A Unified Theory of SGD: Variance Reduction, Sampling, Quantization and Coordinate Descent

Eduard Gorbunov
MIPT, IITP RAS, RANEPA, Russia

Filip Hanzely
KAUST, Saudi Arabia

Peter Richtárik
KAUST, Saudi Arabia

Abstract

In this paper we introduce a unified analysis of a large family of variants of proximal stochastic gradient descent (SGD) which so far have required different intuitions, convergence analyses, have different applications, and which have been developed separately in various communities. We show that our framework includes methods with and without the following tricks, and their combinations: variance reduction, importance sampling, mini-batch sampling, quantization, and coordinate sub-sampling. As a by-product, we obtain the first unified theory of SGD and randomized coordinate descent (RCD) methods, the first unified theory of variance reduced and non-variance-reduced SGD methods, and the first unified theory of quantized and non-quantized methods. A key to our approach is a parametric assumption on the iterates and stochastic gradients. In a single theorem we establish a linear convergence result under this assumption and strong-quasi convexity of the loss function. Whenever we recover an existing method as a special case, our theorem gives the best known complexity result. Our approach can be used to motivate the development of new useful methods, and offers pre-proved convergence guarantees. To illustrate the strength of our approach, we develop five new variants of SGD, and through numerical experiments demonstrate some of their properties.

1 Introduction

In this paper we are interested in the optimization problem

$$\min_{x\in\mathbb{R}^d} f(x) + R(x), \tag{1}$$

where f is convex, differentiable with Lipschitz gradient, and $R : \mathbb{R}^d \to \mathbb{R} \cup \{+\infty\}$ is a proximable (proper closed convex) regularizer. In particular, we focus on situations when it is prohibitively expensive to compute the gradient of f, while an unbiased estimator of the gradient can be computed efficiently. This is typically the case for stochastic optimization problems, i.e., when

$$f(x) = \mathbf{E}_{\xi\sim\mathcal{D}}\left[f_\xi(x)\right], \tag{2}$$

where ξ is a random variable, and $f_\xi : \mathbb{R}^d \to \mathbb{R}$ is smooth for all ξ. Stochastic optimization problems are of key importance in statistical supervised learning theory. In this setup, x represents a machine learning model described by d parameters (e.g., logistic regression or a deep neural network), $\mathcal{D}$ is an unknown distribution of labelled examples, $f_\xi(x)$ represents the loss of model x on datapoint ξ, and f is the generalization error. Problem (1) seeks to find the model x minimizing the generalization error. In statistical learning theory one assumes that while $\mathcal{D}$ is not known, samples $\xi \sim \mathcal{D}$ are available. In such a case, $\nabla f(x)$ is not computable, while $\nabla f_\xi(x)$, which is an unbiased estimator of the gradient of f at x, is easily computable.

Another prominent example, one of special interest in this paper, are functions f which arise as averages of a very large number of smooth functions:

$$f(x) = \frac{1}{n}\sum_{i=1}^n f_i(x). \tag{3}$$

This problem often arises by approximation of the stochastic optimization loss function (2) via Monte Carlo integration, and is in this context known as the empirical risk minimization (ERM) problem. ERM is currently the dominant paradigm for solving supervised learning problems (Shalev-Shwartz and Ben-David, 2014). If index i is chosen uniformly at random

Proceedings of the 23^{rd}International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

from $[n] \coloneqq \{1, 2, \dots, n\}$, $\nabla f_i(x)$ is an unbiased estimator of $\nabla f(x)$. Typically, $\nabla f(x)$ is about n times more expensive to compute than $\nabla f_i(x)$.

Lastly, in some applications, especially in distributed training of supervised models, one considers problem (3), with n being the number of machines, and each f_i also having a finite sum structure, i.e.,

$$f_i(x) = \frac{1}{m} \sum_{j=1}^{m} f_{ij}(x), \tag{4}$$

where m corresponds to the number of training examples stored on machine i.

2 The Many Faces of Stochastic Gradient Descent

Stochastic gradient descent (SGD) (Robbins and Monro, 1951; Nemirovski et al., 2009; Vaswani et al., 2019) is a state-of-the-art algorithmic paradigm for solving optimization problems (1) in situations when f is either of structure (2) or (3). In its generic form, (proximal) SGD defines the new iterate by subtracting a multiple of a stochastic gradient from the current iterate, and subsequently applying the proximal operator of R:

$$x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k). \tag{5}$$

Here, g^k is an unbiased estimator of the gradient (i.e., a stochastic gradient),

$$\mathbf{E}\left[g^k \mid x^k\right] = \nabla f(x^k), \tag{6}$$

and $\mathrm{prox}_{\gamma R}(x) \coloneqq \mathrm{argmin}_u \{\gamma R(x) + \frac{1}{2}\|u - x\|^2\}$. However, and this is the starting point of our journey in this paper, there are *infinitely many* ways of obtaining a random vector g^k satisfying (6). On the one hand, this gives algorithm designers the flexibility to *construct* stochastic gradients in various ways in order to target desirable properties such as convergence speed, iteration cost, parallelizability and generalization. On the other hand, this poses considerable challenges in terms of convergence analysis. Indeed, if one aims to, as one should, obtain the sharpest bounds possible, dedicated analyses are needed to handle each of the particular variants of SGD.

Vanilla[1] **SGD.** The flexibility in the design of efficient strategies for constructing g^k has led to a creative renaissance in the optimization and machine learning communities, yielding a large number of immensely powerful new variants of SGD, such as those employing *importance sampling* (Zhao and Zhang, 2015; Needell et al., 2015), and *mini-batching* (Konečný et al., 2016). These efforts are subsumed by the recently developed and remarkably sharp analysis of SGD under *arbitrary sampling* paradigm (Gower et al., 2019), first introduced in the study of randomized coordinate descent methods by (Richtárik and Takáč, 2016). The arbitrary sampling paradigm covers virtually all stationary mini-batch and importance sampling strategies in a unified way, thus making headway towards theoretical unification of two separate strategies for constructing stochastic gradients. For strongly convex f, the SGD methods analyzed in (Gower et al., 2019) converge linearly to a neighbourhood of the solution $x^* = \arg\min_x f(x)$ for a fixed stepsize $\gamma^k = \gamma$. The size of the neighbourhood is proportional to the second moment of the stochastic gradient at the optimum ($\sigma^2 \coloneqq \frac{1}{n}\sum_{i=1}^n \|\nabla f_i(x^*)\|^2$), to the stepsize ($\gamma$), and inversely proportional to the modulus of strong convexity. The effect of various sampling strategies, such as importance sampling and mini-batching, is twofold: i) improvement of the linear convergence rate by enabling larger stepsizes, and ii) modification of σ^2. However, none of these strategies[2] is able to completely eliminate the adverse effect of σ^2. That is, SGD with a fixed stepsize does not reach the optimum, unless one happens to be in the overparameterized case characterized by the identity $\sigma^2 = 0$.

Variance reduced SGD. While sampling strategies such as importance sampling and mini-batching reduce the variance of the stochastic gradient, in the finite-sum case (3) a new type of *variance reduction* strategies has been developed over the last few years (Roux et al., 2012; Defazio et al., 2014; Johnson and Zhang, 2013; Shalev-Shwartz and Zhang, 2013; Qu et al., 2015; Nguyen et al., 2017; Kovalev et al., 2019). These variance-reduced SGD methods differ from the sampling strategies discussed before in a significant way: they can iteratively *learn* the stochastic gradients at the optimum, and in so doing are able to eliminate the adverse effect of the gradient noise $\sigma^2 > 0$ which, as mentioned above, prevents the iterates of vanilla SGD from converging to the optimum. As a result, for strongly convex f, these new variance-reduced SGD methods converge linearly to x^*, with a fixed stepsize. At the moment, these variance-reduced variants require a markedly different convergence theory from the vanilla variants of SGD. An exception to this is the situation when $\sigma^2 = 0$ as then variance reduction is not needed; indeed, vanilla SGD already converges to the optimum, and with a fixed stepsize. We end the discussion here by remarking that this *hints* at a possible existence of a more unified theory, one that would include both vanilla and variance-reduced SGD.

[1] In this paper, by *vanilla* SGD we refer to SGD variants with or without importance sampling and mini-batching, but *excluding* variance-reduced variants, such as SAGA (Defazio et al., 2014) and SVRG (Johnson and Zhang, 2013).

[2] Except for the full batch strategy, which is prohibitively expensive.

