LoCoDL: Communication-Efficient Distributed Learning with Local Training and Compression

Anonymous Author(s)
Affiliation
Address
email

Abstract

In Distributed optimization and Learning, and even more in the modern framework of federated learning, communication, which is slow and costly, is critical. We introduce LoCoDL, a communication-efficient algorithm that leverages the two popular and effective techniques of Local training, which reduces the communication frequency, and Compression, in which short bitstreams are sent instead of full-dimensional vectors of floats. LoCoDL works with a large class of unbiased compressors that includes widely-used sparsification and quantization methods. LoCoDL provably benefits from local training and compression and enjoys a doubly-accelerated communication complexity, with respect to the condition number of the functions and the model dimension, in the general heterogenous regime with strongly convex functions. This is confirmed in practice, with LoCoDL outperforming existing algorithms.

1 Introduction

Performing distributed computations is now pervasive in all areas of science. Notably, Federated Learning (FL) consists in training machine learning models in a distributed and collaborative way (Konečný et al., 2016a,b; McMahan et al., 2017; Bonawitz et al., 2017). The key idea in this rapidly growing field is to exploit the wealth of information stored on distant devices, such as mobile phones or hospital workstations. The many challenges to face in FL include data privacy and robustness to adversarial attacks, but communication-efficiency is likely to be the most critical (Kairouz et al., 2021; Li et al., 2020a; Wang et al., 2021). Indeed, in contrast to the centralized setting in a datacenter, in FL the clients perform parallel computations but also communicate back and forth with a distant orchestrating server. Communication typically takes place over the internet or cell phone network, and can be slow, costly, and unreliable. It is the main bottleneck that currently prevents large-scale deployment of FL in mass-market applications.

Two strategies to reduce the communication burden have been popularized by the pressing needs of FL: 1) **Local Training (LT)**, which consists in reducing the communication frequency. That is, instead of communicating the output of every computation step involving a (stochastic) gradient call, several such steps are performed between successive communication rounds. 2) **Communication Compression (CC)**, in which compressed information is sent instead of full-dimensional vectors. We review the literature of LT and CC in Section 1.2.

We propose a new randomized algorithm named LoCoDL, which features LT and unbiased CC for communication-efficient FL and distributed optimization. It is variance-reduced (Hanzely & Richtárik, 2019; Gorbunov et al., 2020a; Gower et al., 2020), so that it converges to an exact solution. It provably benefits from the two mechanisms of LT and CC: the communication complexity is doubly accelerated, with a better dependency on the condition number of the functions and on the dimension of the model.

Submitted to 38th Conference on Neural Information Processing Systems (NeurIPS 2024). Do not distribute.

1.1 Problem and Motivation

We study distributed optimization problems of the form

$$\min_{x\in\mathbb{R}^d} \ \frac{1}{n}\sum_{i=1}^{n} f_i(x) + g(x), \tag{1}$$

where $d \geq 1$ is the model dimension and the functions $f_i : \mathbb{R}^d \to \mathbb{R}$ and $g : \mathbb{R}^d \to \mathbb{R}$ are *smooth*, so their gradients will be called. We consider the server-client model in which $n \geq 1$ clients do computations in parallel and communicate back and forth with a server. The private function f_i is owned by and stored on client $i \in [n] := \{1, \ldots, n\}$. Problem (1) models empirical risk minimization, of utmost importance in machine learning (Sra et al., 2011; Shalev-Shwartz & Ben-David, 2014). More generally, minimizing a sum of functions appears in virtually all areas of science and engineering. Our goal is to solve Problem (1) in a communication-efficient way, in the general **heterogeneous** setting in which the functions f_i, as well as g, can be *arbitrarily different*: we do not make any assumption on their similarity whatsoever.

We consider in this work the strongly convex setting — an analysis with nonconvex functions would certainly require very different proof techniques, which we currently do not know how to derive. That is, the following holds:

Assumption 1.1 (strongly convex functions)**.** The functions f_i and g are all L-smooth and μ-strongly convex, for some $0 < \mu \leq L$.[1] Then we denote by $x^\star$ the solution of the strongly convex problem (1), which exists and is unique. We define the condition number $\kappa := \frac{L}{\mu}$.

Problem (1) can be viewed as the minimization of the average of the n functions $(f_i + g)$, which can be performed using calls to $\nabla(f_i + g) = \nabla f_i + \nabla g$. We do not use this straightforward interpretation. Instead, let us illustrate the interest of having the **additional function** g in (1), using 4 different viewpoints. We stress that we can handle the case $g = 0$, as discussed in Section 3.1.

• Viewpoint 1: *regularization*. The function g can be a regularizer. For instance, if the functions f_i are convex, adding $g = \frac{\mu}{2}\|\cdot\|^2$ for a small $\mu > 0$ makes the problem μ-strongly convex.

• Viewpoint 2: *shared dataset*. The function g can model the cost of a common dataset, or a piece thereof, that is known to all clients.

• Viewpoint 3: *server-aided training*. The function g can model the cost of a core dataset, known only to the server, which makes calls to ∇g. This setting has been investigated in several works, with the idea that using a small auxiliary dataset representative of the global data distribution, the server can correct for the deviation induced by partial participation (Zhao et al., 2018; Yang et al., 2021, 2023). We do not focus on this setting, because we deal with the general heterogeneous setting in which g and the f_i are not meant to be similar in any sense, and in our work g is handled by the clients, not by the server.

• Viewpoint 4: *a new mathematical and algorithmic principle*. This is the idea that led to the construction of LoCoDL, and we detail it in Section 2.1.

In LoCoDL, the clients make all gradient calls; that is, Client i makes calls to ∇f_i and ∇g.

1.2 State of the Art

We review the latest developments on communication-efficient algorithms for distributed learning, making use of LT, CC, or both. Before that, we note that we should distinguish uplink, or clients-to-server, from downlink, or server-to-clients, communication. Uplink is usually slower than downlink communication, since uploading different messages in parallel to the server is slower than broadcasting the same message to an arbitrary number of clients. This can be due to cache memory and aggregation speed constraints of the server, as well as asymmetry of the service provider's systems or protocols used on the internet or cell phone network. In this work, we focus on the **uplink communication complexity**, which is the bottleneck in practice. Indeed, the goal is to

[1] A differentiable function $f : \mathbb{R}^d \to \mathbb{R}$ is said to be L-smooth if ∇f is L-Lipschitz continuous; that is, for every $x \in \mathbb{R}^d$ and $y \in \mathbb{R}^d$, $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ (the norm is the Euclidean norm throughout the paper). f is said to be μ-strongly convex if $f - \frac{\mu}{2}\|\cdot\|^2$ is convex.

exploit parallelism to obtain better performance when n increases. Precisely, with LoCoDL, the uplink communication complexity decreases from $\mathcal{O}\left(d\sqrt{\kappa}\log\epsilon^{-1}\right)$ when n is small to $\mathcal{O}\big(\sqrt{d}\sqrt{\kappa}\log\epsilon^{-1}\big)$ when n is large, where the condition number κ is defined in Assumption 1.1, see Corollary 3.2. Many works have considered bidirectional compression, which consists in compressing the messages sent both ways (Gorbunov et al., 2020b; Philippenko & Dieuleveut, 2020; Liu et al., 2020; Philippenko & Dieuleveut, 2021; Condat & Richtárik, 2022; Gruntkowska et al., 2023; Tyurin & Richtárik, 2023b) but to the best of our knowledge, this has no impact on the downlink complexity, which cannot be reduced further than $\mathcal{O}\left(d\sqrt{\kappa}\log\epsilon^{-1}\right)$, just because there is no parallelism to exploit in this direction. Thus, we focus our analysis on theoretical and algorithmic techniques to reduce the uplink communication complexity, which we call communication complexity in short, and we ignore downlink communication.

Communication Compression (CC) consists in applying some lossy scheme that compresses vectors into messages of small bit size, which are communicated. For instance, the well-known rand-k compressor selects k coordinates of the vector uniformly at random, for some $k \in [d] := \{1, \dots, d\}$. k can be as small as 1, in which case the compression factor is d, which can be huge. Some compressors, such as rand-k, are unbiased, whereas others are biased; we refer to Beznosikov et al. (2020); Albasyoni et al. (2020); Horváth et al. (2022); Condat et al. (2022b) for several examples and a discussion of their properties. The introduction of DIANA by Mishchenko et al. (2019) was a major milestone, as this algorithm converges linearly with the large class of unbiased compressors defined in Section 1.3 and also considered in LoCoDL. The communication complexity $\mathcal{O}\big(d\kappa\log\epsilon^{-1}\big)$ of the basic Gradient Descent (GD) algorithm is reduced with DIANA to $\mathcal{O}\big((\kappa+d)\log\epsilon^{-1}\big)$ when n is large, see Table 2. DIANA was later extended in several ways (Horváth et al., 2022; Gorbunov et al., 2020a; Condat & Richtárik, 2022). An accelerated version of DIANA called ADIANA based on Nesterov Accelerated GD has been proposed (Li et al., 2020b) and further analyzed in He et al. (2023); it has the state-of-the-art theoretical complexity.

Algorithms converging linearly with biased compressors have also been proposed, such as EF21 (Richtárik et al., 2021; Fatkhullin et al., 2021; Condat et al., 2022b), but the acceleration potential is less understood than with unbiased compressors. Algorithms with CC such as MARINA (Gorbunov et al., 2021) and DASHA (Tyurin & Richtárik, 2023a) have been proposed for nonconvex optimization, but their analysis requires a different approach and there is a gap in the achievable performance: their complexity depends on $\frac{\omega\kappa}{\sqrt{n}}$ instead of $\frac{\omega\kappa}{n}$ with DIANA, where ω characterizes the compression error variance, see (2). Therefore, we focus on the convex setting and leave the nonconvex study for future work.

