
FedSKETCH: Communication-Efficient Federated Learning via Sketching

Anonymous Author(s)

Affiliation

Address

email

Abstract

1 Communication complexity and data privacy are the two key challenges in Federated Learning (FL) where the goal is to perform a distributed learning through a
2 large volume of devices. In this work, we introduce two new algorithms, namely
3 FedSKETCH and FedSKETCHGATE, to address jointly both challenges and which
4 are, respectively, intended to be used for homogeneous and heterogeneous data distribution settings. Our algorithms are based on a key and novel sketching technique,
5 called HEAPRIX that is unbiased, compresses the accumulation of local gradients
6 using count sketch, and exhibits communication-efficiency properties leveraging
7 low-dimensional sketches. We provide sharp convergence guarantees of our
8 algorithms and validate our theoretical findings with various sets of experiments.
9
10

1 Introduction

12 Federated Learning (FL) is a recently emerging framework for distributed large scale machine
13 learning problems. In FL, data is distributed across devices [23, 33] and due to privacy concerns,
14 users are only allowed to communicate with the parameter server. Formally, the optimization problem
15 across p distributed devices is defined as follows:

$$\min_{\mathbf{x} \in \mathbb{R}^d, \sum_{j=1}^p q_j = 1} f(\mathbf{x}) \triangleq \sum_{j=1}^p q_j F_j(\mathbf{x}), \quad (1)$$

16 where $F_j(\mathbf{x}) = \mathbb{E}_{\xi \in \mathcal{D}_j} [L_j(\mathbf{x}, \xi)]$ is the local cost function at device j , $q_j \triangleq \frac{n_j}{n}$, n_j is the number
17 of data shards at device j and $n = \sum_{j=1}^p n_j$ is the total number of data samples, ξ is a random
18 variable distributed according to probability distribution \mathcal{D}_j , and L_j is a loss function that measures
19 the performance of model \mathbf{x} at device j . We note that, while for the homogeneous setting we
20 assume $\{\mathcal{D}_j\}_{j=1}^p$ have the same distribution across devices and $L_i = L_j$, $1 \leq (i, j) \leq p$, in the
21 heterogeneous setting, these distributions and loss functions L_j can vary from a device to another.

22 There are several challenges that need to be addressed in FL in order to efficiently learn a global
23 model that performs well in average for all devices:

24 – *Communication-efficiency*: There are often many devices communicating with the server, thus
25 incurring immense communication overhead. One approach to reduce communication round is using
26 *local SGD with periodic averaging* [50, 39, 48, 44] which periodically averages models after a few
27 local updates, contrary to baseline SGD [6] where gradient averaging is performed at each iteration.
28 Local SGD has been proposed in [33, 23] under the FL setting and its convergence analysis is studied
29 in [39, 44, 50, 48], later on improved in the followup references [3, 12, 21, 40] for homogeneous
30 setting. It is further extended to heterogeneous setting [12, 20, 47, 30, 38, 31]. The second approach to
31 deal with communication cost aims at reducing the size of communicated message per communication
32 round, such as local gradient quantization [1, 4, 42, 45, 46] or sparsification [2, 32, 41, 40].

33 – *Data heterogeneity*: Since locally generated data in each device may come from different distribution,
34 local computations involved in FL setting can lead to poor convergence error in practice [27, 31].

To mitigate the negative impact of data heterogeneity, [13, 16, 31, 20] suggest applying variance reduction or gradient tracking techniques along local computations.

–*Privacy* [11, 14]: Privacy has been widely addressed by injecting an additional layer of randomness to respect differential-privacy property [34] or using cryptography-based approaches under secure multi-party computation [5]. Further study of challenges can be found in recent surveys [27] and [18].

To tackle the aforementioned challenges in FL jointly, sketching based algorithms [7, 9, 22, 25] are promising approaches. For instance, to reduce communication cost, [17] develops a distributed SGD algorithm using sketching along providing its convergence analysis in the homogeneous setting, and establish a communication complexity of order $\mathcal{O}(\log(d))$ per round, where d is the dimension of the vector of parameters compared to $\mathcal{O}(d)$ complexity per round of baseline mini-batch SGD. Yet, the proposed sketching scheme in [17], built from a communication-efficiency perspective, is based on a deterministic procedure which requires access to the exact information of the gradients, thus not meeting the privacy-preserving criteria. This systemic issue is partially addressed in [37].

Focusing on privacy, [26] derives a single framework in order to tackle these issues jointly and introduces `DiffSketch` algorithm, based on the Count Sketch operator, yet does not provide its convergence analysis. Additionally, the estimation error of `DiffSketch` is higher than the sketching scheme in [17] which may end up in poor convergence.

Our main contributions are summarized as follows:

- We provide a new algorithm – `HEAPRIX` – and theoretically show that it reduces the cost of communication between devices and server, based on unbiased sketching without requiring the broadcast of exact values of gradients to the server. Based on `HEAPRIX`, we develop general algorithms for communication-efficient and sketch-based FL, namely `FedSKETCH` and `FedSKETCHGATE` for homogeneous and heterogeneous data distribution settings respectively.
- We establish non-asymptotic convergence bounds for convex, Polyak-Łojasiewicz (PL) and non-convex functions in Theorems 1 and 2 in both homogeneous and heterogeneous cases, and highlight an improvement in the number of iteration to reach a stationary point. We also provide a convergence analysis for the `PRIVIX/DiffSketch`¹ algorithm proposed in [26].
- We illustrate the benefits of `FedSKETCH` and `FedSKETCHGATE` over baseline methods through a set of experiments. The latter shows the advantages of the `HEAPRIX` compression method achieving comparable test accuracy as Federated SGD (`FedSGD`) while compressing the information exchanged between devices and server.

Notation: We denote the number of communication rounds and bits per round and per device by R and B respectively. The count sketch of vector \mathbf{x} is designated by $\mathbf{S}(\mathbf{x})$. $[p]$ denotes the set $\{1, \dots, p\}$.

2 Compression using Count Sketch

In this paper, we exploit the commonly used `Count Sketch` [7] which uses two sets of functions that encode any input vector \mathbf{x} into a hash table $\mathbf{S}_{m \times t}(\mathbf{x})$. Pairwise independent hash functions $\{h_{j,1 \leq j \leq t} : [d] \rightarrow m\}$ are used along with another set of pairwise independent sign hash functions $\{\text{sign}_{j,1 \leq j \leq t} : [d] \rightarrow \{+1, -1\}\}$ to map entries of \mathbf{x} (x_i , $1 \leq i \leq d$) into t different columns of $\mathbf{S}_{m \times t}$, wherein to lower the dimension of the input vector we usually have $d \gg mt$. The final update reads $\mathbf{S}[j][h_j(i)] = \mathbf{S}[j][h_j(i)] + \text{sign}_j(i)x_i$ for any $1 \leq j \leq t$. There are various types of sketching algorithms which are developed based on count sketching that we develop in the following subsections. See the Appendix for the detailed Count Sketch algorithm.

2.1 Sketching based Unbiased Compressor

We define an unbiased compressor as follows:

Definition 1 (Unbiased compressor). We call randomized function, $C : \mathbb{R}^d \rightarrow \mathbb{R}^d$ an unbiased compression operator with $\Delta \geq 1$, if

$$\mathbb{E}[C(\mathbf{x})] = \mathbf{x} \quad \text{and} \quad \mathbb{E}[\|C(\mathbf{x})\|_2^2] \leq \Delta \|\mathbf{x}\|_2^2.$$

We denote this class of compressors by $\mathbb{U}(\Delta)$.

¹We use `PRIVIX` and `DiffSketch` [26] interchangeably throughout the paper.

82 This definition leads to the following property

$$\mathbb{E} \left[\|\mathbf{C}(\mathbf{x}) - \mathbf{x}\|_2^2 \right] \leq (\Delta - 1) \|\mathbf{x}\|_2^2 .$$

83 Note that if we let $\Delta = 1$ then our algorithm reduces to the case of no compression. This property
84 allows us to control the noise of the compression.

85 An instance of such unbiased compressor is PRIVIX which obtains an estimate of input \mathbf{x} from a
86 count sketch noted $\mathbf{S}(\mathbf{x})$. In this algorithm, to query the quantity x_i , the i -th element of the vector
87 \mathbf{x} , we compute the median of t approximated values specified by the indices of $h_j(i)$ for $1 \leq j \leq t$,
88 see [26], or Algorithm 6 in the Appendix (for more details). The following property of count sketch
89 would be useful for our theoretical analysis.

90 **Property 1** ([26]). *For any $\mathbf{x} \in \mathbb{R}^d$, we have:*

91 *Unbiased estimation: As in [26], we have $\mathbb{E}_{\mathbf{S}} [\text{PRIVIX}[\mathbf{S}(\mathbf{x})]] = \mathbf{x}$.*

92 *Bounded variance: For the given $m < d$, $t = \mathcal{O}(\ln(\frac{d}{\delta}))$ with probability $1 - \delta$ we have:*

$$\mathbb{E}_{\mathbf{S}} \left[\|\text{PRIVIX}[\mathbf{S}(\mathbf{x})] - \mathbf{x}\|_2^2 \right] \leq \frac{c \times d}{m} \|\mathbf{x}\|_2^2 ,$$

93 where c ($e \leq c < m$) is a positive constant independent of the dimension of the input, d .

94 We note that bounded variance assumption does not necessary implies any compression as d could be
95 relatively large. Thus, with probability $1 - \delta$ we obtain $\text{PRIVIX} \in \mathbb{U}(1 + c\frac{d}{m})$. $\Delta = 1 + c\frac{d}{m}$ implies
96 that if $m \rightarrow d$, then $\Delta \rightarrow 1 + c$, indicating a noisy reconstruction. The reference [26] shows that if the
97 data is normally distributed, PRIVIX is differentially private [10], up to additional assumptions and
98 algorithmic design.

99 2.2 Sketching based Biased Compressor

100 A biased compressor is defined as follows:

101 **Definition 2** (Biased compressor). *A (randomized) function, $C : \mathbb{R}^d \rightarrow \mathbb{R}^d$ belongs to $\mathbb{C}(\Delta, \alpha)$, a*
102 *class of compression operators with $\alpha > 0$ and $\Delta \geq 1$, if*

$$\mathbb{E} \left[\|\alpha \mathbf{x} - C(\mathbf{x})\|_2^2 \right] \leq \left(1 - \frac{1}{\Delta} \right) \|\mathbf{x}\|_2^2 ,$$

103 The reference [15] proves that $\mathbb{U}(\Delta) \subset \mathbb{C}(\Delta, \alpha)$. An example of bi-
104 ased compression via sketching and using top_m operation is given below:
105

106 Following [17], HEAVYMIX with sketch size
107 $\Theta(m \log(\frac{d}{\delta}))$ is a biased compressor with
108 $\alpha = 1$ and $\Delta = d/m$ with probability $\geq 1 - \delta$,
109 meaning that it reconstruct the $\tilde{\mathbf{g}}$ from input
110 vector \mathbf{g} . In other words, with probability
111 $1 - \delta$, $\text{HEAVYMIX} \in \mathbb{C}(\frac{d}{m}, 1)$. We note
112 that Algorithm 1 is a variation of the sketch-
113 ing algorithm developed in [17] with distinc-
114 tion that HEAVYMIX does not require a second
115 round of communication to obtain the exact
116 values of top_m . This is mainly because in
117 SKETCGED-SGD [17] the server has to obtain
118 the exact values of *the average of sketches*; however HEAVYMIX obtains exact value locally, thus
119 does not require a second round of communication. Additionally, while a sketching algorithm
120 implementing HEAVYMIX has smaller estimation error compared to PRIVIX, it requires having access
121 to the exact values of top_m , therefore not benefiting from privacy properties contrary to PRIVIX. In
122 the following we introduce HEAPRIX which is built upon HEAVYMIX and PRIVIX methods.

Algorithm 1 HEAVYMIX

- 1: **Inputs:** $\mathbf{S}(\mathbf{g})$; parameter m
 - 2: Query the vector $\tilde{\mathbf{g}} \in \mathbb{R}^d$ from $\mathbf{S}(\mathbf{g})$:
 - 3: Query $\hat{\ell}_2^2 = (1 \pm 0.5) \|\mathbf{g}\|^2$ from sketch $\mathbf{S}(\mathbf{g})$
 - 4: $\forall j$ query $\hat{\mathbf{g}}_j^2 = \tilde{\mathbf{g}}_j^2 \pm \frac{1}{2m} \|\mathbf{g}\|^2$ from sketch $\mathbf{S}(\mathbf{g})$
 - 5: $H = \{j | \hat{\mathbf{g}}_j \geq \frac{\hat{\ell}_2^2}{m}\}$ and $NH = \{j | \hat{\mathbf{g}}_j < \frac{\hat{\ell}_2^2}{m}\}$
 - 6: $\text{Top}_m = H \cup \text{rand}_{\ell}(NH)$, where $\ell = m - |H|$
 - 7: Get exact values of Top_m
 - 8: **Output:** $\tilde{\mathbf{g}} : \forall j \in \text{Top}_m : \tilde{\mathbf{g}}_j = \mathbf{g}_j$ else $\mathbf{g}_j = 0$
-

123 2.3 Sketching based Induced Compressor

124 Due to Theorem 3 in [15], which illustrates that we can convert the biased compressor into an
125 unbiased one such that, for $C_1 \in \mathbb{C}(\Delta_1)$ with $\alpha = 1$, if you choose $C_2 \in \mathbb{U}(\Delta_2)$, then in-
126 duced compressor $C : \mathbf{x} \mapsto C_1(\mathbf{x}) + C_2(\mathbf{x} - C_1(\mathbf{x}))$ belongs to $\mathbb{U}(\Delta)$ with $\Delta = \Delta_2 + \frac{1 - \Delta_2}{\Delta_1}$.

Based on this notion, Algorithm 2 proposes an induced sketching algorithm by utilizing HEAVYMIX and PRIVIX for C_1 and C_2 respectively where the reconstruction of input \mathbf{x} is performed using hash table \mathbf{S} and \mathbf{x} , similar to PRIVIX and HEAVYMIX. Note that if $m \rightarrow d$, then $C(\mathbf{x}) \rightarrow \mathbf{x}$, implying that the convergence rate can be improved by decreasing the size of compression m .

Corollary 1. *Based on Theorem 3 of [15], HEAPRIX in Algorithm 2 satisfies $C(\mathbf{x}) \in \mathcal{U}(c \frac{d}{m})$.*

Benefits of HEAPRIX: Corollary 1 states that, unlike PRIVIX, HEAPRIX compression noise can be made as small as possible using larger hash size. In the distributed setting, contrary to SKETCHED-SGD [17] where decompressing is happening at the server, HEAPRIX does not require having access to exact top_m values of the input as it is based on HEAVYMIX, which helps preserving privacy. In other words, HEAPRIX leverages the best of both: the *unbiasedness* of PRIVIX while using *heavy hitters* as in HEAVYMIX.

3 FedSKETCH and FedSKETCHGATE

We introduce two new algorithms for both homogeneous and heterogeneous settings.

3.1 Homogeneous Setting

In FedSKETCH, the number of local updates, between two consecutive communication rounds, at device j is denoted by τ . Unlike [13], server node does not store any global model, rather, device j has two models: $\mathbf{x}^{(r)}$ and $\mathbf{x}_j^{(\ell, r)}$, which are respectively the local and global models. We develop FedSKETCH in Algorithm 3. A variant of this algorithm implementing HEAPRIX is also described in Algorithm 3. We remark that for this variant, we need to have an additional communication round between server and worker j to aggregate $\delta_j^{(r)} \triangleq \mathbf{S}_j [\text{HEAVYMIX}(\mathbf{S}^{(r)})]$ (Lines 3 and 3) to compute $\mathbf{S}^{(r)} = \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S}_j^{(r)}$. The main difference between FedSKETCH and DiffSketch in [26] is that we use distinct local and global learning rates. Furthermore, unlike [26], we do not add local Gaussian noise.