Distributed SGD, quantization and variance reduction. When SGD is implemented in a distributed fashion, the problem is often expressed in the form (3), where n is the number of workers/nodes, and f_i corresponds to the loss based on data stored on node i. Depending on the number of data points stored on each node, it may or may not be efficient to compute the gradient of f_i in each iteration. In general, SGD is implemented in this way: each node i first computes a stochastic gradient g_i^k of f_i at the current point x^k (maintained individually by each node). These gradients are then aggregated by a master node (Shamir et al., 2014; Konečný and Richtárik, 2018), in-network by a switch (Sapio et al., 2019), or a different technique best suited to the architecture used. To alleviate the communication bottleneck, various lossy update compression strategies such as quantization (Seide et al., 2014; Gupta et al., 2015; Zhang et al., 2017), sparsification (Konečný and Richtárik, 2018; Alistarh et al., 2018; Wangni et al., 2018) and dithering (Alistarh et al., 2017) were proposed. The basic idea is for each worker to apply a randomized transformation $Q : \mathbb{R}^d \to \mathbb{R}^d$ to g_i^k, resulting in a vector which is still an unbiased estimator of the gradient, but one that can be communicated with fewer bits. Mathematically, this amounts to injecting additional noise into the already noisy stochastic gradient g_i^k. The field of quantized SGD is still young, and even some basic questions remained open until recently. For instance, there was no distributed quantized SGD capable of provably solving (1) until the DIANA algorithm (Mishchenko et al., 2019a) was introduced. DIANA applies quantization to *gradient differences*, and in so doing is able to learn the gradients at the optimum, which makes it able to work for any regularizer R. DIANA has some structural similarities with SEGA (Hanzely et al., 2018)—the first coordinate descent type method which works for non-separable regularizers—but a more precise relationship remains elusive. When the functions of f_i are of a finite-sum structure as in (4), one can apply variance reduction to reduce the variance of the stochastic gradients g_i^k together with quantization, resulting in the VR-DIANA method (Horváth et al., 2019). This is the first distributed quantized SGD method which provably converges to the solution of (1)+(4) with a fixed stepsize.

Randomized coordinate descent (RCD). Lastly, in a distinctly separate strain, there are SGD methods for the coordinate/subspace descent variety (Nesterov, 2012). While it is possible to see *some* RCD methods as special cases of (5)+(6), most of them do not follow this algorithmic template. First, standard RCD methods use different stepsizes for updating different coordinates (Qu and Richtárik, 2016), and this seems to be crucial to their success. Second, until the recent discovery of the SEGA method, RCD methods were not able to converge with non-separable regularizers. Third, RCD methods are naturally variance-reduced in the $R = 0$ case as partial derivatives at the optimum are all zero. As a consequence, attempts at creating variance-reduced RCD methods seem to be futile. Lastly, RCD methods are typically analyzed using different techniques. While there are deep links between standard SGD and RCD methods, these are often indirect and rely on duality (Shalev-Shwartz and Zhang, 2013; Csiba and Richtárik, 2018; Gower and Richtárik, 2015).

3 Contributions

As outlined in the previous section, the world of SGD is vast and beautiful. It is formed by many largely disconnected islands populated by elegant and efficient methods, with their own applications, intuitions, and convergence analysis techniques. While some links already exist (e.g., the unification of importance sampling and mini-batching variants under the arbitrary sampling umbrella), there is no comprehensive general theory. It is becoming increasingly difficult for the community to understand the relationships between these variants, both in theory and practice. New variants are yet to be discovered, but it is not clear what tangible principles one should adopt beyond intuition to aid the discovery. This situation is exacerbated by the fact that a number of different assumptions on the stochastic gradient, of various levels of strength, is being used in the literature.

The main contributions of this work include:

• **Unified analysis.** In this work we propose a *unifying theoretical framework* which covers all of the variants of SGD outlined in Section 2. As a by-product, we obtain the *first unified analysis* of vanilla and variance-reduced SGD methods. For instance, our analysis covers as special cases vanilla SGD methods from (Nguyen et al., 2018) and (Gower et al., 2019), variance-reduced SGD methods such as SAGA (Defazio et al., 2014), L-SVRG (Hofmann et al., 2015; Kovalev et al., 2019) and JacSketch (Gower et al., 2018). Another by-product is *the unified analysis of* SGD *methods which include* RCD. For instance, our theory covers the subspace descent method SEGA (Hanzely et al., 2018) as a special case. Lastly, our framework is general enough to capture the phenomenon of *quantization*. For instance, we obtain the DIANA and VR-DIANA methods in special cases.

• **Generalization of existing methods.** An important yet *relatively* minor contribution of our work is that it enables *generalization* of knowns methods. For instance, some particular methods we consider, such as L-SVRG (Alg 10) (Kovalev et al., 2019), were not analyzed in the proximal ($R \neq 0$) case before. To illustrate how this can be done within our framework, we do it here for L-SVRG. Further, most[3] of the methods we analyze can be extended to the *arbitrary sampling* paradigm.

- **Sharp rates.** In all known special cases, the rates obtained from our general theorem (Theorem 4.1) are the *best known rates* for these methods.

- **New methods.** Our general analysis provides estimates for a possibly infinite array of new and yet-to-be-developed variants of SGD. One only needs to verify that Assumption 4.1 holds, and a complexity estimate is readily furnished by Theorem 4.1. Selected existing and new methods that fit our framework are summarized in Table 1. This list is for illustration only, we believe that future work by us and others will lead to its rapid expansion.

- **Experiments.** We show through extensive experimentation that some of the *new* and *generalized* methods proposed here and analyzed via our framework have some intriguing practical properties when compared against appropriately selected existing methods.

4 Main Result

We first introduce the key assumption on the stochastic gradients g^k enabling our general analysis (Assumption 4.1), then state our assumptions on f (Assumption 4.2), and finally state and comment on our unified convergence result (Theorem 4.1).

Notation. We use the following notation. $\langle x, y\rangle := \sum_i x_i y_i$ is the standard Euclidean inner product, and $\|x\| := \langle x, x\rangle^{1/2}$ is the induced ℓ_2 norm. For simplicity we assume that (1) has a unique minimizer, which we denote x^*. Let $D_f(x, y)$ denote the *Bregman divergence* associated with f: $D_f(x, y) := f(x) - f(y) - \langle \nabla f(y), x - y\rangle$. We often write $[n] := \{1, 2, \dots, n\}$.

4.1 Key assumption

Our first assumption is of key importance. It is mainly an assumption on the sequence of stochastic gradients $\{g^k\}$ generated by an arbitrary randomized algorithm. Besides unbiasedness (see (7)), we require two recursions to hold for the iterates x^k and the stochastic gradients g^k of a randomized method. We allow for flexibility by casting these inequalities in a parametric manner.

Assumption 4.1. Let $\{x^k\}$ be the random iterates produced by proximal SGD (Algorithm in Eq (5)). We first assume that the stochastic gradients g^k are unbiased

$$\mathbf{E}\left[g^k \mid x^k\right] = \nabla f(x^k), \tag{7}$$

for all $k \geq 0$. Further, we assume that there exist nonnegative constants A, B, C, D_1, D_2, ρ and a (possibly) random sequence $\{\sigma_k^2\}_{k\geq 0}$ such that the following two relations hold[4]

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right] \leq 2AD_f(x^k, x^*) + B\sigma_k^2 + D_1, \tag{8}$$

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid \sigma_k^2\right] \leq (1-\rho)\sigma_k^2 + 2CD_f(x^k, x^*) + D_2, \tag{9}$$

The expectation above is with respect to the randomness of the algorithm.

The unbiasedness assumption (7) is standard. The key innovation we bring is inequality (8) coupled with (9). We argue, and justify this statement by furnishing many examples in Section 5, that these inequalities capture the essence of a wide array of existing and some new SGD methods, including vanilla, variance reduced, arbitrary sampling, quantized and coordinate descent variants. Note that in the case when $\nabla f(x^*) = 0$ (e.g., when $R = 0$), the inequalities in Assumption 4.1 reduce to

$$\mathbf{E}\left[\left\|g^k\right\|^2 \mid x^k\right] \leq 2A(f(x^k) - f(x^*)) + B\sigma_k^2 + D_1, \tag{10}$$

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid \sigma_k^2\right] \leq (1-\rho)\sigma_k^2 + 2C(f(x^k) - f(x^*)) + D_2. \tag{11}$$

Similar inequalities can be found in the analysis of stochastic first-order methods. However, this is the first time that such inequalities are generalized, equipped with parameters, and elevated to the status of an assumption that can be used on its own, independently from any other details defining the underlying method that generated them.

To give a further intuition about inequalities (8) and (9), we shall note that sequence σ_k usually represents the portion of noise that can gradually decrease over the course of optimization while constants D_1, D_2 represent a static noise. On the other hand, constants A, C are usually related to some measure of smoothness of the objective. For instance, the parameters for (deterministic) gradient descent can be chosen as $A = L, B = C = D_1 = D_2 = \sigma_k^2 = \rho = 0$. For an overview of parameter choices for specific instances of (5), see Table 2. Note also that the choice of parameters of (8) and (9) is not unique, however this has no impact on convergence rates we provide.

[3] Our analysis allows for arbitrary sampling of all methods except of those using partial derivatives such as SEGA or N-SEGA. We shall note that arbitrary sampling for SEGA was developed concurrently in (Hanzely and Richtárik, 2019b). Note that (Hanzely and Richtárik, 2019b) proposes many novel variance reduced algorithms, for some of which we can obtain best rates. A detailed discussion and comparison to (Hanzely and Richtárik, 2019b) is provided in Remark A.4 in the Appendix

[4] For convex and L-smooth f, one can show that $\|\nabla f(x) - \nabla f(y)\|^2 \leq 2LD_f(x, y)$. Hence, D_f can be used as a measure of proximity for the gradients.