Local Training (LT) is a simple but remarkably efficient idea: the clients perform multiple Gradient Descent (GD) steps, instead of only one, between successive communication rounds. The intuition behind is that this leads to the communication of richer information, so that the number of communication rounds to reach a given accuracy is reduced. We refer to Mishchenko et al. (2022) for a comprehensive review of LT-based algorithms, which include the popular FedAvg and Scaffold algorithms of McMahan et al. (2017) and Karimireddy et al. (2020), respectively. Mishchenko et al. (2022) made a breakthrough by proposing Scaffnew, the first LT-based variance-reduced algorithm that not only converges linearly to the exact solution in the strongly convex setting, but does so with accelerated communication complexity $\mathcal{O}(d\sqrt{\kappa}\log\epsilon^{-1})$. In Scaffnew, communication can occur randomly after every iteration, but occurs only with a small probability p. Thus, there are in average p^{-1} local steps between successive communication rounds. The optimal dependency on $\sqrt{\kappa}$ (Scaman et al., 2019) is obtained with $p = 1/\sqrt{\kappa}$. LoCoDL has the same probabilistic LT mechanism as Scaffnew but does not revert to it when compression is disabled, because of the additional function g and tracking variables y and v. A different approach to LT was developed by Sadiev et al. (2022a) with the APDA-Inexact algorithm, and generalized to handle partial participation by Grudzień et al. (2023) with the 5GCS algorithm: in both algorithms, the local GD steps form an inner loop in order to compute a proximity operator inexactly.

Combining LT and CC while retaining their benefits is very challenging. In our strongly convex and heterogeneous setting, the methods Qsparse-local-SGD (Basu et al., 2020) and FedPAQ (Reisizadeh et al., 2020) do not converge linearly. FedCOMGATE features LT + CC and converges linearly (Haddadpour et al., 2021), but its complexity $\mathcal{O}(d\kappa\log\epsilon^{-1})$ does not show any acceleration. We can mention that random reshuffling, a technique that can be seen as a type of LT, has been combined with CC in Sadiev et al. (2022b); Malinovsky & Richtárik (2022). Recently, Condat et al. (2022a) managed to design a specific compression technique compatible with the LT mechanism of Scaffnew, leading to CompressedScaffnew, the first LT + CC algorithm exhibiting a doubly-accelerated complexity, namely $\mathcal{O}\big(\big(\sqrt{d}\sqrt{\kappa}+\frac{d\sqrt{\kappa}}{\sqrt{n}}+d\big)\log\epsilon^{-1}\big)$, as reported in Table 2. However, CompressedScaffnew uses a specific linear compression scheme that requires shared randomness; that is, all clients have to agree on a random permutation of the columns of the global compression pattern. No other compressor can be used, which notably rules out any type of quantization.

1.3 A General Class of Unbiased Random Compressors

For every $\omega \geq 0$, we define the $\mathbb{U}(\omega)$ as the set of random compression operators $\mathcal{C} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ that are unbiased, i.e. $\mathbb{E}[\mathcal{C}(x)] = x$, and satisfy, for every $x \in \mathbb{R}^d$,

$$\mathbb{E}\Big[\|\mathcal{C}(x) - x\|^2\Big] \leq \omega \|x\|^2 . \tag{2}$$

In addition, given a collection $(\mathcal{C}_i)_{i=1}^n$ of compression operators in $\mathbb{U}(\omega)$ for some $\omega \geq 0$, in order to characterize their joint variance, we introduce the constant $\omega_{\mathrm{av}} \geq 0$ such that, for every $x_i \in \mathbb{R}^d$, $i \in [n]$, we have

$$\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^n \big(\mathcal{C}_i(x_i) - x_i\big)\right\|^2\right] \leq \frac{\omega_{\mathrm{av}}}{n}\sum_{i=1}^n \|x_i\|^2 . \tag{3}$$

The inequality (3) is not an additional assumption: it is satisfied with $\omega_{\mathrm{av}} = \omega$ by convexity of the squared norm. But the convergence rate will depend on ω_{av}, which is typically much smaller than ω. In particular, if the compressors $\mathcal{C}_i$ are mutually independent, the variance of their sum is the sum of their variances, and (3) is satisfied with $\omega_{\mathrm{av}} = \frac{\omega}{n}$.

1.4 Challenge and Contributions

This work addresses the following question: *Can we combine LT and CC with any compressors in the generic class $\mathbb{U}(\omega)$ defined in the previous section, and fully benefit from both techniques by obtaining a doubly-accelerated communication complexity?*

We answer this question in the affirmative. LoCoDL has the same probabilistic LT mechanism as Scaffnew and features CC with compressors in $\mathbb{U}(\omega)$ with arbitrarily large $\omega \geq 0$, with proved linear convergence under Assumption 1.1, without further requirements. By choosing the communication probability and the variance ω appropriately, double acceleration is obtained. Thus, LoCoDL achieves the same theoretical complexity as CompressedScaffnew, but allows for a large class of compressors instead of the cumbersome permutation-based compressor of the latter. In particular, with compressors performing sparsification and quantization, LoCoDL outperforms existing algorithms, as we show by experiments in Section 4. This is remarkable, since ADIANA, based on Nesterov acceleration and not LT, has an even better theoretical complexity when n is larger than d, see Table 2, but this is not reflected in practice: ADIANA is clearly behind LoCoDL in our experiments. Thus, LoCoDL sets new standards in terms of communication efficiency.

2 Proposed Algorithm LoCoDL

2.1 Principle: Double Lifting of the Problem to a Consensus Problem

In LoCoDL, every client stores and updates *two* local model estimates. They will all converge to the same solution $x^\star$ of (1). This construction comes from two ideas.

Local steps with local models. In algorithms making use of LT, such as FedAvg, Scaffold and Scaffnew, the clients store and update local model estimates x_i. When communication occurs, an estimate of their average is formed by the server and broadcast to all clients. They all resume their computations with this new model estimate.

Compressing the difference between two estimates. To implement CC, a powerful idea is to compress not the vectors themselves, but *difference vectors* that converge to zero. This way, the algorithm is variance-reduced; that is, the compression error vanishes at convergence. The technique of compressing the difference between a gradient vector and a control variate is at the core of

Table 1: Communication complexity in number of communication rounds to reach ϵ-accuracy for linearly-converging algorithms allowing for CC with independent compressors in $\mathbb{U}(\omega)$ for any $\omega \geq 0$. Since the compressors are independent, $\omega_{\text{av}} = \frac{\omega}{n}$. We provide the leading asymptotic factor and ignore log factors such as $\log \epsilon^{-1}$. The state of the art is highlighted in green.

Algorithm	Com. complexity in # rounds	case $\omega = \mathcal{O}(n)$	case $\omega = \Theta(n)$
DIANA	$(1+\frac{\omega}{n})\kappa + \omega$	$\kappa + \omega$	$\kappa + \omega$
EF21	$(1+\omega)\kappa$	$(1+\omega)\kappa$	$(1+\omega)\kappa$
5GCS-CC	$\left(1+\sqrt{\omega}+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$
ADIANA[1]	$\left(1+\frac{\omega^{3/4}}{n^{1/4}}+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$\left(1+\frac{\omega^{3/4}}{n^{1/4}}\right)\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$
ADIANA[2]	$\left(1+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$\left(1+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$
lower bound[2]	$\left(1+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$\left(1+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$
LoCoDL	$\left(1+\sqrt{\omega}+\frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa}+\omega(1+\frac{\omega}{n})$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$	$(1+\sqrt{\omega})\sqrt{\kappa}+\omega$

[1] This is the complexity derived in the original paper Li et al. (2020b).
[2] This is the complexity derived by a refined analysis in the preprint He et al. (2023), where a matching lower bound is also derived.

Table 2: (Uplink) communication complexity in number of reals to reach ϵ-accuracy for linearly-converging algorithms allowing for CC, with an optimal choice of unbiased compressors. We provide the leading asymptotic factor and ignore log factors such as $\log \epsilon^{-1}$. The state of the art is highlighted in green.

Algorithm	complexity in # reals	case $n = \mathcal{O}(d)$
DIANA	$(1+\frac{d}{n})\kappa + d$	$\frac{d}{n}\kappa + d$
EF21	$d\kappa$	$d\kappa$
5GCS-CC	$\left(\sqrt{d}+\frac{d}{\sqrt{n}}\right)\sqrt{\kappa}+d$	$\frac{d}{\sqrt{n}}\sqrt{\kappa}+d$
ADIANA	$\left(1+\frac{d}{\sqrt{n}}\right)\sqrt{\kappa}+d$	$\frac{d}{\sqrt{n}}\sqrt{\kappa}+d$
CompressedScaffnew	$\left(\sqrt{d}+\frac{d}{\sqrt{n}}\right)\sqrt{\kappa}+d$	$\frac{d}{\sqrt{n}}\sqrt{\kappa}+d$
FedCOMGATE	$d\kappa$	$d\kappa$
LoCoDL	$\left(\sqrt{d}+\frac{d}{\sqrt{n}}\right)\sqrt{\kappa}+d$	$\frac{d}{\sqrt{n}}\sqrt{\kappa}+d$

algorithms such as DIANA and EF21. Here, we want to compress differences between model estimates, not gradient estimates. That is, we want Client i to compress the difference between x_i and another model estimate that converges to the solution $x^\star$ as well. We see the need of an additional model estimate that plays the role of an anchor for compression. This is the variable y common to all clients in LoCoDL, which compress $x_i - y$ and send these compressed differences to the server.

Combining the two ideas. Accordingly, an equivalent reformulation of (1) is the consensus problem with $n+1$ variables

$$\min_{x_1,\ldots,x_n,y} \frac{1}{n}\sum_{i=1}^{n} f_i(x_i) + g(y) \;\text{ s.t. }\; x_1 = \cdots = x_n = y.$$

The primal–dual optimality conditions are $x_1 = \cdots = x_n = y$, $0 = \nabla f_i(x_i) - u_i \; \forall i \in [n]$, $0 = \nabla g(y) - v$, and $0 = u_1 + \cdots + u_n + nv$ (dual feasibility), for some dual variables $u_1, \ldots, u_n, v$ introduced in LoCoDL, that always satisfy the dual feasibility condition.