Algorithmic comparison with [13] An important feature of our algorithm is that due to a lower dimension of the count sketch, the resulting averages ($\mathbf{S}^{(r)}$ and $\tilde{\mathbf{S}}^{(r)}$) received by the server are also of lower dimension. Therefore, these algorithms exploit a bidirectional compression

during the communication from server to device back and forth. As a result, for the case of large quantization error $\omega = \theta(\frac{d}{m})$ as shown in [13], our algorithms can outperform FedCOM and FedCOMGATE developed in [13] if sufficiently large hash tables are used and the uplink communication cost is high. Furthermore, while, in [13], server stores a global model and aggregates the partial gradients from devices which can enable the server to extract some information regarding the device's data, in

Algorithm 2 HEAPRIX

1: **Inputs:** $\mathbf{x} \in \mathbb{R}^d, t, m, \mathbf{S}_{m \times t}, h_j (1 \leq i \leq t), \text{sign}_j (1 \leq i \leq t), \text{parameter } m$
2: Approximate $\mathbf{S}(\mathbf{x})$ using HEAVYMIX
3: Approximate $\mathbf{S}(\mathbf{x} - \text{HEAVYMIX}[\mathbf{S}(\mathbf{x})])$ with PRIVIX
4: **Output:**
 $\text{HEAVYMIX}[\mathbf{S}(\mathbf{x})] + \text{PRIVIX}[\mathbf{S}(\mathbf{x} - \text{HEAVYMIX}[\mathbf{S}(\mathbf{x})])]$.

Algorithm 3 FedSKETCH(R, τ, η, γ)

1: **Inputs:** $\mathbf{x}^{(0)}$: initial model shared by local devices, global and local learning rates γ and η , respectively
2: **for** $r = 0, \dots, R - 1$ **do**
3: **parallel for device** $j \in \mathcal{K}^{(r)}$ **do:**
4: **if PRIVIX variant:**
 $\Phi^{(r)} \triangleq \text{PRIVIX}[\mathbf{S}^{(r-1)}]$
5: **if HEAPRIX variant:**
 $\Phi^{(r)} \triangleq \text{HEAVYMIX}[\mathbf{S}^{(r-1)}] + \text{PRIVIX}[\mathbf{S}^{(r-1)} - \tilde{\mathbf{S}}^{(r-1)}]$
6: Set $\mathbf{x}^{(r)} = \mathbf{x}^{(r-1)} - \gamma \Phi^{(r)}$ and $\mathbf{x}_j^{(0, r)} = \mathbf{x}^{(r)}$
7: **for** $\ell = 0, \dots, \tau - 1$ **do**
8: Sample a mini-batch $\xi_j^{(\ell, r)}$ and compute $\tilde{\mathbf{g}}_j^{(\ell, r)}$
9: Update $\mathbf{x}_j^{(\ell+1, r)} = \mathbf{x}_j^{(\ell, r)} - \eta \tilde{\mathbf{g}}_j^{(\ell, r)}$
10: **end for**
11: Device j broadcasts $\mathbf{S}_j^{(r)} \triangleq \mathbf{S}_j(\mathbf{x}_j^{(0, r)} - \mathbf{x}_j^{(\tau, r)})$.
12: Server **computes** $\mathbf{S}^{(r)} = \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S}_j^{(r)}$.
13: Server **broadcasts** $\mathbf{S}^{(r)}$ to devices in randomly drawn devices $\mathcal{K}^{(r)}$.
14: **if HEAPRIX variant:**
15: Second round of communication: $\delta_j^{(r)} := \mathbf{S}_j[\text{HEAVYMIX}(\mathbf{S}^{(r)})]$ and broadcasts $\tilde{\mathbf{S}}^{(r)} \triangleq \frac{1}{k} \sum_{j \in \mathcal{K}} \delta_j^{(r)}$ to devices in set $\mathcal{K}^{(r)}$
16: **end parallel for**
17: **end**
18: **Output:** $\mathbf{x}^{(R-1)}$

contrast, in our algorithms server does not store the global model and only broadcasts the average sketches. Thus, sketching-based server-devices communication algorithms such as ours do not reveal the exact values of the inputs, to preserve privacy as a by-product.

Remark 1. As pointed out in [15], while induced compressors transform a biased compressor into unbiased one, as a drawback it doubles communication cost since the devices need to send $C_1(\mathbf{x})$ and $C_2(\mathbf{x} - C_1(\mathbf{x}))$ separately. We note that in the special case of HEAPRIX, due to the use of sketching, the extra communication round cost is compensated with lower number of bits per round thanks to the lower dimension of sketching.

3.2 Heterogeneous Setting

In this section, we focus on the optimization problem of (1) in the special case of $q_1 = \dots = q_p = \frac{1}{p}$ with full device participation ($k = p$). These results can be extended to the scenario where devices are sampled. For non i.i.d. data, the FedSKETCH algorithm, designed for homogeneous setting, may fail to perform well in practice. The main reason is that in FL, devices are using local stochastic descent direction which could be different than global descent direction when the data distribution are non-identical. Therefore, to mitigate the effect of data heterogeneity, we introduce a new algorithm called FedSKETCHGATE described in Algorithm 4. This algorithm leverages the idea of gradient tracking applied in [13] (with compression) and a special case of $\gamma = 1$ without compression [31]. The main idea is that using an approximation of global gradient, $\mathbf{c}_j^{(r)}$ allows to correct the local gradient direction. For the FedSKETCHGATE with PRIVIX variant, the correction vector $\mathbf{c}_j^{(r)}$ at device j and communication round r is computed in Line 4. While using HEAPRIX compression, FedSKETCHGATE also updates $\tilde{\mathbf{S}}^{(r)}$ via Line 4.

Remark 2. Most of the existing communication-efficient algorithms with compression only consider communication-efficiency from devices to server. However, Algorithms 3 and 4 also improve the communication efficiency from server to devices since it exploits low-dimensional sketches (and averages), communicated from the server to devices.

For both FedSKETCH and FedSKETCHGATE algorithms, unlike PRIVIX, HEAPRIX variant requires a second round of communication. Therefore, in Cross-Device FL setting, where there could be millions of devices, HEAPRIX variant may not be practical, and we note that it could be more suitable for Cross-Silo FL setting.

4 Convergence Analysis

We first state commonly used assumptions required in the following convergence analysis (reminder of our notations can be found Table 1 of the Appendix).

Assumption 1 (Smoothness and Lower Boundedness). *The local objective function $f_j(\cdot)$ of device j is differentiable for $j \in [p]$ and L -smooth, i.e., $\|\nabla f_j(\mathbf{x}) - \nabla f_j(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$, $\forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d$. Moreover, the optimal objective function $f(\cdot)$ is bounded below by $f^* := \min_{\mathbf{x}} f(\mathbf{x}) > -\infty$.*

Algorithm 4 FedSKETCHGATE(R, τ, η, γ)

```

1: Inputs:  $\mathbf{x}^{(0)} = \mathbf{x}_j^{(0)}$  shared by all local devices,
   global and local learning rates  $\gamma$  and  $\eta$ .
2: for  $r = 0, \dots, R - 1$  do
3:   parallel for device  $j = 1, \dots, p$  do
4:     if PRIVIX variant:
        $\mathbf{c}_j^{(r)} = \mathbf{c}_j^{(r-1)} - \frac{1}{\tau} [\text{PRIVIX}(\mathbf{S}^{(r-1)}) - \text{PRIVIX}(\mathbf{S}_j^{(r-1)})]$ 
       where  $\Phi^{(r)} \triangleq \text{PRIVIX}(\mathbf{S}^{(r-1)})$ 
     if HEAPRIX variant:
        $\mathbf{c}_j^{(r)} = \mathbf{c}_j^{(r-1)} - \frac{1}{\tau} (\Phi^{(r)} - \Phi_j^{(r)})$ 
5:   Set  $\mathbf{x}^{(r)} = \mathbf{x}^{(r-1)} - \gamma \Phi^{(r)}$  and  $\mathbf{x}_j^{(0,r)} = \mathbf{x}^{(r)}$ 
6:   for  $\ell = 0, \dots, \tau - 1$  do
7:     Sample mini-batch  $\xi_j^{(\ell,r)}$  and compute  $\tilde{\mathbf{g}}_j^{(\ell,r)}$ 
8:      $\mathbf{x}_j^{(\ell+1,r)} = \mathbf{x}_j^{(\ell,r)} - \eta (\tilde{\mathbf{g}}_j^{(\ell,r)} - \mathbf{c}_j^{(r)})$ 
9:   end for
10:  Device  $j$  broadcasts  $\mathbf{S}_j^{(r)} \triangleq \mathbf{S}(\mathbf{x}_j^{(0,r)} - \mathbf{x}_j^{(\tau,r)})$ .
11:  Server computes  $\mathbf{S}^{(r)} = \frac{1}{p} \sum_{j=1}^p \mathbf{S}_j^{(r)}$  and broad-
    casts  $\mathbf{S}^{(r)}$  to all devices.
12:  if HEAPRIX variant:
13:    Device  $j$  computes  $\Phi_j^{(r)} \triangleq \text{HEAPRIX}[\mathbf{S}_j^{(r)}]$ 
14:    Second round of communication to obtain  $\delta_j^{(r)} :=$ 
       $\mathbf{S}_j(\text{HEAVYMIX}[\mathbf{S}^{(r)}])$ 
15:    Broadcasts  $\tilde{\mathbf{S}}^{(r)} \triangleq \frac{1}{p} \sum_{j=1}^p \delta_j^{(r)}$  to devices
16:  end parallel for
17: end
18: Output:  $\mathbf{x}^{(R-1)}$ 

```

Assumption 1 is common in stochastic optimization. We present our results for PL, convex and general non-convex objectives. [19] show that PL condition implies strong convexity property with same module (PL objectives can also be non-convex, hence strong convexity does not imply PL condition necessarily).

4.1 Convergence of FEDSKETCH

We now focus on the homogeneous case where data is i.i.d. among local devices, and therefore, the stochastic local gradient of each worker is an unbiased estimator of the global gradient. We have:

Assumption 2 (Bounded Variance). *For all $j \in [m]$, we can sample an independent mini-batch ℓ_j of size $|\Xi_j^{(\ell,r)}| = b$ and compute an unbiased stochastic gradient $\tilde{\mathbf{g}}_j = \nabla f_j(\mathbf{x}; \Xi_j)$, $\mathbb{E}_{\Xi_j}[\tilde{\mathbf{g}}_j] = \nabla f(\mathbf{x}) = \mathbf{g}$ with the variance bounded is bounded by a constant σ^2 , i.e., $\mathbb{E}_{\Xi_j}[\|\tilde{\mathbf{g}}_j - \mathbf{g}\|^2] \leq \sigma^2$.*

Theorem 1. *Suppose Assumptions 1-2 hold. Given $0 < m \leq d$ and considering Algorithm 3 with sketch size $B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ and $\gamma \geq k$, with probability $1 - \delta$ we have:*

*In the **non-convex** case, $\{\mathbf{x}^{(r)}\}_{r \geq 0}$ satisfies $\frac{1}{R} \sum_{r=0}^{R-1} \mathbb{E}[\|\nabla f(\mathbf{x}^{(r)})\|_2^2] \leq \epsilon$ if:*

- **FS-PRIVIX**, for $\eta = \frac{1}{L\gamma} \sqrt{\frac{k}{R\tau(\frac{cd}{mk} + 1)}}$: $R = O(1/\epsilon)$ and $\tau = O((d+m)/(mk\epsilon))$.

- **FS-HEAPRIX**, for $\eta = \frac{1}{L\gamma} \sqrt{\frac{k}{R\tau(\frac{cd-m}{mk} + 1)}}$: $R = O(1/\epsilon)$ and $\tau = O(d/(mk\epsilon))$.

*In the **PL or strongly convex** case, $\{\mathbf{x}^{(r)}\}_{r \geq 0}$ satisfies $\mathbb{E}[f(\mathbf{x}^{(R-1)}) - f(\mathbf{x}^{(*)})] \leq \epsilon$ if we set:*

- **FS-PRIVIX**, for $\eta = \frac{1}{2L(cd/mk+1)\tau\gamma}$: $R = O((d/mk+1)\kappa \log(1/\epsilon))$ and $\tau = O((d/m+1)/(d/m+k)\epsilon)$.

- **FS-HEAPRIX**, for $\eta = \frac{1}{2L((cd-m)/mk+1)\tau\gamma}$: $R = O(((d-m)/mk+1)\kappa \log(1/\epsilon))$ and $\tau = O(d/m/(((d-m-1)+k)\epsilon))$.

*In the **Convex** case, $\{\mathbf{x}^{(r)}\}_{r \geq 0}$ satisfies $\mathbb{E}[f(\mathbf{x}^{(R-1)}) - f(\mathbf{x}^{(*)})] \leq \epsilon$ if we set:*

- **FS-PRIVIX**, for $\eta = \frac{1}{2L(cd/mk+1)\tau\gamma}$: $R = O(L(1+d/mk)/\epsilon \log(1/\epsilon))$ and $\tau = O((d/m+1)^2/(k(d/mk+1)^2\epsilon^2))$.

- **FS-HEAPRIX**, for $\eta = \frac{1}{2L((cd-m)/mk+1)\tau\gamma}$: $R = O(L(1+(d-m)/mk)/\epsilon \log(1/\epsilon))$ and $\tau = O((d/m)^2/(k([d-m]/mk+1)^2\epsilon^2))$.

The bounds in Theorem 1 suggest that in homogeneous setting if we set $d = m$ (no compression), the number of communication rounds to achieve the ϵ error matches with the number of iterations required to achieve the same error under a centralized setting. Additionally, computational complexity scales down with number of sampled devices. To stress on the further impact of using sketching, we also compare our results with prior works in terms of total number of communicated bits per device.

Comparison with [17] From privacy aspect, we note [17] requires for server to have access to exact values of top_m gradients, hence do not preserve privacy, whereas our schemes do not need those exact values. From communication cost point of view, for strongly convex objective and compared to [17], we improve the total communication per worker from $RB = O\left(\frac{d}{\epsilon} \log\left(\frac{d}{\delta\sqrt{\epsilon}} \max\left(\frac{d}{m}, \frac{1}{\sqrt{\epsilon}}\right)\right)\right)$ to

$$RB = O\left(\kappa\left(\frac{d-m}{k} + m\right) \log \frac{1}{\epsilon} \log\left(\frac{\kappa d}{\delta}\left(\frac{d-m}{mk} + 1\right) \log \frac{1}{\epsilon}\right)\right).$$

We note that while reducing communication cost, our scheme requires $\tau = O(d/m(k(\frac{d}{mk}+1)\epsilon)) > 1$, which scales down with the number of sampled devices, k . Moreover, unlike [17], we do not use bounded gradient assumption. Therefore, we obtain stronger result with weaker assumptions. Regarding general non-convex objectives, our result improves the total communication cost per worker in [17] from $RB = O\left(\max(\frac{1}{\epsilon^2}, \frac{d^2}{k^2\epsilon}) \log(\frac{d}{\delta} \max(\frac{1}{\epsilon^2}, \frac{d^2}{k^2\epsilon}))\right)$ for *only one device* to $RB =$

272 $O(\frac{m}{\epsilon} \log(\frac{d}{\epsilon\delta}))$. We also highlight that we can obtain similar rates for Algorithm 3 in heterogeneous
 273 environment if we make the additional assumption of uniformly bounded gradient.

274 **Note:** Such improved communication cost over prior related works is due to joint exploitation of
 275 *sketching*, to reduce the dimension of communicated messages, and the use of *local updates*, to
 276 reduce the total number of communication rounds leading to a specific convergence error.

