective $\tilde{f}$ defined in (1) is L_α-smooth for $L_\alpha \overset{def}{=} \frac{1}{n}\sum_{i=1}^{n} \alpha_i^2 L_i$. If each f_i is convex, then $\tilde{f}$ is also convex. If each f_i is μ_i-strongly convex, then $\tilde{f}$ is μ_α strongly convex for $\mu_\alpha \overset{def}{=} \frac{1}{n}\sum_{i=1}^{n} \alpha_i^2 \mu_i$.*

Our next result offers some insight into the variance-regularizing effect of the α_i: in particular, when all the α_i are equal, increasing α in (FedMix) directly decreases the variance of the deployed local models from their mean. As discussed in Section 2.1, this is a key property of local descent methods that the FedMix formulation captures.

Proposition 3. *Suppose that $\alpha_1 = \alpha_2 = \ldots = \alpha_n = \beta$ in the FedMix formulation (FedMix). Let $T_1(x), T_2(x), \ldots, T_n(x)$ be the deployed models defined in (5). If $y_1, \ldots, y_n$ are vectors in $\mathbb{R}^d$ and $\bar{y}$ is their mean, we define $V(y_1, \ldots, y_n)$ as the population variance $V(y_1, \ldots, y_n) \overset{def}{=} \frac{1}{n}\sum_{i=1}^{n} \|y_i - \bar{y}\|^2$. Then, $V(T_1(x), T_2(x), \ldots, T_n(x)) = (1-\beta)^2\, V(x_1, x_2, \ldots, x_n)$.*

One-shot learning is a learning paradigm where we may use only a single round of communication to solve the federated learning problem (Guha et al., 2019; Salehkaleybar et al., 2019). When the personalization parameters are small enough, we can provably solve the FedMix problem with a single round of communication by computing a certain weighted average of the local solutions $x_1, x_2, \ldots, x_n$.

Theorem 1. *Suppose that each objective f_i is L_i-smooth, let $\hat{L} \overset{def}{=} \frac{1}{n}\sum_{i=1}^{n} L_i$. Given the pure local models $x_1, x_2, \ldots, x_n$, define the weighted average $x^{\text{avg}} \overset{def}{=} \sum_{i=1}^{n} w_i x_i$, $w_i \overset{def}{=} \frac{\alpha_i^2 L_i}{n L_\alpha}$, $L_\alpha \overset{def}{=} \frac{1}{n}\sum_{i=1}^{n} \alpha_i^2 L_i$. We further define the constants $D \overset{def}{=} \max_{i,j=1,\ldots,n, i\neq j} \|x_i - x_j\|$, and, $V \overset{def}{=} \sum_{i=1}^{n} w_i \|x_i - x^{\text{avg}}\|^2$. Fix any $\epsilon > 0$. Assume that either $\max_{i=1,\ldots,n} \alpha_i \leq \sqrt{2\epsilon}/\sqrt{\hat{L}D}$, or $\alpha_i = \beta$ for all i and $\beta \leq \sqrt{2\epsilon}/\sqrt{\hat{L}D}$. Then x^{avg} is an ϵ-approximate minimizer of (FedMix).*

For α_i larger than this, we need more communication rounds. In the next subsection, we describe how distributed gradient descent can be used to solve the problem.

3.1 Distributed gradient descent

The simplest approach to solving (FedMix) is via distributed gradient descent (DGD): given the local models $x_1, x_2, \ldots, x_n$ (precomputed before starting the process) and an initial global model x^0, we run the update $x^{k+1} = x^k - \frac{\gamma}{n} \sum_{i=1}^{n} \alpha_i \nabla f_i \left(\alpha_i x^k + (1-\alpha_i) x_i\right)$. The next theorem shows that under smoothness and strong convexity, DGD converges linearly to the (FedMix) solution.

Theorem 2. *Suppose that each f_i in (FedMix) is L_i-smooth and μ_i-strongly convex. Define x^{avg}, L_α, $\hat{L}$, V and D as in Theorem 1. Suppose that we run DGD for K iterations starting from $x^0 = x^{\text{avg}}$. Then the following hold:*

i) If the α_i are allowed to be arbitrary, then for $\alpha_{\max} \overset{def}{=} \max_{i=1,\ldots,n} \alpha_i$ we have $\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \left(1 - \frac{\mu_\alpha}{L_\alpha}\right)^K \frac{\alpha_{\max}^2 \hat{L} D}{2}$.

ii) Let $\hat{\mu} \overset{def}{=} \frac{1}{n} \sum_{i=1}^{n} \mu_i$. If $\alpha_i = \beta$ for all i, then

$$\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \left(1 - \frac{\hat{\mu}}{\hat{L}}\right)^K \frac{\beta^2 \hat{L} V}{2}. \tag{7}$$

There are four ways of making the right hand side in (7) (the communication complexity) small:

- **Communicate more.** Increase the number of communications K.
- **Homogeneous data**. The variance $V = \sum_{i=1}^{n} w_i \|x_i - x^{\text{avg}}\|^2$ can be seen as a measure of data heterogeneity. More homogeneous data means smaller V, which leads to better performance.
- **Train simpler models.** Focusing attention on models with smaller $\hat{L}$ (adjust model design), or larger $\hat{\mu}$ (e.g., add more regularization).
- **Put more weight on local models.** If we prefer local models to the global model, then α_i is small, and hence fewer communications are needed to achieve any given accuracy.

Armed with Theorem 2, we make good on our promise in Section 1.1 and show that FedMix can be solved using *any* communication budget. Looking at (7) we see that for any fixed $\epsilon > 0$ we have $\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \epsilon$ as long as $\beta \leq Aq^k$, where $A = \sqrt{2\epsilon/(\hat{L}V)}$ and $q = 1/\sqrt{1 - \frac{\hat{\mu}}{\hat{L}}}$.

Putting this together leads to the following observations:

- If $\beta = 0$, the problem can be solved with 0 communications (i.e. , each device i independently computes the pure local model x_i).
- If $0 < \beta \leq A$, the problem can be solved with 1 communication (i.e., compute x^{avg}). This follows from Theorem 1, and also from the more general result Theorem 2 by setting $K = 0$.
- If $A < \beta \leq Aq$, the problem can be solved with 2 communications (1 communication to compute $x^0 = x^{avg}$, followed by one iteration of distributed GD).
- If $Aq^{k-1} < \beta \leq Aq^k$, the problem can be solved with $k+1$ communications (1 communication to compute $x^0 = x^{avg}$, followed by K iterations of distributed gradient descent).
- If $\beta = 1$, we need 1 communication to compute $x^0 = x^{avg}$, followed by $k \geq \frac{\bar{L}}{\bar{\mu}} \log \frac{\bar{L}V}{2\varepsilon}$ iterations of distributed gradient descent. This is recovers the standard communication complexity of gradient descent needed to find the optimal solution of the average risk minimization problem (ERM).

In the supplementary, we develop other algorithms for solving (FedMix) such as distributed gradient descent with compression (Alistarh et al., 2017) and DIANA (Mishchenko et al., 2019). We note that because (FedMix) has a standard finite-sum form, many more algorithms can be used to solve it, e.g. accelerated minibatch SGD (Cotter et al., 2011) or SARAH (Nguyen et al., 2017).

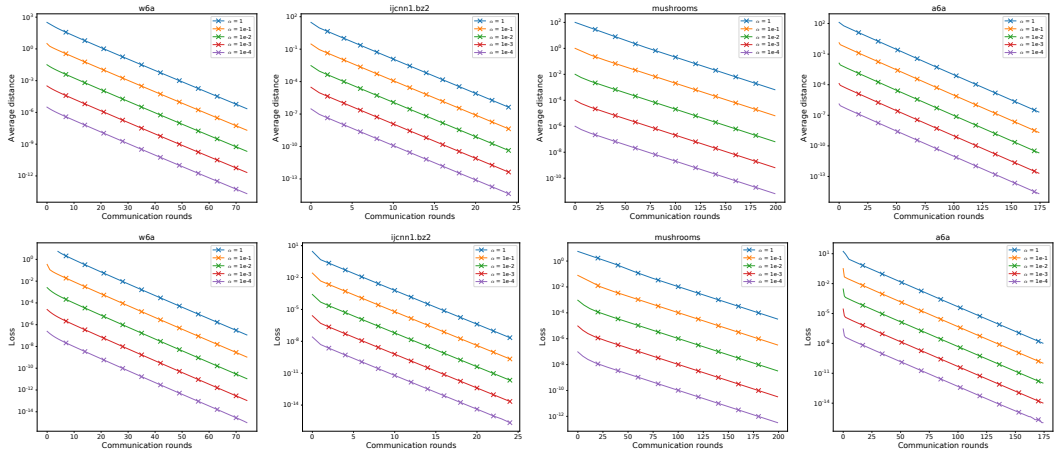


Figure 2: Squared averaged distance $1/n \sum_{i=1}^{n} \|x_i - x_i^*\|^2$ and loss $f(x) - f^*$ vs. # of communication rounds of Gradient Descent for logistic regression with l_2 regularizer. $\alpha_i = \alpha$ is set to the value indicated in the legend.

4 Experiments

Logistic regression with l_2 regularizer. For our first experiment, we consider a setup where each device runs regularized logistic regression: $f_i(x) := \frac{1}{k_i} \sum_{j=1}^{k_i} \left[\log\left(1 + \exp\left(-a_{i,j}^\top x\right)\right)\right] + \lambda/2\|x\|^2$, where $a_{i,j} \in \mathbb{R}^d$ are given for all $j = 1 \dots k_i$, k_i is a number of data points associated with device i and λ is a regularization parameter. The objective is smooth and strongly convex.

We use four datasets from LIBSVM (Chang and Lin, 2011) for this task: `w6a`, `mushrooms`, `ijcnn1.bz2`, `a6a`. We set all α_i's to be equal. We divide data equally between all machines while preserving the order of data points such that i-th machine owns data with indices $\lfloor (i-1)r/n \rfloor + 1$ up to $\lfloor ir/n \rfloor$, where r is the total number of data points. We set the regularization parameter λ to 0.1. To find pure local models for each machine we run gradient descent with step size $1/L_i$, where L_i is f_i's smoothness parameter until the norm of the gradient is below 10^{-6}. We set the condition number to be $\kappa = \frac{1}{n} \sum_{i=1}^{n} L_i / \lambda$.

In this experiment, we investigate the convergence of gradient descent and look into the dependence between convergence and value of α. As expected, Figure 2 confirms that smaller values of α lead to better convergence as we rely more on local solutions and thus start closer to the optimal solution. Also note that the speed of the convergence appears to be constant among different values of α which is also predicted by our theory as the same α on each machine does not affect the conditioning of the global problem, see Proposition 2. We also use the DIANA algorithm (Mishchenko et al., 2019) with the random sparsification (also known as Rand-k compression), where set k coordinates to zero at random before communicating gradients to the server. Figure 3 shows the effect of varying k on the convergence of the method in terms of communication rounds. We consider 7 values of k linearly spaced between 1 and d. Similar to Figure 2, we observe that smaller values of α lead to better convergence. The rate at which the algorithm converges linearly is controlled by the compression constant $\omega + 1 = d/k$ or the effective conditioning $\kappa(\omega+1)/n$. This is in line with the theory for DIANA (Mishchenko et al., 2019). For completeness, we include convex experiments with unregularized logistic loss, extra experiments for all the combinations of 5 datasets and 3 algorithms – gradient descent, compressed gradient descent, and DIANA, as well as a detailed experimental description in the supplementary material.

Generalization experiment 1: Fitting Sine Functions. Following Finn et al. (2017) and Zhou et al. (2019), we show the generalization advantages of FedMix on the following regression problem. We define i-th client's function $f_i(x) = a_i \sin(x + b_i)$, where amplitude a_i and phase b_i lie in the intervals $[0.1, 0.5]$ and $[0, 2\pi]$, respectively. For each client, we fix a_i and b_i and sample 50 points uniformly at random from the interval $[-5.0, 5.0]$. We measure regression fit in terms of mean squared error (MSE) loss. To train a model each client adopts a neural net with 2 hidden layers of size 40 with tanh activation. Further technical details are deferred to the appendix. For the experiment,

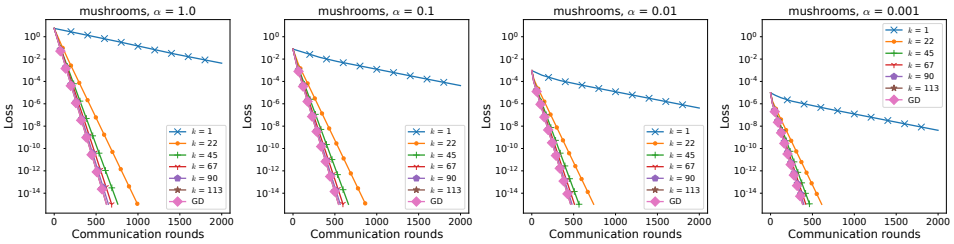


Figure 3: Loss $f(x) - f^*$ vs. # of communication rounds of DIANA for logistic regression problem l_2 regularizer, k is a sparsification parameter of Random-k compressor.

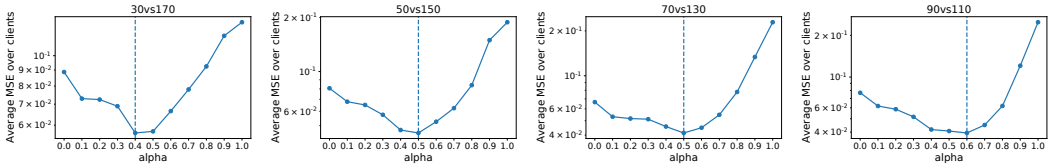


Figure 4: Average MSE vs. personalization parameter α.

we first sample 2 pairs $\{a_i, b_i\}$ and each of 200 clients is assigned one pair, we investigate different proportion– $(30, 170), (50, 150), (70, 130), (90, 110)$. We then train our FedMix formulation with $\alpha_i = \alpha = 0.1, 0.2, \ldots, 1$. For testing for each client generates a new dataset of size 2000. Figure 4 shows average MSE over clients against different values of α for different proportions. As this figure indicates, optimal α for which test average MSE is minimal can dramatically outperform the edge cases of either global model for all tasks or personalized model trained only on the local dataset.

Generalization experiment 2: Comparison to FOMAML and Reptile. Inspired by Reddi et al. (2021), we conduct a similar experiment to compare generalization capabilities, i.e., test accuracy, of FedMix and its two baselines FOMAML (Finn et al., 2017), and Reptile (Nichol et al., 2018). For the first experiment (see Figure 1a), we take 500 train data points of two clients (with client ids '00000267' and '00000459') from the Stack Overflow dataset (TensorFlow Developers, 2021) and divide them among 100 workers so that there are 50 workers with 10 train data points from the first client and another 50 with 10 data points from the second client. For the second experiment (see Figure 1b), a worker gets 90 train data points from a distinct client. For both experiments, each objective component f_i is a cross-entropy loss for multi-class logistic regression. Further technical details and the hyperparameters tuning for a fair comparison can be found in the supplementary. In the test phase, for each client, we used a hold-out testing dataset of size 300 (the same dataset has been used for workers related to the same client in the first experiment). It can be observed from Figure 1a, that for wide range of α_i, $\alpha_i \in \{0.2, 0.4, 0.6, 0.8\}$ FedMix exhibits a better generalization than its classical meta-learning competitors–FOMAML and Reptile, and it can lead to improvement of up to 11% in recall@5. Figure 1b shows that in the more real-world scenario FedMix outperforms FOMAML and Reptile while showing its best test accuracy in non-edge α.