2.2 Description of LoCoDL

LoCoDL is a randomized primal–dual algorithm, shown as Algorithm 1. At every iteration, for every $i \in [n]$ in parallel, Client i first constructs a prediction $\hat{x}_i^t$ of its updated local model estimate, using a GD step with respect to f_i corrected by the dual variable u_i^t. It also constructs a prediction $\hat{y}^t$ of the updated model estimate, using a GD step with respect to g corrected by the dual variable v^t.

Algorithm 1 LoCoDL

1: **input:** stepsizes $\gamma > 0$, $\chi > 0$, $\rho > 0$; probability $p \in (0, 1]$; variance factor $\omega \geq 0$; local initial estimates $x_1^0, \ldots, x_n^0 \in \mathbb{R}^d$, initial estimate $y^0 \in \mathbb{R}^d$, initial control variates $u_1^0, \ldots, u_n^0 \in \mathbb{R}^d$ and $v \in \mathbb{R}^d$ such that $\frac{1}{n}\sum_{i=1}^n u_i^0 + v^0 = 0$.
2: **for** $t = 0, 1, \ldots$ **do**
3:     **for** $i = 1, \ldots, n$, at clients in parallel, **do**
4:         $\hat{x}_i^t := x_i^t - \gamma \nabla f_i(x_i^t) + \gamma u_i^t$
5:         $\hat{y}^t := y^t - \gamma \nabla g(y^t) + \gamma v^t$ // the clients store and update identical copies of y^t, v^t, $\hat{y}^t$
6:         flip a coin $\theta^t \in \{0, 1\}$ with $\mathrm{Prob}(\theta^t = 1) = p$
7:         **if** $\theta^t = 1$ **then**
8:             $d_i^t := \mathcal{C}_i^t\big(\hat{x}_i^t - \hat{y}^t\big)$
9:             send d_i^t to the server
10:             at server: aggregate $\bar{d}^t := \frac{1}{2n}\sum_{j=1}^n d_j^t$ and broadcast $\bar{d}^t$ to all clients
11:             $x_i^{t+1} := (1-\rho)\hat{x}_i^t + \rho(\hat{y}^t + \bar{d}^t)$
12:             $u_i^{t+1} := u_i^t + \frac{p\chi}{\gamma(1+2\omega)}\big(\bar{d}^t - d_i^t\big)$
13:             $y^{t+1} := \hat{y}^t + \rho\bar{d}^t$
14:             $v^{t+1} := v^t + \frac{p\chi}{\gamma(1+2\omega)}\bar{d}^t$
15:         **else**
16:             $x_i^{t+1} := \hat{x}_i^t$, $y^{t+1} = \hat{y}^t$, $u_i^{t+1} := u_i^t$, $v^{t+1} := v^t$
17:         **end if**
18:     **end for**
19: **end for**

Since g is known by all clients, they all maintain and update identical copies of the variables y and v. If there is no communication, which is the case with probability $1-p$, x_i and y are updated with these predicted estimates, and the dual variables u_i and v are unchanged. If communication occurs, which is the case with probability p, the clients compress the differences $\hat{x}_i^t - \hat{y}^t$ and send these compressed vectors to the server, which forms $\bar{d}^t$ equal to one half of their average. Then the variables x_i are updated using a convex combination of the local predicted estimates $\hat{x}_i^t$ and the global but noisy estimate $\hat{y}^t + \bar{d}^t$. y is updated similarly. Finally, the dual variables are updated using the compressed differences minus their weighted average, so that the dual feasibility condition remains satisfied. The model estimates x_i^t, $\hat{x}_i^t$, y^t, $\hat{y}^t$ all converge to $x^\star$, so that their differences, as well as the compressed differences as a consequence of (2), converge to zero. This is the key property that makes the algorithm variance-reduced. We consider the following assumption.

Assumption 2.1 (class of compressors)**.** In LoCoDL the compressors $\mathcal{C}_i^t$ are all in $\mathbb{U}(\omega)$ for some $\omega \geq 0$. Moreover, for every $i \in [n]$, $i' \in [n]$, $t \geq 0$, $t' \geq 0$, $\mathcal{C}_i^t$ and $\mathcal{C}_{i'}^{t'}$ are independent if $t \neq t'$ ($\mathcal{C}_i^t$ and $\mathcal{C}_{i'}^t$ at the same iteration t need not be independent). We define $\omega_{\mathrm{av}} \geq 0$ such that for every $t \geq 0$, the collection $(\mathcal{C}_i^t)_{i=1}^n$ satisfies (3).

Remark 2.2 (partial participation). LoCoDL allows for a form of partial participation if we set $\rho = 1$. Indeed, in that case, at steps 11 and 13 of the algorithm, all local variables x_i as well as the common variable y are overwritten by the same up-to-date model $\hat{y}^t + \bar{d}^t$. So, it does not matter that for a non-participating client i with $d_i^t = 0$, the $\hat{x}_i^{t'}$ were not computed for the $t' \leq t$ since its last participation, as they are not used in the process. However, a non-participating client should still update its local copy of y at every iteration. This can be done when ∇g is much cheaper to compute that ∇f_i, as is the case with $g = \frac{\mu}{2}\|\cdot\|^2$. A non-participating client can be completely idle for a certain period of time, but when it resumes participating, it should receive the last estimates of x, y and v from the server as it lost synchronization.

3 Convergence and Complexity of LoCoDL

Theorem 3.1 (linear convergence of LoCoDL)**.** *Suppose that Assumptions 1.1 and 2.1 hold. In* LoCoDL*, suppose that* $0 < \gamma < \frac{2}{L}$, $2\rho - \rho^2(1+\omega_{\mathrm{av}}) - \chi \geq 0$. *For every* $t \geq 0$, *define the Lyapunov* *function*

$$\Psi^t \coloneqq \frac{1}{\gamma}\left(\sum_{i=1}^n \left\|x_i^t - x^\star\right\|^2 + n\left\|y^t - x^\star\right\|^2\right) + \frac{\gamma(1+2\omega)}{p^2\chi}\left(\sum_{i=1}^n \left\|u_i^t - u_i^\star\right\|^2 + n\left\|v^t - v^\star\right\|^2\right), \tag{4}$$

where $v^\star \coloneqq \nabla g(x^\star)$ *and* $u_i^\star \coloneqq \nabla f_i(x^\star)$. *Then* LoCoDL *converges linearly: for every* $t \geq 0$,

$$\mathbb{E}\left[\Psi^t\right] \leq \tau^t \Psi^0, \quad \textit{where} \quad \tau \coloneqq \max\left((1-\gamma\mu)^2, (1-\gamma L)^2, 1 - \frac{p^2\chi}{1+2\omega}\right) < 1. \tag{5}$$

In addition, for every $i \in [n]$, $(x_i^t)_{t\in\mathbb{N}}$ *and* $(y^t)_{t\in\mathbb{N}}$ *converge to* $x^\star$, $(u_i^t)_{t\in\mathbb{N}}$ *converges to* $u_i^\star$, *and* $(v^t)_{t\in\mathbb{N}}$ *converges to* $v^\star$, *almost surely.*

We place ourselves in the conditions of Theorem 3.1. We observe that in (5), the larger χ, the better, so given ρ we should set $\chi = 2\rho - \rho^2(1+\omega_{\mathrm{av}})$. Then, choosing ρ to maximize χ yields

$$\chi = \rho = \frac{1}{1+\omega_{\mathrm{av}}}. \tag{6}$$

We now study the complexity of LoCoDL with χ and ρ chosen as in (6) and $\gamma = \Theta(\frac{1}{L})$. We remark that LoCoDL has the same rate $\tau^\sharp \coloneqq \max(1-\gamma\mu, \gamma L - 1)^2$ as mere distributed gradient descent, as long as p^{-1}, ω and ω_{av} are small enough to have $1 - \frac{p^2\chi}{1+2\omega} \leq \tau^\sharp$. This is remarkable: communicating with a low frequency and compressed vectors does not harm convergence at all, until some threshold. The iteration complexity of LoCoDL to reach ϵ-accuracy, i.e. $\mathbb{E}[\Psi^t] \leq \epsilon\Psi^0$, is

$$\mathcal{O}\left(\left(\kappa + \frac{(1+\omega_{\mathrm{av}})(1+\omega)}{p^2}\right)\log \epsilon^{-1}\right). \tag{7}$$

By choosing

$$p = \min\left(\sqrt{\frac{(1+\omega_{\mathrm{av}})(1+\omega)}{\kappa}}, 1\right), \tag{8}$$

the iteration complexity becomes $\mathcal{O}\Big(\big(\kappa + \omega(1+\omega_{\mathrm{av}})\big)\log \epsilon^{-1}\Big)$ and the communication complexity in number of communication rounds is p times the iteration complexity, that is

$$\mathcal{O}\left(\left(\sqrt{\kappa(1+\omega_{\mathrm{av}})(1+\omega)} + \omega(1+\omega_{\mathrm{av}})\right)\log \epsilon^{-1}\right).$$

If the compressors are mutually independent, $\omega_{\mathrm{av}} = \frac{\omega}{n}$ and the communication complexity can be equivalently written as

$$\mathcal{O}\left(\left(\left(1 + \sqrt{\omega} + \frac{\omega}{\sqrt{n}}\right)\sqrt{\kappa} + \omega\left(1 + \frac{\omega}{n}\right)\right)\log \epsilon^{-1}\right),$$

as shown in Table 1.