277 4.2 Convergence of FedSKETCHGATE

278 We start with bounded local variance assumption:

279 **Assumption 3** (Bounded Local Variance). *For all $j \in [p]$, we can sample an independent mini-*
 280 *batch Ξ_j of size $|\xi_j| = b$ and compute an unbiased stochastic gradient $\tilde{\mathbf{g}}_j = \nabla f_j(\mathbf{x}; \Xi_j)$ with*
 281 *$\mathbb{E}_{\Xi}[\tilde{\mathbf{g}}_j] = \nabla f_j(\mathbf{x}) = \mathbf{g}_j$. Moreover, the variance of local stochastic gradients is bounded such that*
 282 *$\mathbb{E}_{\Xi}[\|\tilde{\mathbf{g}}_j - \mathbf{g}_j\|^2] \leq \sigma^2$.*

283 **Theorem 2.** *Suppose Assumptions 1 and 3 hold. Given $0 < m \leq d$, and considering*
 284 *FedSKETCHGATE in Algorithm 4 with sketch size $B = O(m \log(\frac{dR}{\delta}))$ and $\gamma \geq p$ with proba-*
 285 *bility $1 - \delta$ we have*

286 *In the **non-convex** case, $\eta = \frac{1}{L\gamma} \sqrt{\frac{mp}{R\tau(cd)}}$, $\{\mathbf{x}^{(r)}\}_{r=0}^{\infty}$ satisfies $\frac{1}{R} \sum_{r=0}^{R-1} \mathbb{E}[\|\nabla f(\mathbf{x}^{(r)})\|_2^2] \leq \epsilon$ if:*

287 • **FS-PRIVIX:**

$$R = O((d + m)/m\epsilon) \quad \text{and} \quad \tau = O(1/(p\epsilon)).$$

288 • **FS-HEAPRIX:** $R = O(d/m\epsilon)$ and $\tau = O(1/(p\epsilon))$.

289 *In the **PL or Strongly convex** case, $\{\mathbf{x}^{(r)}\}_{r=0}^{\infty}$ satisfies $\mathbb{E}[f(\mathbf{x}^{(R-1)}) - f(\mathbf{x}^{(*)})] \leq \epsilon$ if:*

290 • **FS-PRIVIX**, for $\eta = 1/(2L(\frac{cd}{m} + 1)\tau\gamma)$: $R = O((\frac{d}{m} + 1)\kappa \log(1/\epsilon))$ and $\tau = O(1/(p\epsilon))$

291 • **FS-HEAPRIX**, for $\eta = m/(2cLd\tau\gamma)$: $R = O((\frac{d}{m})\kappa \log(1/\epsilon))$ and $\tau = O(1/(p\epsilon))$.

292 *In the **convex** case, $\{\mathbf{x}^{(r)}\}_{r=0}^{\infty}$ satisfies $\mathbb{E}[f(\mathbf{x}^{(R-1)}) - f(\mathbf{x}^{(*)})] \leq \epsilon$ if:*

293 • **FS-PRIVIX**, for $\eta = 1/(2L(cd/m + 1)\tau\gamma)$: $R = O(L(d/m + 1)\epsilon \log(1/\epsilon))$ and $\tau =$
 294 $O(1/(p\epsilon^2))$.

295 • **FS-HEAPRIX**, for $\eta = m/(2Lcd\tau\gamma)$: $R = O(L(d/m)\epsilon \log(1/\epsilon))$ and $\tau = O(1/(p\epsilon^2))$.

296 Theorem 2 implies that the number of communication rounds and local updates are similar to the
 297 corresponding quantities in homogeneous setting except for the non-convex case where the number
 298 of rounds also depends on the compression rate (summarized Table 2-3 of the Appendix).

299 4.3 Comparison with Prior Methods

300 Before comparing with prior works, we highlight that privacy is another purpose of using unbiased
 301 sketching in addition to communication efficiency. Therefore, our main competing schemes are
 302 distributed algorithms based on sketching. Nonetheless, for the sake of showing the effectiveness of
 303 our algorithms, we also compare with prior non-sketching based distributed algorithms ([20, 3, 36,
 304 13]) in Section B of Appendix.

305 **Comparison with [26].** Note that our convergence analysis does not rely on the bounded gradient
 306 assumption. We also improve both the number of communication rounds R and the size of transmitted
 307 bits B per communication round. Additionally, we highlight that, while [26] provides a convergence
 308 analysis for convex objectives, our analysis holds for PL (thus strongly convex case), general convex
 309 and general non-convex objectives.

310 **Comparison with [37].** Due to gradient tracking, our algorithm tackles data heterogeneity issue,
 311 while algorithms in [37] does not particularly. As a consequence, in FedSKETCHGATE each device
 312 has to store an additional state vector compared to [37]. Yet, as our method is built upon an
 313 unbiased compressor, server does not need to store any additional error correction vector. The
 314 convergence results for both of two variants of FenchSGD in [37] rely on the uniform bounded gradient
 315 assumption which may not be applicable with L -smoothness assumption when data distribution
 316 is highly heterogeneous, as in FL, see [21], while our bounds do not assume such boundedness.

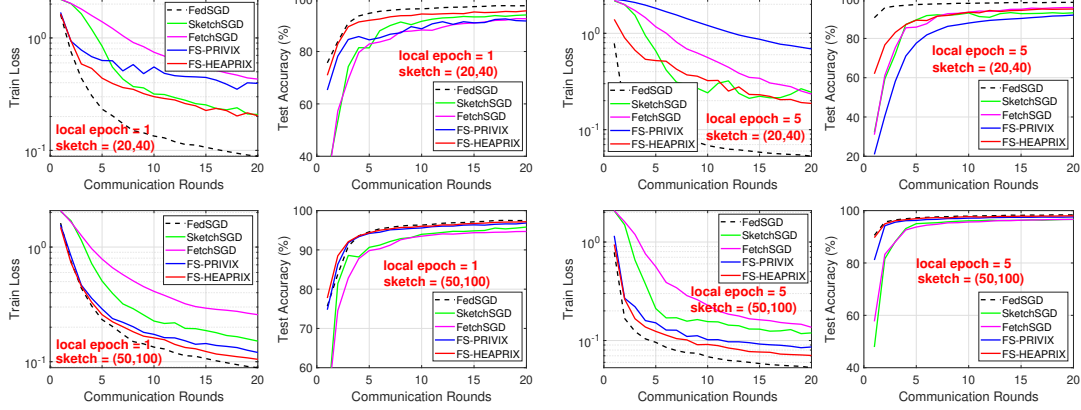


Figure 1: Homogeneous case: Comparison of compressed optimization methods on LeNet CNN.

Besides, Theorem 1 [37] assumes that *Contraction Holds* for the sequence of gradients which may not hold in practice, yet based on this strong assumption, their total communication cost (RB) in order to achieve ϵ error is $RB = O\left(m \max\left(\frac{1}{\epsilon^2}, \frac{d^2 - dm}{m^2 \epsilon}\right) \log\left(\frac{d}{\delta} \max\left(\frac{1}{\epsilon^2}, \frac{d^2 - dm}{m^2 \epsilon}\right)\right)\right)$. For the sake of comparison we let the compression ratio in [37] to be $\frac{m}{d}$. In contrast, without any extra assumptions, our results in Theorem 2 for PRIVIX and HEAPRIX are respectively $RB = O\left(\frac{(d+m)}{\epsilon} \log\left(\frac{d^2}{\epsilon \delta} + d\right)\right)$ and $RB = O\left(\frac{d}{\epsilon} \log\left(\frac{d^2}{\epsilon m \delta}\right)\right)$ which improves the total communication cost of Theorem 1 in [37] under regimes such that $\frac{1}{\epsilon} \geq d$ or $d \gg m$. Theorem 2 in [37] is based the *Sliding Window Heavy Hitters* assumption, which is similar to the gradient diversity assumption in [29, 12]. Under that assumption the total communication cost is shown to be $RB = O\left(\frac{m \max(I^{2/3}, 2 - \alpha)}{\epsilon^3 \alpha} \log\left(\frac{d \max(I^{2/3}, 2 - \alpha)}{\epsilon^3 \delta}\right)\right)$ where I is a constant related to the window of gradients. We improve this bound under weaker assumptions in a regime where $\frac{I^{2/3}}{\epsilon^2} \geq d$. We also provide bounds for PL, convex and non-convex objectives contrary to [37]. Finally, we note that algorithms in [37] are using momentum at server. While we do not use it explicitly, we can modify our algorithms to include momentum easily.

5 Numerical Study

In this section, we provide empirical results on MNIST benchmark dataset to demonstrate the effectiveness of our proposed algorithms. We train LeNet-5 Convolutional Neural Network (CNN) architecture introduced in [24], with 60 000 parameters. We compare Federated SGD (FedSGD) as the full-precision baseline, along with four sketching methods SketchSGD [17], FetchSGD [37], and two FedSketch variants FS-PRIVIX and FS-HEAPRIX. Note that in Algorithm 3, FS-PRIVIX with global learning rate $\gamma = 1$ is equivalent to the DiffSketch algorithm proposed in [29]. Also, SketchSGD is slightly modified to compress the change in local weights (instead of local gradient in every iteration), and FetchSGD is implemented with second round of communication for fairness. (The original proposal does not include second round of communication, which performs worse with small sketch size.) As suggested in [37], the momentum factor of FetchSGD is set to 0.9, and we also follow some recommended implementation tricks to improve its performance, which are detailed in the Appendix. The number of workers is set to 50 and we report the results for 1 and 5 local epochs. A local epoch is finished when all workers go through their local data samples once. The local batch size is 30. In each round, we randomly choose half of the devices to be active. We tune the learning rates (η and γ , if applicable) over log-scale and report the best results, for both *homogeneous* and *heterogeneous* setting. In the former case, each device receives uniformly drawn data samples, and in the latter, it only receives samples from one or two classes among ten.

Homogeneous case. In Figure 1, we provide the training loss and test accuracy with different number of local epochs and sketch size, $(t, k) = (20, 40)$ and $(50, 100)$. Note that, these two choices of sketch size correspond to a $75\times$ and $12\times$ compression ratio, respectively. We conclude

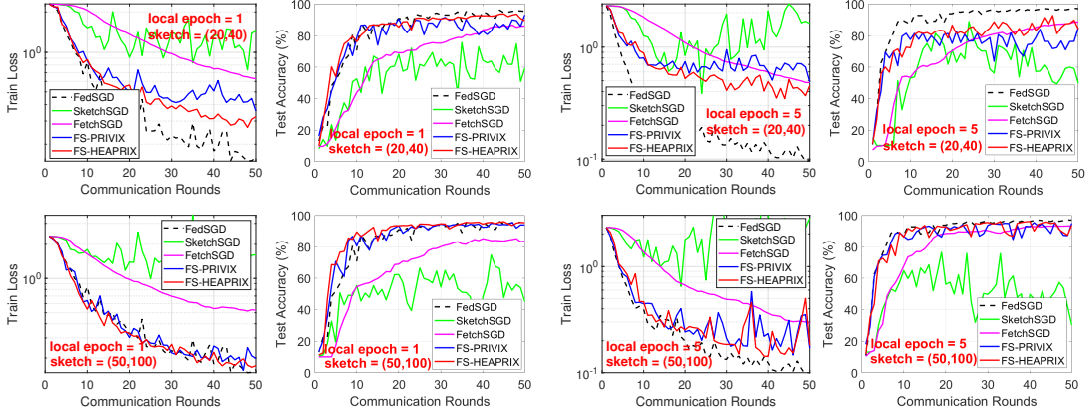


Figure 2: Heterogeneous case: Comparison of compressed optimization algorithms on LeNet CNN.

- In general, increasing compression ratio would sacrifice learning performance. In all cases, FS-HEAPRIX performs the best in terms of both training objective and test accuracy, among all compressed methods.
- FS-HEAPRIX is better than FS-PRIVIX, especially with small sketches (high compression ratio). FS-HEAPRIX yields acceptable extra test error compared to full-precision FedSGD, particularly when considering the high compression ratio (e.g., $75\times$).
- From the training loss, we see that the performance of FS-HEAPRIX improves when the number of local updates increases. *That is, the proposed method is able to further reduce the communication cost by reducing the number of rounds required for communication.* This is also consistent with our theoretical findings.

In general, our proposed FS-HEAPRIX outperforms all competing methods, and a sketch size of $(50, 100)$ is sufficient to approach the accuracy of full-precision FedSGD.

Heterogeneous case. We plot similar set of results in Figure 2 for non-i.i.d. data distribution, which leads to more twists and turns in the training curves. We see that SketchSGD performs very poorly in the heterogeneous case, which is improved by error tracking and momentum in FetchSGD, as expected. However, both of these methods are worse than our proposed FedSketchGATE methods, which can achieve similar generalization accuracy as full-precision FedSGD, even with small sketch size (i.e., $75\times$ compression with 1 local epoch). Note that, slower convergence and worse generalization of FedSGD in non-i.i.d. data distribution case is also reported in e.g. [33, 8].

We also notice in Figure 2 the edge of FS-HEAPRIX over FS-PRIVIX in terms of training loss and test accuracy. However, we see that in the heterogeneous setting, more local updates tend to undermine the learning performance, especially with small sketch size. Nevertheless, when the sketch size is not too small, i.e., $(50, 100)$, FS-HEAPRIX can still provide comparable test accuracy as FedSGD in both cases. Our empirical study demonstrates that FedSketch (and FedSketchGATE) frameworks are able to perform well in homogeneous (resp. heterogeneous) settings, with high compression rate. In particular, FedSketch methods are beneficial over SketchedSGD [17] and FetchSGD [37] in all cases. FS-HEAPRIX performs the best among all the tested compressed algorithms, which in many cases achieves similar generalization accuracy as full-precision FedSGD with small sketch size.

6 Conclusion

In this paper, we introduced FedSKETCH and FedSKETCHGATE algorithms for homogeneous and heterogeneous data distribution setting respectively for Federated Learning wherein communication between server and devices is only performed using count sketch. Our algorithms, thus, provide communication-efficiency and privacy, through random hashes based sketches. We analyze the convergence error for *non-convex*, *PL* and *general convex* objective functions in the scope of Federated Optimization. We provide insightful numerical experiments showcasing the advantages of our FedSKETCH and FedSKETCHGATE methods over current federated optimization algorithm. The proposed algorithms outperform competing compression method and can achieve comparable test accuracy as Federated SGD, with high compression ratio.

References

- [1] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1709–1720, Long Beach, 2017.
- [2] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli. The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5973–5983, Montréal, Canada, 2018.
- [3] D. Basu, D. Data, C. Karakus, and S. N. Diggavi. Qsparse-local-sgd: Distributed SGD with quantization, sparsification and local computations. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 14668–14679, Vancouver, Canada, 2019.
- [4] J. Bernstein, Y. Wang, K. Azizzadenesheli, and A. Anandkumar. SIGNSGD: compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pages 559–568, Stockholmsmässan, Stockholm, Sweden, 2018.
- [5] K. Bonawitz, V. Ivanov, B. Kreuter, A. Marcedone, H. B. McMahan, S. Patel, D. Ramage, A. Segal, and K. Seth. Practical secure aggregation for privacy-preserving machine learning. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS)*, pages 1175–1191, Dallas, TX, 2017.
- [6] L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 161–168, Vancouver, Canada, 2008.
- [7] M. Charikar, K. C. Chen, and M. Farach-Colton. Finding frequent items in data streams. *Theoretical Computer Science*, 312(1):3–15, 2004. doi: 10.1016/S0304-3975(03)00400-6. URL [https://doi.org/10.1016/S0304-3975\(03\)00400-6](https://doi.org/10.1016/S0304-3975(03)00400-6).
- [8] X. Chen, X. Li, and P. Li. Toward communication efficient adaptive gradient method. In *ACM-IMS Foundations of Data Science Conference (FODS)*, Seattle, WA, 2020.
- [9] G. Cormode and S. Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.
- [10] C. Dwork. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II*, volume 4052 of *Lecture Notes in Computer Science*, pages 1–12. Springer, 2006.
- [11] R. C. Geyer, T. Klein, and M. Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.
- [12] F. Haddadpour and M. Mahdavi. On the convergence of local descent methods in federated learning. *arXiv preprint arXiv:1910.14425*, 2019.
- [13] F. Haddadpour, M. M. Kamani, A. Mokhtari, and M. Mahdavi. Federated learning with compression: Unified analysis and sharp guarantees. *arXiv preprint arXiv:2007.01154*, 2020.
- [14] S. Hardy, W. Henecka, H. Ivey-Law, R. Nock, G. Patrini, G. Smith, and B. Thorne. Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption. *arXiv preprint arXiv:1711.10677*, 2017.
- [15] S. Horváth and P. Richtárik. A better alternative to error feedback for communication-efficient distributed learning. *arXiv preprint arXiv:2006.11077*, 2020.
- [16] S. Horváth, D. Kovalev, K. Mishchenko, S. Stich, and P. Richtárik. Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*, 2019.
- [17] N. Ivkin, D. Rothchild, E. Ullah, V. Braverman, I. Stoica, and R. Arora. Communication-efficient distributed SGD with sketching. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 13144–13154, Vancouver, Canada, 2019.
- [18] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.
- [19] H. Karimi, J. Nutini, and M. Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases (ECML-PKDD)*, pages 795–811, Riva del Garda, Italy, 2016.