References

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: communication-efficient SGD via gradient quantization and encoding. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 1709–1720, 2017. (Cited on page 6)

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011. (Cited on page 7)

Zachary Charles and Jakub Konečný. Convergence and accuracy trade-offs in federated learning and meta-learning. In Arindam Banerjee and Kenji Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 2575–2583. PMLR, 13–15 Apr 2021. (Cited on page 4)

Shuxiao Chen, Qinqing Zheng, Qi Long, and Weijie J. Su. A theorem of the alternative for personalized federated learning. *arXiv preprint arXiv:2103.01901*, 2021. (Cited on page 3)

Kurtland Chua, Qi Lei, and Jason D. Lee. How fine-tuning allows for effective meta-learning. *arXiv preprint arXiv:2105.02221*, 2021. (Cited on page 4)

Liam Collins, Aryan Mokhtari, and Sanjay Shakkottai. Why does MAML outperform ERM? an optimization perspective. *arXiv preprint arXiv:2010.14672*, 2020. (Cited on page 4)

Andrew Cotter, Ohad Shamir, Nati Srebro, and Karthik Sridharan. Better mini-batch algorithms via accelerated gradient methods. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pages 1647–1655, 2011. (Cited on page 6)

Yuyang Deng, Mohammad Mahdi Kamani, and Mehrdad Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020. (Cited on pages 3 and 36)

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019. (Cited on page 37)

Canh T. Dinh, Nguyen H. Tran, and Tuan Dung Nguyen. Personalized federated learning with moreau envelopes. *arXiv preprint arXiv:2006.08848*, 2021a. (Cited on page 3)

Canh T. Dinh, Tung T. Vu, Nguyen H. Tran, Minh N. Dao, and Hongyu Zhang. FedU: A unified framework for federated multi-task learning with laplacian regularization. *arXiv preprint arXiv:2102.07148*, 2021b. (Cited on page 3)

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Personalized federated learning: A meta-learning approach. *arXiv preprint arXiv:2002.07948*, 2020a. (Cited on page 3)

Alireza Fallah, Aryan Mokhtari, and Asuman E. Ozdaglar. On the convergence theory of gradient-based model-agnostic meta-learning algorithms. In Silvia Chiappa and Roberto Calandra, editors, *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 1082–1092. PMLR, 2020b. (Cited on page 5)

Alireza Fallah, Aryan Mokhtari, and Asuman Ozdaglar. Generalization of model-agnostic meta-learning algorithms: Recurring and unseen tasks. *arXiv preprint arXiv:2102.03832*, 2021. (Cited on page 4)

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1126–1135. PMLR, 2017. (Cited on pages 4, 7, and 8)

Katelyn Gao and Ozan Sener. Modeling and optimization trade-off in meta-learning. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. (Cited on page 4)

Eduard Gorbunov, Filip Hanzely, and Peter Richtárik. A unified theory of sgd: Variance reduction, sampling, quantization and coordinate descent, 2019. (Cited on pages 21, 22, 23, and 26)

Neel Guha, Ameet Talwalkar, and Virginia Smith. One-shot federated learning. *arXiv preprint arXiv:1902.11175*, 2019. (Cited on page 5)

Filip Hanzely and Peter Richtárik. Federated learning of a mixture of global and local models. *arXiv:2002.05516*, 2020. (Cited on pages 3 and 4)

Yutao Huang, Lingyang Chu, Zirui Zhou, Lanjun Wang, Jiangchuan Liu, Jian Pei, and Yong Zhang. Personalized cross-silo federated learning on non-iid data. *arXiv preprint arXiv:2007.03797*, 2021. (Cited on page 3)

Yihan Jiang, Jakub Konečný, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *arXiv preprint arXiv:1909.12488*, 2019. (Cited on pages 3 and 4)

Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D'Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javidi, Gauri Joshi, Mikhail Khodak, Jakub Konečný, Aleksandra Korolova, Farinaz Koushanfar, Sanmi Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Mariana Raykova, Hang Qi, Daniel Ramage, Ramesh Raskar, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Felix X. Yu, Han Yu, and Sen Zhao. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019. (Cited on pages 1, 2, and 3)

Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world, 2020. (Cited on pages 31 and 32)

Ahmed Khaled, Othmane Sebbouh, Nicolas Loizou, Robert M. Gower, and Peter Richtárik. Unified analysis of stochastic gradient methods for composite convex and smooth optimization, 2020. (Cited on pages 22, 28, and 31)

Jakub Konečný, H. Brendan McMahan, Felix X. Yu, Peter Richtárik, Ananda Theertha Suresh, and Dave Bacon. Federated learning: Strategies for improving communication efficiency. In *NIPS Private Multi-Party Machine Learning Workshop*, 2016. (Cited on pages 2 and 3)

Dingwei Li, Qinglong Chang, Lixue Pang, Yanfang Zhang, Xudong Sun, Jikun Ding, and Liang Zhang. More industry-friendly: Federated learning with high efficient design. *arXiv preprint arXiv:2012.08809*, 2020. (Cited on page 2)

Tian Li, Anit Kumar Sahu, Ameet Talwalkar, and Virginia Smith. Federated learning: Challenges, methods, and future directions. *arXiv preprint arXiv:1908.07873*, 2019. (Cited on page 2)

Zhize Li and Peter Richtárik. A unified analysis of stochastic gradient methods for nonconvex federated optimization, 2020. (Cited on page 33)

Grigory Malinovskiy, Dmitry Kovalev, Elnur Gasanov, Laurent Condat, and Peter Richtárik. From local SGD to local fixed-point methods for federated learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 6692–6701. PMLR, 2020. (Cited on page 3)

Yishay Mansour, Mehryar Mohri, Jae Ro, and Ananda Theertha Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020. (Cited on pages 3 and 36)

Konstantin Mishchenko, Eduard Gorbunov, Martin Takác, and Peter Richtárik. Distributed learning with compressed gradient differences. *arXiv preprint arXiv:1901.09269*, 2019. (Cited on pages 6, 7, and 21)

Yurii Nesterov. *Lectures on Convex Optimization*. Springer International Publishing, 2018. doi: 10.1007/978-3-319-91578-4. (Cited on pages 14 and 20)

Lam M. Nguyen, Jie Liu, Katya Scheinberg, and Martin Takác. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 2613–2621. PMLR, 2017. (Cited on page 6)

Alex Nichol, Joshua Achiam, and John Schulman. On first-order meta-learning algorithms. *arXiv preprint arXiv:1803.02999*, 2018. (Cited on page 8)

Reese Pathak and Martin J. Wainwright. FedSplit: an algorithmic framework for fast federated optimization. *arXiv preprint arXiv:2005.05238*, 2020. (Cited on page 3)

Sashank J. Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and Hugh Brendan McMahan. Adaptive federated optimization. In *International Conference on Learning Representations*, 2021. (Cited on pages 8 and 37)

Amirhossein Reisizadeh, Farzan Farnia, Ramtin Pedarsani, and Ali Jadbabaie. Robust federated learning: the case of affine distribution shifts. *arXiv preprint arXiv:2006.08907*, 2020. (Cited on page 3)

Saber Salehkaleybar, Arsalan Sharifnassab, and S. Jamaloddin Golestani. One-shot federated learning: Theoretical limits and algorithms to achieve them. *arXiv preprint arXiv:1905.04634*, 2019. (Cited on page 5)

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*, pages 1–13, 2020. doi: 10.1109/TNNLS.2020.3015958. (Cited on page 3)

TensorFlow Developers. TensorFlow Federated Stack Overflow dataset, 2021. URL https://www.tensorflow.org/federated/api_docs/python/tff/simulation/datasets/stackoverflow/load_data. (Cited on page 8)

Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. *arXiv preprint arXiv:1910.10252*, 2019. (Cited on page 4)

Blake Woodworth, Kumar Kshitij Patel, and Nathan Srebro. Minibatch vs local SGD for heterogeneous distributed learning. *arXiv preprint arXiv:2006.04735*, 2020. (Cited on page 3)

Zheng Xu, Karan Singhal, Zachary Charles, Ziyu Liu, Advait Gadhikar, and Shanshan Wu. Federated optimization. https://github.com/google-research/federated/tree/master/optimization, 2021. (Cited on page 37)

Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. Salvaging federated learning by local adaptation. *arXiv preprint arXiv:2002.04758*, 2020. (Cited on pages 1 and 3)

Edvin Listo Zec, Olof Mogren, John Martinsson, Leon René Sütfeld, and Daniel Gillblad. Specialized federated learning using a mixture of experts. *arXiv preprint arXiv:2010.02056*, 2021. (Cited on pages 3 and 36)

Pan Zhou, Xiaotong Yuan, Huan Xu, Shuicheng Yan, and Jiashi Feng. Efficient meta learning via minibatch proximal update. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 1534–1544. Curran Associates, Inc., 2019. (Cited on page 7)

APPENDIX

Contents

1 Introduction **1**

1.1 Key properties of the FedMixture framework 2

1.2 Related work 3

2 The FedMix formulation **3**

2.1 Motivation 1: from Local GD to FedMix 3

2.2 Motivation 2: from model-agnostic meta-learning to FedMix 4

3 Theory and algorithms **5**

3.1 Distributed gradient descent 6

4 Experiments **7**

5 Basic facts and notation **14**

6 Missing proofs for Sections 2 and 3 **15**

6.1 Proof of Proposition 1 15

6.2 Proofs for algorithm-independent results 15

6.2.1 Proof of Proposition 2 15

6.2.2 A proposition for bounding the gradient norm 16

6.2.3 Proof of Proposition 3 17

6.3 Result on one shot averaging 17

6.4 Distributed Gradient Descent 19

7 Other algorithms **21**

7.1 Strongly convex objectives 21

7.1.1 DCGD 21

7.1.2 DIANA 24

7.2 Convex objectives 28

7.2.1 DCGD 28

7.2.2 DIANA 30

7.3 Nonconvex objectives 31

7.3.1 DCGD 31

7.3.2 DIANA 33

8 Discussion of other model mixture methods **36**

9 Experimental Details **37**

10 Extra Experiments **39**

11 Limitations **47**

5 Basic facts and notation

The convexity of the norm $\|\cdot\|$ and Jensen's inequality implies the triangle inequality for norms:

$$\left\|\sum_{i=1}^{n} a_i\right\| \leq \sum_{i=1}^{n} \|a_i\| . \tag{8}$$

We say that a function g is L_g-smooth if for any $x, y \in \mathbb{R}^d$ we have

$$\|g(x) - g(y)\| \leq L_g \|x - y\| . \tag{9}$$

Note that (9) implies

$$g(x) \leq g(y) + \langle \nabla g(y), x - y \rangle + \frac{L_g}{2}\|x - y\|^2. \tag{10}$$

We say that a function g is μ_g-strongly convex for $\mu_g > 0$ if for all $x, y \in \mathbb{R}^d$ we have

$$g(x) \geq g(y) + \langle \nabla g(y), x - y \rangle + \frac{\mu_g}{2}\|x - y\|^2. \tag{11}$$

If (11) holds with $\mu_g = 0$, we say that g is convex.

We say $\mathcal{C} \in \mathbb{B}^d(\omega)$ is a compression operator if $\mathcal{C}$ is unbiased (i.e., $\mathbb{E}[\mathcal{C}(x)] = x$ for all $x \in \mathbb{R}^d$) and if the second moment is bounded as

$$\mathbb{E}\|\mathcal{C}(x) - x\|^2 \leq \omega\|x\|^2 \; \forall x \in \mathbb{R}^d. \tag{12}$$

Note that if $\mathcal{C} \in \mathbb{B}^d(\omega)$, then

$$\mathbb{E}\|\mathcal{C}(x)\|^2 \leq (1 + \omega)\|x\|^2, \forall x \in \mathbb{R}^d. \tag{13}$$

Let f be a convex function. Then the Bregman divergence associated with f for points $x, y \in \mathbb{R}^d$ is defined in the following way:

$$D_f(x, y) = f(x) - f(y) - \langle \nabla f(y), x - y \rangle.$$

Nesterov (2018) shows that if function f is convex and L-smooth, then for all x and y

$$\|\nabla f(x) - \nabla f(y)\|^2 \leq 2LD_f(x, y). \tag{14}$$

Let $a, b \in \mathbb{R}^n$ be arbitrary vectors. Then, it holds that

$$\|a + b\|^2 \leq 2(\|a\|^2 + \|b\|^2). \tag{15}$$

6 Missing proofs for Sections 2 and 3

This section collects the proofs of all propositions and theorems mentioned in the paper.

6.1 Proof of Proposition 1

In Section 2.2 we provided the motivation for FedMix through the lens of fine-tuning. We used the following proposition on fine-tuning quadratics:

Proposition 1. *Suppose that we run gradient descent for H steps on the quadratic objective $f_i = \frac{1}{2}x^T A_i x - b_i^T x + c$ starting from x^0 with stepsize $\gamma > 0$. Suppose that the stepsize satisfies $\gamma \leq \frac{1}{L_i}$, where $L_i = \lambda_{\max}(A_i)$, and suppose that A_i is positive definite. Then the final iterate x_i^H can be written as*

$$x_i^H = \left(I - J_i^H\right) x_i + J_i^H x^0, \tag{16}$$

where x_i minimizes f_i and $J_i \in \mathbb{R}^{d\times d}$ is a matrix with maximum eigenvalue smaller than 1, i.e. $\lambda_{\max}(J) < 1$.

Proof. The gradient descent update is

$$\begin{aligned} x_i^{t+1} &= x_i^t - \gamma \nabla f(x_i^t) \\ &= x_i^t - \gamma(A_i x_i^t - b_i) \\ &= (I - \gamma A_i)x_i^t + \gamma b_i. \end{aligned} \tag{17}$$

Note that because x_i minimizes f_i, we have $\nabla f_i(x_i) = 0$ by first-order optimality. Hence,

$$\nabla f_i(x_i) = 0 \Longleftrightarrow A_i x_i = b_i.$$

Using this in (17),

$$x_i^{t+1} = (I - \gamma A_i)x_i^t + \gamma A_i x_i$$

Subtracting x_i from both sides,

$$\begin{aligned} x_i^{t+1} - x_i &= (I - \gamma A_i)x_i^t + (\gamma A_i - I)x_i \\ &= (I - \gamma A_i)(x_i^t - x_i). \end{aligned}$$

Iterating the above equality for H steps we get

$$x_i^H - x_i = (I - \gamma A_i)^H (x^0 - x_i).$$

Rearranging the terms we get (16) with $J_i \overset{\text{def}}{=} (I - \gamma A_i)$. Observe that when $\gamma \leq \frac{1}{\lambda_{\max}(A_i)}$ and $\lambda_{\min}(A_i) > 0$ we have that $\lambda_{\max}(I - \gamma A_i) < 1$. □

6.2 Proofs for algorithm-independent results

6.2.1 Proof of Proposition 2

Proposition 2. *Suppose that each objective f_i is L_i-smooth. That is, for any $x, y \in \mathbb{R}^d$ we have*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_i \|x - y\|. \tag{18}$$

Then the FedMix objective $\tilde{f}$ defined in (1) is L_α-smooth for $L_\alpha \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i$. If each f_i is convex, then $\tilde{f}$ is also convex. If each f_i is μ_i-strongly convex, then $\tilde{f}$ is μ_α strongly convex for $\mu_\alpha \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^n \alpha_i^2 \mu_i$.

Proof. We separate the proofs in two cases: when each f_i is smooth, and when each f_i is (strongly) convex.

(i) Suppose that each f_i is L_i-smooth, and let $x, y \in \mathbb{R}^d$. Then by direct computation we have

$$\begin{aligned}\left\|\nabla \tilde{f}(x) - \nabla \tilde{f}(y)\right\| &= \left\|\frac{1}{n}\sum_{i=1}^{n} \alpha_i \left[\nabla f_i(\alpha_i x + (1-\alpha_i)x_i) - \nabla f_i(\alpha_i y + (1-\alpha_i)y_i)\right]\right\| \\ &\overset{(8)}{\leq} \frac{1}{n}\sum_{i=1}^{n} \|\alpha_i\| \left\|\nabla f_i(\alpha_i x + (1-\alpha_i)\, x_i) - \nabla f_i(\alpha_i y + (1-\alpha_i)y_i)\right\| \\ &\overset{(18)}{\leq} \frac{1}{n}\sum_{i=1}^{n} \alpha_i^2 L_i \|x-y\| = L_\alpha \|x-y\|,\end{aligned}$$

hence $\tilde{f}$ is L_α-smooth.

(ii) Suppose that each f_i is μ_i-strongly convex for $\mu_i \geq 0$ (where $\mu_i = 0$ corresponds to just convexity). Then for $x, h \in \mathbb{R}^d$ we have

$$\begin{aligned}\tilde{f}(x+h) &= \frac{1}{n}\sum_{i=1}^{n} f_i(\alpha_i(x+h) + (1-\alpha_i)x_i) \\ &\overset{(11)}{\geq} \frac{1}{n}\sum_{i=1}^{n} \left[f_i(\alpha_i x + (1-\alpha_i)x_i) + \langle \nabla f_i(\alpha_i x + (1-\alpha_i)x_i), \alpha_i h\rangle + \frac{\mu_i}{2}\|\alpha_i h\|^2\right] \\ &= \frac{1}{n}\sum_{i=1}^{n} f_i(\alpha_i x + (1-\alpha_i)x_i) + \left\langle \frac{1}{n}\sum_{i=1}^{n} \alpha_i \nabla f_i(\alpha_i x + (1-\alpha_i)x_i), h\right\rangle \\ &\qquad + \frac{1}{n}\sum_{i=1}^{n} \frac{\alpha_i^2 \mu_i}{2}\|h\|. \\ &= \tilde{f}(x) + \left\langle \nabla \tilde{f}(x), h\right\rangle + \frac{\mu_\alpha}{2}\|h\|^2,\end{aligned}$$

hence $\tilde{f}$ is μ_α-strongly convex if all the μ_i are positive (resp. convex if they are equal to 0).

□

6.2.2 A proposition for bounding the gradient norm

Proposition 2 provides us with a simple way to bound the gradient of $\tilde{f}$ in terms of the distance to the optimum x^α:

Proposition 4. *Define x^α as the solution to the FedMix problem (FedMix):*

$$x^\alpha \overset{def}{=} \operatorname*{argmin}_{x\in\mathbb{R}^d} \left[\tilde{f}(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(\alpha_i x + (1-\alpha_i)x_i)\right].$$

Define L_α as in Proposition 2. Then for any $x \in \mathbb{R}^d$ we have

$$\left\|\nabla \tilde{f}(x)\right\| \leq L_\alpha \|x - x^\alpha\|, \tag{19}$$

and

$$\tilde{f}(x) - \tilde{f}(x^\alpha) \leq \frac{L_\alpha}{2}\|x - x^\alpha\|^2. \tag{20}$$

Proof. By Proposition 2, we have for any $x, y \in \mathbb{R}^d$ that

$$\left\|\nabla \tilde{f}(x) - \nabla \tilde{f}(y)\right\| \leq L_\alpha \|x - y\|.$$

Putting $y = x^\alpha$ and using that $\nabla \tilde{f}(x^\alpha) = 0$ we get (19). The L_α-smoothness of $\tilde{f}$ implies

$$\tilde{f}(x) \leq \tilde{f}(y) + \left\langle \nabla \tilde{f}(y), x - y\right\rangle + \frac{L_\alpha}{2}\|x - y\|^2.$$

Putting $y = x^\alpha$ recovers (20). □

6.2.3 Proof of Proposition 3

Proposition 3. *Suppose that $\alpha_1 = \alpha_2 = \ldots = \alpha_n = \beta$ in the FedMix formulation (FedMix). Let $T_1(x), T_2(x), \ldots, T_n(x)$ be the deployed models defined in (5). If $y_1, \ldots, y_n$ are vectors in $\mathbb{R}^d$ and $\bar{y}$ is their mean, we define $V(y_1, \ldots, y_n)$ as the population variance $V(y_1, \ldots, y_n) \overset{def}{=} \frac{1}{n}\sum_{i=1}^n \|y_i - \bar{y}\|^2$. Then,*

$$V(T_1(x), T_2(x), \ldots, T_n(x)) = (1-\beta)^2 V(x_1, x_2, \ldots, x_n).$$

Proof. By direct computation observe

$$\begin{aligned}
V(T_1(x), T_2(x), \ldots, T_n(x)) &= \frac{1}{n}\sum_{i=1}^n \left\| T_i(x) - \frac{1}{n}\sum_{j=1}^n T_j(x) \right\|^2 \\
&= \frac{1}{n}\sum_{i=1}^n \left\| \beta x + (1-\beta)x_i - \frac{1}{n}\sum_{j=1}^n (\beta x + (1-\beta)x_j) \right\|^2 \\
&= \frac{1}{n}\sum_{i=1}^n \left\| (1-\beta)\left(x_i - \frac{1}{n}\sum_{j=1}^n x_j \right) \right\|^2 \\
&= (1-\beta)^2 V(x_1, x_2, \ldots, x_n).
\end{aligned}$$

□

6.3 Result on one shot averaging

Before proving Theorem 1, we will need the following lemma which shows that the gradient of the FedMix and the functional suboptimality can be bound using a weighted average of the iterate norms:

Lemma 1. *Suppose that each f_i is L_i-smooth. Then for any $x \in \mathbb{R}^d$ we have,*

$$\left\| \nabla \tilde{f}(x) \right\| \leq \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i \|x - x_i\|, \tag{21}$$

and,

$$\tilde{f}(x) \leq \frac{1}{n}\sum_{i=1}^n f_i(x_i) + \frac{1}{2n}\sum_{i=1}^n \alpha_i^2 L_i \|x - x_i\|^2. \tag{22}$$

Proof. Using the fact that $\nabla f_i(x_i) = 0$ for all i, and applying the triangle inequality (8) and the L_i-smoothness of f_i, we get

$$\begin{aligned}
\|\nabla f(x)\| &= \left\| \frac{1}{n}\sum_{i=1}^n \alpha_i \nabla f_i(\alpha_i x + (1-\alpha_i)x_i) \right\| \\
&= \left\| \frac{1}{n}\sum_{i=1}^n \alpha_i \left[\nabla f_i(\alpha_i x + (1-\alpha_i)x_i) - \nabla f_i(x_i) \right] \right\| \\
&\overset{(8)}{\leq} \frac{1}{n}\sum_{i=1}^n |\alpha_i| \left\| \nabla f_i(\alpha_i x + (1-\alpha_i)x_i) - \nabla f_i(x_i) \right\| \\
&\overset{(9)}{\leq} \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i \|x - x_i\|.
\end{aligned}$$

Using L_i-smoothness of f_i and that $\nabla f_i(x_i) = 0$, we get

$$\begin{aligned}
f_i(\alpha_i x + (1-\alpha_i)x_i) &= f_i(x_i + \alpha_i(x - x_0)) \\
&\overset{(10)}{\leq} f_i(x_i) + \langle \nabla f_i(x_i), \alpha_i(x - x_i) \rangle + \frac{\alpha_i^2 L_i}{2}\|x - x_i\|^2 \\
&= f_i(x_i) + \frac{\alpha_i^2 L_i}{2}\|x - x_i\|^2.
\end{aligned}$$

Averaging the above inequality yields (22). □

As communicated earlier in the paper, we can solve (FedMix) by taking a weighted average of the pure local models x_i if the α_i are small enough. This is a consequence of preprocessing step where we compute x_i, which is communication-free, and also a consequence of the new formulation, where for small α the FedMix objective is less responsive to argument change (see Proposition 2).