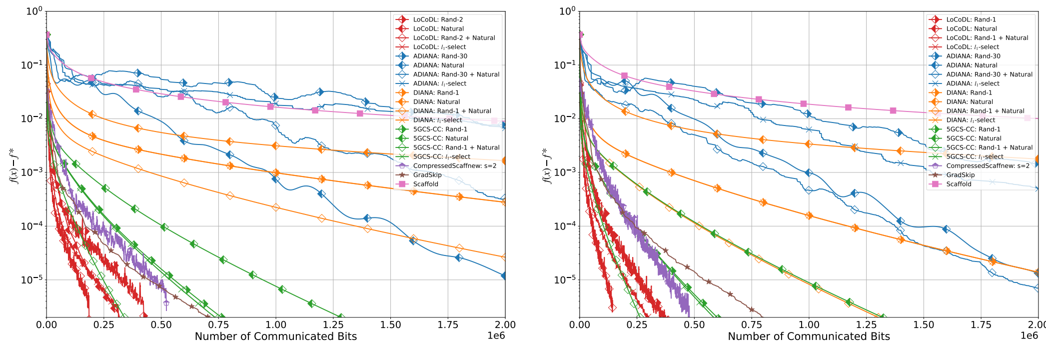
Let us consider the example of independent $\mathtt{rand}\text{-}k$ compressors, for some $k \in [d]$. We have $\omega = \frac{d}{k} - 1$. Therefore, the communication complexity in numbers of reals is k times the complexity in number of rounds; that is, $\mathcal{O}\left(\left(\left(\sqrt{kd} + \frac{d}{\sqrt{n}}\right)\sqrt{\kappa} + d\left(1 + \frac{d}{kn}\right)\right)\log \epsilon^{-1}\right)$. We can now choose k to minimize this complexity: with $k = \lceil \frac{d}{n} \rceil$, it becomes $\mathcal{O}\left(\left(\left(\sqrt{d} + \frac{d}{\sqrt{n}}\right)\sqrt{\kappa} + d\right)\log \epsilon^{-1}\right)$, as shown in Table 2. Let us state this result:

Corollary 3.2. *In the conditions of Theorem 3.1, suppose in addition that the compressors* $\mathcal{C}_i^t$ *are independent* $\mathtt{rand}\text{-}k$ *compressors with* $k = \lceil \frac{d}{n} \rceil$. *Suppose that* $\gamma = \Theta(\frac{1}{L})$, $\chi = \rho = \frac{n}{n-1+d/k}$, *and*

$$p = \min\left(\sqrt{\frac{dk(n-1)+d^2}{nk^2\kappa}}, 1\right). \tag{9}$$

Then the uplink communication complexity in number of reals of LoCoDL *is*

$$\mathcal{O}\left(\left(\sqrt{d}\sqrt{\kappa} + \frac{d\sqrt{\kappa}}{\sqrt{n}} + d\right)\log \epsilon^{-1}\right). \tag{10}$$



(a) $n = 87$ (b) $n = 288$

Figure 1: Comparison of several algorithms with several compressors on logistic regression with the 'a5a' dataset from the LibSVM, which has $d = 122$ and 6,414 data points. We chose different values of n to illustrate the two regimes $n < d$ and $n > d$, as discussed at the end of Section 3.

This is the same complexity as CompressedScaffnew (Condat et al., 2022a). However, it is obtained with simple independent compressors, which is much more practical than the permutation-based compressors with shared randomness of CompressedScaffnew. Moreover, this complexity can be obtained with other types of compressors, and further reduced, when reasoning in number of bits and not only reals, by making use of quantization (Albasyoni et al., 2020), as we illustrate by experiments in the next section.

We can distinguish 2 regimes:

1. In the "large d small n" regime, i.e. $n = \mathcal{O}(d)$, the communication complexity of LoCoDL in (10) becomes $\mathcal{O}\left(\left(\frac{d\sqrt{\kappa}}{\sqrt{n}} + d\right) \log \epsilon^{-1}\right)$. This is the state of the art, as reported in Table 2.

2. In the "large n small d" regime, i.e. $n = \Omega(d)$, the communication complexity of LoCoDL in (10) becomes $\mathcal{O}\left(\left(\sqrt{d}\sqrt{\kappa} + d\right) \log \epsilon^{-1}\right)$. If n is even larger with $n = \Omega(d^2)$, ADIANA achieves the even better complexity $\mathcal{O}\left((\sqrt{\kappa} + d) \log \epsilon^{-1}\right)$.

Yet, in the experiments we ran with different datasets and values of d, n, κ, LoCoDL outperforms the other algorithms, including ADIANA, in all cases.

3.1 The Case $g = 0$

We have assumed the presence of a function g in Problem (1), whose gradient is called by all clients. In this section, we show that we can handle the case where such a function is not available. So, let us assume that we want to minimize $\frac{1}{n}\sum_{i=1}^n f_i$, with the functions f_i satisfying Assumption 1.1. We now define the functions $\tilde{f}_i := f_i - \frac{\mu}{4}\|\cdot\|^2$ and $\tilde{g} := \frac{\mu}{4}\|\cdot\|^2$. They are all $\tilde{L}$-smooth and $\tilde{\mu}$-strongly convex, with $\tilde{L} := L - \frac{\mu}{2}$ and $\tilde{\mu} := \frac{\mu}{2}$. Moreover, it is equivalent to minimize $\frac{1}{n}\sum_{i=1}^n f_i$ or $\frac{1}{n}\sum_{i=1}^n \tilde{f}_i + \tilde{g}$. We can then apply LoCoDL to the latter problem. At Step 5, we simply have $y^t - \gamma\nabla\tilde{g}(y^t) = (1 - \frac{\gamma\mu}{2})y^t$. The rate in (5) applies with L and μ replaced by $\tilde{L}$ and $\tilde{\mu}$, respectively. Since $\kappa \le \tilde{\kappa} := \frac{\tilde{L}}{\tilde{\mu}} \le 2\kappa$, the asymptotic complexities derived above also apply to this setting. Thus, the presence of g in Problem (1) is not restrictive at all, as the only property of g that matters is that it has the same amount of strong convexity as the f_is.

4 Experiments

We evaluate the performance of our proposed method LoCoDL and compare it with several other methods that also allow for CC and converge linearly to $x^\star$. We also include GradSkip (Maranjyan et al., 2023) and Scaffold (McMahan et al., 2017) in our comparisons. We focus on a regularized logistic regression problem, which has the form (1) with

$$f_i(x) = \frac{1}{m}\sum_{s=1}^{m} \log\Big(1+\exp\left(-b_{i,s}a_{i,s}^\top x\right)\Big) + \frac{\mu}{2}\|x\|^2 \qquad (11)$$

and $g = \frac{\mu}{2}\|x\|^2$, where n is the number of clients, m is the number of data points per client, $a_{i,s} \in \mathbb{R}^d$ and $b_{i,s} \in \{-1, +1\}$ are the data samples, and μ is the regularization parameter, set so that $\kappa = 10^4$. For all algorithms other than LoCoDL, for which there is no function g, the functions f_i in (11) have a twice higher μ, so that the problem remains the same.

We considered several datasets from the LibSVM library (Chang & Lin, 2011) (3-clause BSD license). We show the results with the 'a5a' dataset in Figure 1 and with other datasets in the Appendix. We prepared each dataset by first shuffling it, then distributing it equally among the n clients (since m in (11) is an integer, the remaining datapoints were discarded). We used four different compression operators in the class $\mathbb{U}(\omega)$, for some $\omega \geq 0$:

- rand-k for some $k \in [d]$, which communicates $32k + k\lceil\log_2(d)\rceil$ bits. Indeed, the k randomly chosen values are sent in the standard 32-bits IEEE floating-point format, and their locations are encoded with $k\lceil\log_2(d)\rceil$ additional bits. We have $\omega = \frac{d}{k} - 1$.

- Natural Compression (Horváth et al., 2022), a form of quantization in which floats are encoded into 9 bits instead of 32 bits. We have $\omega = \frac{1}{8}$.

- A combination of rand-k and Natural Compression, in which the k chosen values are encoded into 9 bits, which yields a total of $9k + k\lceil\log_2(d)\rceil$ bits. We have $\omega = \frac{9d}{8k} - 1$.

- The l_1-selection compressor, defined as $C(x) = \text{sign}(x_j)\|x\|_1 e_j$, where j is chosen randomly in $[d]$, with the probability of choosing $j' \in [d]$ equal to $|x_{j'}|/\|x\|_1$, and e_j is the j-th standard unit basis vector in $\mathbb{R}^d$. $\text{sign}(x_j)\|x\|_1$ is sent as a 32-bits float and the location of j is indicated with $\lceil\log_2(d)\rceil$, so that this compressor communicates $32 + \lceil\log_2(d)\rceil$ bits. Like with rand-1, we have $\omega = d - 1$.

The compressors at different clients are independent, so that $\omega_{\text{av}} = \frac{\omega}{n}$ in (3).

We can see that LoCoDL, when combined with rand-k and Natural Compression, converges faster than all other algorithms, with respect to the total number of communicated bits per client. We chose two different numbers n of clients, one with $n < d$ and another one with $n > 2d$, since the compressor of CompressedScaffnew is different in the two cases $n < 2d$ and $n > 2d$ (Condat et al., 2022a). LoCoDL outperforms CompressedScaffnew in both cases. As expected, all methods exhibit faster convergence with larger n. Remarkably, ADIANA, which has the best theoretical complexity for large n, improves upon DIANA but is not competitive with the LT-based methods CompressedScaffnew, 5GCS-CC, and LoCoDL. This illustrates the power of doubly-accelerated methods based on a successful combination of LT and CC. In this class, our new proposed LoCoDL algorithm shines. For all algorithms, we used the theoretical parameter values given in their available convergence results (Corollary 3.2 for LoCoDL). We tried to tune the parameter values, such as k in rand-k and the (average) number of local steps per round, but this only gave minor improvements. For instance, ADIANA in Figure 1 was a bit faster with the best value of $k = 20$ than with $k = 30$. Increasing the learning rate γ led to inconsistent results, with sometimes divergence.