4.2 Main theorem

For simplicity, we shall assume throughout that f is (μ, x^*)-strongly quasi-convex, which is a generalization of μ-strong convexity. We leave an analysis under different assumptions on f to future work.

Assumption 4.2 ((μ, x^*)-strong quasi-convexity). There exists $\mu > 0$ such that $f : \mathbb{R}^d \to \mathbb{R}$ satisfies the following inequality for all $x \in \mathbb{R}^d$:

$$f(x^*) \geq f(x) + \langle \nabla f(x), x^* - x \rangle + \frac{\mu}{2} \left\| x^* - x \right\|^2. \quad (12)$$

We are now ready to present the key lemma of this paper which states per iteration recurrence to analyze (5). Due to space limitations, we present the proof in Section 4 of the Appendix.

Lemma 4.1. Let Assumptions 4.1 and 4.2 be satisfied. Then the following inequality holds for all $k \geq 0$:

$$\begin{aligned}
&\mathbf{E}\left[\left\|x^{k+1} - x^*\right\|^2\right] + M\gamma^2 \mathbf{E}\left[\sigma_{k+1}^2\right] \\
&\quad + 2\gamma\left(1 - \gamma(A + CM)\right)\mathbf{E}\left[D_f(x^k, x^*)\right] \\
&\leq (1 - \gamma\mu)\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] + (1 - \rho)\, M\gamma^2 \mathbf{E}\left[\sigma_k^2\right] \\
&\quad + B\gamma^2 \mathbf{E}\left[\sigma_k^2\right] + (D_1 + MD_2)\gamma^2.
\end{aligned}$$

Using recursively Lemma 4.1, we obtain the convergence rate of proximal SGD, which we state as Theorem 4.1.

Theorem 4.1. Let Assumptions 4.1 and 4.2 be satisfied. Choose constant M such that $M > \frac{B}{\rho}$. Choose a stepsize satisfying

$$0 < \gamma \leq \min\left\{\frac{1}{\mu}, \frac{1}{A + CM}\right\}. \quad (13)$$

Then the iterates $\{x^k\}_{k \geq 0}$ of proximal `SGD` (Algorithm (5)) satisfy

$$\begin{aligned}
\mathbf{E}\left[V^k\right] \leq &\max\left\{(1 - \gamma\mu)^k, \left(1 + \frac{B}{M} - \rho\right)^k\right\} V^0 \\
&+ \frac{(D_1 + MD_2)\gamma^2}{\min\left\{\gamma\mu, \rho - \frac{B}{M}\right\}}, \quad (14)
\end{aligned}$$

where the Lyapunov function V^k is defined by $V^k := \left\|x^k - x^*\right\|^2 + M\gamma^2\sigma_k^2$.

This theorem establishes a linear rate for a wide range of proximal `SGD` methods up to a certain oscillation radius, controlled by the additive term in (14), and namely, by parameters D_1 and D_2. As we shall see in Section A (refer to Table 2), the main difference between the vanilla and variance-reduced `SGD` methods is that while the former satisfy inequality (9) with $D_1 > 0$ or $D_2 > 0$, which in view of (14) prevents them from reaching the optimum x^* (using a fixed stepsize), the latter methods satisfy inequality (9) with $D_1 = D_2 = 0$, which in view of (14) enables them to reach the optimum.

5 The Classic, The Recent and The Brand New

In this section we deliver on the promise from the introduction and show how many existing and some new variants of `SGD` fit our general framework (see Table 1).

An overview. As claimed, our framework is powerful enough to include vanilla methods (✗ in the "VR" column) as well as variance-reduced methods (✓ in the "VR" column), methods which generalize to arbitrary sampling (✓ in the "AS" column), methods supporting gradient quantization (✓ in the "Quant" column) and finally, also `RCD` type methods (✓ in the "RCD" column).

For existing methods we provide a citation; new methods developed in this paper are marked accordingly. Due to space restrictions, all algorithms are described (in detail) in the Appendix; we provide a link to the appropriate section for easy navigation. While these details are important, the main message of this paper, i.e., the generality of our approach, is captured by Table 1. The "Result" column of Table 1 points to a corollary of Theorem 4.1; these corollaries state in detail the convergence statements for the various methods. In all cases where known methods are recovered, these corollaries of Theorem 4.1 recover the best known rates.

Parameters. From the point of view of Assumption 4.1, the methods listed in Table 1 exhibit certain patterns. To shed some light on this, in Table 2 we summarize the values of these parameters.

Note, for example, that for all methods the parameter A is non-zero. Typically, this a multiple of an appropriately defined smoothness parameter (e.g., L is the Lipschitz constant of the gradient of f, $\mathcal{L}$ and $\mathcal{L}_1$ in `SGD-SR`[5], `SGD-star` and `JacSketch` are *expected smoothness* parameters). In the three variants of the `DIANA` method, ω captures the variance of the quantization operator Q. That is, one assumes that $\mathbf{E}Q(x) = x$

[5] `SGD-SR` is first `SGD` method analyzed in the *arbitrary sampling* paradigm. It was developed using the *stochastic reformulation* approach (whence the "SR") pioneered in (Richtárik and Takáč, 2017) in a numerical linear algebra setting, and later extended to develop the `JacSketch` variance-reduction technique for finite-sum optimization (Gower et al., 2018).

Table 1: List of specific existing (in some cases generalized) and new methods which fit our general analysis framework. VR = variance reduced method, AS = arbitrary sampling, Quant = supports gradient quantization, RCD = randomized coordinate descent type method. [a] Special case of `SVRG` with 1 outer loop only; [b] Special case of `DIANA` with 1 node and quantization of exact gradient.

Problem	Method	Alg #	Citation	VR?	AS?	Quant?	RCD?	Section	Result
(1)+(2)	`SGD`	Alg 1	Nguyen et al. (2018)	✗	✗	✗	✗	A.1	Cor A.1
(1)+(3)	`SGD-SR`	Alg 2	Gower et al. (2019)	✗	✓	✗	✗	A.2	Cor A.2
(1)+(3)	`SGD-MB`	Alg 3	**NEW**	✗	✗	✗	✗	A.3	Cor A.3
(1)+(3)	`SGD-star`	Alg 4	**NEW**	✓	✓	✗	✗	A.4	Cor A.4
(1)+(3)	`SAGA`	Alg 5	Defazio et al. (2014)	✓	✗	✗	✗	A.5	Cor A.5
(1)+(3)	`N-SAGA`	Alg 6	**NEW**	✗	✗	✗	✗	A.6	Cor A.6
(1)	`SEGA`	Alg 7	Hanzely et al. (2018)	✓	✗	✗	✓	A.7	Cor A.7
(1)	`N-SEGA`	Alg 8	**NEW**	✗	✗	✗	✓	A.8	Cor A.8
(1)+(3)	`SVRG`[a]	Alg 9	Johnson and Zhang (2013)	✓	✗	✗	✗	A.9	Cor A.9
(1)+(3)	`L-SVRG`	Alg 10	Hofmann et al. (2015)	✓	✗	✗	✗	A.10	Cor A.10
(1)+(3)	`DIANA`	Alg 11	Mishchenko et al. (2019a)	✗	✗	✓	✗	A.11	Cor A.11
(1)+(3)	`DIANA`[b]	Alg 12	Mishchenko et al. (2019a)	✓	✗	✓	✗	A.11	Cor A.12
(1)+(3)	`Q-SGD-SR`	Alg 13	**NEW**	✗	✓	✓	✗	A.12	Cor A.13
(1)+(3)+(4)	`VR-DIANA`	Alg 14	Horváth et al. (2019)	✓	✗	✓	✗	A.13	Cor A.15
(1)+(3)	`JacSketch`	Alg 15	Gower et al. (2018)	✓	✓✗	✗	✗	A.14	Cor A.16

Table 2: The parameters for which the methods from Table 1 (special cases of (5)) satisfy Assumption 4.1. The meaning of the expressions appearing in the table, as well as their justification is defined in detail in the Appendix (Section A).