Theorem 1. *Suppose that each objective f_i is L_i-smooth, let $\hat{L} \overset{def}{=} \frac{1}{n}\sum_{i=1}^n L_i$. Given the pure local models $x_1, x_2, \dots, x_n$, define the weighted average*

$$x^{\text{avg}} \overset{def}{=} \sum_{i=1}^n w_i x_i, \qquad w_i \overset{def}{=} \frac{\alpha_i^2 L_i}{n L_\alpha}, \qquad L_\alpha \overset{def}{=} \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i. \tag{23}$$

We further define the constants

$$D \overset{def}{=} \max_{i,j=1,\dots,n, i\neq j} \|x_i - x_j\|^2, \qquad \text{and} \qquad V \overset{def}{=} \sum_{i=1}^n w_i \|x_i - x^{\text{avg}}\|^2 \tag{24}$$

Fix any $\epsilon > 0$. Assume that either $\max_{i=1,\dots,n} \alpha_i \leq \sqrt{2\epsilon}/\sqrt{\hat{L}D}$, or $\alpha_i = \beta$ for all i and $\beta \leq \sqrt{2\epsilon}/\sqrt{\hat{L}D}$. Then x^{avg} is an ϵ-approximate minimizer of Problem (FedMix). That is,

$$\tilde{f}(x^{\text{avg}}; \alpha_1, \dots, \alpha_n, x_1, \dots, x_n) - \min_{x\in\mathbb{R}^d} \tilde{f}(x; \alpha_1, \dots, \alpha_n, x_1, \dots, x_n) \leq \epsilon.$$

Proof. Lemma 1 provides the following bound on the suboptimality of any point x:

$$\tilde{f}(x) \leq \frac{1}{n}\sum_{i=1}^n f_i(x_i) + \mathcal{E}_\alpha(x), \tag{25}$$

where

$$\mathcal{E}_\alpha(x) \overset{\text{def}}{=} \frac{1}{2n}\sum_{i=1}^n \alpha_i^2 L_i \|x - x_i\|^2. \tag{26}$$

Note that since the vectors $\{x_i\}$ are known, the expression $\mathcal{E}_\alpha(x)$ can be minimized in x, leading to a weighted average of the pure local models:

$$x^{avg} \overset{\text{def}}{=} \sum_{i=1}^n w_i x_i, \quad w_i \overset{\text{def}}{=} \frac{\alpha_i^2 L_i}{n L_\alpha}, \quad L_\alpha \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i. \tag{27}$$

Note that x^{avg}, can be computed using a single communication round. By plugging (27) into (26), we can evaluate the error of the average model:

$$\mathcal{E}(x^{avg}) = \frac{L_\alpha}{2}\sum_{i=1}^n w_i \|x^{avg} - x_i\|^2 = \frac{L_\alpha V_\alpha}{2}, \tag{28}$$

where

$$V_\alpha \overset{\text{def}}{=} \sum_{i=1}^n w_i \|x^{avg} - x_i\|^2 = \mathbb{E}\|x_i - \mathbb{E}x_i\|^2 \tag{29}$$

can be interpreted as the *variance of the local optimal models* $\{x_i\}$. We now give two ways how $\mathcal{E}(x^{avg})$ can be bounded:

- Assume the customization parameters $\{\alpha_i\}$ are allowed to be arbitrary. Since x^{avg} is in the convex hull of the set $\{x_1, \dots, x_n\}$, we have $\|x^{avg} - x_i\| \leq \max_{i,j} \|x_i - x_j\|$. Therefore,

$$V_\alpha \leq D \overset{\text{def}}{=} \max_{i,j} \|x_i - x_j\|^2. \tag{30}$$

  Moreover,

$$L_\alpha = \frac{1}{n}\sum_{i=1}^n \alpha_i^2 L_i \leq \alpha_{\max}^2 \bar{L}, \tag{31}$$

  where $\alpha_{\max} \overset{\text{def}}{=} \max_i \alpha_i$ and $\bar{L} \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^n L_i$. By plugging (30) and (31) into (28), we get

$$\mathcal{E}(x^{avg}) \leq \frac{\alpha_{\max}^2 \bar{L} D}{2}. \tag{32}$$

Algorithm 1 Distributed gradient descent for FedMix.

Require: Number of communication rounds K, stepsize γ, initial point x^0

1: **for** $k = 0, 1, \ldots, K-1$ **do**
2: **for** $i = 1, \ldots, n$ in parallel **do**
3: Compute $\nabla \tilde{f}_i(x^k) = \alpha_i \nabla f_i\left(\alpha_i x^k + (1-\alpha_i)\, x_i\right)$ and communicate it to server.
4: Average and broadcast the new iterate

$$x^{k+1} = x^k - \frac{\gamma}{n} \sum_{i=1}^{n} \nabla \tilde{f}_i(x^k).$$

- Assume the customization parameters $\{\alpha_i\}$ are all equal: $\alpha_i = \beta$ for all i. Then $w_i = \frac{L_i}{\sum_j L_j}$, and hence x^{avg} and $V_\alpha = V$ are independent of β. Since $L_\alpha = \beta^2 \bar{L}$, by plugging these expressions into (28), we get

$$\mathcal{E}(x^{avg}) \leq \frac{\beta^2 \bar{L} V}{2}. \tag{33}$$

  Note that $V \leq D$.

Plugging (32) or (33) into (25) yields the following observation: fix any $\epsilon > 0$ and assume one of the following conditions holds

- $\alpha_{\max} \leq \sqrt{\frac{2\epsilon}{LD}}$.
- $\alpha_i = \beta$ for all i and $\beta \leq \sqrt{\frac{2\epsilon}{LV}}$.

Then the weighted average of the local optimal models x^{avg} satisfies

$$\tilde{f}(x^{\text{avg}}) \leq \frac{1}{n} \sum_{i=1}^{n} f_i(x_i) + \epsilon \leq \frac{1}{n} \sum_{i=1}^{n} f_i(x^\alpha) + \epsilon = \tilde{f}(x^\alpha) + \epsilon,$$

where $x^\alpha = \operatorname{argmin}_{x \in \mathbb{R}^d} \tilde{f}(x)$. □

6.4 Distributed Gradient Descent

Gradient descent is a simple but yet informative way to solve many optimization problems. Here we provide the proof for the convergence of DGD.

Theorem 2. *Suppose that each f_i in (FedMix) is L_i-smooth and μ_i-strongly convex. Define x^{avg}, L_α, $\hat{L}$, V and D as in Theorem 1. Suppose that we run DGD for K iterations starting from $x^0 = x^{\text{avg}}$. Then the following hold:*

i) If the α_i are allowed to be arbitrary, then for $\alpha_{\max} \overset{def}{=} \max_{i=1,\ldots,n} \alpha_i$ we have

$$\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \left(1 - \frac{\mu_\alpha}{L_\alpha}\right)^k \frac{\alpha_{\max}^2 \hat{L} D}{2}.$$

ii) If $\alpha_i = \beta$ for all i, then

$$\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \left(1 - \frac{\hat{\mu}}{\hat{L}}\right)^k \frac{\beta^2 \hat{L} V}{2},$$

where $\hat{\mu} \overset{def}{=} \frac{1}{n} \sum_{i=1}^{n} \mu_i$.

Proof. Recall the standard result that gradient descent for an L-smooth and μ-strongly convex objective g satisfies for any initial point x^0

$$g(x^k) - g_* \leq (1 - \gamma\mu)^k \left(g(x_0) - g_*\right), \tag{34}$$

where $g_* = \min_{x \in \mathbb{R}^d} g(x)$. For a proof, see (Nesterov, 2018). Note that by Proposition 2 we have that $\tilde{f}$ is L_α-smooth and μ_α-strongly convex. Specializing (34) to this case yields

$$\tilde{f}(x^k) - \min_{x \in \mathbb{R}^d} \tilde{f}(x) \leq \left(1 - \frac{\mu_\alpha}{L_\alpha}\right)^k \left(\tilde{f}(x_0) - \min_{x \in \mathbb{R}^d} \tilde{f}(x)\right) \tag{35}$$

Note that our initialization is the same x^{avg} from Theorem 1. Plugging the bounds of that theorem into (35) yields the theorem's claims. □

Algorithm	Assumption	Convergence guarantee	Corollary
DCGD	smoothness, str cvx	$\mathbb{E}\|x^k - x^*\|^2 \leq (1-\gamma_0\overline{\mu})^k C_1 + C_2$ $\mathbb{E}[\tilde{f}(x^k) - \tilde{f}^*] \leq ((1-\gamma_0\overline{\mu})^k C_1 + C_2)\alpha^2$	1
DIANA	smoothness, str cvx	$\mathbb{E}\|x^k - x^*\|^2 \leq (1-\rho)^k C$ $\mathbb{E}[\tilde{f}(x^k) - \tilde{f}^*] \leq (1-\rho)^k C\alpha^2$	2
DCGD	smoothness, cvx	$\mathbb{E}[\tilde{f}(\overline{x}^k) - \tilde{f}(x^*)] \leq \frac{1}{k}C_1\alpha^2 + C_2\alpha$	3
DIANA	smoothness, cvx	$\mathbb{E}[\tilde{f}(\overline{x}^k) - \tilde{f}(x^*)] \leq \frac{1}{k}(C_1\alpha^2 + C_2\alpha)$	4
DCGD	smoothness	$\min\limits_{0\leq t\leq k-1} \mathbb{E}\|\nabla\tilde{f}(x^t)\|^2 \leq \frac{(1+C_1\gamma_0^2)^k C_2}{\gamma_0 k}\alpha^2$	5
DIANA	smoothness	$\mathbb{E}\|\nabla\tilde{f}(\hat{x})\|^2 \leq \frac{C}{k}\alpha^2$	6

Table 1: Convergence results for Distributed Compressed Gradient Descent and DIANA in different settings. All the constants are independent of α.

7 Other algorithms

In this section, we include additional algorithms to solve (FedMix) formulation, namely, DCGD (Gorbunov et al., 2019) and DIANA (Mishchenko et al., 2019). In Table 1, we display convergence guarantees for these algorithms. One can see that similarly to our results in Section 3 both of the algorithms requires less iterations in terms of convergence in both local deploy iterates $T_i(x^k)$'s and functional value $\tilde{f}(x^k)$. In addition, for the standard (ERM) problem, i.e., $\alpha = 1$, we recover the best known convergence guarantees. Below, we provide a derivation of these claims.

7.1 Strongly convex objectives

7.1.1 DCGD

We firstly introduce the DCGD algorithm followed by general convergence results that we later exploit to obtain a convergence guarantee for DCGD applied to (FedMix).

Algorithm 2 Distributed Compressed Gradient Descent with different noise levels ω_i

Require: $x^0 \in \mathbb{R}^d$, learning rate γ
1: **for** $k = 0, 1, 2, \ldots$ **do**
2: Broadcast x^k to all workers
3: **for** $i = 1, \ldots, n$ in parallel **do**
4: Evaluate $\nabla f_i(x^k)$
5: $g_i^k = \mathcal{C}_i(\nabla f_i(x^k))$
6: $g^k = \frac{1}{n}\sum_{i=1}^n g_i^k$
7: $x^{k+1} = x^k - \gamma g^k$

Lemma 2. *Suppose each f_i is L_i-smooth and convex, and f is L-smooth. Let $\mathcal{C}_i : \mathbb{R}^d \to \mathbb{R}^d$ be randomized compression operators satisfying $\mathcal{C}_i \in \mathbb{B}^d(\omega_i)$. Let $g_k = \frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla f_i(x^k))$. Then*

$$\mathbb{E}\|g^k - \nabla f(x^*)\|^2 \leq 2\left(L + \frac{2\max\{L_i\omega_i\}}{n}\right) D_f(x^k, x^*) + \sigma_{DCGD}, \tag{36}$$

where $\sigma_{DCGD} = \frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^*)\|^2$.

Proof. By applying bias-variance decomposition to $\mathbb{E}\|g^k - \nabla f(x^*)\|^2$, we get

$$\mathbb{E}\|g^k - \nabla f(x^*)\|^2 = \|\nabla f(x^k) - \nabla f(x^*)\|^2 + \mathbb{E}\|g^k - \nabla f(x^k)\|^2.$$

Since all functions are convex holds, function f as a linear combination of convex functions is also convex. That is why the first term enjoys the 'classic' bound for *convex* and smooth functions expressed in Bregman divergence between x^k and x^* (see inequality 14):

$$\|\nabla f(x^k) - \nabla f(x^*)\|^2 \leq 2LD_f(x^k, x^*). \tag{37}$$

We start looking into the second term by expanding the quadratic:

$$
\begin{aligned}
&\mathbb{E}\|g^k - \nabla f(x^k)\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla f_i(x^k)) - \frac{1}{n}\sum_{i=1}^n \nabla f_i(x^k)\right\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n (\mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k))\right\|^2 \\
&= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\|\mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k)\|^2 + \\
&\quad \frac{1}{n^2}\sum_{i\neq j} \mathbb{E}\langle \mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k), \mathcal{C}_j(\nabla f_j(x^k)) - \nabla f_j(x^k)\rangle.
\end{aligned}
$$

Since $\mathcal{C}_i \in \mathbb{B}(\omega_i)\ \forall i$ is drawn independently, the expected value of each scalar product in the second sum becomes the scalar product of expected values, each of which is zero due to unbiasedness of a compression operator (see Section 5):

$$
\begin{aligned}
&\mathbb{E}\langle \mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k), \mathcal{C}_j(\nabla f_j(x^k)) - \nabla f_j(x^k)\rangle \\
&= \langle \underbrace{\mathbb{E}\mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k)}_{=0}, \underbrace{\mathbb{E}\mathcal{C}_j(\nabla f_j(x^k)) - \nabla f_j(x^k)}_{=0}\rangle = 0.
\end{aligned}
$$

Then, we apply 12 (the second property of compressed operators) to get the final upper-bound on $\mathbb{E}\|g^k - \nabla f(x^k)\|^2$.

$$
\mathbb{E}\|g^k - \nabla f(x^k)\|^2 = \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\|\mathcal{C}_i(\nabla f_i(x^k)) - \nabla f_i(x^k)\|^2 \leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k)\|^2.
$$

Curiously but not surprisingly, the result resembles the Law of large numbers (indeed, if each squared gradient norm and noise level are bounded above by values R and ω_0 respectively, then the right side is bounded by $\frac{\omega_0 R}{n}$, which converges to zero as n converges to infinity), what partially verifies the correctness of the proof. The error $\mathbb{E}\|g^k - \nabla f(x^k)||^2$ is now estimated by squared gradient norms at iterate x^k, which is, in general, a random point generated by CGD. Since we already have some dependence on Bregman divergence in 37, which perfectly fits assumptions for unified theories from papers Gorbunov et al. (2019) and Khaled et al. (2020) required in later theorems, we express the term in the right side of the last inequality in Bregman divergence, too. By subtracting and adding the same vector $\nabla f_i(x^*)$ inside the norm operator we get

$$
\begin{aligned}
&\frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k)\|^2 = \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*) + \nabla f_i(x^*)\|^2 \\
&\leq \frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*)\|^2 + \frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^*)\|^2,
\end{aligned}
$$

where in the second line we used (15). Applying again the bound from 14 to the first term and using the linearity of Bregman divergence we have

$$
\begin{aligned}
&\frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*)\|^2 \leq \frac{2}{n^2}\sum_{i=1}^n \omega_i 2L_i D_{f_i}(x^k, x^*) \\
&\leq \frac{4\max\{L_i\omega_i\}}{n^2}\sum_{i=1}^n D_{f_i}(x^k, x^*) = \frac{4\max\{L_i\omega_i\}}{n} D_f(x^k, x^*).
\end{aligned}
$$