- [20] S. P. Karimireddy, S. Kale, M. Mohri, S. J. Reddi, S. U. Stich, and A. T. Suresh. Scaffold: Stochastic controlled averaging for on-device federated learning. *arXiv preprint arXiv:1910.06378*, 2019.
- [21] A. Khaled, K. Mishchenko, and P. Richtárik. Tighter theory for local SGD on identical and heterogeneous data. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 4519–4529, Online [Palermo, Sicily, Italy], 2020.
- [22] J. Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7(4):373–397, 2003.
- [23] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.
- [24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- [25] P. Li, K. W. Church, and T. Hastie. One sketch for all: Theory and application of conditional random sampling. In *Advances in Neural Information Processing Systems (NIPS)*, pages 953–960, Vancouver, Canada, 2008.
- [26] T. Li, Z. Liu, V. Sekar, and V. Smith. Privacy for free: Communication-efficient learning with differential privacy using sketches. *arXiv preprint arXiv:1911.00972*, 2019.
- [27] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Process. Mag.*, 37(3):50–60, 2020.
- [28] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.
- [29] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith. Federated optimization in heterogeneous networks. In *Proceedings of Machine Learning and Systems (MLSys)*, Austin, TX, 2020.
- [30] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang. On the convergence of fedavg on non-iid data. In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia, 2020.
- [31] X. Liang, S. Shen, J. Liu, Z. Pan, E. Chen, and Y. Cheng. Variance reduced local sgd with lower communication complexity. *arXiv preprint arXiv:1912.12844*, 2019.
- [32] Y. Lin, S. Han, H. Mao, Y. Wang, and B. Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.
- [33] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 1273–1282, Fort Lauderdale, FL, 2017.
- [34] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning differentially private recurrent language models. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*, Vancouver, Canada, 2018.
- [35] C. Philippenko and A. Dieuleveut. Artemis: tight convergence guarantees for bidirectional compression in federated learning. *arXiv preprint arXiv:2006.14591*, 2020.
- [36] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani. Fedpaq: A communication-efficient federated learning method with periodic averaging and quantization. In *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 2021–2031, Online [Palermo, Sicily, Italy], 2020.
- [37] D. Rothchild, A. Panda, E. Ullah, N. Ivkin, I. Stoica, V. Braverman, J. Gonzalez, and R. Arora. FetchSGD: Communication-efficient federated learning with sketching. *arXiv preprint arXiv:2007.07682*, 2020.
- [38] A. K. Sahu, T. Li, M. Sanjabi, M. Zaheer, A. Talwalkar, and V. Smith. On the convergence of federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.
- [39] S. U. Stich. Local sgd converges fast and communicates little. In *Proceedings of the 7th International Conference on Learning Representations (ICLR)*, New Orleans, LA, 2019.

- 495 [40] S. U. Stich and S. P. Karimireddy. The error-feedback framework: Better rates for sgd with
496 delayed gradients and compressed communication. *arXiv preprint arXiv:1909.05350*, 2019.
- 497 [41] S. U. Stich, J.-B. Cordonnier, and M. Jaggi. Sparsified sgd with memory. In *Advances in Neural*
498 *Information Processing Systems (NeurIPS)*, pages 4447–4458, Montréal, Canada, 2018.
- 499 [42] H. Tang, S. Gan, C. Zhang, T. Zhang, and J. Liu. Communication compression for decentralized
500 training. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7652–7662,
501 Montréal, Canada, 2018.
- 502 [43] H. Tang, C. Yu, X. Lian, T. Zhang, and J. Liu. Doublesqueeze: Parallel stochastic gradient
503 descent with double-pass error-compensated compression. In *International Conference on*
504 *Machine Learning*, pages 6155–6165. PMLR, 2019.
- 505 [44] J. Wang and G. Joshi. Cooperative sgd: A unified framework for the design and analysis of
506 communication-efficient sgd algorithms. *arXiv preprint arXiv:1808.07576*, 2018.
- 507 [45] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li. Terngrad: Ternary gradients
508 to reduce communication in distributed deep learning. In *Advances in neural information*
509 *processing systems (NIPS)*, pages 1509–1519, Long Beach, CA, 2017.
- 510 [46] J. Wu, W. Huang, J. Huang, and T. Zhang. Error compensated quantized sgd and its applications
511 to large-scale distributed optimization. *arXiv preprint arXiv:1806.08054*, 2018.
- 512 [47] H. Yu, R. Jin, and S. Yang. On the linear speedup analysis of communication efficient momen-
513 tum SGD for distributed non-convex optimization. In *Proceedings of the 36th International*
514 *Conference on Machine Learning (ICML)*, pages 7184–7193, Long Beach, CA, 2019.
- 515 [48] H. Yu, S. Yang, and S. Zhu. Parallel restarted SGD with faster convergence and less communi-
516 cation: Demystifying why model averaging works for deep learning. In *The Thirty-Third AAAI*
517 *Conference on Artificial Intelligence (AAAI)*, pages 5693–5700, Honolulu, HI, 2019.
- 518 [49] S. Zheng, Z. Huang, and J. T. Kwok. Communication-efficient distributed blockwise momentum
519 sgd with error-feedback. *arXiv preprint arXiv:1905.10936*, 2019.
- 520 [50] F. Zhou and G. Cong. On the convergence properties of a k-step averaging stochastic gradient
521 descent algorithm for nonconvex optimization. In *Proceedings of the Twenty-Seventh Inter-*
522 *national Joint Conference on Artificial Intelligence (IJCAI)*, pages 3219–3227, Stockholm,
523 Sweden, 2018.

524 Checklist

- 525 1. For all authors...
- 526 (a) Do the main claims made in the abstract and introduction accurately reflect the paper's
527 contributions and scope? [Yes]
- 528 (b) Did you describe the limitations of your work? [Yes]
- 529 (c) Did you discuss any potential negative societal impacts of your work? [N/A]
- 530 (d) Have you read the ethics review guidelines and ensured that your paper conforms to
531 them? [Yes]
- 532 2. If you are including theoretical results...
- 533 (a) Did you state the full set of assumptions of all theoretical results? [Yes]
- 534 (b) Did you include complete proofs of all theoretical results? [Yes]
- 535 3. If you ran experiments...
- 536 (a) Did you include the code, data, and instructions needed to reproduce the main exper-
537 imental results (either in the supplemental material or as a URL)? [Yes] The code is
538 available upon demand.
- 539 (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they
540 were chosen)? [Yes]
- 541 (c) Did you report error bars (e.g., with respect to the random seed after running experi-
542 ments multiple times)? [No] Results are averaged over several runs.
- 543 (d) Did you include the total amount of compute and the type of resources used (e.g., type
544 of GPUs, internal cluster, or cloud provider)? [N/A]
- 545 4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
- 546 (a) If your work uses existing assets, did you cite the creators? [Yes]
- 547 (b) Did you mention the license of the assets? [N/A]
- 548 (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
- 549
- 550 (d) Did you discuss whether and how consent was obtained from people whose data you're
551 using/curating? [N/A]
- 552 (e) Did you discuss whether the data you are using/curating contains personally identifiable
553 information or offensive content? [N/A]
- 554 5. If you used crowdsourcing or conducted research with human subjects...
- 555 (a) Did you include the full text of instructions given to participants and screenshots, if
556 applicable? [N/A]
- 557 (b) Did you describe any potential participant risks, with links to Institutional Review
558 Board (IRB) approvals, if applicable? [N/A]
- 559 (c) Did you include the estimated hourly wage paid to participants and the total amount
560 spent on participant compensation? [N/A]

561 Appendix for FedSKETCH: Communication-Efficient Federated 562 Learning via Sketching

563 The appendix is organized as follows: Section A recalls important notations used throughout the
564 paper and provides the formulation of related algorithms used in the main paper and omitted for the
565 sake of the page limit. We present in Section B of this supplementary file, a through comparison with
566 notable related works. Section C contains the proofs of our results and Section D presents additional
567 numerical runs.

568 A Notations and Definitions

569 **Notation.** Here we denote the count sketch of the vector \mathbf{x} by $\mathbf{S}(\mathbf{x})$ and with an abuse of notation,
570 we indicate the expectation over the randomness of count sketch with $\mathbb{E}_{\mathbf{S}}[\cdot]$. We illustrate the random
571 subset of the devices selected by the central server with \mathcal{K} with size $|\mathcal{K}| = k \leq p$, and we represent
572 the expectation over the device sampling with $\mathbb{E}_{\mathcal{K}}[\cdot]$.

Table 1: Table of Notations

| | | |
|--------------------------|--------------|---|
| p | \triangleq | Number of devices |
| k | \triangleq | Number of sampled devices for homogeneous setting |
| $\mathcal{K}^{(r)}$ | \triangleq | Set of sampled devices in communication round r |
| d | \triangleq | Dimension of the model |
| τ | \triangleq | Number of local updates |
| R | \triangleq | Number of communication rounds |
| B | \triangleq | Size of transmitted bits |
| $R \times B$ | \triangleq | Total communication cost per device |
| κ | \triangleq | Condition number |
| ϵ | \triangleq | Target accuracy |
| μ | \triangleq | PL constant |
| m | \triangleq | Number of bins of hash tables |
| $\mathbf{S}(\mathbf{x})$ | \triangleq | Count sketch of the vector \mathbf{x} |
| $\mathbb{U}(\Delta)$ | \triangleq | Class of unbiased compressor, see Definition 1 |

573 **Definition 3** (Polyak-Łojasiewicz). *A function $f(\mathbf{x})$ satisfies the Polyak-Łojasiewicz(PL) condition*
574 *with constant μ if $\frac{1}{2}\|\nabla f(\mathbf{x})\|_2^2 \geq \mu(f(\mathbf{x}) - f(\mathbf{x}^*))$, $\forall \mathbf{x} \in \mathbb{R}^d$ with \mathbf{x}^* is an optimal solution.*

575 A.1 Count sketch

576 In this paper, we exploit the commonly used Count Sketch [7] which is described in Algorithm 5.

Algorithm 5 Count Sketch (CS) [7]

```

1: Inputs:  $\mathbf{x} \in \mathbb{R}^d, t, k, \mathbf{S}_{m \times t}, h_j(1 \leq i \leq t), \text{sign}_j(1 \leq i \leq t)$ 
2: Compress vector  $\mathbf{x} \in \mathbb{R}^d$  into  $\mathbf{S}(\mathbf{x})$ :
3: for  $x_i \in \mathbf{x}$  do
4:   for  $j = 1, \dots, t$  do
5:      $\mathbf{S}[j][h_j(i)] = \mathbf{S}[j-1][h_{j-1}(i)] + \text{sign}_j(i) \cdot x_i$ 
6:   end for
7: end for
8: return  $\mathbf{S}_{m \times t}(\mathbf{x})$ 

```

577 **A.2 PRIVIX method and compression error of HEAPRIX**

578 For the sake of completeness we review PRIVIX algorithm that is also mentioned in [26] as follows:

Algorithm 6 PRIVIX/DiffSketch [26]: Unbiased compressor based on sketching.

```

1: Inputs:  $x \in \mathbb{R}^d, t, m, \mathbf{S}_{m \times t}, h_j(1 \leq i \leq t), \text{sign}_j(1 \leq i \leq t)$ 
2: Query  $\tilde{x} \in \mathbb{R}^d$  from  $\mathbf{S}(x)$ :
3: for  $i = 1, \dots, d$  do
4:    $\tilde{x}[i] = \text{Median}\{\text{sign}_j(i) \cdot \mathbf{S}[j][h_j(i)] : 1 \leq j \leq t\}$ 
5: end for
6: Output:  $\tilde{x}$ 

```

579 Regarding the compression error of sketching we restate the following Corollary from the main body
580 of this paper:

581 **Corollary 2.** *Based on Theorem 3 of [15] and using Algorithm 2, we have $C(x) \in \mathbb{U}(c \frac{d}{m})$. This*
582 *shows that unlike PRIVIX (Algorithm 6) the compression noise can be made as small as possible*
583 *using large size of hash table.*

584 *Proof.* The proof simply follows from Theorem 3 in [15] and Algorithm 2 by setting $\Delta_1 = c \frac{d}{m}$
585 and $\Delta_2 = 1 + c \frac{d}{m}$ we obtain $\Delta = \Delta_2 + \frac{1-\Delta_2}{\Delta_1} = c \frac{d}{m} = O\left(\frac{d}{m}\right)$ for the compression error of
586 HEAPRIX. □

587 B Summary of comparison of our results with prior works

588 For the purpose of further clarification, we summarize the comparison of our results with related
589 works. We recall that p is the number of devices, d is the dimension of the model, κ is the condition
590 number, ϵ is the target accuracy, R is the number of communication rounds, and τ is the number of
591 local updates. We start with the homogeneous setting comparison. Comparison of our results and
592 existing ones for homogeneous and heterogeneous setting are given respectively Table 2 and Table 3.