Method	A	B	ρ	C	D_1	D_2
`SGD`	$2L$	0	1	0	$2\sigma^2$	0
`SGD-SR`	$2\mathcal{L}$	0	1	0	$2\sigma^2$	0
`SGD-MB`	$\frac{A'+L(\tau-1)}{\tau}$	0	1	0	$\frac{D'}{\tau}$	0
`SGD-star`	$2\mathcal{L}$	0	1	0	0	0
`SAGA`	$2L$	2	$1/n$	L/n	0	0
`N-SAGA`	$2L$	2	$1/n$	L/n	$2\sigma^2$	$\frac{\sigma^2}{n}$
`SEGA`	$2dL$	$2d$	$1/d$	L/d	0	0
`N-SEGA`	$2dL$	$2d$	$1/d$	L/d	$2d\sigma^2$	$\frac{\sigma^2}{d}$
`SVRG`[a]	$2L$	2	0	0	0	0
`L-SVRG`	$2L$	2	p	Lp	0	0
`DIANA`	$\left(1+\frac{2\omega}{n}\right)L$	$\frac{2\omega}{n}$	α	$L\alpha$	$\frac{(1+\omega)\sigma^2}{n}$	$\alpha\sigma^2$
`DIANA`[b]	$(1+2\omega)L$	2ω	α	$L\alpha$	0	0
`Q-SGD-SR`	$2(1+\omega)\mathcal{L}$	0	1	0	$2(1+\omega)\sigma^2$	0
`VR-DIANA`	$\left(1+\frac{4\omega+2}{n}\right)L$	$\frac{2(\omega+1)}{n}$	α	$\left(\frac{1}{m}+4\alpha\right)L$	0	0
`JacSketch`	$2\mathcal{L}_1$	$\frac{2\lambda_{\max}}{n}$	$\lambda_{\min}$	$\frac{\mathcal{L}_2}{n}$	0	0

and $\mathbf{E}\|Q(x)-x\|^2 \leq \omega\|x\|^2$ for all $x \in \mathbb{R}^d$. In view of (13), large A means a smaller stepsize, which slows down the rate. Likewise, the variance ω also affects the parameter B, which in view of (14) also has an adverse effect on the rate. Further, as predicted by Theorem 4.1, whenever either $D_1 > 0$ or $D_2 > 0$, the corresponding method converges to an oscillation region only. These methods are not variance-reduced. All symbols used in Table 2 are defined in the appendix, in the same place where the methods are described and analyzed.

Five new methods. To illustrate the usefulness of our general framework, we develop *5 new variants* of `SGD` never explicitly considered in the literature before (see Table 1). Here we briefly motivate them; details can be found in the Appendix.

• `SGD-MB` (Algorithm 3). This method is specifically designed for functions of the finite-sum structure (4). As we show through experiments, this is a powerful mini-batch `SGD` method, with mini-batches formed with replacement as follows: in each iteration, we repeatedly (τ times) and independently pick $i \in [n]$ with probability $p_i > 0$. Stochastic gradient g^k is then formed by averaging the stochastic gradients $\nabla f_i(x^k)$ for all selected indices i (including each i as many times as this index was selected). This allows for a more practical importance mini-batch sampling implementation than what was until now possible (see Remark A.1 in the Appendix for more details and experiment in Figure 1).

- `SGD-star` (Algorithm 4). This new method forms a bridge between vanilla and variance-reduced `SGD` methods. While not practical, it sheds light on the role of variance reduction. Again, we consider functions of the finite-sum form (4). This methods answers the following question: assuming that the gradients $\nabla f_i(x^*)$, $i \in [n]$ are *known*, can they be used to design a more powerful `SGD` variant? The answer is *yes*, and `SGD-star` is the method. In its most basic form, `SGD-star` constructs the stochastic gradient via $g^k = \nabla f_i(x^k) - \nabla f_i(x^*) + \nabla f(x^*)$, where $i \in [n]$ is chosen uniformly at random. Inferring from Table 2, where $D_1 = D_2 = 0$, this method converges to x^*, and not merely to some oscillation region. Variance-reduced methods essentially work by iteratively constructing increasingly more accurate *estimates* of $\nabla f_i(x^*)$. Typically, the term σ_k^2 in the Lyapunov function of variance reduced methods will contain a term of the form $\sum_i \left\|h_i^k - \nabla f_i(x^*)\right\|^2$, with h_i^k being the estimators maintained by the method. Remarkably, `SGD-star` was never explicitly considered in the literature before.

- `N-SAGA` (Algorithm 6). This is a novel variant of `SAGA` (Defazio et al., 2014), one in which one does not have access to the gradients of f_i, but instead only has access to *noisy* stochastic estimators thereof (with noise σ^2). Like `SAGA`, `N-SAGA` is able to reduce the variance inherent in the finite sum structure (4) of the problem. However, it necessarily pays the price of noisy estimates of ∇f_i, and hence, just like vanilla `SGD` methods, is ultimately unable to converge to x^*. The oscillation region is governed by the noise level σ^2 (refer to D_1 and D_2 in Table 2). This method will be of practical importance for problems where each f_i is of the form (2), i.e., for problems of the "average of expectations" structure. Batch versions of `N-SAGA` would be well suited for distributed optimization, where each f_i is owned by a different worker, as in such a case one wants the workers to work in parallel.

- `N-SEGA` (Algorithm 8). This is a *noisy* extension of the `RCD`-type method `SEGA`, in complete analogy with the relationship between `SAGA` and `N-SAGA`. Here we assume that we only have noisy estimates of partial derivatives (with noise σ^2). This situation is common in derivative-free optimization, where such a noisy estimate can be obtained by taking (a random) finite difference approximation (Nesterov, 2017). Unlike `SEGA`, `N-SEGA` only converges to an oscillation region the size of which is governed by σ^2.

- `Q-SGD-SR` (Algorithm 13). This is a quantized version of `SGD-SR`, which is the first `SGD` method analyzed in the arbitrary sampling paradigm. As such, `Q-SGD-SR` is a vast generalization of the celebrated `QSGD` method (Alistarh et al., 2017).

6 Experiments

In this section we numerically verify the claims from the paper. We present only a fraction of experiments here, the rest is contained in Appendix B. Besides an extended version of experiment described here, we also provide experiments on `SGD-star`, as well as about `N-SEGA` (recall that both are new methods).

In Section A.3, we describe in detail the `SGD-MB` method already outlined before. The main advantage of `SGD-MB` is that the sampling procedure it employs can be implemented in just $\mathcal{O}(\tau \log n)$ time. In contrast, even the simplest without-replacement sampling which selects each function into the minibatch with a prescribed probability independently (we will refer to it as independent `SGD`) requires n calls of a uniform random generator. We demonstrate numerically that `SGD-MB` has essentially identical iteration complexity to independent `SGD` in practice. We consider logistic regression with Tikhonov regularization. For a fixed expected sampling size τ, consider two options for the probability of sampling the i-th function:

(i) $\frac{\tau}{n}$, or

(ii) $\frac{\|a_i\|^2+\lambda}{\delta+\|a_i\|^2+\lambda}$, where δ is such that[6] $\sum_{i=1}^n \frac{\|a_i\|^2+\lambda}{\delta+\|a_i\|^2+\lambda} = 1$.

The results can be found in Figure 1, where we also report the choice of stepsize γ and the choice of τ in the legend and title of the plot, respectively.

Indeed, iteration complexity of `SGD-MB` and independent `SGD` is almost identical. Since the cost of each iteration of `SGD-MB` is cheaper[7], we conclude superiority of `SGD-MB` to independent `SGD`.

7 Limitations and Extensions

Although our approach is rather general, we still see several possible directions for future extensions, includ-

[6] An `RCD` version of this sampling was proposed in (Hanzely and Richtárik, 2019a); it was shown to be superior to uniform sampling both in theory and practice.

[7] The relative difference between iteration costs of `SGD-MB` and independent `SGD` can be arbitrary, especially for the case when cost of evaluating $\nabla f_i(x)$ is cheap, n is huge and $n \gg \tau$. In such case, cost of one iteration of `SGD-MB` is $\tau \mathrm{Cost}(\nabla f_i) + \tau \log(n)$ while the cost of one iteration of independent `SGD` is $\tau \mathrm{Cost}(\nabla f_i) + n$.

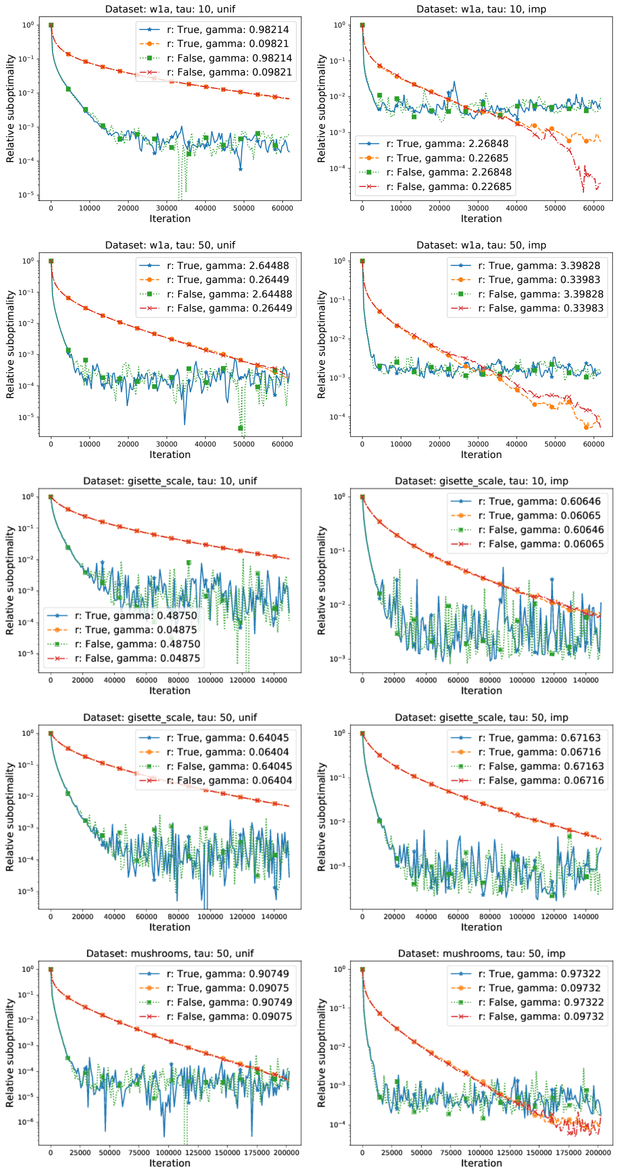

Figure 1: `SGD-MB` and independent `SGD` applied on LIBSVM (Chang and Lin, 2011). Title label "unif" corresponds to probabilities chosen by (i) while label "imp" corresponds to probabilities chosen by (ii). Lastly, legend label "r" corresponds to "replacement" with value "True" for `SGD-MB` and value "False" for independent `SGD`.

ing:

- We believe our results can be extended to *weakly convex* functions. However, producing a comparable result in the *nonconvex* case remains a major open problem.