Incorporation of the results above gives the statement. □

Proposition 5. *Suppose each f_i is L_i-smooth and μ_i-strong. Assume $\alpha_i \equiv \alpha \in \mathbb{R}$ for all i. For any $x^0 \in \mathbb{R}^d$, it holds that*

$$\|x^0 - x^*\|^2 \leq \frac{1}{\overline{\mu}} \overline{L_i \|x^0 - x_i\|^2}. \tag{38}$$

Proof.

$$\|x^0 - x^*\|^2 \overset{(11)}{\leq} \frac{2}{\mu_\alpha}(\tilde{f}(x^0) - \tilde{f}(x^*)) \overset{\text{Proposition 2}}{=} \frac{2}{\overline{\mu}\alpha^2}(\tilde{f}(x^0) - \tilde{f}(x^*)) \overset{(25)}{\leq} \frac{2}{\overline{\mu}\alpha^2}\mathcal{E}_\alpha(x^0)$$
$$\overset{(26)}{=} \frac{1}{\overline{\mu}\alpha^2}\frac{1}{n}\sum_{i=1}^n L_i \alpha^2 \|x^0 - x_i\|^2 = \frac{1}{\overline{\mu}}\overline{L_i\|x^0 - x_i\|^2}.$$

□

Proposition 6. *Suppose each f_i is L_i-smooth and μ_i-strong. Assume $\alpha_i \equiv \alpha \in \mathbb{R}$ for all i. Then,*

$$\overline{\|x_i - x^*\|^2} \leq \frac{\max_i L_i}{\overline{\mu}} \max_{i,j} \|x_i - x_j\|^2. \tag{39}$$

Proof. According to Proposition 5, for any x_l^* where $l \in \{1, \ldots, n\}$ it follows that

$$\|x_l - x^*\|^2 \leq \frac{1}{\overline{\mu}}\overline{L_i\|x_l - x_i\|^2} \leq \frac{1}{\overline{\mu}}\max_i L_i \cdot \overline{\|x_l - x_i\|^2} \leq \frac{1}{\overline{\mu}}\max_i L_i \cdot \max_{i,j}\|x_j - x_i\|^2.$$

Taking the average of both sides finishes the proof. □

Theorem 3. *Assume all conditions of Lemma 2 hold, and each function f_i is μ_i-strongly convex. Then, if $\gamma \leq \frac{1}{L_\alpha + \frac{2\max\{L_i\alpha_i^2\omega_i\}}{n}}$, then*

$$\mathbb{E}\|x^k - x^*\|^2 \leq (1 - \gamma\mu_\alpha)^k \|x^0 - x^*\|^2 + \frac{2\gamma}{\mu_\alpha n^2}\sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x^*))]\|^2. \tag{40}$$

Proof. The lemma is a direct corollary of Theorem 4.1 from Gorbunov et al. (2019) and Proposition 2 with constants $A = L_\alpha + \frac{2\max\{L_i\alpha_i^2\omega_i\}}{n}$, $D_1 = \sigma_{DCGD} = \frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x^*))]\|^2$, $B = 0, \sigma_k^2 \equiv 0, \rho = 1, C = 0, D_2 = 0$. □

Corollary 1. *Assume all conditions of Theorem 3 hold, and $\alpha_i \equiv \alpha\ \forall i$. If $\gamma = \frac{1}{\alpha^2\left(\overline{L} + \frac{2\omega\max_i L_i}{n}\right)} = \gamma_0 \frac{1}{\alpha^2}$, then*

$$\mathbb{E}\|x^k - x^*\|^2 \leq (1 - \gamma_0\overline{\mu})^k \frac{1}{\overline{\mu}}\overline{L_i\|x^0 - x_i\|^2} + \frac{2\gamma_0\omega(\max_i L_i)^2 \max_{i,j}\|x_i - x_j\|^2}{\overline{\mu}^2 n} \tag{41}$$

and

$$\mathbb{E}[\tilde{f}(x^k) - \tilde{f}^*] \leq ((1 - \gamma_0\overline{\mu})^k C_1 + C_2)\alpha^2, \tag{42}$$

where $C_1 = \frac{\overline{L_i}}{2\overline{\mu}}\overline{L_i\|x^0 - x_i\|^2}$ and $C_2 = \frac{\overline{L_i}\gamma_0\omega(\max_i L_i)^2 \max_{i,j}\|x_i - x_j\|^2}{\overline{\mu}^2 n}$.

Proof. Notice that

$$\sum_{i=1}^n \omega_i \|\nabla[f_i(\alpha_i x^* + (1-\alpha_i)x_i)]\|^2 = \sum_{i=1}^n \omega_i \|\alpha_i \nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i)\|^2$$
$$= \omega\alpha^2 \sum_{i=1}^n \|\nabla f_i(\alpha x^* + (1-\alpha)x_i)\|^2 \overset{\nabla f_i(x_i)=0}{=} \omega\alpha^2 \sum_{i=1}^n \|\nabla f_i(\alpha x^* + (1-\alpha)x_i) - \nabla f_i(x_i)\|^2$$
$$\overset{(9)}{\leq} \omega\alpha^2 \sum_{i=1}^n L_i \|\alpha(x^* - x_i)\|^2 \leq \omega\alpha^4 \max_i L_i n \overline{\|x^* - x_i\|^2} \overset{(39)}{\leq} \frac{\omega\alpha^4 n(\max_i L_i)^2}{\overline{\mu}} \max_{i,j}\|x_i - x_j\|^2.$$

This brings us to the following bound on the neigbourhood:

$$\begin{aligned}&\frac{2\gamma}{\mu_\alpha n^2}\sum_{i=1}^n \omega_i\|\nabla[f_i(T_i(x^*))]\|^2 = \frac{2\gamma_0}{\alpha^4\overline{\mu}n^2}\sum_{i=1}^n \omega_i\|\nabla[f_i(T_i(x^*))]\|^2\\ &\leq \frac{2\gamma_0\omega(\max_i L_i)^2\max_{i,j}\|x_i-x_j\|^2}{\overline{\mu}^2 n}.\end{aligned} \tag{43}$$

We further investigate the rate of convergence

$$1-\gamma\mu_\alpha \overset{\text{Proposition 2}}{=} 1-\gamma_0\frac{1}{\alpha^2}\overline{\mu}\alpha^2 = 1-\gamma_0\overline{\mu}. \tag{44}$$

As can be seen, the rate of convergence does not depend on α.

Combining last two results we obtain the following dependence on α for CGD convergence:

$$\begin{aligned}&\mathbb{E}\|x^k-x^*\|^2\\ &\overset{(40)}{\leq} (1-\gamma\mu_\alpha)^k\|x^0-x^*\|^2 + \frac{2\gamma}{\mu_\alpha n^2}\sum_{i=1}^n \omega_i\|\nabla[f_i(T_i(x^*))]\|^2\\ &\overset{(44)}{=} (1-\gamma_0\overline{\mu})^k\|x^0-x^*\|^2 + \frac{2\gamma}{\mu_\alpha n^2}\sum_{i=1}^n \omega_i\|\nabla[f_i(T_i(x^*))]\|^2\\ &\overset{(38)}{\leq} (1-\gamma_0\overline{\mu})^k\frac{1}{\overline{\mu}}\overline{L_i\|x^0-x_i\|^2} + \frac{2\gamma}{\mu_\alpha n^2}\sum_{i=1}^n \omega_i\|\nabla[f_i(T_i(x^*))]\|^2\\ &\overset{(43)}{\leq} (1-\gamma_0\overline{\mu})^k\frac{1}{\overline{\mu}}\overline{L_i\|x^0-x_i\|^2} + \frac{2\gamma_0\omega(\max_i L_i)^2\max_{i,j}\|x_i-x_j\|^2}{\overline{\mu}^2 n}\\ &= (1-\gamma_0\overline{\mu})^k\widetilde{C}_1 + \widetilde{C}_2,\end{aligned} \tag{45}$$

where constants $\widetilde{C}_1 = \frac{1}{\overline{\mu}}\overline{L_i\|x^0-x_i\|^2}$ and $\widetilde{C}_2 = \frac{2\gamma_0\omega(\max_i L_i)^2\max_{i,j}\|x_i-x_j\|^2}{\overline{\mu}^2 n}$ do not depend on α.

In terms of suboptimality convergence for CGD we have

$$\mathbb{E}[\tilde{f}(x^k)-\tilde{f}^*] \overset{(10)}{\leq} \frac{L_\alpha}{2}\mathbb{E}\|x^k-x^*\|^2 \overset{L_\alpha \text{ def}}{=} \frac{\overline{L}}{2}\alpha^2\mathbb{E}\|x^k-x^*\|^2 \overset{(45)}{\leq} ((1-\gamma_0\overline{\mu})^k C_1 + C_2)\alpha^2, \tag{46}$$

where $C_1 = \frac{\widetilde{C}_1\overline{L}}{2}$ and $C_2 = \frac{\widetilde{C}_2\overline{L}}{2}$. □

7.1.2 DIANA

We follow the same procedure as for the previous subsection. Below, we introduce DIANA algorithm, followed by general theorem, which is then applied to (FedMix).

Algorithm 3 DIANA with different noise levels ω_i

Require: $x^0, h_1^0, \ldots, h_n^0 \in \mathbb{R}^d$, $h^0 = \frac{1}{n}\sum_{i=1}^n h_i^0$

for k = 0, 1, 2, ... **do**
  Broadcast x^k to all workers
  for $i = 1, \ldots, n$ in parallel **do**
    $\Delta_i^k = \nabla f_i(x^k) - h_i^k$
    Sample $\hat{\Delta}_i^k \sim \mathcal{C}_i(\Delta_i^k)$
    $h_i^{k+1} = h_i^k + \beta_i\hat{\Delta}_i^k$
    $\hat{g}_i^k = h_i^k + \hat{\Delta}_i^k$
  $g^k = \frac{1}{n}\sum_{i=1}^n \hat{g}_i^k = h^k + \frac{1}{n}\sum_{i=1}^n \hat{\Delta}_i^k$
  $x^{k+1} = x^k - \gamma g^k$
  $h^k = \frac{1}{n}\sum_{i=1}^n h_i^{k+1} = h^k + \frac{1}{n}\sum_{i=1}^n \beta_i\hat{\Delta}_i^k$

Lemma 3. *Suppose each f_i is L_i-smooth and convex, and f is L-smooth. Let $\mathcal{C}_i : \mathbb{R}^d \to \mathbb{R}^d$ be randomized compression operators satisfying $\mathcal{C}_i \in \mathbb{B}^d(\omega_i)$. Let $g_k = \frac{1}{n}\sum_{i=1}^n h_i^k + \mathcal{C}_i(\nabla f_i(x^k) - h_i^k)$. Then*

$$\mathbb{E}\|g^k - \nabla f(x^*)\|^2 \leq 2\left(L + \frac{2\max\{L_i\omega_i\}}{n}\right) D_f(x^k, x^*) + \frac{2}{n}\sigma_k^2, \tag{47}$$

where $\sigma_k^2 = \frac{1}{n}\sum_{i=1}^n \omega_i \|h_i^k - \nabla f_i(x^*)\|^2$.

Proof. We start with bias-variance decomposition:

$$\mathbb{E}\|g^k - \nabla f(x^*)\|^2 = \|\nabla f(x^k) - \nabla f(x^*)\|^2 + \mathbb{E}\|g^k - \nabla f(x^k)\|^2, \tag{48}$$

where the first term in RHS bounded due to convexity and smoothness of function f: $\|\nabla f(x^k) - \nabla f(x^*)\|^2 \overset{(14)}{\leq} 2LD_f(x^k, x^*)$. Function f is, indeed, convex as a linear combination of convex functions f_i.

When expanding the second term in RHS of equation 48, we encounter scalar products, each of which is zero in expectation due to independence of $\mathcal{C}_i$ for all i and unbiasedness of a compression operator (see the definition in Section 5).

$$\begin{aligned}
&\mathbb{E}\|g^k - \nabla f(x^k)\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n (h_i^k + \mathcal{C}_i(\nabla f_i(x^k) - h_i^k)) - \frac{1}{n}\sum_{i=1}^n \nabla f_i(x^k)\right\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla f_i(x^k) - h_i^k) - (\nabla f_i(x^k) - h_i^k)\right\|^2 \\
&= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\|\mathcal{C}_i(\nabla f_i(x^k) - h_i^k) - (\nabla f_i(x^k) - h_i^k)\|^2 \\
&\quad + \frac{1}{n^2}\sum_{i\neq j} \underbrace{\mathbb{E}\langle \mathcal{C}_i(\nabla f_i(x^k) - h_i^k) - (\nabla f_i(x^k) - h_i^k), \mathcal{C}_j(\nabla f_j(x^k) - h_j^k) - (\nabla f_j(x^k) - h_j^k)\rangle}_{=0}.
\end{aligned}$$

Then, we apply bounded variance property of compression operators:

$$\begin{aligned}
\mathbb{E}\|g^k - \nabla f(x^k)\|^2 &= \frac{1}{n^2}\sum_{i=1}^n \mathbb{E}\|\mathcal{C}_i(\nabla f_i(x^k) - h_i^k) - (\nabla f_i(x^k) - h_i^k)\|^2 \\
&\leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - h_i^k\|^2.
\end{aligned}$$

We subtract and add $\nabla f_i(x^*)$ inside each norm operator to split $\nabla f_i(x^k)$ and h_i^k from each other using (15):

$$\begin{aligned}
\mathbb{E}\|g^k - \nabla f(x^k)\|^2 &\leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - h_i^k\|^2 \\
&= \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*) - (h_i^k - \nabla f_i(x^*))\|^2 \\
&\leq \frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*)\|^2 + \frac{2}{n^2}\sum_{i=1}^n \omega_i \|h_i^k - \nabla f_i(x^*)\|^2
\end{aligned}$$

While leaving the second term in the last line unchanged (it is basically $2\sigma_k/n$), we apply (14) to each norm of gradient differences in the first term and use max-function over $L_i\omega_i$ to take out Bregman divergences:

$$\begin{aligned}
&\frac{2}{n^2}\sum_{i=1}^n \omega_i \|\nabla f_i(x^k) - \nabla f_i(x^*)\|^2 \leq \frac{2}{n^2}\sum_{i=1}^n \omega_i 2L_i D_{f_i}(x^k, x^*) \\
&\leq \frac{4\max\{\omega_i L_i\}}{n^2}\sum_{i=1}^n D_{f_i}(x^k, x^*) = \frac{4\max\{\omega_i L_i\}}{n} D_f(x^k, x^*),
\end{aligned}$$

where in the last part we used linearity property of Bregman divergence. It remains to incorporate all results to get the statement. □

Lemma 4. *Assuming all conditions of Lemma 3 hold, let* $h_i^{k+1} = h_i^k + \beta_i \mathcal{C}_i(\nabla f_i(x^k) - h_i^k)$ *and* $\beta_i \leq \frac{1}{\omega_i+1}$*, where* $i \in \{1, \ldots, n\}$*. Then,*

$$\sigma_{k+1}^2 \leq (1 - \min \beta_i)\sigma_k^2 + 2\max\{\beta_i \omega_i L_i\} D_f(x^k, x^*). \tag{49}$$

Proof. Gorbunov et al. (2019) state for $\beta_i \leq \frac{1}{\omega_i+1}$

$$\mathbb{E}\|h_i^{k+1} - \nabla f_i(x^*)\|^2 \leq (1-\beta_i)\|h_i^k - \nabla f_i(x^*)\|^2 + 2\beta_i L_i D_{f_i}(x^k, x^*). \tag{50}$$

From this it follows that

$$\begin{aligned}
\sigma_{k+1}^2 &= \frac{1}{n}\sum_{i=1}^n \omega_i \|h_i^{k+1} - \nabla f_i(x^*)\|^2 \\
&\leq \frac{1}{n}\sum_{i=1}^n (1-\beta_i)\omega_i \|h_i^k - \nabla f_i(x^*)\|^2 + \frac{1}{n}\sum_{i=1}^n 2\beta_i \omega_i L_i D_{f_i}(x^k, x^*) \\
&\leq (1 - \min \beta_i)\sigma_k^2 + \frac{1}{n}\sum_{i=1}^n 2\beta_i \omega_i L_i D_{f_i}(x^k, x^*) \\
&\leq (1 - \min \beta_i)\sigma_k^2 + 2\max\{\beta_i \omega_i L_i\}\frac{1}{n}\sum_{i=1}^n D_{f_i}(x^k, x^*) \\
&= (1 - \min \beta_i)\sigma_k^2 + 2\max\{\beta_i \omega_i L_i\} D_f(x^k, x^*).
\end{aligned}$$

□

Theorem 4. *Assume all conditions of Lemmas 3, 4 hold, and each function* f_i *is* μ_i*-strongly convex. If* $\gamma \leq \frac{1}{L_\alpha + \frac{2\max\{L_i \alpha_i^2 \omega_i\}}{n} + \frac{4\max\{\beta_i \omega_i L_i \alpha_i^2\}}{n \min \beta_i}}$*, then*

$$\mathbb{E}[\mathcal{D}^k] \leq \max\left\{(1-\gamma\mu_\alpha)^k, \left(1 - \frac{1}{2}\min \beta_i\right)^k\right\}\mathcal{D}^0, \tag{51}$$

where $\mathcal{D}^k = \|x^k - x^*\|^2 + \frac{4}{n \min \beta_i}\gamma^2 \sigma_k^2$ *and* L_α *is defined as in Proposition 2.*

Proof. The lemma is a direct corollary of Theorem 4.1 from Gorbunov et al. (2019), 3 and 4 and strong convexity with constants: $A = L_\alpha + \frac{2\max\{L_i \alpha_i^2 \omega_i\}}{n}$, $B = \frac{2}{n}$, $D_1 = 0$, $\rho = \min \beta_i$, $C = \max\{\beta_i \omega_i L_i \alpha_i^2\}$, $D_2 = 0$, $M = \frac{4}{n \min \beta_i} = \frac{2B}{\rho} \geq \frac{4(1+\max \omega_i)}{n}$. □

Corollary 2. *Assume all conditions of Theorem 4 hold and* $\alpha_i \equiv \alpha \ \forall i$*. If* $\gamma = \frac{1}{\alpha^2} \cdot \frac{1}{\overline{L} + \frac{2\max\{L_i \omega_i\}}{n} + \frac{4\max\{\beta_i \omega_i L_i\}}{n \min \beta_i}} =: \gamma_0 \frac{1}{\alpha^2}$*, then*

$$\mathbb{E}\|x^k - x^*\|^2 \leq \max\left\{(1-\gamma_0\overline{\mu})^k, \left(1 - \frac{1}{2}\min \beta_i\right)^k\right\}\left(1 + \frac{4}{n \min \beta_i}\gamma_0^2 \overline{\omega_i L_i^2}\right)\frac{1}{\overline{\mu}}\overline{L_i \|x^0 - x_i\|^2} \tag{52}$$

and

$$\mathbb{E}[\tilde{f}(x^k)] - \tilde{f}^* \leq \max\left\{(1-\gamma_0\overline{\mu})^k, \left(1 - \frac{1}{2}\min \beta_i\right)^k\right\} C\alpha^2, \tag{53}$$

where $C = \frac{\overline{L_i}}{2}\left(1 + \frac{4}{n \min \beta_i}\gamma_0^2 \overline{\omega_i L_i^2}\right)\frac{1}{\overline{\mu}}\overline{L_i \|x^0 - x_i\|^2}$.