Table 2: Comparison of results with compression and periodic averaging in the homogeneous setting. UG and PP stand for Unbounded Gradient and Privacy Property respectively.

| Reference | PL/Strongly Convex | UG | PP |
|--------------------------|--|----|----|
| Ivkin et al. [17] | $R = O\left(\max\left(\frac{d}{m\sqrt{\epsilon}}, \frac{1}{\epsilon}\right)\right), \tau = 1, B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ $pRB = O\left(\frac{pd}{m\epsilon} \log\left(\frac{d}{\delta\sqrt{\epsilon}} \max\left(\frac{d}{m}, \frac{1}{\sqrt{\epsilon}}\right)\right)\right)$ | ✗ | ✗ |
| Theorem 1 | $R = O\left(\kappa\left(\frac{d-m}{mk} + 1\right) \log\left(\frac{1}{\epsilon}\right)\right), \tau = O\left(\frac{d}{k\left(\frac{d}{k} + m\right)\epsilon}\right), B = O\left(m \log\left(\frac{dR}{\delta}\right)\right)$ $kRB = O\left(m\kappa(d-m+mk) \log\left(\frac{1}{\epsilon}\right) \log\left(\frac{\kappa(d\frac{d-m}{mk} + d) \log\left(\frac{1}{\epsilon}\right)}{\delta}\right)\right)$ | ✓ | ✓ |

Table 3: Comparison of results with compression and periodic averaging in the heterogeneous setting. UG and PP stand for Unbounded Gradient and Privacy Property respectively.

| Reference | non-convex | General Convex | UG | PP |
|---|--|---|----|----|
| Basu et al. [3] (with $\gamma = m/d$) | $R = O\left(\frac{d}{m\epsilon^{1.5}}\right)$ $\tau = O\left(\frac{m}{pd\sqrt{\epsilon}}\right)$ $B = O(d)$ $RB = O\left(\frac{d^2}{m\epsilon^{1.5}}\right)$ | — | ✗ | ✗ |
| Li et al. [26] | — | $R = O\left(\frac{d}{m\epsilon^2}\right)$ $\tau = 1$ $B = O\left(m \log\left(\frac{d^2}{m\epsilon^2\delta}\right)\right)$ | ✗ | ✓ |
| Rothchild et al. [37] | $R = O\left(\max\left(\frac{1}{\epsilon^2}, \frac{d^2-md}{m^2\epsilon}\right)\right)$ $\tau = 1$ $B = O\left(m \log\left(\frac{d}{\delta} \max\left(\frac{1}{\epsilon^2}, \frac{d^2-md}{m^2\epsilon}\right)\right)\right)$ $RB = O\left(m \max\left(\frac{1}{\epsilon^2}, \frac{d^2-md}{m^2\epsilon}\right) \log\left(\frac{d}{\delta} \max\left(\frac{1}{\epsilon^2}, \frac{d^2-md}{m^2\epsilon}\right)\right)\right)$ | — | ✗ | ✗ |
| Rothchild et al. [37] | $R = O\left(\frac{\max(I^{2/3}, 2-\alpha)}{\epsilon^3}\right)$ $\tau = 1$ $B = O\left(\frac{m}{\alpha} \log\left(\frac{d \max(I^{2/3}, 2-\alpha)}{\epsilon^3\delta}\right)\right)$ $RB = O\left(\frac{m \max(I^{2/3}, 2-\alpha)}{\epsilon^3\alpha} \log\left(\frac{d \max(I^{2/3}, 2-\alpha)}{\epsilon^3\delta}\right)\right)$ | — | ✗ | ✗ |
| Theorem 2 | $R = O\left(\frac{d}{m\epsilon}\right)$ $\tau = O\left(\frac{1}{p\epsilon}\right)$ $B = O\left(m \log\left(\frac{d^2}{m\epsilon\delta}\right)\right)$ $RB = O\left(\frac{d}{\epsilon} \log\left(\frac{d^2}{m\epsilon\delta} \log\left(\frac{1}{\epsilon}\right)\right)\right)$ | $R = O\left(\frac{d}{m\epsilon} \log\left(\frac{1}{\epsilon}\right)\right)$ $\tau = O\left(\frac{1}{p\epsilon^2}\right)$ $B = O\left(m \log\left(\frac{d^2}{m\epsilon\delta}\right)\right)$ | ✓ | ✓ |

593 **Comparison with [13] and [36]** Convergence analysis of algorithms in [13] relies on unbiased compression,
594 while in this paper our FL algorithm based on HEAPRIX enjoys from unbiased compression
595 with equivalent biased compression variance. Moreover, we highlight that the convergence analysis
596 of FedCOMGATE is based on the extra assumption of boundedness of the difference between the
597 average of compressed vectors and compressed averages of vectors. However, we do not need this
598 extra assumption as it is satisfied naturally due to linearity of sketching. Finally, as pointed out
599 in Remark 2, our algorithms enjoy from a bidirectional compression property, unlike FedCOMGATE
600 in general. Furthermore, since results in [13] improve the communication complexity of FedPAQ
601 algorithm, developed in [36], hence FedSKETCH and FedSKETCHGATE improves the communication
602 complexity obtained in [36].

603 **Comparison with [3].** We note that the algorithm in [3] uses a composed compression and quantiza-
604 tion while our algorithm is solely based on compression. So, in order to compare with algorithms
605 in [3] we only consider Qsparse-local-SGD with compression and we let compression factor $\gamma = \frac{m}{d}$
606 (to compare with the same compression ratio induced with sketch size of mt). For strongly convex
607 objective in Qsparse-local-SGD to achieve convergence error of ϵ they require $R = O\left(\kappa \frac{d}{m\sqrt{\epsilon}}\right)$ and
608 $\tau = O\left(\frac{m}{pd\sqrt{\epsilon}}\right)$, which is improved to $R = O\left(\frac{\kappa d}{m} \log(1/\epsilon)\right)$ and $\tau = O\left(\frac{1}{p\epsilon}\right)$ for PL objectives.
609 Similarly, for non-convex objective [3] requires $R = O\left(\frac{d}{m\epsilon^{1.5}}\right)$ and $\tau = O\left(\frac{m}{pd\sqrt{\epsilon}}\right)$, which is
610 improved to $R = O\left(\frac{d}{m\epsilon}\right)$ and $\tau = O\left(\frac{1}{p\epsilon}\right)$. We note that we reduce communication rounds at the
611 cost of increasing number of local updates (which scales down with number of devices, p). Addi-
612 tionally, we highlight that our FedSKETCHGATE exploits the gradient tracking idea to deal with data
613 heterogeneity, while algorithms in [3] does not develop such mechanism and may suffer from poor
614 convergence in heterogeneous setting. We also note that setting $\tau = 1$ and using top_m compressor,
615 the QSPARSE-local-SGD algorithm becomes similar to distributed SGD with sketching as they both
616 use the error feedback framework to improve the compression variance. Finally, since the average of
617 sparse vectors may not be sparse in general the number of transmitted bits from server to devices in
618 QSPARSE-Local-SGD in [3] may not be sparse in general ($B = O(d)$), however our algorithms enjoy
619 from bidirectional compression properly due to lower dimension and linearity properties of sketch-
620 ing ($B = O(m \log(\frac{Rd}{\delta}))$). Therefore, the total number of bits per device for strongly convex and
621 non-convex objective is improved respectively from $RB = O\left(\kappa \frac{d^2}{m\sqrt{\epsilon}}\right)$ and $RB = O\left(\frac{d^2}{m\epsilon^{1.5}}\right)$
622 in [3] to $RB = O\left(\kappa d \log(\frac{\kappa d^2}{m\delta} \log(\frac{1}{\epsilon})) \log(1/\epsilon)\right) = O\left(\kappa d \max\left(\log(\frac{\kappa d^2}{m\delta}), \log^2(1/\epsilon)\right)\right)$ and
623 $RB = O\left(\log(\frac{d^2}{m\epsilon\delta}) \frac{d}{\epsilon}\right)$.

624 Additionally, as we noted using sketching for transmission implies two way communication from
625 master to devices and vice versa. Therefore, in order to show efficacy of our algorithm we compare
626 our convergence analysis with the obtained rates in the following related work:

627 **Comparison with [35].** The reference [35] considers two-way compression from parameter server to
628 devices and vice versa. They provide the convergence rate of $R = O\left(\frac{\omega^{\text{Up}} \omega^{\text{Down}}}{\epsilon^2}\right)$ for strongly-objective
629 functions where ω^{Up} and ω^{Down} are uplink and downlink's compression noise (specializing to our
630 case for the sake of comparison $\omega^{\text{Up}} = \omega^{\text{Down}} = \theta(d)$) for general heterogeneous data distribution.
631 In contrast, while our algorithms are using bidirectional compression due to use of sketching for
632 communication, our convergence rate for strongly-convex objective is $R = O(\kappa \mu^2 d \log(\frac{1}{\epsilon}))$ with
633 probability $1 - \delta$.

634 We would like to also mention that there prior studies such as [43] and [49] that analyze the two-way
635 compression, but since [35] is the state-of-the-art on this topic we only compared our results with
636 these papers.

637 C Theoretical Proofs

638 We will use the following fact (which is also used in [30, 12]) in proving results.

639 **Fact 3** ([30, 12]). *Let $\{x_i\}_{i=1}^p$ denote any fixed deterministic sequence. We sample a multiset \mathcal{P} (with*
 640 *size K) uniformly at random where x_j is sampled with probability q_j for $1 \leq j \leq p$ with replacement.*
 641 *Let $\mathcal{P} = \{i_1, \dots, i_K\} \subset [p]$ (some i_j s may have the same value). Then*

$$\mathbb{E}_{\mathcal{P}} \left[\sum_{i \in \mathcal{P}} x_i \right] = \mathbb{E}_{\mathcal{P}} \left[\sum_{k=1}^K x_{i_k} \right] = K \mathbb{E}_{\mathcal{P}} [x_{i_k}] = K \left[\sum_{j=1}^p q_j x_j \right] \quad (2)$$

642 For the sake of the simplicity, we review an assumption for the quantization/compression, that
 643 naturally holds for PRIVIX and HEAPRIX.

644 **Assumption 4** ([13]). *The output of the compression operator $Q(\mathbf{x})$ is an unbiased estimator of*
 645 *its input \mathbf{x} , and its variance grows with the squared of the squared of ℓ_2 -norm of its argument, i.e.,*
 646 $\mathbb{E}[Q(\mathbf{x})] = \mathbf{x}$ *and* $\mathbb{E}[\|Q(\mathbf{x}) - \mathbf{x}\|^2] \leq \omega \|\mathbf{x}\|^2$.

647 We note that the sketching PRIVIX and HEAPRIX, satisfy Assumption 4 with $\omega = c \frac{d}{m}$ and $\omega =$
 648 $c \frac{d}{m} - 1$ respectively with probability $1 - \frac{\delta}{R}$ per communication round. Therefore, all the results in
 649 Theorem 1, by taking union over the all probabilities of each communication rounds, are concluded
 650 with probability $1 - \delta$ by plugging $\omega = c \frac{d}{m}$ and $\omega = c \frac{d}{m} - 1$ respectively into the corresponding
 651 convergence bounds.

652 C.1 Proof of Theorem 1

653 In this section, we study the convergence properties of our FedSKETCH method presented in Algo-
 654 rithm 3. Before developing the proofs for FedSKETCH in the homogeneous setting, we first mention
 655 the following intermediate lemmas.

656 **Lemma 1.** *Using unbiased compression and under Assumption 2, we have the following bound:*

$$\mathbb{E}_{\mathcal{K}} \left[\mathbb{E}_{\mathbf{S}, \xi^{(r)}} \left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2 \right] \right] = \mathbb{E}_{\xi^{(r)}} \mathbb{E}_{\mathbf{S}} \left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2 \right] \leq \tau \left(\frac{\omega}{k} + 1 \right) \sum_{j=1}^m q_j \left[\sum_{c=0}^{\tau-1} \|\mathbf{g}_j^{(c,r)}\|^2 + \sigma^2 \right] \quad (3)$$

Proof.

$$\begin{aligned} & \mathbb{E}_{\xi^{(r)} | \mathbf{w}^{(r)}} \mathbb{E}_{\mathcal{K}} \left[\mathbb{E}_{\mathbf{S}} \left[\left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right\|^2 \right] \right] \\ &= \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\mathbb{E}_{\mathbf{S}} \left[\left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \underbrace{\mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right)}_{\tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)}} \right\|^2 \right] \right] \right] \\ &\stackrel{\textcircled{1}}{=} \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\left[\left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)} - \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbb{E}_{\mathbf{S}} [\tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)}] \right\|^2 + \left\| \mathbb{E}_{\mathbf{S}} \left[\frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)} \right] \right\|^2 \right] \right] \right] \\ &\stackrel{\textcircled{2}}{=} \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\mathbb{E}_{\mathbf{S}} \left[\left\| \frac{1}{k} \left[\sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_{\mathbf{S},j}^{(r)} - \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right] \right\|^2 + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right\|^2 \right] \right] \right] \end{aligned}$$

$$\begin{aligned}
&= \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\left[\text{Var}_{\mathbf{S}} \left[\frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)} \right] + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right\|^2 \right] \right] \right] \\
&= \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\left[\frac{1}{k^2} \sum_{j \in \mathcal{K}} \text{Var}_{\mathbf{S}_j} \left[\tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)} \right] + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right\|^2 \right] \right] \right] \\
&\leq \mathbb{E}_{\xi^{(r)}} \left[\mathbb{E}_{\mathcal{K}} \left[\left[\frac{1}{k^2} \sum_{j \in \mathcal{K}} \omega \left\| \tilde{\mathbf{g}}_j^{(r)} \right\|^2 + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right\|^2 \right] \right] \right] \\
&= \left[\mathbb{E}_{\xi} \left[\left[\frac{1}{k} \sum_{j \in \mathcal{K}} \omega \left\| \tilde{\mathbf{g}}_j^{(r)} \right\|^2 + \mathbb{E}_{\mathcal{K}} \mathbb{E}_{\xi^{(r)}} \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right\|^2 \right] \right] \right] \\
&= \left[\mathbb{E}_{\xi} \left[\left[\frac{\omega}{k} \sum_{j=1}^p q_j \left\| \tilde{\mathbf{g}}_j^{(r)} \right\|^2 + \mathbb{E}_{\mathcal{K}} \left[\text{Var} \left(\frac{1}{k} \sum_{j \in \mathcal{K}} \tilde{\mathbf{g}}_j^{(r)} \right) + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{g}_j^{(r)} \right\|^2 \right] \right] \right] \right] \\
&= \frac{\omega}{k} \sum_{j=1}^p q_j \mathbb{E}_{\xi} \left\| \tilde{\mathbf{g}}_j^{(r)} \right\|^2 + \mathbb{E}_{\mathcal{K}} \left[\frac{1}{k^2} \sum_{j \in \mathcal{K}} \text{Var} \left(\tilde{\mathbf{g}}_j^{(r)} \right) + \left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{g}_j^{(r)} \right\|^2 \right] \\
&\leq \frac{\omega}{k} \sum_{j=1}^p q_j \mathbb{E}_{\xi} \left\| \tilde{\mathbf{g}}_j^{(r)} \right\|^2 + \mathbb{E}_{\mathcal{K}} \left[\frac{1}{k^2} \sum_{j \in \mathcal{K}} \tau \sigma^2 + \frac{1}{k} \sum_{j \in \mathcal{K}} \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] \\
&= \frac{\omega}{k} \sum_{j=1}^p q_j \left[\text{Var} \left(\tilde{\mathbf{g}}_j^{(r)} \right) + \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] + \left[\frac{\tau \sigma^2}{k} + \sum_{j=1}^p q_j \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] \\
&\leq \frac{\omega}{k} \sum_{j=1}^p q_j \left[\tau \sigma^2 + \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] + \left[\frac{\tau \sigma^2}{k} + \sum_{j=1}^p q_j \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] \\
&= (\omega + 1) \frac{\tau \sigma^2}{k} + \left(\frac{\omega}{k} + 1 \right) \left[\sum_{j=1}^p q_j \left\| \mathbf{g}_j^{(r)} \right\|^2 \right] \tag{4}
\end{aligned}$$

657 where ① holds due to $\mathbb{E} \left[\left\| \mathbf{x} \right\|^2 \right] = \text{Var}[\mathbf{x}] + \left\| \mathbb{E}[\mathbf{x}] \right\|^2$, ② is due to $\mathbb{E}_{\mathbf{S}} \left[\frac{1}{p} \sum_{j=1}^p \tilde{\mathbf{g}}_{\mathbf{S}_j}^{(r)} \right] = \frac{1}{p} \sum_{j=1}^m \tilde{\mathbf{g}}_j^{(r)}$.

658 Next we show that from Assumptions 3, we have

$$\mathbb{E}_{\xi^{(r)}} \left[\left\| \tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)} \right\|^2 \right] \leq \tau \sigma^2 \tag{5}$$

659 To do so, note that

$$\begin{aligned}
\text{Var} \left(\tilde{\mathbf{g}}_j^{(r)} \right) &= \mathbb{E}_{\xi^{(r)}} \left[\left\| \tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)} \right\|^2 \right] \stackrel{\text{①}}{=} \mathbb{E}_{\xi^{(r)}} \left[\left\| \sum_{c=0}^{\tau-1} \left[\tilde{\mathbf{g}}_j^{(c,r)} - \mathbf{g}_j^{(c,r)} \right] \right\|^2 \right] = \text{Var} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \\
&\stackrel{\text{②}}{=} \sum_{c=0}^{\tau-1} \text{Var} \left(\tilde{\mathbf{g}}_j^{(c,r)} \right) \\
&= \sum_{c=0}^{\tau-1} \mathbb{E} \left[\left\| \tilde{\mathbf{g}}_j^{(c,r)} - \mathbf{g}_j^{(c,r)} \right\|^2 \right] \\
&\stackrel{\text{③}}{\leq} \tau \sigma^2 \tag{6}
\end{aligned}$$

660 where in ① we use the definition of $\tilde{\mathbf{g}}_j^{(r)}$ and $\mathbf{g}_j^{(r)}$, in ② we use the fact that mini-batches are chosen
661 in i.i.d. manner at each local machine, and ③ immediately follows from Assumptions 2.