- It would be further interesting to unify our theory with *biased* gradient estimators. If this was possible, one could recover methods as `SAG` (Roux et al., 2012) in special cases, or obtain rates for the zero-order optimization. We have some preliminary results in this direction already.

- Although our theory allows for non-uniform stochasticity, it does not recover the best known rates for `RCD` type methods with *importance sampling*. It would be thus interesting to provide a more refined analysis capable of capturing importance sampling phenomena more accurately.

- An extension of Assumption 4.1 to *iteration dependent* parameters A, B, C, D_1, D_2, ρ would enable an array of new methods, such as `SGD` with decreasing stepsizes. Such an extension is rather very straightforward.

- It would be interesting to provide a unified analysis of stochastic methods with *acceleration* and *momentum*. In fact, (Kulunchakov and Mairal, 2019) provide (separately) a unification of some methods with and without variance reduction. Hence, an attempt to combine our insights with their approach seems to be a promising starting point in these efforts.

Acknowledgments

The research of Eduard Gorbunov was supported by RFBR, project number 18-31-20005 mol_a_ved.

References

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. (2017). QSGD: Communication-efficient SGD via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pages 1709–1720.

Alistarh, D., Hoefler, T., Johansson, M., Konstantinov, N., Khirirat, S., and Renggli, C. (2018). The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems*, pages 5977–5987.

Chang, C.-C. and Lin, C.-J. (2011). LibSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27.

Csiba, D. and Richtárik, P. (2018). Coordinate descent face-off: primal or dual? In *JMLR Workshop and Conference Proceedings, The 29th International Conference on Algorithmic Learning Theory*.

Defazio, A., Bach, F., and Lacoste-Julien, S. (2014). SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems*, pages 1646–1654.

Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. (2019). SGD: General analysis and improved rates. *arXiv preprint arXiv:1901.09401*.

Gower, R. M. and Richtárik, P. (2015). Randomized iterative methods for linear systems. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1660–1690.

Gower, R. M. and Richtárik, P. (2015). Stochastic dual ascent for solving linear systems. *arXiv:1512.06890*.

Gower, R. M., Richtárik, P., and Bach, F. (2018). Stochastic quasi-gradient methods: Variance reduction via Jacobian sketching. *arXiv preprint arXiv:1805.02632*.

Gupta, S., Agrawal, A., Gopalakrishnan, K., and Narayanan, P. (2015). Deep learning with limited numerical precision. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pages 1737–1746. JMLR.org.

Hanzely, F., Mishchenko, K., and Richtárik, P. (2018). SEGA: Variance reduction via gradient sketching. In *Advances in Neural Information Processing Systems 31*, pages 2082–2093.

Hanzely, F. and Richtárik, P. (2019a). Accelerated coordinate descent with arbitrary sampling and best rates for minibatches. In *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 304–312. PMLR.

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

Hofmann, T., Lucchi, A., Lacoste-Julien, S., and McWilliams, B. (2015). Variance reduced stochastic gradient descent with neighbors. In *Advances in Neural Information Processing Systems*, pages 2305–2313.

Horváth, S., Kovalev, D., Mishchenko, K., Stich, S., and Richtárik, P. (2019). Stochastic distributed learning with gradient quantization and variance reduction. *arXiv preprint arXiv:1904.05115*.

Johnson, R. and Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems 26*, pages 315–323.

Konečný, J., Lu, J., Richtárik, P., and Takáč, M. (2016). Mini-batch semi-stochastic gradient descent in the proximal setting. *IEEE Journal of Selected Topics in Signal Processing*, 10(2):242–255.

Konečný, J. and Richtárik, P. (2018). Randomized distributed mean estimation: accuracy vs communication. *Frontiers in Applied Mathematics and Statistics*, 4(62):1–11.

Kovalev, D., Horváth, S., and Richtárik, P. (2019). Don't jump through hoops and remove those loops: SVRG and Katyusha are better without the outer loop. *arXiv preprint arXiv:1901.08689*.

Kulunchakov, A. and Mairal, J. (2019). Estimate sequences for variance-reduced stochastic composite optimization. *arXiv preprint arXiv:1905.02374*.

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

Mishchenko, K., Hanzely, F., and Richtárik, P. (2019b). 99% of distributed optimization is a waste of time: The issue and how to fix it. *arXiv preprint arXiv:1901.09437*.

Needell, D., Srebro, N., and Ward, R. (2015). Stochastic gradient descent, weighted sampling, and the randomized Kaczmarz algorithm. *Mathematical Programming*, 155(1–2):549–573.

Nemirovski, A., Juditsky, A., Lan, G., and Shapiro, A. (2009). Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609.

Nesterov, Y. (2012). Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362.

Nesterov, Y. (2017). Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566.

Nguyen, L., Nguyen, P. H., van Dijk, M., Richtárik, P., Scheinberg, K., and Takáč, M. (2018). SGD and Hogwild! Convergence without the bounded gradients assumption. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3750–3758, Stockholmsmässan, Stockholm Sweden. PMLR.

Nguyen, L. M., Liu, J., Scheinberg, K., and Takáč, M. (2017). SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 2613–2621. PMLR.

Qu, Z. and Richtárik, P. (2016). Coordinate descent with arbitrary sampling I: Algorithms and complexity. *Optimization Methods and Software*, 31(5):829–857.

Qu, Z., Richtárik, P., and Zhang, T. (2015). Quartz: Randomized dual coordinate ascent with arbitrary sampling. In *Advances in Neural Information Processing Systems 28*, pages 865–873.

Richtárik, P. and Takáč, M. (2016). On optimal probabilities in stochastic coordinate descent methods. *Optimization Letters*, 10(6):1233–1243.

Richtárik, P. and Takáč, M. (2017). Stochastic reformulations of linear systems: algorithms and convergence theory. *arXiv:1706.01108*.

Robbins, H. and Monro, S. (1951). A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407.

Roux, N. L., Schmidt, M., and Bach, F. (2012). A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems*, pages 2663–2671.

Sapio, A., Canini, M., Ho, C., Nelson, J., Kalnis, P., Kim, C., Krishnamurthy, A., Moshref, M., Ports, D. R. K., and Richtárik, P. (2019). Scaling distributed machine learning with in-network aggregation. *arXiv preprint ArXiv:1903.06701*.

Seide, F., Fu, H., Droppo, J., Li, G., and Yu, D. (2014). 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. pages 1058–1062. ISCA.

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

Shalev-Shwartz, S. and Zhang, T. (2013). Stochastic dual coordinate ascent methods for regularized loss. *Journal of Machine Learning Research*, 14(1):567–599.

Shamir, O., Srebro, N., and Zhang, T. (2014). Communication-efficient distributed optimization using an approximate Newton-type method. In *Proceedings of the 31st International Conference on Machine Learning, PMLR*, volume 32, pages 1000–1008.

Tran-Dinh, Q., Pham, N. H., Phan, D. T., and Nguyen, L. M. (2019). Hybrid stochastic gradient descent algorithms for stochastic nonconvex optimization. *arXiv preprint arXiv:1905.05920*.

Vaswani, S., Bach, F., and Schmidt, M. (2019). Fast and faster convergence of SGD for over-parameterized models and an accelerated perceptron. In *22nd International Conference on Artificial Intelligence and Statistics*, volume 89 of *PMLR*, pages 1195–1204.

Wangni, J., Wang, J., Liu, J., and Zhang, T. (2018). Gradient sparsification for communication-efficient distributed optimization. In *Advances in Neural Information Processing Systems*, pages 1306–1316.

Wen, W., Xu, C., Yan, F., Wu, C., Wang, Y., Chen, Y., and Li, H. (2017). Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *Advances in Neural Information Processing Systems*, pages 1509–1519.

Zhang, H., Li, J., Kara, K., Alistarh, D., Liu, J., and Zhang, C. (2017). ZipML: Training linear models with end-to-end low precision, and a little bit of deep learning. In Precup, D. and Teh, Y. W., editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 4035–4043, International Convention Centre, Sydney, Australia. PMLR.