Proof. Let us investigate the quantity $\mathcal{D}^0$. We assume memory h_0^i in DIANA algorithm is equal to $\nabla[f_i(T_i(x^0))]$ for $i \in \{1, \ldots, n\}$. Plugging in definitions and initializations we obtain expanded definition of $\mathcal{D}^0$:

$$
\begin{aligned}
\mathcal{D}^0 &\overset{\text{def}}{=} \|x^0 - x^*\|^2 + \frac{4}{n \min \beta_i} \gamma^2 \sigma_0^2 \overset{\gamma = \gamma_0 \frac{1}{\alpha^2}}{=} \|x^0 - x^*\|^2 + \frac{4}{n \min \beta_i} \gamma_0^2 \frac{1}{\alpha^4} \sigma_0^2 \\
&\overset{\sigma_0^2 \text{ def}}{=} \|x^0 - x^*\|^2 + \frac{4}{n \min \beta_i} \gamma_0^2 \frac{1}{\alpha^4} \frac{1}{n} \sum_{i=1}^n \omega_i \|h_0^i - \nabla[f_i(T_i(x^*))]\|^2 \\
&\overset{h_0^i \text{initialisation}}{=} \|x^0 - x^*\|^2 + \frac{4}{n \min \beta_i} \gamma_0^2 \frac{1}{\alpha^4} \frac{1}{n} \sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x^0))] - \nabla[f_i(T_i(x^*))]\|^2 \\
&.
\end{aligned}
$$

We further upper bound the sum in the last term using smoothness properties of functions f_i:

$$
\begin{aligned}
&\frac{1}{n} \sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x^0))] - \nabla[f_i(T_i(x^*))]\|^2 \\
&= \frac{1}{n} \sum_{i=1}^n \omega_i \|\nabla[f_i(\alpha x^0 + (1-\alpha) x_i^*)] - \nabla[f_i(\alpha x^* + (1-\alpha) x_i^*)]\|^2 \\
&= \frac{1}{n} \sum_{i=1}^n \omega_i \|\alpha(\nabla f_i(\alpha x^0 + (1-\alpha) x_i^*) - \nabla f_i(\alpha x^* + (1-\alpha) x_i^*))\|^2 \\
&= \alpha^2 \frac{1}{n} \sum_{i=1}^n \omega_i \|\nabla f_i(\alpha x^0 + (1-\alpha) x_i^*) - \nabla f_i(\alpha x^* + (1-\alpha) x_i^*)\|^2 \\
&\leq \alpha^2 \frac{1}{n} \sum_{i=1}^n \omega_i L_i^2 \|\alpha(x^0 - x^*)\|^2 = \alpha^4 \overline{\omega_i L_i^2} \|x^0 - x^*\|^2.
\end{aligned}
$$

Plugging this result back into the definition of $\mathcal{D}^0$ and applying further bounds we get

$$
\begin{aligned}
\mathcal{D}^0 &\leq \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \|x^0 - x^*\|^2 \\
&\overset{(11)}{\leq} \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \frac{2}{\mu_\alpha} (\tilde{f}(x^0) - \tilde{f}(x^*)) \\
&\overset{\text{Proposition 2}}{=} \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \frac{2}{\overline{\mu} \alpha^2} (\tilde{f}(x^0) - \tilde{f}(x^*)) \\
&\overset{(25)}{\leq} \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \frac{2}{\overline{\mu} \alpha^2} \mathcal{E}_\alpha(x^0) \\
&\overset{(26)}{=} \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \frac{1}{\overline{\mu} \alpha^2} \frac{1}{n} \sum_{i=1}^n L_i \alpha^2 \|x^0 - x_i\|^2 \\
&= \left(1 + \frac{4}{n \min \beta_i} \gamma_0^2 \overline{\omega_i L_i^2}\right) \frac{1}{\overline{\mu}} \overline{L_i \|x^0 - x_i\|^2},
\end{aligned}
$$

what shows that $\mathcal{D}^0$ can be upper bounded by value *independent* of α. Since $M\gamma^2\sigma_k^2 \geq 0$ $\mathcal{D}^k = \|x^k - x^*\|^2 + M\gamma^2\sigma_k^2 \geq \|x^k - x^*\|^2$, and that is why

$$
\begin{aligned}
\mathbb{E}\|x^k - x^*\|^2 \leq \mathbb{E}\mathcal{D}^k &\leq \max\left\{(1-\gamma\mu_\alpha)^k, \left(1 - \frac{1}{2}\min\beta_i\right)^k\right\}\mathcal{D}^0 \\
&= \max\left\{(1 - \left(\gamma_0\frac{1}{\alpha^2}\right)(\overline{\mu}\alpha^2))^k, \left(1 - \frac{1}{2}\min\beta_i\right)^k\right\}\mathcal{D}^0 \\
&= \max\left\{(1-\gamma_0\overline{\mu})^k, \left(1 - \frac{1}{2}\min\beta_i\right)^k\right\}\mathcal{D}^0 \\
&\leq \max\left\{(1-\gamma_0\overline{\mu})^k, \left(1 - \frac{1}{2}\min\beta_i\right)^k\right\}\left(1 + \frac{4}{n\min\beta_i}\gamma_0^2\overline{\omega_i L_i^2}\right)\frac{1}{\overline{\mu}}\overline{L_i\|x^0 - x_i\|^2},
\end{aligned}
$$

what means that the convergence of iterates is completely independent from α. To derive the result for the convergence of functional values we use L_α-smoothness of function $\tilde{f}$ (see Proposition 2): $\mathbb{E}[\tilde{f}(x^k)] - \tilde{f}^* \leq \frac{L_\alpha}{2}\mathbb{E}\|x^k - x^*\|^2 = \alpha^2\frac{\overline{L_i}}{2}\mathbb{E}\|x^k - x^*\|^2$.

□

7.2 Convex objectives

7.2.1 DCGD

As for the previous section, we first introduce the algorithm, followed by general theorem, which is then applied to (FedMix).

Theorem 5. *Assume all conditions of Lemma 2 hold. Let* $0 < \gamma \leq \frac{1}{4\left(L_\alpha + \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}\right)}$. *Then,*

$$
\mathbb{E}[\tilde{f}(\overline{x}^k) - \tilde{f}(x^*)] \leq \frac{2(\tilde{f}(x^0) - \tilde{f}(x^*))}{k} + \frac{\|x^0 - x^*\|^2}{\gamma k} + 2\gamma\sigma_{DCGD}, \tag{54}
$$

where $\overline{x}^k = (1/k)\sum\limits_{j=1}^{k} x^j$.

Proof. The theorem is a corollary of Corollary 4.1 from Khaled et al. (2020) with constants

$$
\begin{aligned}
&A = L_\alpha + \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n},\ B = 0,\ \sigma_k^2 \equiv 0, \\
&\rho = 1,\ C = 0,\ D_2 = 0, \\
&D_1 = \sigma_{DCGD} = \frac{2}{n^2}\sum_{i=1}^{n}\omega_i\|\nabla[f_i(T_i(x^*))]\|^2 \\
&= \frac{2}{n^2}\sum_{i=1}^{n}\omega_i\alpha_i^2\|\nabla f_i(T_i(x^*))\|^2.
\end{aligned}
$$

□

Corollary 3. *Assume all conditions of Lemma 2 hold. Let $\alpha_i = \alpha$, $\omega_i = \omega$ for all i, $\gamma = \frac{1}{4(\overline{L_i}+\frac{2\omega \max_i L_i}{n})\alpha^2}$ and $\sup\limits_{\alpha\in[0,1]} \inf\limits_{x^*\in\mathcal{X}} \|x^0 - x^*\| = R$. Then,*

$$
\begin{aligned}
&\mathbb{E}[\tilde{f}(\overline{x}^k) - \tilde{f}(x^*)] \\
&\leq \alpha^2 \cdot \frac{2}{k}\left(\overline{L_i\|x^0 - x_i\|^2} + 2\left(\overline{L_i} + \frac{2\max\{L_i\omega_i\}}{n}\right)R^2\right) \\
&\qquad + \alpha \cdot \frac{\omega \max_i L_i}{n\overline{L_i} + 2\omega \max_i L_i}\left(\widetilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right) \\
&= \frac{1}{k}C_1\alpha^2 + C_2\alpha.
\end{aligned}
$$

Proof. Let us investigate $\tilde{f}(x^0) - \tilde{f}(x^*)$. Let $\alpha_i \equiv \alpha \in \mathbb{R}$ for all i. Since $\min \frac{1}{n}\sum_{i=1}^n f_i(T_i(x)) \geq \frac{1}{n}\sum_{i=1}^n \min f_i(T_i(x)) = \frac{1}{n}\sum_{i=1}^n \min f_i(x)$ and each f_i is L_i-smooth, we get

$$
\begin{aligned}
\tilde{f}(x^0) - \tilde{f}(x^*) &\leq \tilde{f}(x^0) - \frac{1}{n}\sum_{i=1}^n f_i(x_i) \\
&= \frac{1}{n}\sum_{i=1}^n f_i(\alpha x^0 + (1-\alpha)x_i) - f(x_i) \\
&\overset{(9)}{\leq} \frac{1}{n}\sum_{i=1}^n L_i\|\alpha(x^0 - x_i)\|^2 \\
&= \alpha^2\overline{L_i\|x^0 - x_i\|^2}.
\end{aligned}
$$

This observation shows that the dependence between $\tilde{f}(x^0) - \tilde{f}(x^*)$ and α is quadratic. $\tilde{f}(x^0) - \tilde{f}(x^*)$ diminishes to zero as α goes to zero what proves our intuition: the closer α is to zero, the less steps the algorithm needs to make till convergence.

We verify that the stepsize in the corollary fits the restriction on stepsizes in Theorem 5.

$$
\begin{aligned}
\frac{1}{\gamma} &= 4\left(L_\alpha + \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}\right) \\
&\overset{\alpha_i\equiv\alpha,\ \text{Proposition 2}}{=} 4\left(\overline{L_i} + \frac{2\max\{L_i\omega_i\}}{n}\right)\cdot\alpha^2.
\end{aligned} \tag{55}
$$

Further assuming that $\inf\limits_{x^*\in\mathcal{X}} \|x^0 - x^*\|$ is bounded above for all $\alpha \in [0,1]$ by value $R \in \mathbb{R}$, we obtain the following result:

$$
\frac{1}{\gamma}\|x^0 - x^*\|^2 \leq 4\left(\overline{L_i} + \frac{2\max\{L_i\omega_i\}}{n}\right)R^2\alpha^2.
$$

The neighborhood governing term is $\gamma\sigma_{DCGD}$. Let us assume $\alpha_i \equiv \alpha$ and $\omega_i \equiv \omega$ for all i.

$$
\begin{aligned}
\sigma_{DCGD} &= \frac{2}{n^2}\sum_{i=1}^n \omega_i\alpha_i^2\|\nabla f_i(T_i(x^*))\|^2 \\
&\overset{T_i(x)\ \text{def}}{=} \frac{2}{n^2}\sum_{i=1}^n \omega_i\alpha_i^2\|\nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i)\|^2 \\
&\overset{\alpha_i\equiv\alpha,\ \omega_i\equiv\omega}{=} \frac{2\omega\alpha^2}{n^2}\sum_{i=1}^n \|\nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i)\|^2
\end{aligned}
$$

Since x_i^* minimizes $f_i(x)$, $\nabla f_i(x_i^*) = 0$. Thus,

$$\frac{2\omega\alpha^2}{n^2}\sum_{i=1}^n \|\nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i)\|^2$$
$$= \frac{2\omega\alpha^2}{n^2}\sum_{i=1}^n \|\nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i) - \nabla f_i(x_i)\|^2$$

Inequality (14) further bounds each norm of gradient differences.

$$\frac{2\omega\alpha^2}{n^2}\sum_{i=1}^n \|\nabla f_i(\alpha_i x^* + (1-\alpha_i)x_i) - \nabla f_i(x_i)\|^2$$
$$\leq \frac{4\omega\alpha^2}{n^2}\sum_{i=1}^n L_i(f_i(\alpha_i x^* + (1-\alpha_i)x_i) - f_i(x_i))$$
$$\leq \frac{4\omega\alpha^2 \max_i L_i}{n^2}\sum_{i=1}^n (f_i(\alpha_i x^* + (1-\alpha_i)x_i) - f_i(x_i))$$
$$= \frac{4\omega\alpha^2 \max_i L_i}{n}\left(\tilde{f}(x^*) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)$$

In the rightmost term point x^* depends on α because it is an optimal point of a problem dependent on α. But the observation below explicitly reveals the dependence. Since each f_i is convex,

$$\tilde{f}(x) = \frac{1}{n}\sum_{i=1}^n f_i(\alpha x + (1-\alpha)x_i)$$
$$\leq \alpha\frac{1}{n}\sum_{i=1}^n f_i(x) + (1-\alpha)\frac{1}{n}\sum_{i=1}^n f_i(x_i).$$

Taking the minimum of both sides of inequality above we get to

$$\tilde{f}(x^*) \leq \alpha\tilde{f}(x^*(0)) + (1-\alpha)\frac{1}{n}\sum_{i=1}^n f_i(x_i).$$

We established that

$$\sigma_{DCGD} \leq \frac{4\omega \max_i L_i}{n}\left(\tilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)\alpha^3. \tag{56}$$

Together with equality 55 it brings us to the following bound on the neighbourhood:

$$\gamma\sigma_{DCGD} \leq \frac{\omega \max_i L_i}{n\overline{L_i} + 2\omega \max_i L_i}\left(\tilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)\alpha = C\alpha,$$

where C does not depend on α.

□

7.2.2 DIANA

As for the previous section, we first introduce the algorithm, followed by general theorem, which is then applied to (FedMix).

Theorem 6. *Assume all conditions of Lemmas 3 and 4 hold. Let* $0 < \gamma \leq \frac{1}{4\left(L_\alpha + \frac{2\max_i\{L_i\alpha_i^2\omega_i\}}{n} + \frac{4\max_i\{\beta_i\omega_i L_i\alpha_i^2\}}{n\min_i \beta_i}\right)}$. *Then,*

$$\mathbb{E}[\tilde{f}(\overline{x}^k) - \tilde{f}(x^*)] \leq \left(2(\tilde{f}(x^0) - \tilde{f}(x^*)) + \frac{4\gamma}{n\min\beta_i}\sigma_0^2 + \frac{1}{\gamma}\|x^0 - x^*\|^2\right)\frac{1}{k}. \tag{57}$$

Proof. The theorem is a corollary of Corollary 4.1 from Khaled et al. (2020) with constants

$$A = L_\alpha + \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}, B = \frac{2}{n}, D_1 = 0,$$
$$\rho = \min_i \beta_i, C = \max_i\{\beta_i\omega_i L_i\alpha_i^2\},\ D_2 = 0.$$

□

Corollary 4. *Assume all conditions of Lemmas 3 and 4 hold. Let $\alpha_i = \alpha$, $\omega_i = \omega$, $h_i^0 = 0 \in \mathbb{R}^d$ for all i, $\gamma = \frac{1}{4\left(\overline{L_i}+\frac{6\omega\max_i L_i}{n}\right)\alpha^2}$, $\sup_{\alpha\in[0,1]}\inf_{x^*\in\mathcal{X}}\|x^0 - x^*\| = R$. Then,*

$$\mathbb{E}[f(\overline{x}^k) - f(x^*)] \le \frac{1}{k}(C_1\alpha^2 + C_2\alpha),$$

where $C_1 = 2\left(\overline{L_i\|x^0 - x_i\|^2} + 2\left(\overline{L_i} + \frac{6\omega\max_i L_i}{n}\right)R^2\right)$ *and* $C_2 = \frac{\omega(\omega+1)\max_i L_i}{(n\overline{L_i}+6\omega\max_i L_i)}\left(\tilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)$.

Proof. Let us investigate stepsize γ. The statement of the corrolary requires the stepsize equals its maximum value $\frac{1}{4\left(L_\alpha+\frac{2\max_i\{L_i\alpha_i^2\omega_i\}}{n}+\frac{4\max_i\{\beta_i\omega_i L_i\alpha_i^2\}}{n\min_i\beta_i}\right)}$. Since $\alpha_i = \alpha$, $\omega_i = \omega$, $\beta_i = \frac{1}{1+\omega}$ for all i,

$$\gamma = \frac{1}{4\left(L_\alpha + \frac{2\max_i\{L_i\alpha_i^2\omega_i\}}{n} + \frac{4\max_i\{\beta_i\omega_i L_i\alpha_i^2\}}{n\min_i\beta_i}\right)} = \frac{1}{4\left(\overline{L_i} + \frac{6\omega\max_i L_i}{n}\right)\alpha^2}. \tag{58}$$

From definitions it follows that $\sigma_0^2 = \frac{n}{2}\sigma_{DCGD}$ if $h_i^0 = 0 \in \mathbb{R}^d$. Thus,

$$\sigma_0^2 = \frac{n}{2}\sigma_{DCGD} \overset{(56)}{\le} 2\omega\max_i L_i\left(\tilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)\alpha^3.$$

That is why

$$\frac{4\gamma}{n\min\beta_i}\sigma_0^2 \overset{\beta_i=\frac{1}{\omega}+1}{=} \frac{4\gamma(\omega+1)}{n}\sigma_0^2 \overset{(58)}{=} \frac{\omega+1}{n\left(\overline{L_i} + \frac{6\omega\max_i L_i}{n}\right)\alpha^2}\sigma_0^2$$
$$= \frac{\omega+1}{(n\overline{L_i} + 6\omega\max_i L_i)\alpha^2}\sigma_0^2 \le \frac{2\omega(\omega+1)\max_i L_i}{(n\overline{L_i} + 6\omega\max_i L_i)}\left(\tilde{f}(x^*(0)) - \frac{1}{n}\sum_{i=1}^n f_i(x_i)\right)\alpha.$$

Remaining terms are analyzed as for DCGD in convex case. □

7.3 Nonconvex objectives

In this section we analyze algorithms in the nonconvex case of functions. Throughout this section we assume that x_i^* is a stationary point of function f_i. For the sake of convenience we transfer Lemma 1 from Khaled and Richtárik (2020) to here.