662 Replacing $\mathbb{E}_{\xi^{(r)}} [\|\tilde{\mathbf{g}}_j^{(r)} - \mathbf{g}_j^{(r)}\|^2]$ in (4) by its upper bound in (5) implies that

$$\mathbb{E}_{\xi^{(r)}|\mathbf{w}^{(r)}} \mathbb{E}_{\mathbf{S}, \mathcal{K}} \left[\left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right\|^2 \right] \leq (\omega + 1) \frac{\tau \sigma^2}{k} + \left(\frac{\omega}{k} + 1 \right) \sum_{j=1}^p q_j \|\mathbf{g}_j^{(r)}\|^2 \quad (7)$$

663 Further note that we have

$$\|\mathbf{g}_j^{(r)}\|^2 = \left\| \sum_{c=0}^{\tau-1} \mathbf{g}_j^{(c,r)} \right\|^2 \leq \tau \sum_{c=0}^{\tau-1} \|\mathbf{g}_j^{(c,r)}\|^2 \quad (8)$$

664 where the last inequality is due to $\left\| \sum_{j=1}^n \mathbf{a}_i \right\|^2 \leq n \sum_{j=1}^n \|\mathbf{a}_i\|^2$, which together with (7) leads to
 665 the following bound:

$$\mathbb{E}_{\xi^{(r)}|\mathbf{w}^{(r)}} \mathbb{E}_{\mathbf{S}} \left[\left\| \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right\|^2 \right] \leq (\omega + 1) \frac{\tau \sigma^2}{k} + \tau \left(\frac{\omega}{k} + 1 \right) \sum_{j=1}^p q_j \|\mathbf{g}_j^{(c,r)}\|^2, \quad (9)$$

666 and the proof is complete. \square

667 **Lemma 2.** Under Assumption 1, and according to the FedCOM algorithm the expected inner product
 668 between stochastic gradient and full batch gradient can be bounded with:

$$-\mathbb{E}_{\xi, \mathbf{S}, \mathcal{K}} \left[\left\langle \nabla f(\mathbf{w}^{(r)}), \tilde{\mathbf{g}}^{(r)} \right\rangle \right] \leq \frac{1}{2} \eta \frac{1}{m} \sum_{j=1}^m \sum_{c=0}^{\tau-1} \left[-\|\nabla f(\mathbf{w}^{(r)})\|_2^2 - \|\nabla f(\mathbf{w}_j^{(c,r)})\|_2^2 + L^2 \|\mathbf{w}^{(r)} - \mathbf{w}_j^{(c,r)}\|_2^2 \right] \quad (10)$$

669 *Proof.* We have:

$$\begin{aligned} & -\mathbb{E}_{\{\xi_1^{(t)}, \dots, \xi_m^{(t)} | \mathbf{w}_1^{(t)}, \dots, \mathbf{w}_m^{(t)}\}} \mathbb{E}_{\mathbf{S}, \mathcal{K}} \left[\left\langle \nabla f(\mathbf{w}^{(r)}), \tilde{\mathbf{g}}_{\mathbf{S}, \mathcal{K}}^{(r)} \right\rangle \right] \\ &= -\mathbb{E}_{\{\xi_1^{(t)}, \dots, \xi_m^{(t)} | \mathbf{w}_1^{(t)}, \dots, \mathbf{w}_m^{(t)}\}} \left[\left\langle \nabla f(\mathbf{w}^{(r)}), \eta \sum_{j \in \mathcal{K}} q_j \sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right\rangle \right] \\ &= -\left\langle \nabla f(\mathbf{w}^{(r)}), \eta \sum_{j=1}^m q_j \sum_{c=0}^{\tau-1} \mathbb{E}_{\xi, \mathbf{S}} [\tilde{\mathbf{g}}_{j, \mathbf{S}}^{(c,r)}] \right\rangle \\ &= -\eta \sum_{c=0}^{\tau-1} \sum_{j=1}^m q_j \left\langle \nabla f(\mathbf{w}^{(r)}), \mathbf{g}_j^{(c,r)} \right\rangle \\ &\stackrel{\textcircled{1}}{=} \frac{1}{2} \eta \sum_{c=0}^{\tau-1} \sum_{j=1}^m q_j \left[-\|\nabla f(\mathbf{w}^{(r)})\|_2^2 - \|\nabla f(\mathbf{w}_j^{(c,r)})\|_2^2 + \|\nabla f(\mathbf{w}^{(r)}) - \nabla f(\mathbf{w}_j^{(c,r)})\|_2^2 \right] \\ &\stackrel{\textcircled{2}}{\leq} \frac{1}{2} \eta \sum_{c=0}^{\tau-1} \sum_{j=1}^m q_j \left[-\|\nabla f(\mathbf{w}^{(r)})\|_2^2 - \|\nabla f(\mathbf{w}_j^{(c,r)})\|_2^2 + L^2 \|\mathbf{w}^{(r)} - \mathbf{w}_j^{(c,r)}\|_2^2 \right] \quad (11) \end{aligned}$$

670 where ① is due to $2\langle \mathbf{a}, \mathbf{b} \rangle = \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 - \|\mathbf{a} - \mathbf{b}\|^2$, and ② follows from Assumption 1. \square

671 The following lemma bounds the distance of local solutions from global solution at r th communication
 672 round.

673 **Lemma 3.** Under Assumptions 2 we have:

$$\mathbb{E} \left[\|\mathbf{w}^{(r)} - \mathbf{w}_j^{(c,r)}\|_2^2 \right] \leq \eta^2 \tau \sum_{c=0}^{\tau-1} \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + \eta^2 \tau \sigma^2$$

674 *Proof.* Note that

$$\begin{aligned}
\mathbb{E} \left[\left\| \mathbf{w}^{(r)} - \mathbf{w}_j^{(c,r)} \right\|_2^2 \right] &= \mathbb{E} \left[\left\| \mathbf{w}^{(r)} - \left(\mathbf{w}^{(r)} - \eta \sum_{k=0}^c \tilde{\mathbf{g}}_j^{(k,r)} \right) \right\|_2^2 \right] \\
&= \mathbb{E} \left[\left\| \eta \sum_{k=0}^c \tilde{\mathbf{g}}_j^{(k,r)} \right\|_2^2 \right] \\
&\stackrel{\textcircled{1}}{=} \mathbb{E} \left[\left\| \eta \sum_{k=0}^c \left(\tilde{\mathbf{g}}_j^{(k,r)} - \mathbf{g}_j^{(k,r)} \right) \right\|_2^2 \right] + \mathbb{E} \left[\left\| \eta \sum_{k=0}^c \mathbf{g}_j^{(k,r)} \right\|_2^2 \right] \\
&\stackrel{\textcircled{2}}{=} \eta^2 \sum_{k=0}^c \mathbb{E} \left[\left\| \left(\tilde{\mathbf{g}}_j^{(k,r)} - \mathbf{g}_j^{(k,r)} \right) \right\|_2^2 \right] + (c+1) \eta^2 \sum_{k=0}^c \mathbb{E} \left[\left\| \mathbf{g}_j^{(k,r)} \right\|_2^2 \right] \\
&\leq \eta^2 \sum_{k=0}^{\tau-1} \mathbb{E} \left[\left\| \left(\tilde{\mathbf{g}}_j^{(k,r)} - \mathbf{g}_j^{(k,r)} \right) \right\|_2^2 \right] + \tau \eta^2 \sum_{k=0}^{\tau-1} \mathbb{E} \left[\left\| \mathbf{g}_j^{(k,r)} \right\|_2^2 \right] \\
&\stackrel{\textcircled{3}}{\leq} \eta^2 \sum_{k=0}^{\tau-1} \sigma^2 + \tau \eta^2 \sum_{k=0}^{\tau-1} \mathbb{E} \left[\left\| \mathbf{g}_j^{(k,r)} \right\|_2^2 \right] \\
&= \eta^2 \tau \sigma^2 + \eta^2 \sum_{k=0}^{\tau-1} \tau \left\| \mathbf{g}_j^{(k,r)} \right\|_2^2 \tag{12}
\end{aligned}$$

675 where $\textcircled{1}$ comes from $\mathbb{E}[\mathbf{x}^2] = \text{Var}[\mathbf{x}] + [\mathbb{E}[\mathbf{x}]]^2$ and $\textcircled{2}$ holds because $\text{Var}\left(\sum_{j=1}^n \mathbf{x}_j\right) =$
676 $\sum_{j=1}^n \text{Var}(\mathbf{x}_j)$ for i.i.d. vectors \mathbf{x}_i (and i.i.d. assumption comes from i.i.d. sampling), and fi-
677 nally $\textcircled{3}$ follows from Assumption 2. \square

678 C.1.1 Main result for the non-convex setting

679 Now we are ready to present our result for the homogeneous setting. We first state and prove the
680 result for the general non-convex objectives.

681 **Theorem 4** (non-convex). *For FedSKETCH(τ, η, γ), for all $0 \leq t \leq R\tau - 1$, under Assumptions 1*
682 *to 2, if the learning rate satisfies*

$$1 \geq \tau^2 L^2 \eta^2 + \left(\frac{\omega}{k} + 1 \right) \eta \gamma L \tau \tag{13}$$

683 *and all local model parameters are initialized at the same point $\mathbf{w}^{(0)}$, then the average-squared*
684 *gradient after τ iterations is bounded as follows:*

$$\frac{1}{R} \sum_{r=0}^{R-1} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 \leq \frac{2(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)}))}{\eta \gamma \tau R} + \frac{L \eta \gamma (\omega + 1)}{k} \sigma^2 + L^2 \eta^2 \tau \sigma^2, \tag{14}$$

685 *where $\mathbf{w}^{(*)}$ is the global optimal solution with function value $f(\mathbf{w}^{(*)})$.*

686 *Proof.* Before proceeding with the proof of Theorem 4, we would like to highlight that

$$\mathbf{w}^{(r)} - \mathbf{w}_j^{(\tau,r)} = \eta \sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)}. \tag{15}$$

687 From the updating rule of Algorithm 3 we have

$$\mathbf{w}^{(r+1)} = \mathbf{w}^{(r)} - \gamma \eta \left(\frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0, r}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right) = \mathbf{w}^{(r)} - \gamma \left[\frac{\eta}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right].$$

In what follows, we use the following notation to denote the stochastic gradient used to update the global model at r th communication round

$$\tilde{\mathbf{g}}_{\mathbf{S},\mathcal{K}}^{(r)} \triangleq \frac{\eta}{p} \sum_{j=1}^p \mathbf{S} \left(\frac{\mathbf{w}^{(r)} - \mathbf{w}_j^{(\tau,r)}}{\eta} \right) = \frac{1}{k} \sum_{j \in \mathcal{K}} \mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right).$$

688 and notice that $\mathbf{w}^{(r)} = \mathbf{w}^{(r-1)} - \gamma \tilde{\mathbf{g}}^{(r)}$.

689 Then using the unbiased estimation property of sketching we have:

$$\mathbb{E}_{\mathbf{S}} [\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}] = \frac{1}{k} \sum_{j \in \mathcal{K}} \left[-\eta \mathbb{E}_{\mathbf{S}} \left[\mathbf{S} \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right] \right] = \frac{1}{k} \sum_{j \in \mathcal{K}} \left[-\eta \left(\sum_{c=0}^{\tau-1} \tilde{\mathbf{g}}_j^{(c,r)} \right) \right] \triangleq \tilde{\mathbf{g}}_{\mathbf{S},\mathcal{K}}^{(r)}.$$

690 From the L -smoothness gradient assumption on global objective, by using $\tilde{\mathbf{g}}^{(r)}$ in inequality (15) we
691 have:

$$f(\mathbf{w}^{(r+1)}) - f(\mathbf{w}^{(r)}) \leq -\gamma \langle \nabla f(\mathbf{w}^{(r)}), \tilde{\mathbf{g}}^{(r)} \rangle + \frac{\gamma^2 L}{2} \|\tilde{\mathbf{g}}^{(r)}\|^2 \quad (16)$$

692 By taking expectation on both sides of above inequality over sampling, we get:

$$\begin{aligned} \mathbb{E} \left[\mathbb{E}_{\mathbf{S}} \left[f(\mathbf{w}^{(r+1)}) - f(\mathbf{w}^{(r)}) \right] \right] &\leq -\gamma \mathbb{E} \left[\mathbb{E}_{\mathbf{S}} \left[\langle \nabla f(\mathbf{w}^{(r)}), \tilde{\mathbf{g}}_{\mathbf{S}}^{(r)} \rangle \right] \right] + \frac{\gamma^2 L}{2} \mathbb{E} \left[\mathbb{E}_{\mathbf{S}} \|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2 \right] \\ &\stackrel{(a)}{=} \underbrace{-\gamma \mathbb{E} \left[\langle \nabla f(\mathbf{w}^{(r)}), \tilde{\mathbf{g}}^{(r)} \rangle \right]}_{\text{(I)}} + \underbrace{\frac{\gamma^2 L}{2} \mathbb{E} \left[\mathbb{E}_{\mathbf{S}} \left[\|\tilde{\mathbf{g}}_{\mathbf{S}}^{(r)}\|^2 \right] \right]}_{\text{(II)}}. \end{aligned} \quad (17)$$

693 We proceed to use Lemma 1, Lemma 2, and Lemma 3, to bound terms (I) and (II) in right hand side
694 of (17), which gives

$$\begin{aligned} &\mathbb{E} \left[\mathbb{E}_{\mathbf{S}} \left[f(\mathbf{w}^{(r+1)}) - f(\mathbf{w}^{(r)}) \right] \right] \\ &\leq \gamma \frac{1}{2} \eta \sum_{j=1}^p q_j \sum_{c=0}^{\tau-1} \left[-\left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 - \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + L^2 \eta^2 \sum_{c=0}^{\tau-1} \left[\tau \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + \sigma^2 \right] \right] \\ &\quad + \frac{\gamma^2 L (\frac{\omega}{k} + 1)}{2} \left[\eta^2 \tau \sum_{j=1}^p q_j \sum_{c=0}^{\tau-1} \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 \right] + \frac{\gamma^2 \eta^2 L (\omega + 1)}{2} \frac{\tau \sigma^2}{k} \\ &\stackrel{\textcircled{1}}{\leq} \frac{\gamma \eta}{2} \sum_{j=1}^p q_j \sum_{c=0}^{\tau-1} \left[-\left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 - \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + \tau L^2 \eta^2 \left[\tau \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + \sigma^2 \right] \right] \\ &\quad + \frac{\gamma^2 L (\frac{\omega}{k} + 1)}{2} \left[\eta^2 \tau \sum_{j=1}^p q_j \sum_{c=0}^{\tau-1} \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 \right] + \frac{\gamma^2 \eta^2 L (\omega + 1)}{2} \frac{\tau \sigma^2}{k} \\ &= -\eta \gamma \frac{\tau}{2} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 \\ &\quad - \left(1 - \tau L^2 \eta^2 \tau - \left(\frac{\omega}{k} + 1 \right) \eta \gamma L \tau \right) \frac{\eta \gamma}{2} \sum_{j=1}^p q_j \sum_{c=0}^{\tau-1} \left\| \mathbf{g}_j^{(c,r)} \right\|_2^2 + \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + \gamma (\omega + 1)) \sigma^2 \\ &\stackrel{\textcircled{2}}{\leq} -\eta \gamma \frac{\tau}{2} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 + \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + \gamma (\omega + 1)) \sigma^2, \end{aligned} \quad (18)$$

695 where in $\textcircled{1}$ we incorporate outer summation $\sum_{c=0}^{\tau-1}$, and $\textcircled{2}$ follows from condition

$$1 \geq \tau L^2 \eta^2 \tau + \left(\frac{\omega}{k} + 1 \right) \eta \gamma L \tau.$$