Zhao, P. and Zhang, T. (2015). Stochastic optimization with importance sampling for regularized loss minimization. In *Proceedings of the 32nd International Conference on Machine Learning, PMLR*, volume 37, pages 1–9.

Appendix
A Unified Theory of SGD: Variance Reduction, Sampling, Quantization and Coordinate Descent

Contents

A Special Cases

A.1 Proximal SGD for stochastic optimization

Algorithm 1 SGD

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$, distribution $\mathcal{D}$ over $\xi$
  for $k = 0, 1, 2, \ldots$ do
      Sample $\xi \sim \mathcal{D}$
      $g^k = \nabla f_\xi(x^k)$
      $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

We start with stating the problem, the assumptions on the objective and on the stochastic gradients for SGD (Nguyen et al., 2018). Consider the expectation minimization problem

$$\min_{x\in\mathbb{R}^d} f(x) + R(x), \quad f(x) \coloneqq \mathbf{E}_{\mathcal{D}}\left[f_\xi(x)\right] \tag{15}$$

where $\xi \sim \mathcal{D}$, $f_\xi(x)$ is differentiable and L-smooth almost surely in ξ.

Lemma A.1 shows that the stochastic gradient $g^k = \nabla f_\xi(x^k)$ satisfies Assumption 4.1. The corresponding choice of parameters can be found in Table 2.

Lemma A.1 (Generalization of Lemmas 1,2 from (Nguyen et al., 2018))**.** Assume that $f_\xi(x)$ is convex in x for every ξ. Then for every $x \in \mathbb{R}^d$

$$\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_\xi(x) - \nabla f(x^*)\|^2\right] \le 4L(D_f(x, x^*)) + 2\sigma^2, \tag{16}$$

where $\sigma^2 \coloneqq \mathbf{E}_\xi\left[\|\nabla f_\xi(x^*)\|^2\right]$. If further $f(x)$ is μ-strongly convex with possibly non-convex f_ξ, then for every $x \in \mathbb{R}^d$

$$\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_\xi(x) - \nabla f(x^*)\|^2\right] \le 4L\kappa(D_f(x, x^*)) + 2\sigma^2, \tag{17}$$

where $\kappa = \frac{L}{\mu}$.

Corollary A.1. Assume that $f_\xi(x)$ is convex in x for every ξ and f is μ-strongly quasi-convex. Then SGD with $\gamma \le \frac{1}{2L}$ satisfies

$$\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] \le (1-\gamma\mu)^k \left\|x^0 - x^*\right\|^2 + \frac{2\gamma\sigma^2}{\mu}. \tag{18}$$

If we further assume that $f(x)$ is μ-strongly convex with possibly non-convex $f_\xi(x)$, SGD with $\gamma \le \frac{1}{2L\kappa}$ satisfies (18) as well.

Proof. It suffices to plug parameters from Table 2 into Theorem 4.1. □

Proof of Lemma A.1

The proof is a direct generalization to the one from (Nguyen et al., 2018). Note that

$$
\begin{aligned}
&\frac{1}{2}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f(x^*)\|^2\right]-\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x^*)-\nabla f(x^*)\|^2\right]\\
&\qquad=\frac{1}{2}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f(x^*)\|^2-\|\nabla f_{\xi}(x^*)-\nabla f(x^*)\|^2\right]\\
&\qquad\overset{(77)}{\leq}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f_{\xi}(x^*)\|^2\right]\\
&\qquad\leq 2LD_f(x,x^*).
\end{aligned}
$$

It remains to rearrange the above to get (16). To obtain (17), we shall proceed similarly:

$$
\begin{aligned}
&\frac{1}{2}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f(x^*)\|^2\right]-\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x^*)-\nabla f(x^*)\|^2\right]\\
&\qquad=\frac{1}{2}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f(x^*)\|^2-\|\nabla f_{\xi}(x^*)-\nabla f(x^*)\|^2\right]\\
&\qquad\overset{(77)}{\leq}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f_{\xi}(x^*)\|^2\right]\\
&\qquad\leq L^2\|x-x^*\|^2\\
&\qquad\leq 2\frac{L^2}{\mu}D_f(x,x^*).
\end{aligned}
$$

Again, it remains to rearrange the terms.

A.2 SGD-SR

In this section, we recover convergence result of SGD under expected smoothness property from (Gower et al., 2019). This setup allows obtaining tight convergence rates of SGD under arbitrary stochastic reformulation of finite sum minimization[8].

The stochastic reformulation is a special instance of (15):

$$
\min_{x\in\mathbb{R}^d} f(x)+R(x),\quad f(x)=\mathbf{E}_{\mathcal{D}}\left[f_{\xi}(x)\right],\quad f_{\xi}(x):=\frac{1}{n}\sum_{i=1}^{n}\xi_i f_i(x) \tag{19}
$$

where ξ is a random vector from distribution $\mathcal{D}$ such that for all i: $\mathbf{E}_{\mathcal{D}}\left[\xi_i\right]=1$ and f_i (for all i) is smooth, possibly non-convex function. We next state the expextes smoothness assumption. A specific instances of this assumption allows to get tight convergence rates of SGD, which we recover in this section.

Algorithm 2 SGD-SR

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$, distribution $\mathcal{D}$ over $\xi \in \mathbb{R}^n$ such that $\mathbf{E}_{\mathcal{D}}[\xi]$ is vector of ones
  for $k = 0, 1, 2, \dots$ do
      Sample $\xi \sim \mathcal{D}$
      $g^k = \nabla f_{\xi}(x^k)$
      $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

Assumption A.1 (Expected smoothness)**.** We say that f is $\mathcal{L}$-smooth in expectation with respect to distribution $\mathcal{D}$ if there exists $\mathcal{L}=\mathcal{L}(f,\mathcal{D})>0$ such that

$$
\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\xi}(x)-\nabla f_{\xi}(x^*)\|^2\right]\leq 2\mathcal{L}D_f(x,x^*), \tag{20}
$$

for all $x\in\mathbb{R}^d$. For simplicity, we will write $(f,\mathcal{D})\sim ES(\mathcal{L})$ to say that (20) holds.

[8] For technical details on how to exploit expected smoothness for specific reformulations, see (Gower et al., 2019)

Next, we present Lemma A.2 which shows that choice of constants for Assumption 4.1 from Table 2 is valid.

Lemma A.2 (Generalization of Lemma 2.4, (Gower et al., 2019))**.** If $(f,\mathcal{D}) \sim ES(\mathcal{L})$, then

$$\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_\xi(x) - \nabla f(x^*)\|^2\right] \le 4\mathcal{L}D_f(x,x^*) + 2\sigma^2. \tag{21}$$

where $\sigma^2 \coloneqq \mathbf{E}_{\mathcal{D}}\left[\|\nabla f_\xi(x^*) - \nabla f(x^*)\|^2\right]$.

A direct consequence of Theorem 4.1 in this setup is Corollary A.2.

Corollary A.2. Assume that $f(x)$ is μ-strongly quasi-convex and $(f,\mathcal{D}) \sim ES(\mathcal{L})$. Then SGD-SR with $\gamma^k \equiv \gamma \le \frac{1}{2\mathcal{L}}$ satisfies

$$\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] \le (1-\gamma\mu)^k \left\|x^0 - x^*\right\|^2 + \frac{2\gamma\sigma^2}{\mu}. \tag{22}$$

Proof of Lemma A.2

Here we present the generalization of the proof of Lemma 2.4 from (Gower et al., 2019) for the case when $\nabla f(x^*) \neq 0$. In this proof all expectations are conditioned on x^k.

$$\begin{aligned}
\mathbf{E}\left[\|\nabla f_\xi(x) - \nabla f(x^*)\|^2\right] &= \mathbf{E}\left[\|\nabla f_\xi(x) - \nabla f_\xi(x^*) + \nabla f_\xi(x^*) - \nabla f(x^*)\|^2\right] \\
&\overset{(76)}{\le} 2\mathbf{E}\left[\|\nabla f_\xi(x) - \nabla f_\xi(x^*)\|^2\right] + 2\mathbf{E}\left[\|\nabla f_\xi(x^*) - \nabla f(x^*)\|^2\right] \\
&\overset{(20)}{\le} 4\mathcal{L}D_f(x,x^*) + 2\sigma^2.
\end{aligned}$$

A.3 SGD-MB

In this section, we present a specific practical formulation of (19) which was not considered in (Gower et al., 2019). The resulting algorithm (Algorithm 3) is novel; it was not considered in (Gower et al., 2019) as a specific instance of SGD-SR. The key idea behind SGD-MB is constructing unbiased gradient estimate via with-replacement sampling.