Lemma 5 (Lemma 1 from Khaled and Richtárik (2020))**.** *Suppose $f_i^* = \min f_i(x)$ exists and f_i is L_i-smooth. Then, for any $x \in \mathbb{R}^d$ it holds that*

$$\|\nabla f_i(x)\|^2 \le 2L_i(f_i(x) - f_i^*).$$

Throughout this section we will use notation $f^* \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^n f_i^*$.

7.3.1 DCGD

Theorem 7. *Suppose $f_i^* = \min f_i(x)$ exists and f_i is L_i-smooth for each i. Then, if $\gamma \le \frac{1}{L_\alpha}$,*

$$\min_{0\le t\le k-1}\mathbb{E}\|\nabla\tilde{f}(x^t)\|^2 \le \frac{2\left(1 + \frac{2L_\alpha\gamma^2\max\{L_i\omega_i\alpha_i^2\}}{n}\right)^k}{\gamma k}(\tilde{f}(x^0) - f^\star). \tag{59}$$

Proof. The proof is a direct application of Theorem 2 in Khaled and Richtárik (2020) to our setting. First, let us analyze $\mathbb{E}\|g(x)\|^2$.

$$
\begin{aligned}
\mathbb{E}\|g(x)\|^2 &= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))])\right\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))]) - \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))])\right]\right\|^2 + \left\|\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))])\right]\right\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))]) - \frac{1}{n}\sum_{i=1}^n \nabla[f_i(T_i(x))]\right\|^2 + \left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i(T_i(x))\right\|^2 \\
&= \mathbb{E}\left\|\frac{1}{n}\sum_{i=1}^n \mathcal{C}_i(\nabla[f_i(T_i(x))]) - \frac{1}{n}\sum_{i=1}^n \nabla[f_i(T_i(x))]\right\|^2 + \left\|\nabla \tilde{f}(x)\right\|^2 \\
&= \frac{1}{n^2}\sum_{i=1}^n \|\mathcal{C}_i(\nabla[f_i(T_i(x))]) - \nabla[f_i(T_i(x))]\|^2 + \left\|\nabla \tilde{f}(x)\right\|^2 \\
&\leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x))]\|^2 + \left\|\nabla \tilde{f}(x)\right\|^2.
\end{aligned}
$$

To analyze the first term we refer ourselves to Lemma 5:

$$
\begin{aligned}
\|\nabla[f_i(T_i(x))]\|^2 &\overset{T_i \text{ def}}{=} \|\nabla[f_i(\alpha_i x + (1-\alpha_i)x_i)]\|^2 = \alpha_i^2 \|\nabla f_i(\alpha_i x + (1-\alpha_i)x_i)\|^2 \\
&\overset{\text{Lemma 5}}{\leq} \alpha_i^2 2L_i(f_i(\alpha_i x + (1-\alpha_i)x_i) - f_i^*)
\end{aligned}
$$

Plugging this in the previous inequality we get

$$
\begin{aligned}
\mathbb{E}\|g(x)\|^2 &\leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \|\nabla[f_i(T_i(x))]\|^2 + \left\|\nabla \tilde{f}(x)\right\|^2 \\
&\leq \frac{1}{n^2}\sum_{i=1}^n \omega_i \alpha_i^2 2L_i(f_i(\alpha_i x + (1-\alpha_i)x_i) - f_i^*) + \left\|\nabla \tilde{f}(x)\right\|^2 \\
&\leq \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}\left(\frac{1}{n}\sum_{i=1}^n f_i(\alpha_i x + (1-\alpha_i)x_i) - \frac{1}{n}\sum_{i=1}^n f_i^*\right) + \|\nabla \tilde{f}(x)\|^2 \\
&= \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}(\tilde{f}(x) - f^*) + \|\nabla \tilde{f}(x)\|^2,
\end{aligned}
$$

what means that Assumption 2 from Khaled and Richtárik (2020) holds with $A = \frac{2\max\{L_i\omega_i\alpha_i^2\}}{n}$, $B = 1$, $C = 0$. Then, as Theorem 2 from the same paper states, for stepsize $\gamma \leq \frac{1}{L_\alpha}$ it holds that

$$
\min_{0\leq t\leq k-1} \mathbb{E}\|\nabla \tilde{f}(x^t)\|^2 \leq \frac{2\left(1 + \frac{2L_\alpha\gamma^2\max\{L_i\omega_i\alpha_i^2\}}{n}\right)^k}{\gamma k}(\tilde{f}(x^0) - f^\star).
$$

□

Corollary 5. *Suppose assumptions of Theorem 7 hold, $\alpha_i \equiv \alpha$, $\omega_i \equiv \omega$. Let $\gamma = \gamma_0 \frac{1}{\alpha^2} \leq \frac{1}{\overline{L}_i\alpha^2}$ and $\Delta_0 = \sup_\alpha \tilde{f}(x^0) - f^*$.*

$$
\min_{0\leq t\leq k-1} \mathbb{E}\|\nabla \tilde{f}(x^t)\|^2 \leq \alpha^2 \frac{2\left(1 + \frac{2\overline{L}\omega\gamma_0^2\max_i L_i}{n}\right)^k}{\gamma_0 k}\Delta_0. \tag{60}
$$

Proof. We note that when $\alpha_i \equiv \alpha$, $L_\alpha = \overline{L}\alpha^2$, which follows from the definition of L_α. Plugging equations $L_\alpha = \overline{L}\alpha^2$, $\gamma = \gamma_0 \frac{1}{\alpha^2}$, $\alpha_i \equiv \alpha$ and $\omega_i \equiv \omega$ into the main result of Theorem 7 we get

$$\min_{0\le t\le k-1} \mathbb{E}\|\nabla \tilde{f}(x^t)\|^2 \le \alpha^2 \frac{2\left(1+\frac{2\overline{L}\omega\gamma_0^2 \max_i L_i}{n}\right)^k}{\gamma_0 k}(\tilde{f}(x^0)-f^\star).$$

Noting that $\tilde{f}(x^0) - f^* \le \Delta_0$ we finish the proof. □

7.3.2 DIANA

Theorem 8. *Suppose $f_i^* = \min f_i(x)$ exists and f_i is L_i-smooth for each i. Let $\beta_k \equiv \beta$ and $\omega_k \equiv \omega$ for all $k \in \{1,\dots,n\}$. Suppose $\beta \in \frac{1}{1+\omega}\left[1, \min\left\{1+\frac{1}{2(1+\omega)}, \sqrt{\frac{3}{2}+\omega+\frac{1}{2\omega}}\right\}\right]$ and $\gamma \le \frac{1}{L_\alpha}\cdot\min\left\{\frac{1}{2\sqrt{\eta_0}}, \frac{2}{\sqrt{1+8\eta_0}+1}\right\}$, $\eta_0 = \frac{(1+\omega)\omega(1+2(1+\omega))}{n}$, and $h_i^0 = \nabla[f_i(T_i(x^0))]$. Let $\hat{x}$ be a point chosen uniformly at random among iterates $x^0, x^1, \dots, x^{k-1}$ generated by DIANA. Then,*

$$\mathbb{E}\|\nabla \tilde{f}(\hat{x})\|^2 \le \frac{2}{\gamma k}(\tilde{f}(x^0) - f^*). \tag{61}$$

Proof. The proof is following the proof of Theorem 1 in Li and Richtárik (2020) with small modifications: we rewrite the main proposition in the different way and give slightly different bound on the stepsize. First, Lemma 9 in Li and Richtárik (2020) says that

$$\mathbb{E}\|g^k\|^2 \le \|\nabla f(x^k)\|^2 + \frac{1+\omega}{n}\sigma_k^2, \tag{62}$$

$$\mathbb{E}\sigma_{k+1}^2 \le (1-\rho)\sigma_k^2 + \frac{\omega(1+r)L_\alpha^2\gamma^2}{1+\omega}\|\nabla f(x)\|^2, \tag{63}$$

where

$$\sigma_k^2 = \frac{\omega}{(1+\omega)n}\sum_{i=1}^n \|\nabla[f_i(T_i(x^k))] - h_i^k\|^2 \tag{64}$$

$$\rho = \theta - \frac{\omega(1+r)}{n}L_\alpha^2\gamma^2 \tag{65}$$

$$\theta = \min\{1-\beta^2\omega, 2\beta - \frac{1-\beta}{r} - \beta^2 - \beta^2\omega\} \tag{66}$$

and r is an arbitrary positive number. We choose $r = 2(1+w) > 0$. For the sake of readability, we define function ζ in the following way

$$2\beta - \frac{1-\beta}{2(1+\omega)} - \beta^2 - \beta^2\omega = -(\omega+1)\beta^2 + \left(2+\frac{1}{2(1+\omega)}\right)\beta - \frac{1}{2(1+\omega)} =: \zeta(\beta)$$

That is why we can write that $\theta \in \min\{1-\beta^2\omega, \zeta(\beta)\}$.

Second, let us analyze the lower bound for θ. Note that for β lying in the range $\frac{1}{1+\omega}\left[1, \min\left\{1+\frac{1}{2(1+\omega)}, \sqrt{\frac{3}{2}+\omega+\frac{1}{2\omega}}\right\}\right]$ we have

$$\begin{aligned}1-\beta^2\omega &\ge 1 - \frac{\omega}{(1+\omega)^2}\left(\frac{3}{2}+\omega+\frac{1}{2\omega}\right) = 1 - \frac{\omega}{(1+\omega)^2}\frac{(1+2\omega)(1+\omega)}{2\omega}\\ &= 1 - \frac{1+2\omega}{2(1+\omega)} = \frac{1}{2(1+\omega)},\end{aligned}$$

and

$$\begin{aligned}\zeta\left(\frac{1}{1+\omega}\right) &= -\frac{1}{1+\omega} + \left(2+\frac{1}{2(1+\omega)}\right)\frac{1}{1+\omega} - \frac{1}{2(1+\omega)}\\ &= \frac{1}{2(1+\omega)} + \frac{1}{2(1+\omega)^2} \ge \frac{1}{2(1+\omega)},\end{aligned}$$

and

$$\zeta\left(\frac{1}{1+\omega}+\frac{1}{2(1+\omega)^2}\right)=-\frac{1}{1+\omega}-\frac{1}{(1+\omega)^2}-\frac{1}{4(1+\omega)^3}$$
$$+\left(2+\frac{1}{2(1+\omega)}\right)\left(\frac{1}{1+\omega}+\frac{1}{2(1+\omega)^2}\right)-\frac{1}{2(1+\omega)}$$
$$=\frac{1}{2(1+\omega)}+\frac{1}{2(1+\omega)^2}\geq\frac{1}{2(1+\omega)}.$$

Since ζ is a quadratic function with the ends of the parabola pointed downwards, for all β in the range we have $\zeta(\beta)\geq\frac{1}{2(1+\omega)}$, which altogether means that $\theta\geq\frac{1}{2(1+\omega)}>0$.

If we enforce

$$\frac{\omega(1+r)}{n}L_\alpha^2\gamma^2\leq\frac{1}{4(1+\omega)}$$

or

$$\gamma\leq\frac{1}{2L_\alpha}\sqrt{\frac{n}{(1+\omega)\omega(1+r)}}=\frac{1}{2L_\alpha}\sqrt{\frac{n}{(1+\omega)\omega(1+2(1+\omega))}},$$

then $\rho=\theta-\frac{\omega(1+r)}{n}L_\alpha^2\gamma^2\geq\frac{1}{4(1+\omega)}$.

Now let us switch to the proof of the convergence. We first note that due to smoothness of f_is and inequality 62 we get

$$\mathbb{E}f(x^{k+1})\leq f(x^k)+\mathbb{E}\langle\nabla f(x^k),x^{k+1}-x^k\rangle+\frac{L_\alpha}{2}\mathbb{E}\|x^{k+1}-x^k\|^2$$
$$=f(x^k)-\gamma\|\nabla f(x^k)\|^2+\frac{L_\alpha\gamma^2}{2}\mathbb{E}\|g^k\|^2$$
$$\leq f(x^k)-\left(\gamma-\frac{L_\alpha\gamma^2}{2}\right)\|\nabla f(x^k)\|^2+\frac{L_\alpha\gamma^2}{2}\frac{1+\omega}{n}\sigma_k^2.$$

Let us fix $\xi>0$. Then according to 63 we have

$$\mathbb{E}[f(x^{k+1})-f^*+\xi\sigma_{k+1}^2]$$
$$\leq f(x^k)-f^*-\left(\gamma-\frac{L_\alpha\gamma^2}{2}-\xi\frac{\omega(1+r)L_\alpha^2\gamma^2}{1+\omega}\right)\|\nabla f(x^k)\|^2+\left(\xi(1-\rho)+\frac{L_\alpha\gamma^2}{2}\frac{1+\omega}{n}\right)\sigma_k^2.$$

Let us notate $\Delta^k=f(x^k)-f^*+\xi\sigma_k^2$ and set $\xi=\frac{L_\alpha\gamma^2}{2\rho}\frac{1+\omega}{n}$. Then from previous inequality it follows that

$$\mathbb{E}\Delta^{k+1}\leq\Delta^k-\left(\gamma-\frac{L_\alpha\gamma^2}{2}-\xi\frac{\omega(1+r)L_\alpha^2\gamma^2}{1+\omega}\right)\|\nabla f(x^k)\|^2. \tag{67}$$

Let us define γ' the coefficient in front of $\|\nabla f(x^k)\|^2$ in the last inequality. When is γ' larger than $\frac{\gamma}{2}$?

$$\gamma-\frac{L_\alpha\gamma^2}{2}-\xi\frac{\omega(1+r)L_\alpha^2\gamma^2}{1+\omega}=\gamma-\frac{L_\alpha\gamma^2}{2}-\frac{L_\alpha\gamma^2}{2\rho}\frac{1+\omega}{n}\frac{\omega(1+r)L_\alpha^2\gamma^2}{1+\omega}$$
$$=\gamma-\frac{L_\alpha\gamma^2}{2}-\frac{L_\alpha\gamma^2}{2\rho}\frac{\omega(1+r)L_\alpha^2\gamma^2}{n}=\gamma-\frac{\gamma}{2}\left(L_\alpha\gamma+\frac{L_\alpha^3\gamma^3\omega(1+r)}{\rho n}\right)\geq\frac{\gamma}{2},$$

if

$$L_\alpha\gamma+\frac{L_\alpha^3\gamma^3\omega(1+r)}{\rho n}\leq 1.$$

Let us define $y := L_\alpha \gamma$ and $a := \frac{\omega(1+r)}{n}$. Then according to 65 $\rho = \theta - ay^2$. Plugging this back to the last inequality we get

$$
\begin{aligned}
& y + \frac{ay^3}{\theta - ay^2} \leq 1 \\
& \Leftrightarrow \frac{y\theta - ay^3 + ay^3}{\theta - ay^2} \leq 1 \\
& \Leftrightarrow \frac{y\theta}{\theta - ay^2} \leq 1 \\
& \overset{\theta - ay^2 = \rho > 0}{\Leftrightarrow} y\theta \leq \theta - ay^2 \\
& \overset{\theta > 0}{\Leftrightarrow} \frac{a}{\theta} y^2 + y - 1 \leq 0.
\end{aligned}
$$

The last inequality holds for all y lying between zero and the positive root of the quadratic equation. The positive root is $\frac{\sqrt{1+\frac{4a}{\theta}}-1}{\frac{2a}{\theta}} = \frac{\frac{4a}{\theta}}{\frac{2a}{\theta}\left(\sqrt{1+\frac{4a}{\theta}}+1\right)} = \frac{2}{\sqrt{1+\frac{4a}{\theta}}+1}$. That means that the last inequality holds if

$$
y \leq \frac{2}{\sqrt{1 + \frac{4a}{\theta}} + 1}
$$

or the tightest bound for γ achieved on smallest θ gives

$$
\gamma \leq \frac{1}{L_\alpha} \frac{2}{\sqrt{1 + \frac{4\omega(1+2(1+\omega))}{n\theta_{\min}}} + 1} = \frac{1}{L_\alpha} \frac{2}{\sqrt{1 + \frac{8(1+\omega)\omega(1+2(1+\omega))}{n}} + 1}.
$$

Going back to 67 we finally get

$$
\begin{aligned}
& \mathbb{E}\Delta^{k+1} \leq \Delta^k - \frac{\gamma}{2}\|\nabla f(x^k)\|^2 \\
& \mathbb{E}\|\nabla f(x^k)\|^2 \leq \frac{2}{\gamma}(\mathbb{E}\Delta^k - \mathbb{E}\Delta^{k+1}).
\end{aligned}
$$

That is why

$$
\mathbb{E}\|\nabla f(\hat{x})\|^2 = \frac{1}{k}\sum_{t=0}^{k-1} \mathbb{E}\|\nabla f(x^t)\|^2 \leq \frac{2}{\gamma k}(\Delta^0 - \mathbb{E}\Delta^k) \leq \frac{2}{\gamma k}\Delta^0.
$$

The proof is written for general function $f(x) = \frac{1}{n}\sum_{i=1}^n f_i(x)$. To apply the theorem for our setting, we replace f by $\tilde{f}$. □

Corollary 6. *Suppose all conditions of Theorem 8 hold and $\alpha_i \equiv \alpha$ for all i from $\{1, \ldots, n\}$. Let $\gamma = \frac{1}{\alpha^2}\gamma_0$, where $\gamma_0 = \frac{1}{L}\min\left\{\frac{1}{2\sqrt{\eta_0}}, \frac{2}{\sqrt{1+8\eta_0}+1}\right\}$. Let $\Delta_0 = \sup_\alpha \tilde{f}(x^0) - f^*$ Then,*

$$
\mathbb{E}\|\nabla \tilde{f}(\hat{x})\|^2 \leq \alpha^2 \frac{2}{\gamma_0 k}\Delta_0. \tag{68}
$$

Proof. According to the definition of $L_\alpha = \frac{1}{n}\sum_{i=1}^n L_i\alpha_i^2$, in the case of equal α_is we get $L_\alpha = \overline{L}\alpha^2$. It remains to notice that the stepsize in the corollary condition still satisfies the condition on the stepsize γ in Theorem 8 and $\Delta_0 \geq \tilde{f}(x) - f^*$. □

8 Discussion of other model mixture methods

Deng et al. (2020) propose that each client solves the local problem

$$\min_{v \in \mathbb{R}^d} f_i(\alpha_i v + (1 - \alpha_i) w^*),$$

where w^* is the minimizer of (ERM). This does not result in any personalization since the "personalized" solution on each node is a reparameterization of each local solution x_i. Furthermore, the convergence theory that Deng et al. (2020) develop does not recover the linear convergence of gradient descent at $\alpha_i = 1$.