5 Conclusion

We have proposed LoCoDL, which combines a probabilistic Local Training mechanism similar to the one of Scaffnew and Communication Compression with a large class of unbiased compressors. This successful combination makes LoCoDL highly communication-efficient, with a doubly accelerated complexity with respect to the model dimension d and the condition number of the functions. In practice, LoCoDL outperforms other algorithms, including ADIANA, which has an even better complexity in theory obtained from Nesterov acceleration and not Local Training. This again shows the relevance of the popular mechanism of Local Training, which has been widely adopted in Federated Learning. A venue for future work is to implement bidirectional compression (Liu et al., 2020; Philippenko & Dieuleveut, 2021). We will also investigate extensions of our method with calls to stochastic gradient estimates, with or without variance reduction, as well as partial participation. These two features have been proposed for Scaffnew in Malinovsky et al. (2022) and Condat et al. (2023), but they are challenging to combine with generic compression.

References

Albasyoni, A., Safaryan, M., Condat, L., and Richtárik, P. Optimal gradient compression for distributed and federated learning. preprint arXiv:2010.03246, 2020.

Basu, D., Data, D., Karakus, C., and Diggavi, S. N. Qsparse-Local-SGD: Distributed SGD With Quantization, Sparsification, and Local Computations. *IEEE Journal on Selected Areas in Information Theory*, 1(1):217–226, 2020.

Bertsekas, D. P. *Convex optimization algorithms*. Athena Scientific, Belmont, MA, USA, 2015.

Beznosikov, A., Horváth, S., Richtárik, P., and Safaryan, M. On biased compression for distributed learning. preprint arXiv:2002.12410, 2020.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *Proc. of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1175–1191, 2017.

Chang, C.-C. and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at http://www.csie.ntu.edu.tw/%7Ecjlin/libsvm.

Condat, L. and Richtárik, P. MURANA: A generic framework for stochastic variance-reduced optimization. In *Proc. of the conference Mathematical and Scientific Machine Learning (MSML), PMLR 190*, 2022.

Condat, L. and Richtárik, P. RandProx: Primal-dual optimization algorithms with randomized proximal updates. In *Proc. of International Conference on Learning Representations (ICLR)*, 2023.

Condat, L., Agarský, I., and Richtárik, P. Provably doubly accelerated federated learning: The first theoretically successful combination of local training and compressed communication. preprint arXiv:2210.13277, 2022a.

Condat, L., Li, K., and Richtárik, P. EF-BV: A unified theory of error feedback and variance reduction mechanisms for biased and unbiased compression in distributed optimization. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2022b.

Condat, L., Agarský, I., Malinovsky, G., and Richtárik, P. TAMUNA: Doubly accelerated federated learning with local training, compression, and partial participation. preprint arXiv:2302.09832 presented at the *Int. Workshop on Federated Learning in the Age of Foundation Models in Conjunction with NeurIPS 2023*, 2023.

Fatkhullin, I., Sokolov, I., Gorbunov, E., Li, Z., and Richtárik, P. EF21 with bells & whistles: Practical algorithmic extensions of modern error feedback. preprint arXiv:2110.03294, 2021.

Gorbunov, E., Hanzely, F., and Richtárik, P. A unified theory of SGD: Variance reduction, sampling, quantization and coordinate descent. In *Proc. of 23rd Int. Conf. Artificial Intelligence and Statistics (AISTATS), PMLR 108*, 2020a.

Gorbunov, E., Kovalev, D., Makarenko, D., and Richtárik, P. Linearly converging error compensated SGD. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2020b.

Gorbunov, E., Burlachenko, K., Li, Z., and Richtárik, P. MARINA: Faster non-convex distributed learning with compression. In *Proc. of 38th Int. Conf. Machine Learning (ICML)*, pp. 3788–3798, 2021.

Gower, R. M., Schmidt, M., Bach, F., and Richtárik, P. Variance-reduced methods for machine learning. *Proc. of the IEEE*, 108(11):1968–1983, November 2020.

Grudzień, M., Malinovsky, G., and Richtárik, P. Can 5th Generation Local Training Methods Support Client Sampling? Yes! In *Proc. of Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2023.

Gruntkowska, K., Tyurin, A., and Richtárik, P. EF21-P and friends: Improved theoretical communication complexity for distributed optimization with bidirectional compression. In *Proc. of 40th Int. Conf. Machine Learning (ICML)*, 2023.

Haddadpour, F., Kamani, M. M., Mokhtari, A., and Mahdavi, M. Federated learning with compression: Unified analysis and sharp guarantees. In *Proc. of Int. Conf. Artificial Intelligence and Statistics (AISTATS), PMLR 130*, pp. 2350–2358, 2021.

Hanzely, F. and Richtárik, P. One method to rule them all: Variance reduction for data, parameters and many new methods. preprint arXiv:1905.11266, 2019.

He, Y., Huang, X., and Yuan, K. Unbiased compression saves communication in distributed optimization: When and how much? preprint arXiv:2305.16297, 2023.

Horváth, S., Ho, C.-Y., Horváth, L., Sahu, A. N., Canini, M., and Richtárik, P. Natural compression for distributed deep learning. In *Proc. of the conference Mathematical and Scientific Machine Learning (MSML), PMLR 190*, 2022.

Horváth, S., Kovalev, D., Mishchenko, K., Stich, S., and Richtárik, P. Stochastic distributed learning with gradient quantization and variance reduction. *Optimization Methods and Software*, 2022.

Kairouz, P. et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2), 2021.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S. U., and Suresh, A. T. SCAFFOLD: Stochastic controlled averaging for federated learning. In *Proc. of 37th Int. Conf. Machine Learning (ICML)*, pp. 5132–5143, 2020.

Konečný, J., McMahan, H. B., Ramage, D., and Richtárik, P. Federated optimization: distributed machine learning for on-device intelligence. arXiv:1610.02527, 2016a.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. In *NIPS Private Multi-Party Machine Learning Workshop*, 2016b. arXiv:1610.05492.

Li, T., Sahu, A. K., Talwalkar, A., and Smith, V. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 3(37):50–60, 2020a.

Li, Z., Kovalev, D., Qian, X., and Richtárik, P. Acceleration for compressed gradient descent in distributed and federated optimization. In *Proc. of 37th Int. Conf. Machine Learning (ICML)*, volume PMLR 119, 2020b.

Liu, X., Li, Y., Tang, J., and Yan, M. A double residual compression algorithm for efficient distributed learning. In *Proc. of Int. Conf. Artificial Intelligence and Statistics (AISTATS), PMLR 108*, pp. 133–143, 2020.

Malinovsky, G. and Richtárik, P. Federated random reshuffling with compression and variance reduction. preprint arXiv:arXiv:2205.03914, 2022.

Malinovsky, G., Yi, K., and Richtárik, P. Variance reduced ProxSkip: Algorithm, theory and application to federated learning. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2022.

Maranjyan, A., Safaryan, M., and Richtárik, P. Gradskip: Communication-accelerated local gradient methods with better computational complexity, 2023.

McMahan, H. B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Proc. of Int. Conf. Artificial Intelligence and Statistics (AISTATS), PMLR 54*, 2017.

Mishchenko, K., Gorbunov, E., Takáč, M., and Richtárik, P. Distributed learning with compressed gradient differences. arXiv:1901.09269, 2019.

Mishchenko, K., Malinovsky, G., Stich, S., and Richtárik, P. ProxSkip: Yes! Local Gradient Steps Provably Lead to Communication Acceleration! Finally! In *Proc. of the 39th International Conference on Machine Learning (ICML)*, July 2022.

Philippenko, C. and Dieuleveut, A. Artemis: tight convergence guarantees for bidirectional compression in federated learning. preprint arXiv:2006.14591, 2020.

Philippenko, C. and Dieuleveut, A. Preserved central model for faster bidirectional compression in distributed settings. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2021.

Reisizadeh, A., Mokhtari, A., Hassani, H., Jadbabaie, A., and Pedarsani, R. FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization. In *Proc. of Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, pp. 2021–2031, 2020.

Richtárik, P., Sokolov, I., and Fatkhullin, I. EF21: A new, simpler, theoretically better, and practically faster error feedback. In *Proc. of 35th Conf. Neural Information Processing Systems (NeurIPS)*, 2021.

Sadiev, A., Kovalev, D., and Richtárik, P. Communication acceleration of local gradient methods via an accelerated primal-dual algorithm with an inexact prox. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2022a.

Sadiev, A., Malinovsky, G., Gorbunov, E., Sokolov, I., Khaled, A., Burlachenko, K., and Richtárik, P. Federated optimization algorithms with random reshuffling and gradient compression. preprint arXiv:2206.07021, 2022b.

Scaman, K., Bach, F., Bubeck, S., Lee, Y. T., and Massoulié, L. Optimal convergence rates for convex distributed optimization in networks. *Journal of Machine Learning Research*, 20:1–31, 2019.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

Sra, S., Nowozin, S., and Wright, S. J. *Optimization for Machine Learning*. The MIT Press, 2011.

Tyurin, A. and Richtárik, P. DASHA: Distributed nonconvex optimization with communication compression, optimal oracle complexity, and no client synchronization. In *Proc. of International Conference on Learning Representations (ICLR)*, 2023a.

Tyurin, A. and Richtárik, P. 2Direction: Theoretically faster distributed training with bidirectional communication compression. In *Proc. of Conf. Neural Information Processing Systems (NeurIPS)*, 2023b.

Wang, J. et al. A field guide to federated optimization. preprint arXiv:2107.06917, 2021.

Yang, H., Fang, M., and Liu, J. Achieving linear speedup with partial worker participation in non-IID federated learning. In *Proc. of International Conference on Learning Representations (ICLR)*, 2021.