Consider random variable $\nu \sim \mathcal{D}$ such that

$$\mathbf{P}(\nu = i) = p_i; \qquad \sum_{i=1}^n p_i = 1. \tag{23}$$

Notice that if we define

$$\psi_i(x) \coloneqq \frac{1}{np_i} f_i(x), \qquad i = 1,2,\ldots,n, \tag{24}$$

then

$$f(x) = \frac{1}{n}\sum_{i=1}^n f_i(x) \overset{(24)}{=} \sum_{i=1}^n p_i\psi_i(x) \overset{(23)}{=} \mathbf{E}_{\mathcal{D}}\left[\psi_\nu(x)\right]. \tag{25}$$

So, we have rewritten the finite sum problem (3) into the *equivalent stochastic optimization problem*

$$\min_{x\in\mathbb{R}^d} \mathbf{E}_{\mathcal{D}}\left[\psi_\nu(x)\right]. \tag{26}$$

We are now ready to describe our method. At each iteration k we sample $\nu_i^k, \ldots, \nu_\tau^k \sim \mathcal{D}$ independently $(1 \le \tau \le n)$, and define $g^k \coloneqq \frac{1}{\tau}\sum_{i=1}^\tau \nabla\psi_{\nu_i^k}(x^k)$. Further, we use g^k as a stochastic gradient, resulting in Algorithm 3.

To remain in full generality, consider the following Assumption.

Assumption A.2. There exists constants $A' > 0$ and $D' \ge 0$ such that

$$\mathbf{E}_{\mathcal{D}}\left[\|\nabla\psi_\nu(x)\|^2\right] \le 2A'(f(x) - f(x^*)) + D' \tag{27}$$

for all $x \in \mathbb{R}^d$.

Algorithm 3 `SGD-MB`

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$, distribution $\mathcal{D}$ over $\nu$ such that (23) holds.
  for  $k = 0, 1, 2, \dots$  do
      Sample $\nu_i^k, \dots, \nu_\tau^k \sim \mathcal{D}$ independently
      $g^k = \frac{1}{\tau}\sum_{i=1}^{\tau} \nabla \psi_{\nu_i^k}(x^k)$
      $x^{k+1} = x^k - \gamma g^k$
  end for
```

Note that it is sufficient to have convex and smooth f_i in order to satisfy Assumption A.2, as Lemma A.3 states.

Lemma A.3. Let $\sigma^2 \coloneqq \mathbf{E}_{\mathcal{D}}\left[\left\|\nabla \psi_\nu(x^*)\right\|^2\right]$. If f_i are convex and L_i-smooth, then Assumption A.2 holds for $A' = 2\mathcal{L}$ and $D' = 2\sigma^2$, where

$$\mathcal{L} \leq \max_i \frac{L_i}{np_i}. \tag{28}$$

If moreover $\nabla f_i(x^*) = 0$ for all i, then Assumption A.2 holds for $A' = \mathcal{L}$ and $D' = 0$.

Next, Lemma A.4 states that Algorithm 3 indeed satisfies Assumption 4.1.

Lemma A.4. Suppose that Assumption A.2 holds. Then g^k is unbiased; i.e. $\mathbf{E}_{\mathcal{D}}\left[g^k\right] = \nabla f(x^k)$. Further,

$$\mathbf{E}_{\mathcal{D}}\left[\left\|g^k\right\|^2\right] \leq \frac{2A' + 2L(\tau - 1)}{\tau}(f(x^k) - f(x^*)) + \frac{D'}{\tau}.$$

Thus, parameters from Table 2 are validated. As a direct consequence of Theorem 4.1 we get Corollary A.3.

Corollary A.3. As long as $0 < \gamma \leq \frac{\tau}{A' + L(\tau-1)}$, we have

$$\mathbf{E}\left\|x^k - x^*\right\|^2 \leq (1 - \gamma\mu)^k \left\|x^0 - x^*\right\|^2 + \frac{\gamma D'}{\mu\tau}. \tag{29}$$

Remark A.1. For $\tau = 1$, `SGD-MB` is a special of the method from (Gower et al., 2019), Section 3.2. However, for $\tau > 1$, this is a different method; the difference lies in the with-replacement sampling. Note that with-replacement trick allows for efficient and implementation of independent importance sampling [9] with complexity $\mathcal{O}(\tau \log(n))$. In contrast, implementation of without-replacement importance sampling has complexity $\mathcal{O}(n)$, which can be significantly more expensive to the cost of evaluating $\sum_{i \in S} \nabla f_i(x)$.

Proof of Lemma A.4

Notice first that

$$\begin{aligned}
\mathbf{E}_{\mathcal{D}}\left[g^k\right] &\overset{(24)}{=} \frac{1}{\tau}\sum_{i=1}^{\tau} \mathbf{E}_{\mathcal{D}}\left[\frac{1}{np_{\nu_i^k}} \nabla f_{\nu_i^k}(x^k)\right] \\
&= \mathbf{E}_{\mathcal{D}}\left[\frac{1}{np_\nu}\nabla f_\nu(x^k)\right] \\
&\overset{(23)}{=} \sum_{i=1}^{n} p_i \frac{1}{np_i} \nabla f_i(x^k) \\
&= \nabla f(x_k).
\end{aligned}$$

[9] Distribution of random sets S for which random variables $i \in S$ and $j \in S$ are independent for $j \neq i$.

So, g^k is an unbiased estimator of the gradient $\nabla f(x^k)$. Next,

$$\begin{aligned}
\mathbf{E}_{\mathcal{D}}\left[\left\|g^k\right\|^2\right] &= \mathbf{E}_{\mathcal{D}}\left[\left\|\frac{1}{\tau}\sum_{i=1}^{\tau}\nabla\psi_{\nu_i^k}(x^k)\right\|^2\right] \\
&= \frac{1}{\tau^2}\mathbf{E}_{\mathcal{D}}\left[\sum_{i=1}^{\tau}\left\|\nabla\psi_{\nu_i^k}(x^k)\right\|^2 + 2\sum_{i<j}\left\langle\nabla\psi_{\nu_i^k}(x^k), \nabla\psi_{\nu_j^k}(x^k)\right\rangle\right] \\
&= \frac{1}{\tau}\mathbf{E}_{\mathcal{D}}\left[\left\|\nabla\psi_{\nu}(x^k)\right\|^2\right] + \frac{2}{\tau^2}\sum_{i<j}\left\langle\mathbf{E}_{\mathcal{D}}\left[\nabla\psi_{\nu_i^k}(x^k)\right], \mathbf{E}_{\mathcal{D}}\left[\nabla\psi_{\nu_j^k}(x^k)\right]\right\rangle \\
&= \frac{1}{\tau}\mathbf{E}_{\mathcal{D}}\left[\left\|\nabla\psi_{\nu}(x^k)\right\|^2\right] + \frac{\tau-1}{\tau}\left\|\nabla f(x^k)\right\|^2 \\
&\overset{(27)}{\leq} \frac{2A'(f(x^k)-f(x^*)) + D' + 2L(\tau-1)(f(x^k)-f(x^*))}{\tau}.
\end{aligned}$$

Proof of Lemma A.3

Let $\mathcal{L} = \mathcal{L}(f,\mathcal{D}) > 0$ be any constant for which

$$\mathbf{E}_{\xi\sim\mathcal{D}}\left\|\nabla\phi_\xi(x) - \nabla\phi_\xi(x^*)\right\|^2 \leq 2\mathcal{L}(f(x) - f(x^*)) \tag{30}$$

holds for all $x \in \mathbb{R}^d$. This is the expected smoothness property (for a single item sampling) from (Gower et al., 2019). It was shown in (Gower et al., 2019, Proposition 3.7) that (30) holds, and that $\mathcal{L}$ satisfies (28). The claim now follows by applying (Gower et al., 2019, Lemma 2.4).

A.4 SGD-star

Consider problem (19). Suppose that $\nabla f_i(x^*)$ is known for all i. In this section we present a novel algorithm — SGD-star — which is SGD-SR shifted by the stochastic gradient in the optimum. The method is presented under Expected Smoothness Assumption (20), obtaining general rates under arbitrary sampling. The algorithm is presented as Algorithm 4.

Algorithm 4 SGD-star

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$, distribution $\mathcal{D}$ over $\xi \in \mathbb{R}^n$ such that $\mathbf{E}_{\mathcal{D}}[\xi]$ is vector of ones
  for $k = 0, 1, 2, \ldots$ do
    Sample $\xi \sim \mathcal{D}$
    $g^k = \nabla f_\xi(x^k) - \nabla f_\xi(x^*) + \nabla f(x^*)$
    $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

Suppose that $(f,\mathcal{D}) \sim ES(\mathcal{L})$. Note next that SGD-star is just SGD-SR applied on objective $D_f(x,x^*)$ instead of $f(x)$ when $\nabla f(x^*) = 0$. This careful design of the objective yields $(D_f(\cdot,x^*),\mathcal{D}) \sim ES(\mathcal{L})$ and $\mathbf{E}_{\mathcal{D}}\left[\left\|\nabla_x D_{f_\xi}(x,x^*)\right\|^2 \mid x = x^*\right] = 0$, and thus Lemma (A.2) becomes

Lemma A.5 (Lemma 2.4, (Gower et al., 2019))**.** If $(f,\mathcal{D}) \sim ES(\mathcal{L})$, then

$$\mathbf{E}_{\mathcal{D}}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] \leq 4\mathcal{L}D_f(x^k,x^*). \tag{31}$$

A direct consequence of Corollary (thus also a direct consequence of Theorem 4.1) in this setup is Corollary A.4.

Corollary A.4. Suppose that $(f,\mathcal{D}) \sim ES(\mathcal{L})$. Then SGD-star with $\gamma = \frac{1}{2\mathcal{L}}$ satisfies

$$\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] \leq \left(1 - \frac{\mu}{2\mathcal{L}}\right)^k\left\|x^0 - x^*\right\|^2. \tag{32}$$

Remark A.2. Note that results from this section are obtained by applying results from A.2. Since Section A.3 presets a specific sampling algorithm for SGD-SR, the results can be thus extended to SGD-star as well.

Proof of Lemma A.5

In this proof all expectations are conditioned on x^k.

$$\begin{aligned}\mathbf{E}_{\mathcal{D}}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] &= \mathbf{E}_{\mathcal{D}}\left[\left\|\nabla f_\xi(x^k) - \nabla f_\xi(x^*)\right\|^2\right] \\ &\overset{(20)}{\leq} 4\mathcal{L}D_f(x^k, x^*).\end{aligned}$$

A.5 SAGA

In this section we show that our approach is suitable for SAGA (Defazio et al., 2014) (see Algorithm 5). Consider the finite-sum minimization problem

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) + R(x), \tag{33}$$

where f_i is convex, L-smooth for each i and f is μ-strongly convex.

Algorithm 5 SAGA (Defazio et al., 2014)

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$
  Set $\phi_j^0 = x^0$ for each $j \in [n]$
  for $k = 0, 1, 2, \ldots$ do
      Sample $j \in [n]$ uniformly at random
      Set $\phi_j^{k+1} = x^k$ and $\phi_i^{k+1} = \phi_i^k$ for $i \neq j$
      $g^k = \nabla f_j(\phi_j^{k+1}) - \nabla f_j(\phi_j^k) + \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\phi_i^k)$
      $x^{k+1} = \mathrm{prox}_{\gamma R}\left(x^k - \gamma g^k\right)$
  end for
```