Mansour et al. (2020) introduce a similar method, MAPPER, where they propose to solve

$$\min_{z, \alpha_i, y_i} \frac{1}{n} \sum_{i=1}^{n} f_i(\alpha_i y_i + (1 - \alpha_i) z).$$

Again, this objective is trivially minimized by setting $\alpha_i = 1$, $y_i = \min_{x \in \mathbb{R}^d} f_i(x)$, and $z = 0$ (i.e. with no personalization at all).

Zec et al. (2021) also introduce a similar formulation based on the mixture of experts framework, where they propose to first learn the minimizer x_* of (ERM), learn the optimal local models $x_1, x_2, \ldots, x_n$, and then learn a mixture of both the global and local models (i.e. the α_i) on each client. Unfortunately, this is also ill-defined, as $\alpha_i = 1$ will always perform best on the local training set, and hence if there is no additional data the optimization process cannot improve over the local minimizers $x_1, \ldots, x_n$.

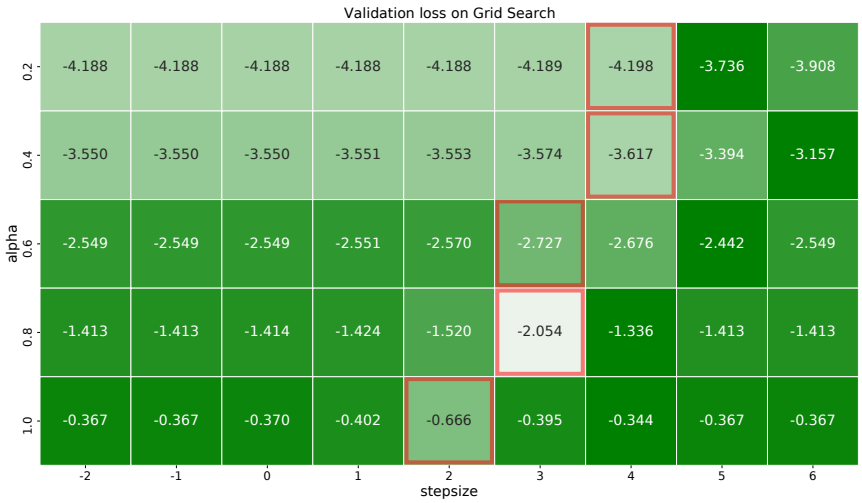


Figure 5: Grid-Search of stepsizes for FedMix for the first Stack Overflow experiment. Values in cells are logarithms of validation losses. Horizontal axis corresponds to logarithm of scaled step-sizes. Red frames correspond to minimum value in a row. Colors are scaled row-wise.

9 Experimental Details

Generalization experiment 1: Fitting Sine Functions. We train each local model i until the gradient norm $\|\nabla f_i(x)\|$ is below 10^{-2}. Stopping criteria for global models is the same but with respect to FedMix formulation: $\|\frac{1}{n}\sum_{i=1}^{n}\nabla f_i(T_i(x))\| < 10^{-2}$. Gradient descent with line search at each iteration was used as a local and a meta optimizer.

Generalization experiment 2: Comparison to FOMAML and Reptile. We preprocess the raw data in the same way as in Reddi et al. (2021) for Stackoverflow logistic regression task, i.e., each feature vector is a bag-of-words representation of a user's sentence, each label vector is a binary vector showing if a sentence relates to a particular question tag or not. Word vocabulary for the feature dataset is restricted to the 10000 most common words. We restrict the task to the 500 most used tags. For preprocessing, we used Tensorflow computational procedures from Xu et al. (2021).

To select 50 clients for the Figure 1, we map first 5000 clients from train dataset to vector space with BERT Devlin et al. (2019) and run k-means with 10 clusters. The first 50 clients from the first cluster have been selected for the experiment.

The hold-out validation dataset is of size 100, for the first experiment, and of size 110, for the second one. To compute pure local models we run gradient descent until the norm of the gradient is less than $[10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}]$, respectively. We train pure local models until the gradient norm is less than 10^{-4}, for the first experiments, and 10^{-5}, for the second, as this tolerance level was observed to have the lowest generalization error on the validation dataset in our experiments.

Then, we use a grid-search to find optimal stepsizes for gradient descent used for training FedMix. Our grid for step-sizes are $[5 \cdot 10^{-2}, 5 \cdot 10^{-1}, \ldots, 5 \cdot 10^{6}]$. The results are presented in Figures 5 and 6.

As table shows, the smaller α is, the higher step-size the task needs to achieve the best generalization, which is in line with our convergence results, see Theorem 2 and Table 1.

To train FOMAML and Reptile, we set the number of inner steps to five and grid search outer and inner loop step-sizes. The explored outer step-sizes are the same as for FedMix. Inner step-sizes iterate over the set $[10^{-3}, 10^{-2}, \ldots, 10]$.

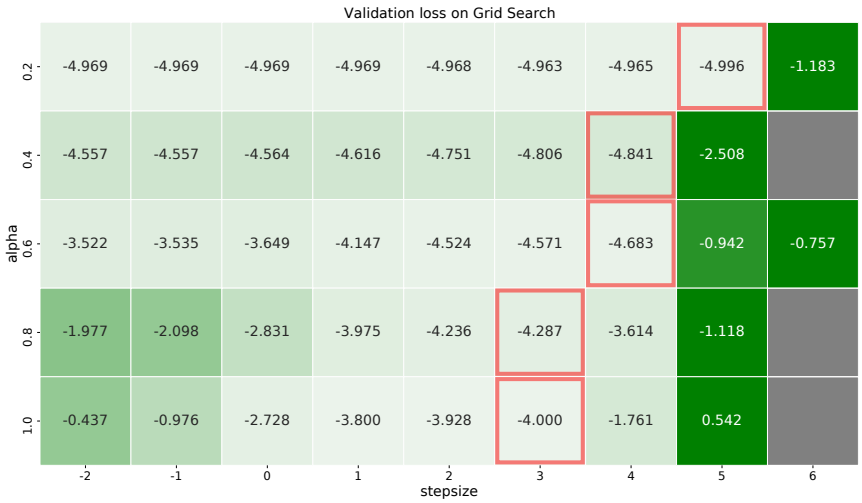


Figure 6: Grid-Search of stepsizes for FedMix for the seconds Stack Overflow experiment. Values in cells are logarithms of validation losses. Horizontal axis corresponds to logarithm of scaled step-sizes. Red frames correspond to minimum value in a row. Colors are scaled row-wise. Grey cells correspond to nan values.

After grid search, we run gradient descent for each value of alpha of FedMix, FOMAML, and Reptile for 10 000 and 50000 iterations in the first and seconds experiments accordigly and report obtained test accuracy.

10 Extra Experiments

Logistic regression with l_2 regularizer. On figures 7, 8, 9, and 10 we present the plots for regularized logistic regression loss for all datasets mentioned in the previous section and for all 3 algorithms: GD, CGD and DIANA. Number of machines n is set to 50 for DIANA and GD, and is set to 8 for CGD. Plots showing the dependence between loss and communication cost show that although in terms of communication rounds DIANA and CGD lose to classical GD, they are better with respect to communication cost, which is of more practical importance. We run all the algorithms with their best theoretical step-sizes.

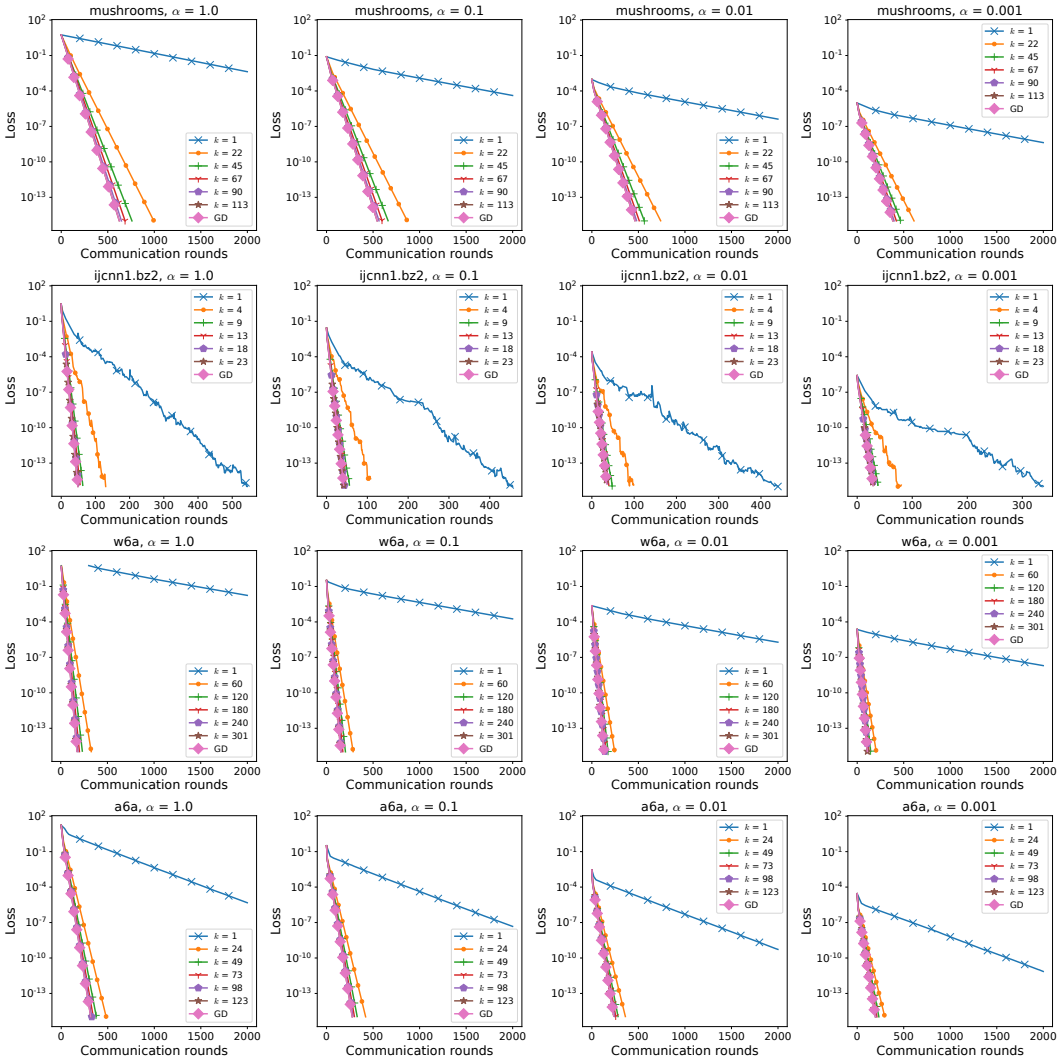


Figure 7: Loss $f(x) - f^*$ vs. # of communication rounds of DIANA and GD for logistic regression problem l_2 regularizer for four datasets, k is a sparsification parameter of Random-k compressor.

Unregularized logistic regression. Unregularized logistic regression task has got the following form $f_i(x) := \frac{1}{k_i}\sum_{j=1}^{k_i} \log\left(1 + \exp\left(-a_{i,j}^\top x\right)\right)$, which is well-known to be a convex problem. Similarly to regularized case, we present four plots (11, 12, 13, and 14) exhibiting convergence of three algorithms in terms of communication rounds and overall communication cost for four LIBSVM datasets. Number of machines n is set to 100 for DIANA and GD, and to 8 for CGD. We run all the algorithms with their best theoretical step-sizes.

Generalization experiment 1: Fitting Sin Functions. On figure 15 we present the rest of the plots of this subsection.

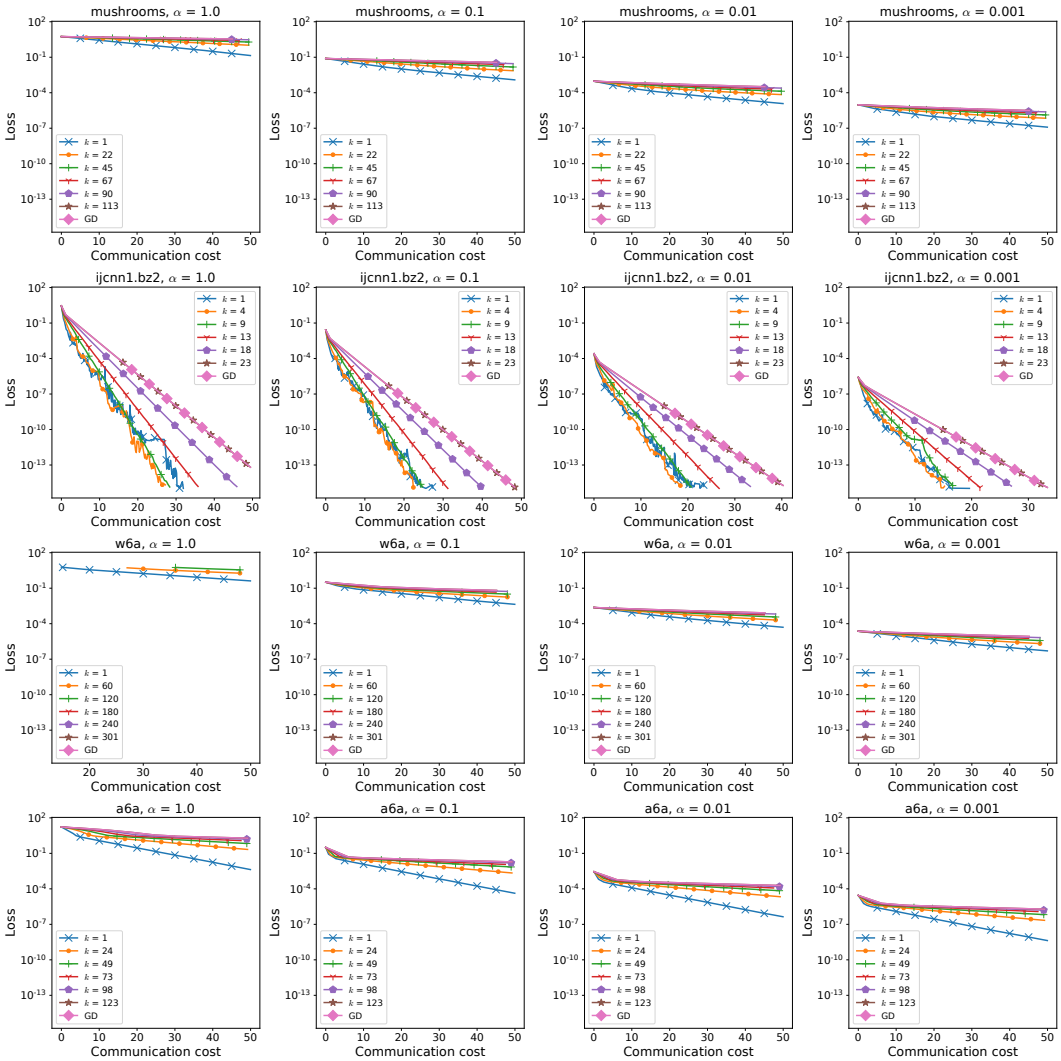


Figure 8: Loss $f(x) - f^*$ vs. communication cost in unit of thousands of float numbers of DIANA and GD for logistic regression problem l_2 regularizer for four datasets, k is a sparsification parameter of Random-k compressor.

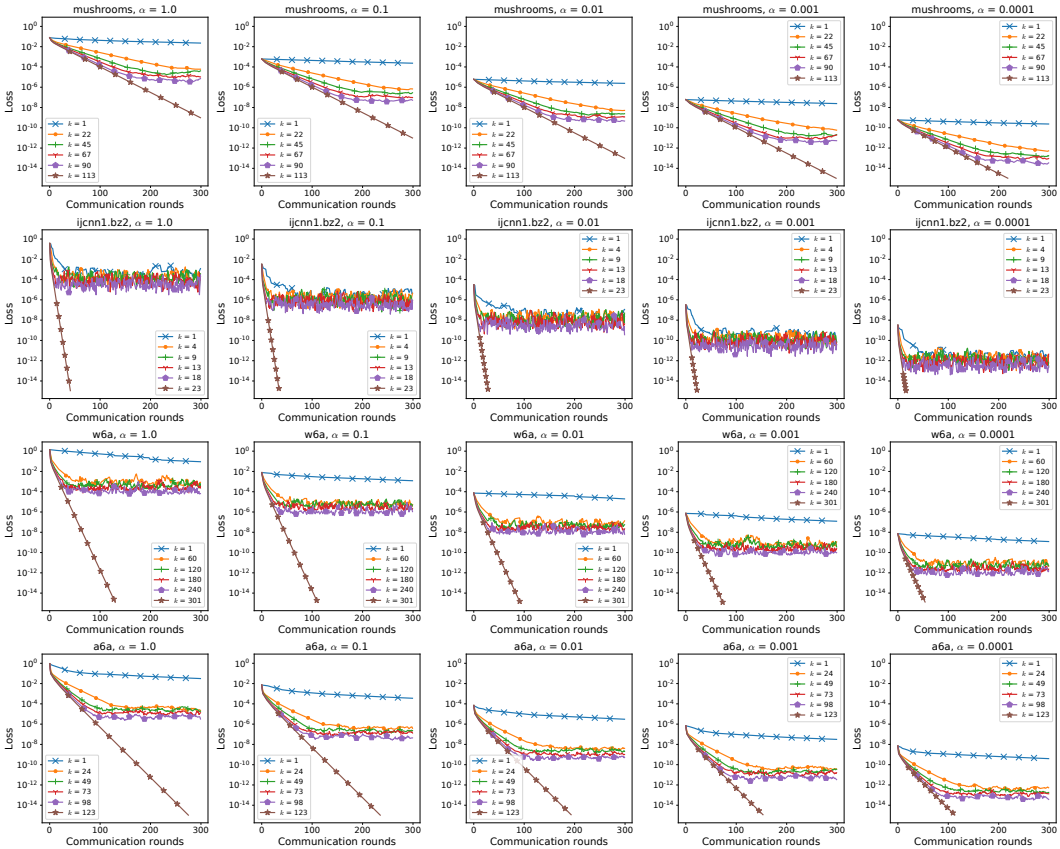


Figure 9: Loss $f(x) - f^*$ vs. # of communication rounds of CGD for logistic regression problem l_2 regularizer for four datasets, k is a sparsification parameter of Random-k compressor.