Yang, H., Qiu, P., Khanduri, P., and Liu, J. On the efficacy of server-aided federated learning against partial client participation. preprint https://openreview.net/forum?id=Dyzhru5NO3u, 2023.

Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. Federated learning with non-iid data. preprint arXiv:1806.00582, 2018.

Appendix

A Proof of Theorem 3.1

We define the Euclidean space $\mathcal{X} \coloneqq \mathbb{R}^d$ and the product space $\boldsymbol{\mathcal{X}} \coloneqq \mathcal{X}^{n+1}$ endowed with the weighted inner product

$$\langle \mathbf{x}, \mathbf{x}' \rangle_{\boldsymbol{\mathcal{X}}} \coloneqq \sum_{i=1}^{n} \langle x_i, x_i' \rangle + n\langle y, y' \rangle, \quad \forall \mathbf{x} = (x_1, \ldots, x_n, y), \mathbf{x}' = (x_1', \ldots, x_n', y'). \tag{12}$$

We define the copy operator $\mathbf{1} : x \in \mathcal{X} \mapsto (x, \ldots, x, x) \in \boldsymbol{\mathcal{X}}$ and the linear operator

$$S : \mathbf{x} \in \boldsymbol{\mathcal{X}} \mapsto \mathbf{1}\bar{x}, \text{ with } \bar{x} = \frac{1}{2n}\left(\sum_{i=1}^{n} x_i + ny\right). \tag{13}$$

S is the orthogonal projector in $\boldsymbol{\mathcal{X}}$ onto the consensus line $\{\mathbf{x} \in \boldsymbol{\mathcal{X}} \,:\, x_1 = \cdots = x_n = y\}$. We also define the linear operator

$$W \coloneqq \mathrm{Id} - S : \mathbf{x} = (x_1, \ldots, x_n, y) \in \boldsymbol{\mathcal{X}} \mapsto (x_1 - \bar{x}, \ldots, x_n - \bar{x}, y - \bar{x}), \text{ with } \bar{x} = \frac{1}{2n}\left(\sum_{i=1}^{n} x_i + ny\right), \tag{14}$$

where Id denotes the identity. W is the orthogonal projector in $\boldsymbol{\mathcal{X}}$ onto the hyperplane $\{\mathbf{x} \in \boldsymbol{\mathcal{X}} \,:\, x_1 + \cdots + x_n + ny = 0\}$, which is orthogonal to the consensus line. As such, it is self-adjoint, positive semidefinite, its eigenvalues are $(1, \ldots, 1, 0)$, its kernel is the consensus line, and its spectral norm is 1. Also, $W^2 = W$. Note that we can write W in terms of the differences $d_i = x_i - y$ and $\bar{d} = \frac{1}{2n}\sum_{i=1}^{n} d_i$:

$$W : \mathbf{x} = (x_1, \ldots, x_n, y) \mapsto \big(d_1 - \bar{d}, \ldots, d_n - \bar{d}, -\bar{d}\big). \tag{15}$$

Since for every $\mathbf{x} = (x_1, \ldots, x_n, y)$, $W\mathbf{x} = \mathbf{0} \coloneqq (0, \ldots, 0, 0)$ if and only if $x_1 = \cdots = x_n = y$, we can reformulate the problem (1) as

$$\min_{\mathbf{x} = (x_1, \ldots, x_n, y) \in \boldsymbol{\mathcal{X}}} \mathbf{f}(\mathbf{x}) \quad \text{s.t.} \quad W\mathbf{x} = \mathbf{0}, \tag{16}$$

where $\mathbf{f}(\mathbf{x}) \coloneqq \sum_{i=1}^{n} f_i(x_i) + ng(y)$. Note that in $\boldsymbol{\mathcal{X}}$, $\mathbf{f}$ is L-smooth and μ-strongly convex, and $\nabla \mathbf{f}(\mathbf{x}) = \big(\nabla f_1(x_1), \ldots \nabla f_n(x_n), \nabla g(y)\big)$.

Let $t \geq 0$. We also introduce vector notations for the variables of the algorithm: $\mathbf{x}^t \coloneqq (x_1^t, \ldots, x_n^t, y^t)$, $\hat{\mathbf{x}}^t \coloneqq (\hat{x}_1^t, \ldots, \hat{x}_n^t, \hat{y}^t)$, $\mathbf{u}^t \coloneqq (u_1^t, \ldots, u_n^t, v^t)$, $\mathbf{u}^\star \coloneqq (u_1^\star, \ldots, u_n^\star, v^\star)$, $\mathbf{w}^t \coloneqq \mathbf{x}^t - \gamma \nabla \mathbf{f}(\mathbf{x}^t)$, $\mathbf{w}^\star \coloneqq \mathbf{x}^\star - \gamma \nabla \mathbf{f}(\mathbf{x}^\star)$, where $\mathbf{x}^\star \coloneqq \mathbf{1}x^\star$ is the unique solution to (16). We also define $\bar{x}^t \coloneqq \frac{1}{2n}(\sum_{i=1}^{n} \hat{x}_i^t + n\hat{y}^t)$ and $\lambda \coloneqq \frac{p\chi}{\gamma(1+2\omega)}$.

Then we can write the iteration of LoCoDL as

$$\left\lfloor \begin{array}{l} \hat{\mathbf{x}}^t \coloneqq \mathbf{x}^t - \gamma \nabla \mathbf{f}(\mathbf{x}^t) + \gamma \mathbf{u}^t = \mathbf{w}^t + \gamma \mathbf{u}^t \\ \text{flip a coin } \theta^t \in \{0,1\} \text{ with } \mathrm{Prob}(\theta^t = 1) = p \\ \textbf{if } \theta^t = 1 \\ \quad \mathbf{d}^t \coloneqq \big(\mathcal{C}_1^t(\hat{x}_1^t - \hat{y}^t), \ldots, \mathcal{C}_n^t(\hat{x}_n^t - \hat{y}^t), 0\big) \\ \quad \bar{d}^t \coloneqq \frac{1}{2n}\sum_{j=1}^{n} d_j^t \\ \quad \mathbf{x}^{t+1} \coloneqq (1-\rho)\hat{\mathbf{x}}^t + \rho \mathbf{1}(\hat{y}^t + \bar{d}^t) \\ \quad \mathbf{u}^{t+1} \coloneqq \mathbf{u}^t + \lambda\big(\mathbf{1}\bar{d}^t - \mathbf{d}^t\big) = \mathbf{u}^t - \lambda W \mathbf{d}^t \\ \textbf{else} \\ \quad \mathbf{x}^{t+1} \coloneqq \hat{\mathbf{x}}^t \\ \quad \mathbf{u}^{t+1} \coloneqq \mathbf{u}^t \\ \textbf{end if} \end{array} \right. \tag{17}$$

We denote by $\mathcal{F}^t$ the σ-algebra generated by the collection of $\mathcal{X}$-valued random variables $\mathbf{x}^0, \mathbf{u}^0, \ldots, \mathbf{x}^t, \mathbf{u}^t$.

Since we suppose that $S\mathbf{u}^0 = \mathbf{0}$ and we have $SW\mathbf{d}^{t'} = \mathbf{0}$ in the update of $\mathbf{u}$, we have $S\mathbf{u}^{t'} = \mathbf{0}$ for every $t' \geq 0$.

If $\theta^t = 1$, we have

$$\begin{aligned}\left\|\mathbf{u}^{t+1} - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 &= \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + \lambda^2 \left\|W\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - 2\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, W\mathbf{d}^t\rangle_{\boldsymbol{\mathcal{X}}} \\ &= \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + \lambda^2 \left\|\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - \lambda^2 \left\|S\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - 2\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \mathbf{d}^t\rangle_{\boldsymbol{\mathcal{X}}},\end{aligned}$$

because $S\mathbf{u}^t = S\mathbf{u}^\star = \mathbf{0}$, so that $\langle \mathbf{u}^t - \mathbf{u}^\star, S\mathbf{d}^t\rangle_{\boldsymbol{\mathcal{X}}} = 0$.

The variance inequality (2) satisfied by the compressors $\mathcal{C}_i^t$ is equivalent to $\mathbb{E}\Big[\|\mathcal{C}_i^t(x)\|^2\Big] \leq (1 + \omega) \|x\|^2$, so that

$$\mathbb{E}\Big[\left\|\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta^t = 1\Big] \leq (1+\omega) \left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2.$$

Also,

$$\mathbb{E}\big[\mathbf{d}^t \mid \mathcal{F}^t, \theta^t = 1\big] = \hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t.$$

Thus,

$$\begin{aligned}\mathbb{E}\Big[\left\|\mathbf{u}^{t+1} - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t\Big] &= (1-p) \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\mathbb{E}\Big[\left\|\mathbf{u}^{t+1} - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta^t = 1\Big] \\ &\leq \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2(1+\omega) \left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - p\lambda^2\mathbb{E}\Big[\left\|S\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta^t = 1\Big] \\ &\quad - 2p\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\rangle_{\boldsymbol{\mathcal{X}}} \\ &= \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2(1+\omega) \left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - p\lambda^2\mathbb{E}\Big[\left\|S\mathbf{d}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta^t = 1\Big] \\ &\quad - 2p\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \hat{\mathbf{x}}^t\rangle_{\boldsymbol{\mathcal{X}}}.\end{aligned}$$