696 Summing up for all R communication rounds and rearranging the terms gives:

$$\frac{1}{R} \sum_{r=0}^{R-1} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 \leq \frac{2(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)}))}{\eta \gamma \tau R} + \frac{L \eta \gamma (\omega + 1)}{k} \sigma^2 + L^2 \eta^2 \tau \sigma^2.$$

697 From the above inequality, is it easy to see that in order to achieve a linear speed up, we need to have
 698 $\eta\gamma = O\left(\frac{\sqrt{k}}{\sqrt{R\tau}}\right)$. \square

699 **Corollary 3** (Linear speed up). *In (14) for the choice of $\eta\gamma = O\left(\frac{1}{L}\sqrt{\frac{k}{R\tau(\omega+1)}}\right)$, and $\gamma \geq k$ the
 700 convergence rate reduces to:*

$$\frac{1}{R} \sum_{r=0}^{R-1} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 \leq O \left(\frac{L\sqrt{(\omega+1)} (f(\mathbf{w}^{(0)}) - f(\mathbf{w}^*))}{\sqrt{kR\tau}} + \frac{(\sqrt{(\omega+1)})^2 \sigma^2}{\sqrt{kR\tau}} + \frac{k\sigma^2}{R\gamma^2} \right). \quad (19)$$

701 *Note that according to (19), if we pick a fixed constant value for γ , in order to achieve an ϵ -accurate
 702 solution, $R = O\left(\frac{1}{\epsilon}\right)$ communication rounds and $\tau = O\left(\frac{\omega+1}{k\epsilon}\right)$ local updates are necessary. We
 703 also highlight that (19) also allows us to choose $R = O\left(\frac{\omega+1}{\epsilon}\right)$ and $\tau = O\left(\frac{1}{k\epsilon}\right)$ to get the same
 704 convergence rate.*

705 **Remark 3.** *Condition in (13) can be rewritten as*

$$\begin{aligned} \eta &\leq \frac{-\gamma L\tau \left(\frac{\omega}{k} + 1\right) + \sqrt{\gamma^2 \left(L\tau \left(\frac{\omega}{k} + 1\right)\right)^2 + 4L^2\tau^2}}{2L^2\tau^2} \\ &= \frac{-\gamma L\tau \left(\frac{\omega}{k} + 1\right) + L\tau \sqrt{\left(\frac{\omega}{k} + 1\right)^2 \gamma^2 + 4}}{2L^2\tau^2} \\ &= \frac{\sqrt{\left(\frac{\omega}{k} + 1\right)^2 \gamma^2 + 4} - \left(\frac{\omega}{k} + 1\right) \gamma}{2L\tau}. \end{aligned} \quad (20)$$

706 *So based on (20), if we set $\eta = O\left(\frac{1}{L\gamma} \sqrt{\frac{k}{R\tau(\omega+1)}}\right)$, it implies that:*

$$R \geq \frac{\tau k}{(\omega+1)\gamma^2 \left(\sqrt{\left(\frac{\omega}{k} + 1\right)^2 \gamma^2 + 4} - \left(\frac{\omega}{k} + 1\right) \gamma \right)^2}. \quad (21)$$

707 *We note that $\gamma^2 \left(\sqrt{\left(\frac{\omega}{k} + 1\right)^2 \gamma^2 + 4} - \left(\frac{\omega}{k} + 1\right) \gamma \right)^2 = \Theta(1) \leq 5$ therefore even for $\gamma \geq m$ we
 708 need to have*

$$R \geq \frac{\tau k}{5(\omega+1)} = O\left(\frac{\tau k}{(\omega+1)}\right). \quad (22)$$

709 *Therefore, for the choice of $\tau = O\left(\frac{\omega+1}{k\epsilon}\right)$, due to condition in (22), we need to have $R = O\left(\frac{1}{\epsilon}\right)$.
 710 Similarly, we can have $R = O\left(\frac{\omega+1}{\epsilon}\right)$ and $\tau = O\left(\frac{1}{k\epsilon}\right)$.*

711 **Corollary 4** (Special case, $\gamma = 1$). *By letting $\gamma = 1$, $\omega = 0$ and $k = p$ the convergence rate in (14)
 712 reduces to*

$$\frac{1}{R} \sum_{r=0}^{R-1} \left\| \nabla f(\mathbf{w}^{(r)}) \right\|_2^2 \leq \frac{2(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^*))}{\eta R\tau} + \frac{L\eta}{p} \sigma^2 + L^2 \eta^2 \tau \sigma^2,$$

713 *which matches the rate obtained in [44]. In this case the communication complexity and the number
 714 of local updates become*

$$R = O\left(\frac{p}{\epsilon}\right), \quad \tau = O\left(\frac{1}{\epsilon}\right),$$

715 *which simply implies that in this special case the convergence rate of our algorithm reduces to the
 716 rate obtained in [44], which indicates the tightness of our analysis.*

717 **C.1.2 Main result for the PL/Strongly convex setting**

718 We now turn to stating the convergence rate for the homogeneous setting under PL condition which
 719 naturally leads to the same rate for strongly convex functions.

720 **Theorem 5** (PL or strongly convex). *For FedSKETCH(τ, η, γ), for all $0 \leq t \leq R\tau - 1$, under
 721 Assumptions 1 to 2 and 3, if the learning rate satisfies*

$$1 \geq \tau^2 L^2 \eta^2 + \left(\frac{\omega}{k} + 1\right) \eta \gamma L \tau$$

722 and if the all the models are initialized with $\mathbf{w}^{(0)}$ we obtain:

$$\mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] \leq (1 - \eta \gamma \mu \tau)^R \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\mu} \left[\frac{1}{2} L^2 \tau \eta^2 \sigma^2 + (1 + \omega) \frac{\gamma \eta L \sigma^2}{2k}\right]$$

723 *Proof.* From (18) under condition:

$$1 \geq \tau L^2 \eta^2 \tau + \left(\frac{\omega}{k} + 1\right) \eta \gamma L \tau$$

724 we obtain:

$$\begin{aligned} \mathbb{E}\left[f(\mathbf{w}^{(r+1)}) - f(\mathbf{w}^{(r)})\right] &\leq -\eta \gamma \frac{\tau}{2} \left\|\nabla f(\mathbf{w}^{(r)})\right\|_2^2 + \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + \gamma(\omega + 1)) \sigma^2 \\ &\leq -\eta \mu \gamma \tau \left(f(\mathbf{w}^{(r)}) - f(\mathbf{w}^{(*)})\right) + \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + \gamma(\omega + 1)) \sigma^2 \end{aligned} \quad (23)$$

725 which leads to the following bound:

$$\mathbb{E}\left[f(\mathbf{w}^{(r+1)}) - f(\mathbf{w}^{(*)})\right] \leq (1 - \eta \mu \gamma \tau) \left[f(\mathbf{w}^{(r)}) - f(\mathbf{w}^{(*)})\right] + \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + (\omega + 1) \gamma) \sigma^2$$

726 By setting $\Delta = 1 - \eta \mu \gamma \tau$ we obtain the following bound:

$$\begin{aligned} &\mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] \\ &\leq \Delta^R \left[f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right] + \frac{1 - \Delta^R}{1 - \Delta} \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + (\omega + 1) \gamma) \sigma^2 \\ &\leq \Delta^R \left[f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right] + \frac{1}{1 - \Delta} \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + (\omega + 1) \gamma) \sigma^2 \\ &= (1 - \eta \mu \gamma \tau)^R \left[f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right] + \frac{1}{\eta \mu \gamma \tau} \frac{L \tau \gamma \eta^2}{2k} (k L \tau \eta + (\omega + 1) \gamma) \sigma^2 \end{aligned} \quad (24)$$

727 □

728 **Corollary 5.** *If we let $\eta \gamma \mu \tau \leq \frac{1}{2}$, $\eta = \frac{1}{2L(\frac{\omega}{k} + 1)\tau \gamma}$ and $\kappa = \frac{L}{\mu}$ the convergence error in Theorem 5,*

729 *with $\gamma \geq k$ results in:*

$$\begin{aligned} &\mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] \\ &\leq e^{-\eta \gamma \mu \tau R} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\mu} \left[\frac{1}{2} \tau L^2 \eta^2 \sigma^2 + (1 + \omega) \frac{\gamma \eta L \sigma^2}{2k}\right] \\ &\leq e^{-\frac{R}{2(\frac{\omega}{k} + 1)\kappa}} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\mu} \left[\frac{1}{2} L^2 \frac{\tau \sigma^2}{L^2 (\frac{\omega}{k} + 1)^2 \gamma^2 \tau^2} + \frac{(1 + \omega) L \sigma^2}{2 (\frac{\omega}{k} + 1) L \tau k}\right] \\ &= O\left(e^{-\frac{R}{2(\frac{\omega}{k} + 1)\kappa}} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{\sigma^2}{(\frac{\omega}{k} + 1)^2 \gamma^2 \mu \tau} + \frac{(\omega + 1) \sigma^2}{\mu (\frac{\omega}{k} + 1) \tau k}\right) \\ &= O\left(e^{-\frac{R}{2(\frac{\omega}{k} + 1)\kappa}} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{\sigma^2}{\gamma^2 \mu \tau} + \frac{(\omega + 1) \sigma^2}{\mu (\frac{\omega}{k} + 1) \tau k}\right) \end{aligned} \quad (25)$$

730 which indicates that to achieve an error of ϵ , we need to have $R = O\left(\left(\frac{\omega}{k} + 1\right) \kappa \log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau =$
731 $\frac{(\omega+1)}{k\left(\frac{\omega}{k}+1\right)\epsilon}$. Additionally, we note that if $\gamma \rightarrow \infty$, yet $R = O\left(\left(\frac{\omega}{k} + 1\right) \kappa \log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = \frac{(\omega+1)}{k\left(\frac{\omega}{k}+1\right)\epsilon}$
732 will be necessary.

733 C.1.3 Main result for the general convex setting

734 **Theorem 6** (Convex). For a general convex function $f(\mathbf{w})$ with optimal solution $\mathbf{w}^{(*)}$, using
735 *FedSKETCH*(τ, η, γ) to optimize $\tilde{f}(\mathbf{w}, \phi) = f(\mathbf{w}) + \frac{\phi}{2} \|\mathbf{w}\|^2$, for all $0 \leq t \leq R\tau - 1$, under
736 Assumptions 1 to 2, if the learning rate satisfies

$$1 \geq \tau^2 L^2 \eta^2 + \left(\frac{\omega}{k} + 1\right) \eta \gamma L \tau$$

737 and if the all the models initiate with $\mathbf{w}^{(0)}$, with $\phi = \frac{1}{\sqrt{k\tau}}$ and $\eta = \frac{1}{2L\gamma\tau(1+\frac{\omega}{k})}$ we obtain:

$$\begin{aligned} \mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] &\leq e^{-\frac{R}{2L(1+\frac{\omega}{k})\sqrt{m\tau}}} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) \\ &\quad + \left[\frac{\sqrt{k}\sigma^2}{8\sqrt{\tau}\gamma^2(1+\frac{\omega}{k})^2} + \frac{(\omega+1)\sigma^2}{4\left(\frac{\omega}{k}+1\right)\sqrt{k\tau}}\right] + \frac{1}{2\sqrt{k\tau}} \|\mathbf{w}^{(*)}\|^2 \end{aligned} \quad (26)$$

738 We note that above theorem implies that to achieve a convergence error of ϵ we need to have
739 $R = O\left(L\left(1 + \frac{\omega}{k}\right) \frac{1}{\epsilon} \log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = O\left(\frac{(\omega+1)^2}{k\left(\frac{\omega}{k}+1\right)^2\epsilon}\right)$.

740 *Proof.* Since $\tilde{f}(\mathbf{w}^{(r)}, \phi) = f(\mathbf{w}^{(r)}) + \frac{\phi}{2} \|\mathbf{w}^{(r)}\|^2$ is ϕ -PL, according to Theorem 5, we have:

$$\begin{aligned} &\tilde{f}(\mathbf{w}^{(R)}, \phi) - \tilde{f}(\mathbf{w}^{(*)}, \phi) \\ &= f(\mathbf{w}^{(r)}) + \frac{\phi}{2} \|\mathbf{w}^{(r)}\|^2 - \left(f(\mathbf{w}^{(*)}) + \frac{\phi}{2} \|\mathbf{w}^{(*)}\|^2\right) \\ &\leq (1 - \eta\gamma\phi\tau)^R \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\phi} \left[\frac{1}{2} L^2 \tau \eta^2 \sigma^2 + (1 + \omega) \frac{\gamma\eta L \sigma^2}{2k}\right] \end{aligned} \quad (27)$$

741 Next rearranging (27) and replacing μ with ϕ leads to the following error bound:

$$\begin{aligned} &f(\mathbf{w}^{(R)}) - f^* \\ &\leq (1 - \eta\gamma\phi\tau)^R \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\phi} \left[\frac{1}{2} L^2 \tau \eta^2 \sigma^2 + (1 + \omega) \frac{\gamma\eta L \sigma^2}{2k}\right] \\ &\quad + \frac{\phi}{2} \left(\|\mathbf{w}^*\|^2 - \|\mathbf{w}^{(r)}\|^2\right) \\ &\leq e^{-(\eta\gamma\phi\tau)R} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \frac{1}{\phi} \left[\frac{1}{2} L^2 \tau \eta^2 \sigma^2 + (1 + \omega) \frac{\gamma\eta L \sigma^2}{2k}\right] + \frac{\phi}{2} \|\mathbf{w}^{(*)}\|^2 \end{aligned}$$

742 Next, if we set $\phi = \frac{1}{\sqrt{k\tau}}$ and $\eta = \frac{1}{2(1+\frac{\omega}{k})L\gamma\tau}$, we obtain that

$$\begin{aligned} &f(\mathbf{w}^{(R)}) - f^* \\ &\leq e^{-\frac{R}{2(1+\frac{\omega}{k})L\sqrt{m\tau}}} \left(f(\mathbf{w}^{(0)}) - f(\mathbf{w}^{(*)})\right) + \sqrt{k\tau} \left[\frac{\sigma^2}{8\tau\gamma^2(1+\frac{\omega}{k})^2} + \frac{(\omega+1)\sigma^2}{4\left(\frac{\omega}{k}+1\right)\tau k}\right] + \frac{1}{2\sqrt{k\tau}} \|\mathbf{w}^{(*)}\|^2, \end{aligned}$$

743 thus the proof is complete. \square

C.2 Proof of Theorem 2

The proof of Theorem 2 follows directly from the results in [13]. We first mention the general Theorem 7 from [13] for general compression noise ω . Next, since the sketching PRIVIX and HEAPRIX, satisfy Assumption 4 with $\omega = c \frac{d}{m}$ and $\omega = c \frac{d}{m} - 1$ respectively with probability $1 - \frac{\delta}{R}$ per communication round, all the results in Theorem 2, conclude from Theorem 7 with probability $1 - \delta$ (by taking union over the all probabilities of each communication rounds with probability $1 - \delta/R$) and plugging $\omega = c \frac{d}{m}$ and $\omega = c \frac{d}{m} - 1$ respectively into the corresponding convergence bounds. For the heterogeneous setting, the results in [13] requires the following extra assumption that naturally holds for the sketching:

Assumption 5 ([13]). *The compression scheme Q for the heterogeneous data distribution setting satisfies the following condition $\mathbb{E}_Q[\|\frac{1}{m} \sum_{j=1}^m Q(\mathbf{x}_j)\|^2 - \|Q(\frac{1}{m} \sum_{j=1}^m \mathbf{x}_j)\|^2] \leq G_q$.*

We note that since sketching is a linear compressor, in the case of our algorithms for heterogeneous setting we have $G_q = 0$.