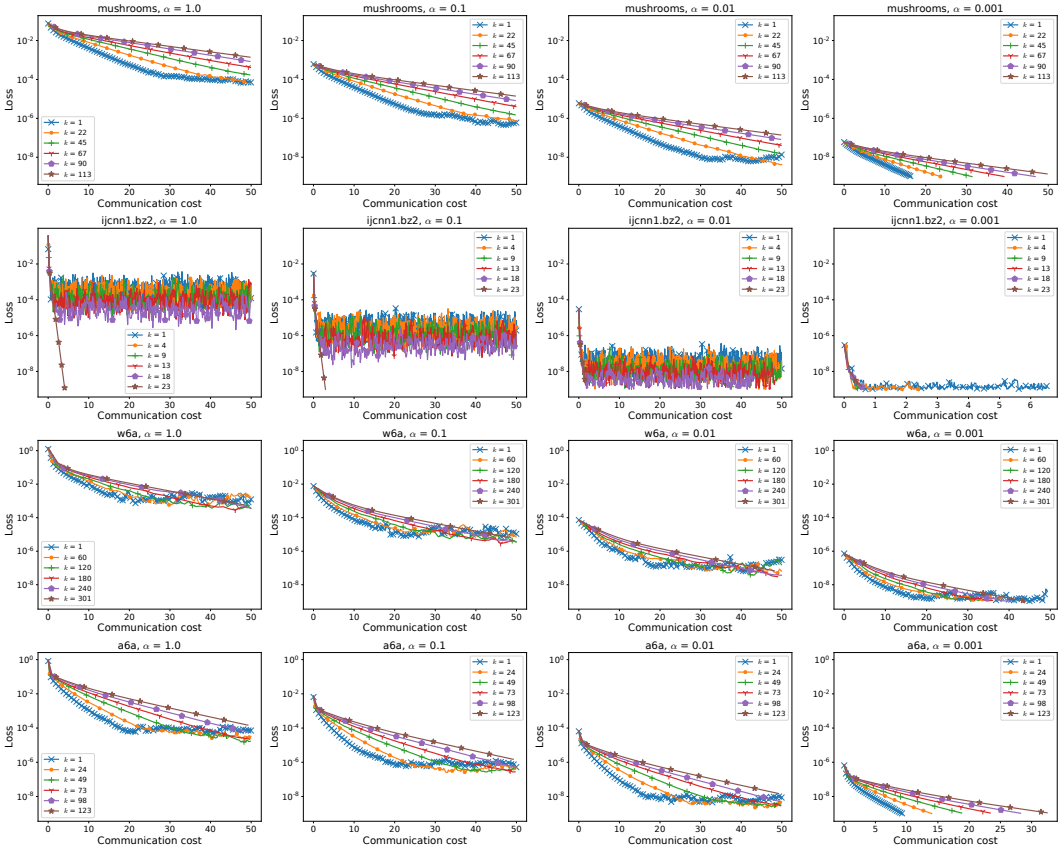


Figure 10: Loss $f(x) - f^*$ vs. communication cost in unit of thousands of float numbers of CGD for logistic regression problem l_2 regularizer for four datasets, k is a sparsification parameter of Random-k compressor.

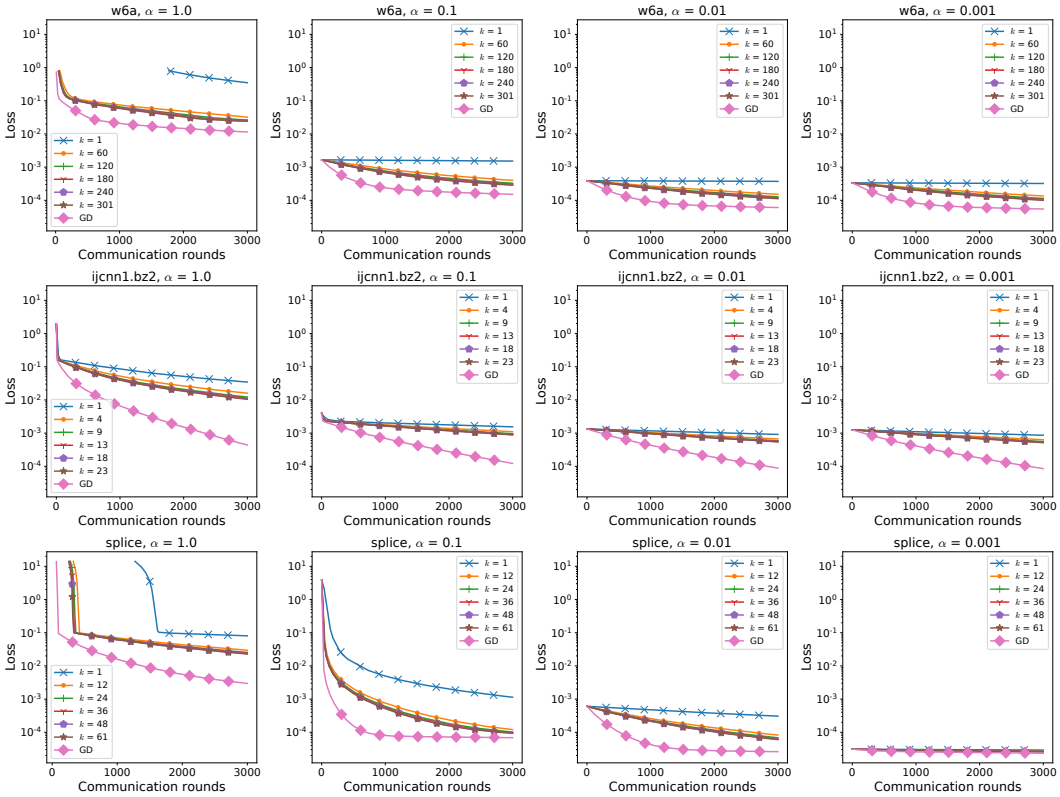


Figure 11: Loss $f(x) - f^*$ vs. # of communication rounds of DIANA and GD for unregularized logistic regression problem for four datasets, k is a sparsification parameter of Random-k compressor.

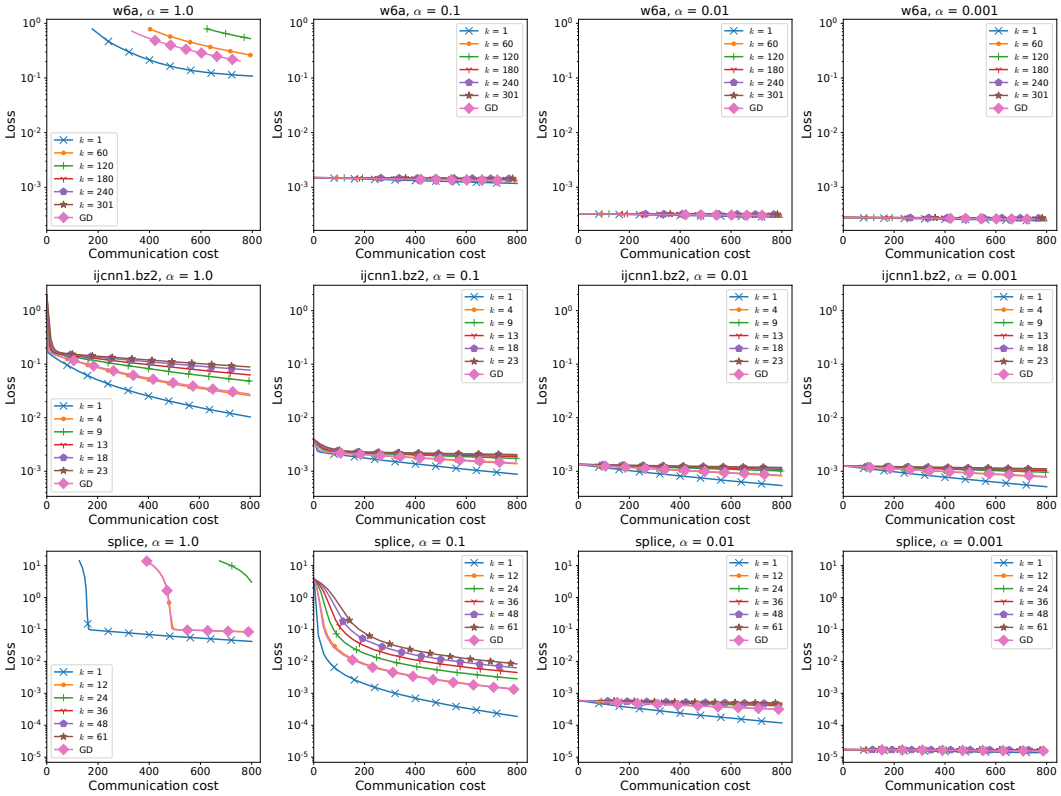


Figure 12: Loss $f(x) - f^*$ vs. communication cost in unit of thousands of float numbers of DIANA and GD for unregularized logistic regression problem for four datasets, k is a sparsification parameter of Random-k compressor.

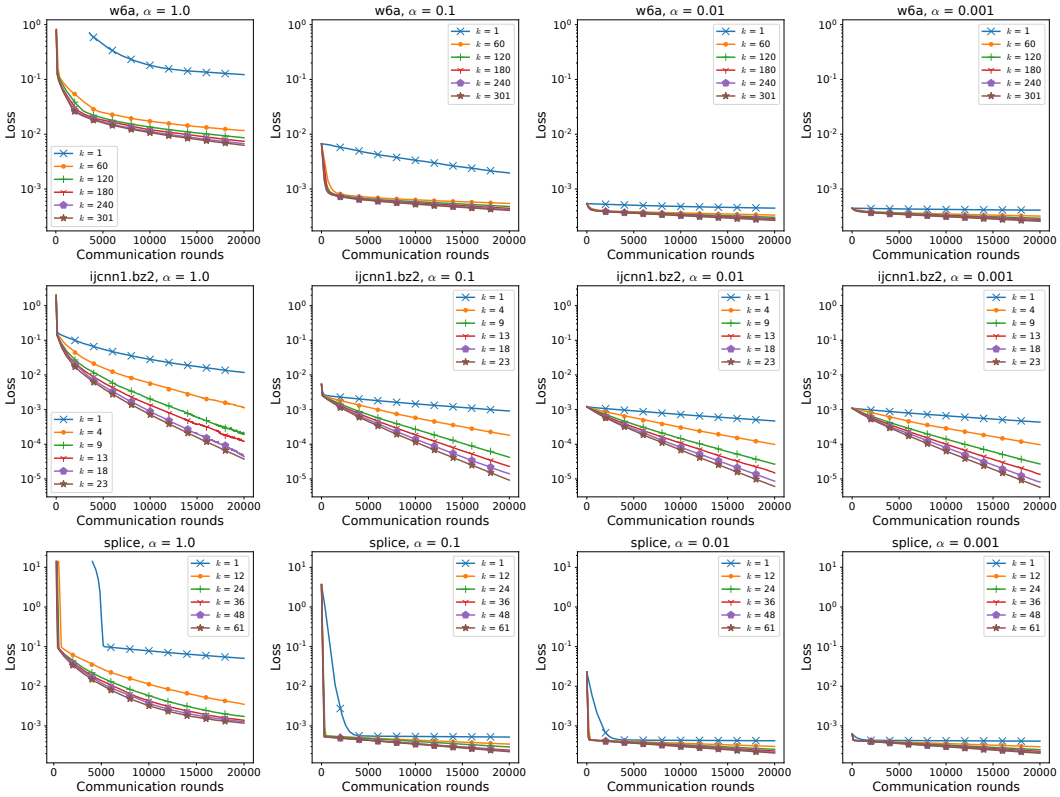


Figure 13: Loss $f(x) - f^*$ vs. # of communication rounds of CGD for unregularized logistic regression problem for four datasets, k is a sparsification parameter of Random-k compressor.

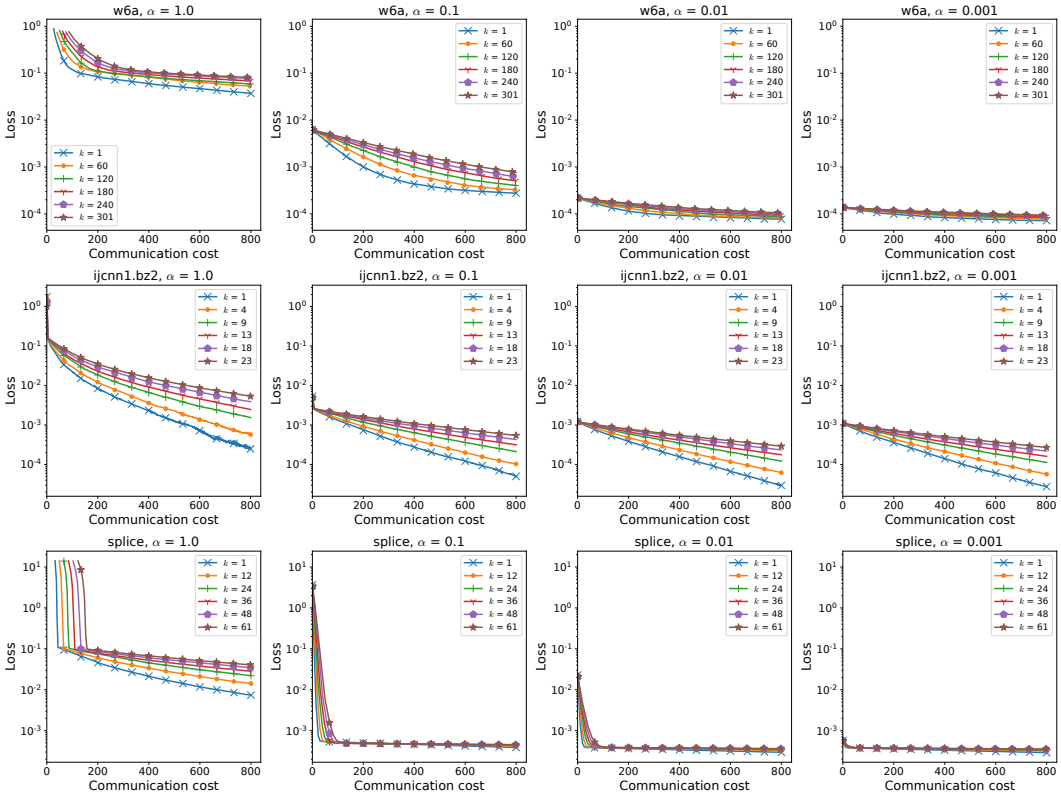


Figure 14: Loss $f(x) - f^*$ vs. communication cost in unit of thousands of float numbers of CGD for unregularized logistic regression problem for four datasets, k is a sparsification parameter of Random-k compressor.

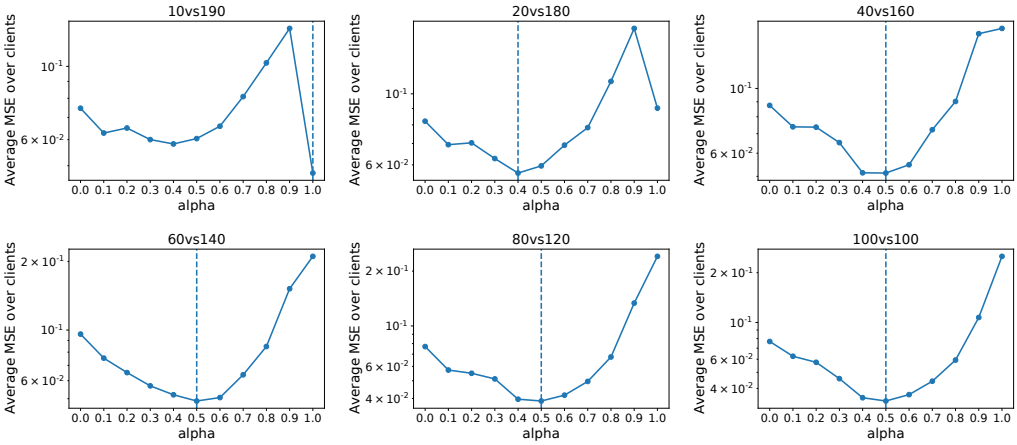


Figure 15: Average MSE vs. personalization parameter α for different splittings.

11 Limitations

Our current (FedMix) formulation is dependent on a computation of the local optimal solutions x_i's. Arguably, this is not well-defined beyond strongly convex models. Therefore, applying (FedMix) formulation to weakly convex or non-convex models might lead to some problems because of the existence of potentially several local optima and selection of x_i affects (FedMix) formulation. One possibility to remedy such issue is to use the ensemble of global and local models with the weights α_i and$(1 - \alpha_i)$, respectively, instead of the linear combination of x and x_i.

Another possible limitation is the availability of local data to compute an optimal local solution as the small number of local data points might lead to a noisy estimate and subsequently, it can potentially lead to bad generalization of (FedMix). To mitigate this limitation, we propose to use some clustering method to group several clients and compute x_i's on clusters rather than single clients.

Further investigation of both of these limitations is out of the scope of our current manuscript and we plan to investigate this in future work.