Moreover, $\mathbb{E}\Big[\|S\mathbf{d}^t\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta^t = 1\Big] \geq \|\mathbb{E}[S\mathbf{d}^t \mid \mathcal{F}^t, \theta^t = 1]\|_{\boldsymbol{\mathcal{X}}}^2 = \|S\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\|_{\boldsymbol{\mathcal{X}}}^2$ and $\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\|_{\boldsymbol{\mathcal{X}}}^2 = \|S\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\|_{\boldsymbol{\mathcal{X}}}^2 + \|W\hat{\mathbf{x}}^t\|_{\boldsymbol{\mathcal{X}}}^2$, so that

$$\begin{aligned}\mathbb{E}\Big[\left\|\mathbf{u}^{t+1} - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t\Big] &\leq \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2(1+\omega) \left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - p\lambda^2 \left\|S\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|^2 \\ &\quad - 2p\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \hat{\mathbf{x}}^t\rangle_{\boldsymbol{\mathcal{X}}} \\ &= \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2\omega \left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2 \left\|W\hat{\mathbf{x}}^t\right\|^2 - 2p\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \hat{\mathbf{x}}^t\rangle_{\boldsymbol{\mathcal{X}}}.\end{aligned}$$

From the Peter–Paul inequality $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$ for any a and b, we have

$$\begin{aligned}\left\|\hat{\mathbf{x}}^t - \mathbf{1}\hat{y}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 &= \sum_{i=1}^n \left\|\hat{x}_i^t - \hat{y}^t\right\|^2 = \sum_{i=1}^n \left\|(\hat{x}_i^t - \bar{x}^t) - (\hat{y}^t - \bar{x}^t)\right\|^2 \\ &\leq \sum_{i=1}^n \left(2\left\|\hat{x}_i^t - \bar{x}^t)\right\|^2 + 2\left\|\hat{y}^t - \bar{x}^t\right\|^2\right) \\ &= 2\left(\sum_{i=1}^n \left\|\hat{x}_i^t - \bar{x}^t)\right\|^2 + n\left\|\hat{y}^t - \bar{x}^t\right\|^2\right) \\ &= 2\left\|\hat{\mathbf{x}}^t - \mathbf{1}\bar{x}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 = 2\left\|W\hat{\mathbf{x}}^t\right\|_{\boldsymbol{\mathcal{X}}}^2. \end{aligned} \tag{18}$$

Hence,

$$\mathbb{E}\Big[\left\|\mathbf{u}^{t+1} - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t\Big] \leq \left\|\mathbf{u}^t - \mathbf{u}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + p\lambda^2(1+2\omega) \left\|W\hat{\mathbf{x}}^t\right\|_{\boldsymbol{\mathcal{X}}}^2 - 2p\lambda\langle \mathbf{u}^t - \mathbf{u}^\star, \hat{\mathbf{x}}^t\rangle_{\boldsymbol{\mathcal{X}}}.$$

On the other hand,

$$\begin{aligned}\mathbb{E}\Big[\left\|\mathbf{x}^{t+1} - \mathbf{x}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta = 1\Big] &= (1-\rho)^2 \left\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 + \rho^2\mathbb{E}\Big[\left\|\mathbf{1}(\hat{y}^t + \bar{d}^t) - \mathbf{x}^\star\right\|_{\boldsymbol{\mathcal{X}}}^2 \mid \mathcal{F}^t, \theta = 1\Big] \\ &\quad + 2\rho(1-\rho)\left\langle \hat{\mathbf{x}}^t - \mathbf{x}^\star, \mathbf{1}\big(\hat{y}^t + \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big) - \mathbf{x}^\star\right\rangle_{\boldsymbol{\mathcal{X}}}.\end{aligned}$$

We have $\mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big] = \frac{1}{2n}\sum_{i=1}^n \hat{x}_i^t - \frac{1}{2}\hat{y}^t = \bar{x}^t - \hat{y}^t$, so that

$$\mathbf{1}\big(\hat{y}^t + \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big) = \mathbf{1}\bar{x}^t = S\hat{\mathbf{x}}^t.$$

In addition,

$$\big\langle \hat{\mathbf{x}}^t - \mathbf{x}^\star, S\hat{\mathbf{x}}^t - \mathbf{x}^\star \big\rangle_{\boldsymbol{\mathcal{X}}} = \big\langle \hat{\mathbf{x}}^t - \mathbf{x}^\star, S(\hat{\mathbf{x}}^t - \mathbf{x}^\star) \big\rangle_{\boldsymbol{\mathcal{X}}} = \big\|S(\hat{\mathbf{x}}^t - \mathbf{x}^\star)\big\|_{\boldsymbol{\mathcal{X}}}^2 .$$

Moreover,

$$\begin{aligned}
\mathbb{E}\Big[\big\|\mathbf{1}(\hat{y}^t + \bar{d}^t) - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t, \theta = 1\Big] &= \big\|\mathbf{1}\big(\hat{y}^t + \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big) - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&\quad + \mathbb{E}\Big[\big\|\mathbf{1}\big(\bar{d}^t - \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big)\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t, \theta = 1\Big] \\
&= \big\|S\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&\quad + 2n\mathbb{E}\Big[\big\|\bar{d}^t - \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big\|^2 \;\mid \mathcal{F}^t, \theta = 1\Big]
\end{aligned}$$

and, using (3),

$$\begin{aligned}
\mathbb{E}\Big[\big\|\bar{d}^t - \mathbb{E}\big[\bar{d}^t \mid \mathcal{F}^t, \theta = 1\big]\big\|^2 \;\mid \mathcal{F}^t, \theta = 1\Big] &\leq \frac{\omega_{\mathrm{av}}}{4n}\sum_{i=1}^n \big\|\hat{x}_i^t - \hat{y}^t\big\|^2 \\
&\leq \frac{\omega_{\mathrm{av}}}{2n}\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 ,
\end{aligned}$$

where the second inequality follows from (18). Hence,

$$\begin{aligned}
\mathbb{E}\Big[\big\|\mathbf{x}^{t+1} - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t, \theta = 1\Big] &\leq (1-\rho)^2\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + \rho^2\big\|S\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + \rho^2\omega_{\mathrm{av}}\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&\quad + 2\rho(1-\rho)\big\|S(\hat{\mathbf{x}}^t - \mathbf{x}^\star)\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&= (1-\rho)^2\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + \rho^2\omega_{\mathrm{av}}\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&\quad + (2\rho - \rho^2)\big\|S(\hat{\mathbf{x}}^t - \mathbf{x}^\star)\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&= (1-\rho)^2\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + \rho^2\omega_{\mathrm{av}}\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 \\
&\quad + (2\rho - \rho^2)\Big(\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - \big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2\Big) \\
&= \big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - \big(2\rho - \rho^2 - \rho^2\omega_{\mathrm{av}}\big)\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}\Big[\big\|\mathbf{x}^{t+1} - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t\Big] &= (1-p)\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + p\mathbb{E}\Big[\big\|\mathbf{x}^{t+1} - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t, \theta^t = 1\Big] \\
&\leq \big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - p\big(2\rho - \rho^2(1+\omega_{\mathrm{av}})\big)\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 .
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\big\|\hat{\mathbf{x}}^t - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 &= \big\|\mathbf{w}^t - \mathbf{w}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + \gamma^2\big\|\mathbf{u}^t - \mathbf{u}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + 2\gamma\langle \mathbf{w}^t - \mathbf{w}^\star, \mathbf{u}^t - \mathbf{u}^\star\rangle_{\boldsymbol{\mathcal{X}}} \\
&= \big\|\mathbf{w}^t - \mathbf{w}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - \gamma^2\big\|\mathbf{u}^t - \mathbf{u}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + 2\gamma\langle \hat{\mathbf{x}}^t - \mathbf{x}^\star, \mathbf{u}^t - \mathbf{u}^\star\rangle_{\boldsymbol{\mathcal{X}}} \\
&= \big\|\mathbf{w}^t - \mathbf{w}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - \gamma^2\big\|\mathbf{u}^t - \mathbf{u}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + 2\gamma\langle \hat{\mathbf{x}}^t, \mathbf{u}^t - \mathbf{u}^\star\rangle_{\boldsymbol{\mathcal{X}}},
\end{aligned}$$

which yields

$$\begin{aligned}
\mathbb{E}\Big[\big\|\mathbf{x}^{t+1} - \mathbf{x}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 \;\mid \mathcal{F}^t\Big] &\leq \big\|\mathbf{w}^t - \mathbf{w}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 - \gamma^2\big\|\mathbf{u}^t - \mathbf{u}^\star\big\|_{\boldsymbol{\mathcal{X}}}^2 + 2\gamma\langle \hat{\mathbf{x}}^t, \mathbf{u}^t - \mathbf{u}^\star\rangle_{\boldsymbol{\mathcal{X}}} \\
&\quad - p\big(2\rho - \rho^2(1+\omega_{\mathrm{av}})\big)\big\|W\hat{\mathbf{x}}^t\big\|_{\boldsymbol{\mathcal{X}}}^2 .
\end{aligned}$$

Hence, with $\lambda = \frac{p\chi}{\gamma(1+2\omega)}$,

$$\begin{aligned}
&\frac{1}{\gamma}\mathbb{E}\Big[\left\|\mathbf{x}^{t+1}-\mathbf{x}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} \mid \mathcal{F}^t\Big] + \frac{\gamma(1+2\omega)}{p^2\chi}\mathbb{E}\Big[\left\|\mathbf{u}^{t+1}-\mathbf{u}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} \mid \mathcal{F}^t\Big] \\
&\quad\leq \frac{1}{\gamma}\left\|\mathbf{w}^t-\mathbf{w}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} - \gamma\left\|\mathbf{u}^t-\mathbf{u}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} + 2\langle\hat{\mathbf{x}}^t, \mathbf{u}^t-\mathbf{u}^\star\rangle_{\boldsymbol{\mathcal{X}}} - \frac{p}{\gamma}\big(2\rho-\rho^2(1+\omega_{\mathrm{av}})\big)\left\|W\hat{\mathbf{x}}^t\right\|^2_{\boldsymbol{\mathcal{X}}} \\
&\qquad + \frac{\gamma(1+2\omega)}{p^2\chi}\left\|\mathbf{u}^t-\mathbf{u}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} + \frac{p\chi}{\gamma}\left\|W\hat{\mathbf{x}}^t\right\|^2_{\boldsymbol{\mathcal{X}}} - 2\langle\mathbf{u}^t-\mathbf{u}^\star, \hat{\mathbf{x}}^t\rangle_{\boldsymbol{\mathcal{X}}} \\
&\quad= \frac{1}{\gamma}\left\|\mathbf{w}^t-\mathbf{w}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} + \frac{\gamma(1+2\omega)}{p^2\chi}\left(1-\frac{p^2\chi}{1+2\omega}\right)\left\|\mathbf{u}^t-\mathbf{u}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} \\
&\qquad - \frac{p}{\gamma}\big(2\rho-\rho^2(1+\omega_{\mathrm{av}})-\chi\big)\left\|W\hat{\mathbf{x}}^t\right\|^2_{\boldsymbol{\mathcal{X}}}.
\end{aligned}$$