Next, we restate the Theorem in [13] here as follows:

Theorem 7. *Consider FedCOMGATE in [13]. If Assumptions 1, 3, 4 and 5 hold, then even for the case the local data distribution of users are different (heterogeneous setting) we have*

- **non-convex:** By choosing stepsizes as $\eta = \frac{1}{L\gamma} \sqrt{\frac{p}{R\tau(\omega+1)}}$ and $\gamma \geq p$, we obtain that the iterates satisfy $\frac{1}{R} \sum_{r=0}^{R-1} \|\nabla f(\mathbf{w}^{(r)})\|_2^2 \leq \epsilon$ if we set $R = O\left(\frac{\omega+1}{\epsilon}\right)$ and $\tau = O\left(\frac{1}{p\epsilon}\right)$.
- **Strongly convex or PL:** By choosing stepsizes as $\eta = \frac{1}{2L(\frac{\omega}{p}+1)\tau\gamma}$ and $\gamma \geq \sqrt{p\tau}$, we obtain that the iterates satisfy $\mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] \leq \epsilon$ if we set $R = O\left((\omega+1)\kappa \log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = O\left(\frac{1}{p\epsilon}\right)$.
- **Convex:** By choosing stepsizes as $\eta = \frac{1}{2L(\omega+1)\tau\gamma}$ and $\gamma \geq \sqrt{p\tau}$, we obtain that the iterates satisfy $\mathbb{E}\left[f(\mathbf{w}^{(R)}) - f(\mathbf{w}^{(*)})\right] \leq \epsilon$ if we set $R = O\left(\frac{L(1+\omega)}{\epsilon} \log\left(\frac{1}{\epsilon}\right)\right)$ and $\tau = O\left(\frac{1}{p\epsilon^2}\right)$.

Proof. Since the sketching methods PRIVIX and HEAPRIX, satisfy the Assumption 4 with $\omega = c \frac{d}{m}$ and $\omega = c \frac{d}{m} - 1$ respectively with probability $1 - \frac{\delta}{R}$ per communication round, we conclude the proofs of Theorem 2 using Theorem 7 with probability $1 - \delta$ (by taking union over all communication rounds) and plugging $\omega = c \frac{d}{m}$ and $\omega = c \frac{d}{m} - 1$ respectively into the convergence bounds. \square

771 **D Numerical Experiments and Additional Results**

772 **D.1 Implementation of FetchSGD**

773 Our implementation of FetchSGD basically follows the original paper (Algorithm 1 in [37]). The
774 only difference is that, in the original algorithm, the local workers compress the gradient (in every
775 local step) and transmit it to the central server. In our setting, we extend to the case with multiple local
776 updates, where the difference in local weights are transmitted (same as the standard FL framework).
777 Also, TopK compression is used to decode the sketches at the central server. We apply the same
778 implementation trick that when accumulating the errors, we only count the non-zero coordinates and
779 leave other coordinates zero for the accumulator. This greatly improves the empirical performance.

780 **D.2 Additional Plots for the MNIST Experiments**

781 **D.2.1 Homogeneous setting**

782 In the homogeneous case, each node has same data distribution. To achieve this setting, we randomly
783 choose samples uniformly from 10 classes of hand-written digits. The train loss and test accuracy
784 are provided in Figure 3, where we report local epochs $\tau = 2$ in addition to the main context (single
785 local update). The number of users is set to 50, and in each round of training we randomly pick half
786 of the nodes to be active (i.e., receiving data and performing local updates). We can draw similar
787 conclusion: FS-HEAPRIX consistently performs better than other competing methods. The test
788 accuracy increases with larger τ in homogeneous setting.

789 **D.2.2 Heterogeneous setting**

790 Analogously, we present experiments on MNIST dataset under heterogeneous data distribution,
791 including $\tau = 2$. We simulate the setting by only sending samples from one digit to each local
792 worker (very few nodes get two classes). We see from Figure 4 that FS-HEAPRIX shows consistent
793 advantage over competing methods. SketchedSGD performs poorly in this case.

794 **D.3 Additional Experiments: CIFAR-10**

795 We conduct similar sets of experiments on CIFAR10 dataset. We also use the simple LeNet CNN
796 structure, as in practice small models are more favorable in federated learning, due to the limitation of
797 mobile devices. The test accuracy is presented in Figure 5 and Figure 6, for respectively homogeneous
798 and heterogeneous data distribution. In general, we retrieve similar information as from MNIST
799 experiments: our proposed FS-HEAPRIX improves FS-PRIVIX and SketchedSGD in all cases. We
800 note that although the test accuracy provided by LeNet cannot reach the state-of-the-art accuracy
801 given by some huge models, it is also informative in terms of comparing the relative performance of
802 different sketching methods.

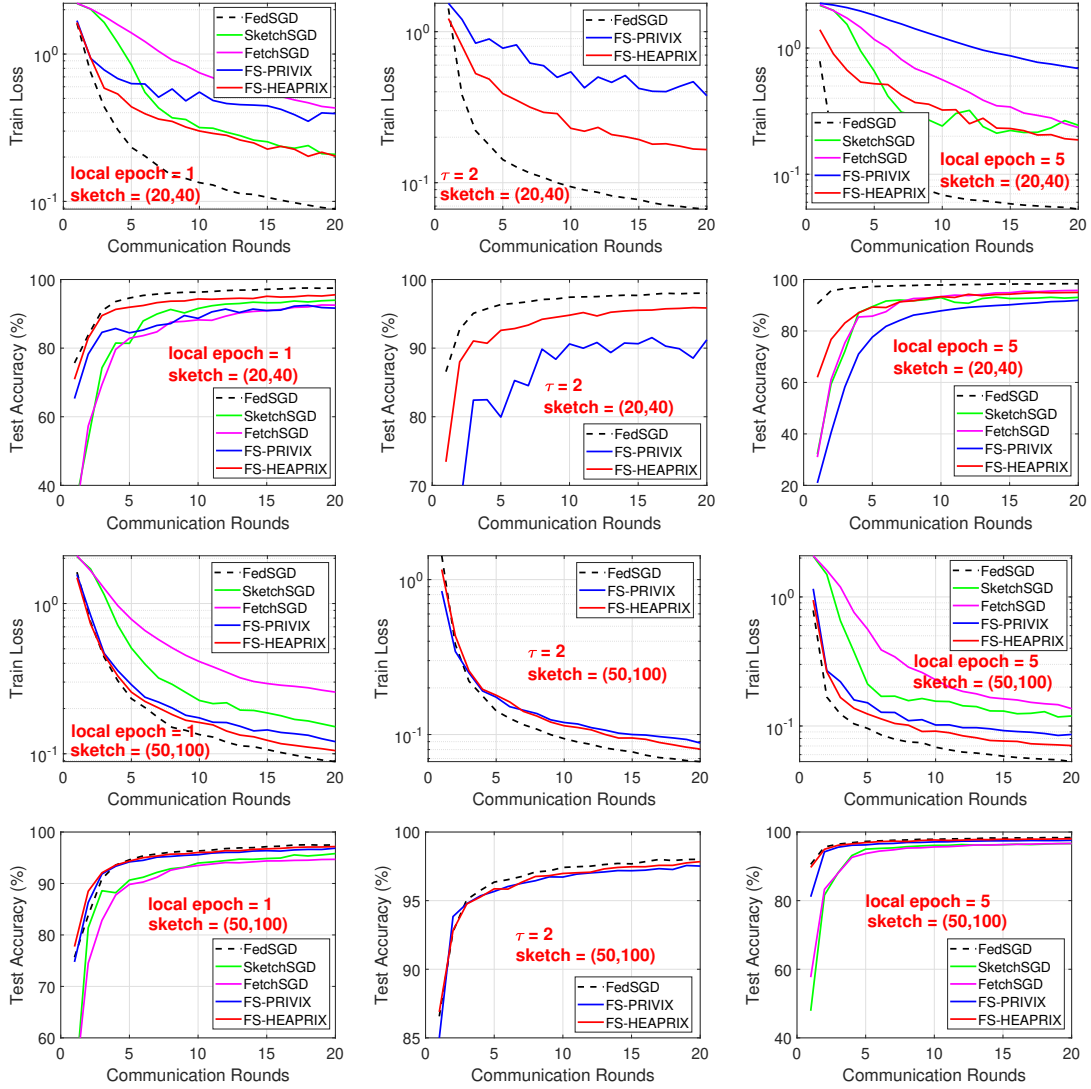


Figure 3: MNIST Homogeneous case: Comparison of compressed optimization methods on LeNet CNN architecture.

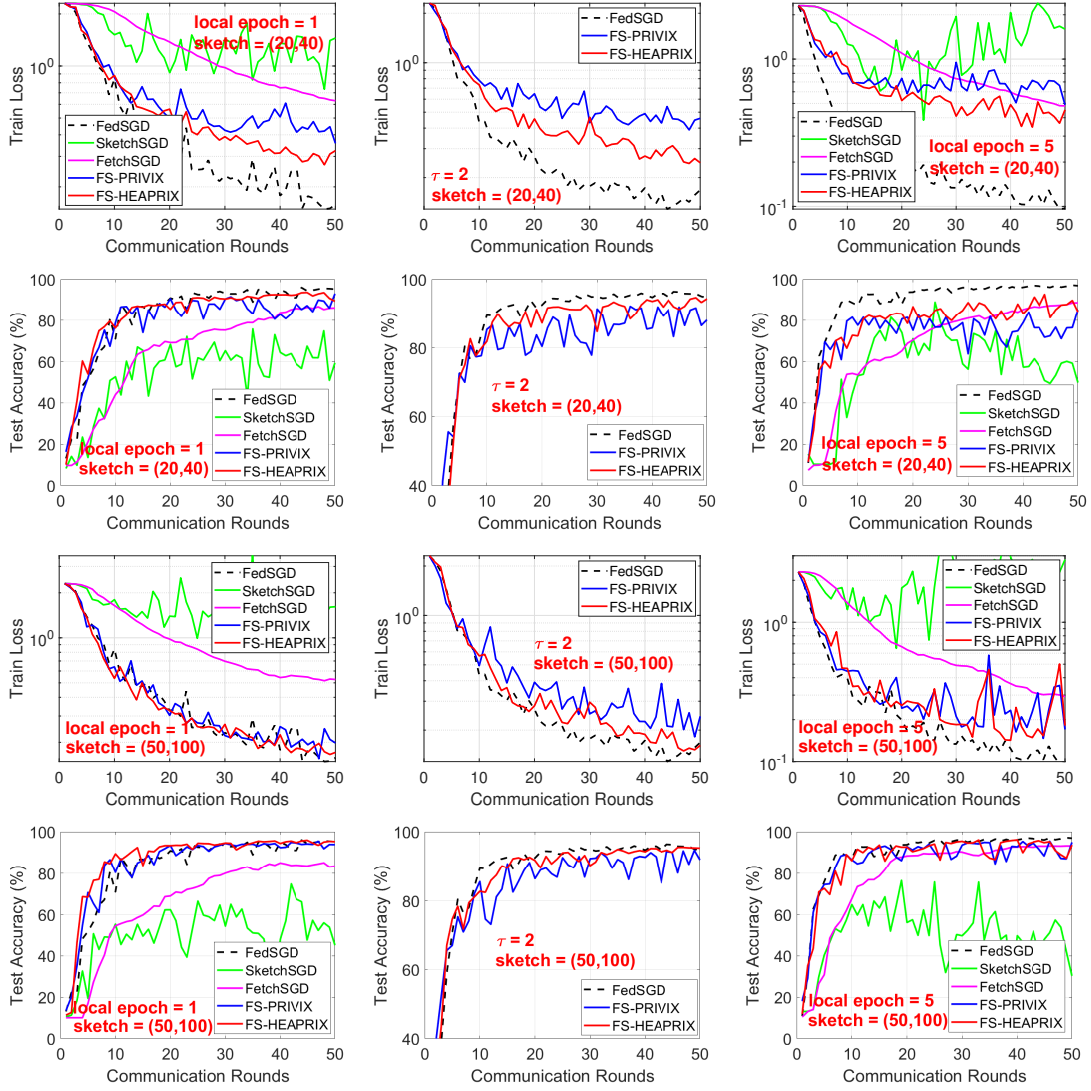


Figure 4: MNIST Heterogeneous case: Comparison of compressed optimization algorithms on LeNet CNN architecture.

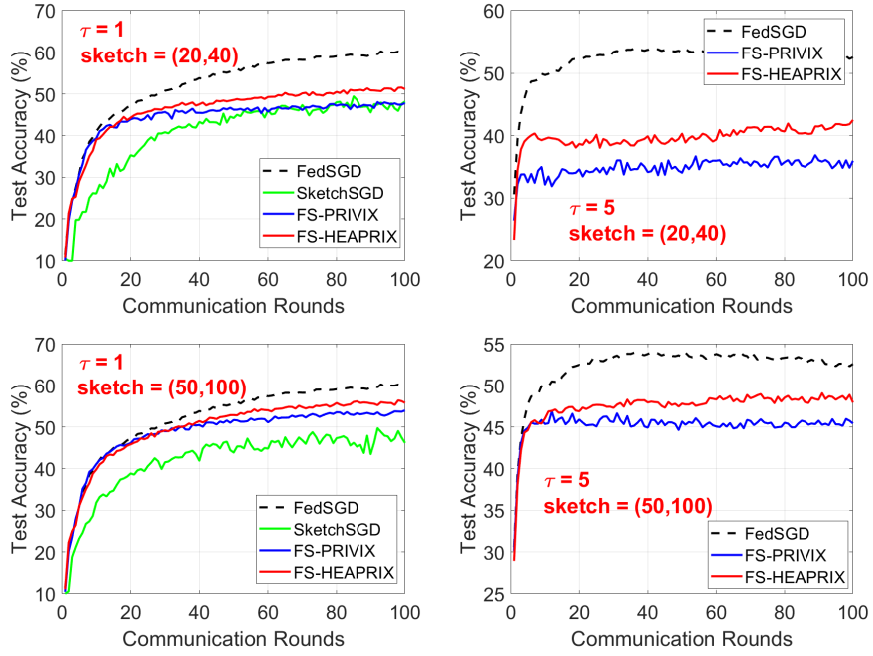


Figure 5: Homogeneous case: CIFAR10: Comparison of compressed optimization methods on LeNet CNN.

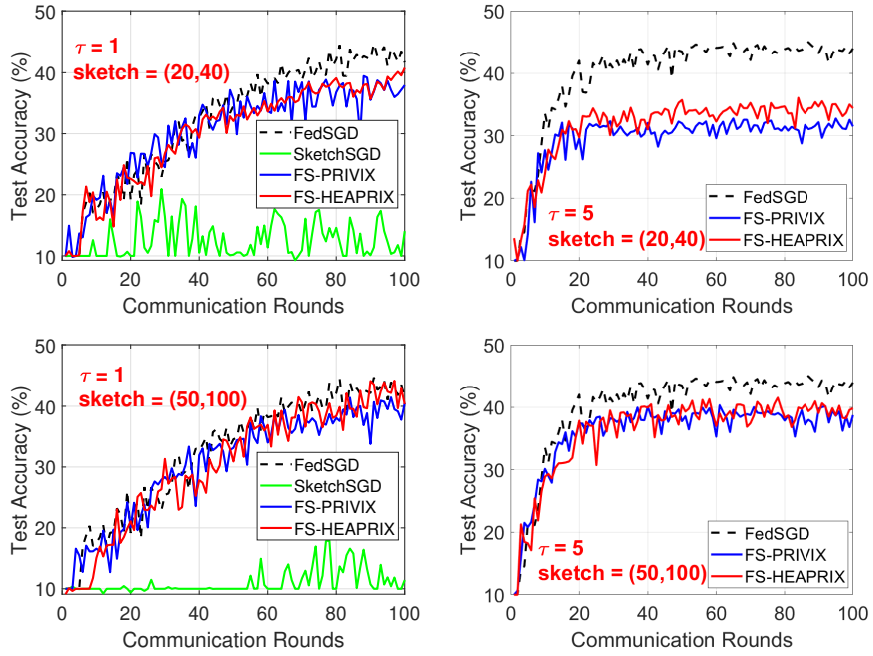


Figure 6: Heterogeneous case: CIFAR10: Comparison of compressed optimization methods on LeNet CNN.