Therefore, assuming that $2\rho - \rho^2(1+\omega_{\mathrm{av}}) - \chi \geq 0$,

$$\mathbb{E}\big[\Psi^{t+1} \mid \mathcal{F}^t\big] \leq \frac{1}{\gamma}\left\|\mathbf{w}^t-\mathbf{w}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} + \left(1-\frac{p^2\chi}{1+2\omega}\right)\frac{\gamma(1+2\omega)}{p^2\chi}\left\|\mathbf{u}^t-\mathbf{u}^\star\right\|^2_{\boldsymbol{\mathcal{X}}}.$$

According to Condat & Richtárik (2023, Lemma 1),

$$\begin{aligned}
\left\|\mathbf{w}^t-\mathbf{w}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} &= \left\|(\mathrm{Id}-\gamma\nabla\mathbf{f})\mathbf{x}^t - (\mathrm{Id}-\gamma\nabla\mathbf{f})\mathbf{x}^\star\right\|^2_{\boldsymbol{\mathcal{X}}} \\
&\leq \max(1-\gamma\mu, \gamma L-1)^2\left\|\mathbf{x}^t-\mathbf{x}^\star\right\|^2_{\boldsymbol{\mathcal{X}}}.
\end{aligned}$$

Hence,

$$\mathbb{E}\big[\Psi^{t+1} \mid \mathcal{F}^t\big] \leq \max\left((1-\gamma\mu)^2, (1-\gamma L)^2, 1-\frac{p^2\chi}{1+2\omega}\right)\Psi^t. \tag{19}$$

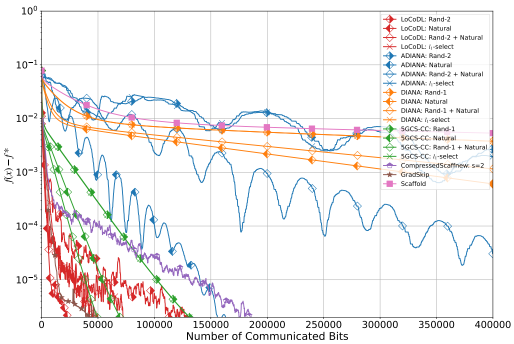
Using the tower rule, we can unroll the recursion in (19) to obtain the unconditional expectation of Ψ^{t+1}.

Using classical results on supermartingale convergence (Bertsekas, 2015, Proposition A.4.5), it follows from (19) that $\Psi^t \to 0$ almost surely. Almost sure convergence of $\mathbf{x}^t$ and $\mathbf{u}^t$ follows.

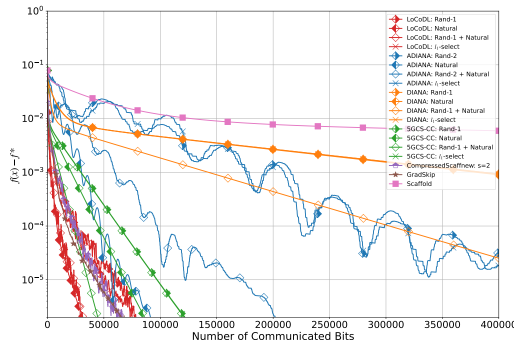
B Additional Experiments

The results for the experiments in Section 4 with the 'diabetes' dataset from the LibSVM library (Chang & Lin, 2011) are shown in Figure 2. The results with the 'w1a' and 'australian' datasets, for the same logistic regression problem with $\kappa = 10^4$, are shown in Figures 3 and 4.

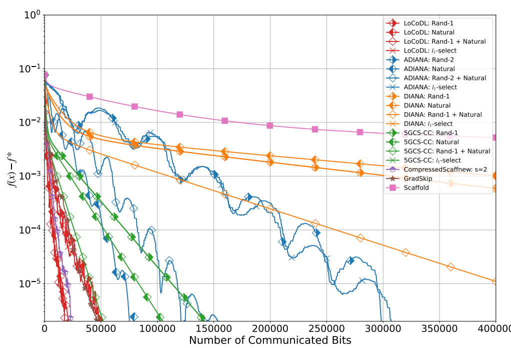
Consistent with our previous findings, LoCoDL outperforms the other algorithms in terms of communication efficiency.



(a) $n = 6$

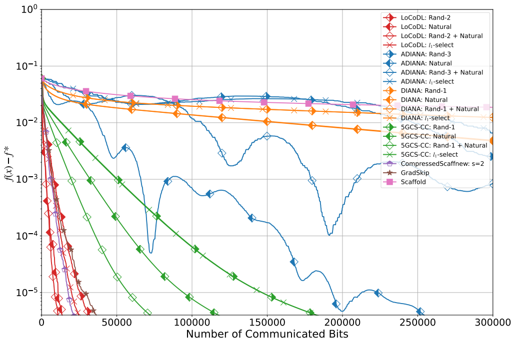


(b) $n = 37$

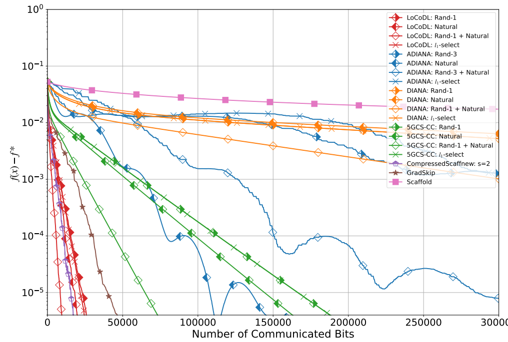


(c) $n = 73$

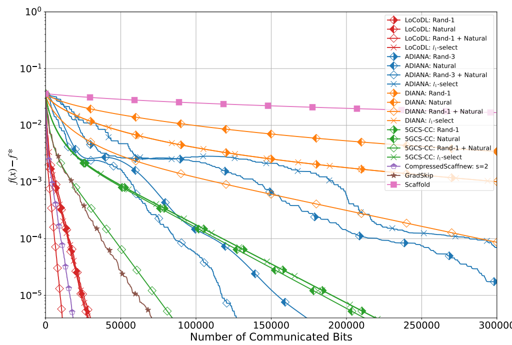
Figure 2: Comparison of several algorithms with several compressors on logistic regression with the 'diabetes' dataset from the LibSVM, which has $d = 8$ and 768 data points. We chose different values of n to illustrate the three regimes $n < d$, $n > d$, $n > d^2$, as discussed at the end of Section 3.



(a) $n = 9$

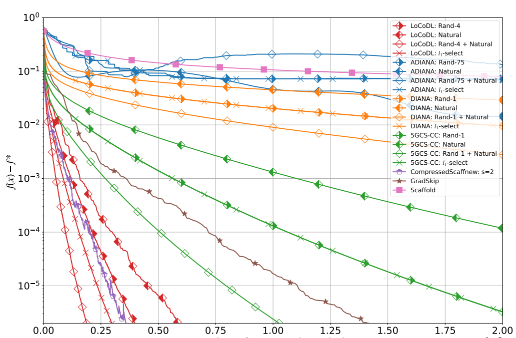


(b) $n = 41$

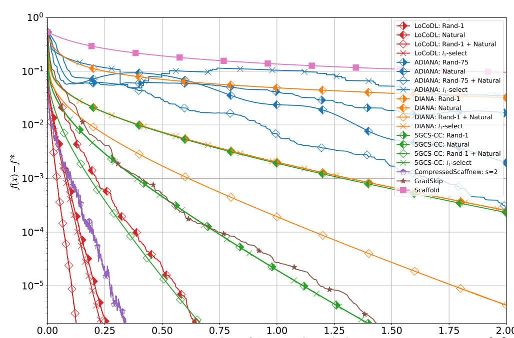


(c) $n = 225$

Figure 3: Comparison of several algorithms with various compressors on logistic regression with the 'australian' dataset from the LibSVM, which has $d = 14$ and 690 data points. We chose different values of n to illustrate the three regimes: $n < d$, $n > d$, $n > d^2$, as discussed at the end of Section 3.



(a) $n = 87$



(b) $n = 619$

Figure 4: Comparison of several algorithms with various compressors on logistic regression with the 'w1a' dataset from the LibSVM, which has $d = 300$ and 2,477 data points. We chose different values of n to illustrate the two regimes, $n < d$ and $n > d$, as discussed at the end of Section 3.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: our contribution is the unique combination of the two key mechanisms of local training and compression, as mentioned in the title and detailed in the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: the limitations are discussed in the conclusion.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: the proofs are in the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: the pseudo-code of our proposed algorithm is given. It is short and easy to implement. The parameter values for the experiments are provided.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The data used in the experiments are publicly available. The code is provided.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provide these details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: the variability with respect to different random realizations plays a minor role in the performance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: we do not focus on the computation time but on the number of communicated bits, which is independent from the hardware. So the experiments can be reproduced on any machine.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This is a theoretical work whose goal is to advance the foundations of optimization and machine learning. The positive impact of developing more efficient algorithms is clear and does not need to be discussed. We do not see any particular negative impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: we use the LibSVM and mention the 3-clause BSD license.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.