Lemma A.6. We have

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right] \leq 4LD_f(x^k, x^*) + 2\sigma_k^2 \tag{34}$$

and

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] \leq \left(1 - \frac{1}{n}\right)\sigma_k^2 + \frac{2L}{n}D_f(x^k, x^*), \tag{35}$$

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

Clearly, Lemma A.6 shows that Algorithm 5 satisfies Assumption 4.1; the corresponding parameter choice can be found in Table 2. Thus, as a direct consequence of Theorem 4.1 with $M = 4n$ we obtain the next corollary.

Corollary A.5. SAGA with $\gamma = \frac{1}{6L}$ satisfies

$$\mathbf{E}V^k \leq \left(1 - \min\left\{\frac{\mu}{6L}, \frac{1}{2n}\right\}\right)^k V^0. \tag{36}$$

Proof of Lemma A.6

Note that Lemma A.6 is a special case of Lemmas 3,4 from (Mishchenko et al., 2019b) without prox term. We reprove it with prox for completeness.

Let all expectations be conditioned on x^k in this proof. Note that L-smoothness and convexity of f_i implies

$$\frac{1}{2L}\left\|\nabla f_i(x) - \nabla f_i(y)\right\|^2 \leq f_i(x) - f_i(y) - \langle\nabla f_i(y), x - y\rangle, \quad \forall x, y \in \mathbb{R}^d, i \in [n]. \tag{37}$$

By definition of g^k we have

$$\begin{aligned}
\mathbf{E}\left[\left\|g^k-\nabla f(x^*)\right\|^2\right] &= \mathbf{E}\left[\left\|\nabla f_j(\phi_j^{k+1})-\nabla f_j(\phi_j^k)+\frac{1}{n}\sum_{i=1}^n \nabla f_i(\phi_i^k)-\nabla f(x^*)\right\|^2\right]\\
&= \mathbf{E}\left[\left\|\nabla f_j(x^k)-\nabla f_j(x^*)+\nabla f_j(x^*)-\nabla f_j(\phi_j^k)+\frac{1}{n}\sum_{i=1}^n \nabla f_i(\phi_i^k)-\nabla f(x^*)\right\|^2\right]\\
&\overset{(76)}{\leq} 2\mathbf{E}\left[\left\|\nabla f_j(x^k)-\nabla f_j(x^*)\right\|^2 \mid x^k\right]\\
&\quad +2\mathbf{E}\left[\left\|\nabla f_j(x^*)-\nabla f_j(\phi_j^k)-\mathbf{E}\left[\nabla f_j(x^*)-\nabla f_j(\phi_j^k)\right]\right\|^2\right]\\
&\overset{(78)+(37)}{\leq} \frac{4L}{n}\sum_{i=1}^n D_{f_i}(x^k,x^*)+2\mathbf{E}\left[\left\|\nabla f_j(x^*)-\nabla f_j(\phi_j^k)\right\|^2 \mid x^k\right]\\
&= 4LD_f(x^k,x^*)+2\underbrace{\frac{1}{n}\sum_{i=1}^n \left\|\nabla f_i(\phi_i^k)-\nabla f_i(x^*)\right\|^2}_{\sigma_k^2}.
\end{aligned}$$

To proceed with (35), we have

$$\begin{aligned}
\mathbf{E}\left[\sigma_{k+1}^2\right] &= \frac{1}{n}\sum_{i=1}^n \mathbf{E}\left[\left\|\nabla f_i(\phi_i^{k+1})-\nabla f_i(x^*)\right\|^2\right]\\
&= \frac{1}{n}\sum_{i=1}^n\left(\frac{n-1}{n}\left\|\nabla f_i(\phi_i^k)-\nabla f_i(x^*)\right\|^2+\frac{1}{n}\left\|\nabla f_i(x^k)-\nabla f_i(x^*)\right\|^2\right)\\
&\overset{(37)}{\leq} \left(1-\frac{1}{n}\right)\frac{1}{n}\sum_{i=1}^n\left\|\nabla f_i(\phi_i^k)-\nabla f_i(x^*)\right\|^2\\
&\quad +\frac{2L}{n^2}\sum_{i=1}^n D_{f_i}(x^k,x^*)\\
&= \left(1-\frac{1}{n}\right)\sigma_k^2+\frac{2L}{n}D_f(x^k,x^*).
\end{aligned}$$

A.6 N-SAGA

Algorithm 6 Noisy SAGA (N-SAGA)

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$
  Set $\psi_j^0 = x^0$ for each $j \in [0]$
  for $k = 0, 1, 2, \ldots$ do
      Sample $j \in [n]$ uniformly at random and $\zeta$
      Set $g_j^{k+1} = g_j(x^k, \xi)$ and $g_i^{k+1} = g_i^k$ for $i \neq j$
      $g^k = g_j(x^k, \xi) - g_j^k + \frac{1}{n}\sum_{i=1}^n g_i^k$
      $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

Note that it can in practice happen that instead of $\nabla f_i(x)$ one can query $g_i(x,\zeta)$ such that $\mathbf{E}_\xi g_i(\cdot,\xi)=\nabla f_i(\cdot)$ and $\mathbf{E}_\xi\left\|g_i(\cdot,\xi)\right\|^2\leq\sigma^2$. This leads to a variant of SAGA which only uses noisy estimates of the stochastic gradients $\nabla_i(\cdot)$. We call this variant N-SAGA (see Algorithm 6).

Lemma A.7. We have

$$\mathbf{E}\left[\left\|g^k-\nabla f(x^*)\right\|^2 \mid x^k\right]\leq 4LD_f(x^k,x^*)+2\sigma_k^2+2\sigma^2, \tag{38}$$

and

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] \leq \left(1 - \frac{1}{n}\right)\sigma_k^2 + \frac{2L}{n}D_f(x^k, x^*) + \frac{\sigma^2}{n}, \tag{39}$$

where $\sigma_k^2 \coloneqq \frac{1}{n}\sum\limits_{i=1}^n \left\|g_i^k - \nabla f_i(x^*)\right\|^2$.

Corollary A.6. Let $\gamma = \frac{1}{6L}$. Then, iterates of Algorithm 6 satisfy

$$\mathbf{E}V^k \leq \left(1 - \min\left(\frac{\mu}{6L}, \frac{1}{2n}\right)\right)^k V^0 + \frac{\sigma^2}{L\min(\mu, \frac{3L}{n})}.$$

Analogous results can be obtained for L-SVRG.

Proof of Lemma A.7

Let all expectations be conditioned on x^k. By definition of g^k we have

$$\begin{aligned}
\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] &\leq \mathbf{E}\left[\left\|g_j(x^k,\zeta) - g_j^k + \frac{1}{n}\sum_{i=1}^n g_i^k - \nabla f(x^*)\right\|^2\right] \\
&= \mathbf{E}\left[\left\|g_j(x^k,\zeta) - \nabla f_j(x^*) + \nabla f_j(x^*) - g_j^k + \frac{1}{n}\sum_{i=1}^n g_i^k - \nabla f(x^*)\right\|^2\right] \\
&\overset{(76)}{\leq} 2\mathbf{E}\left[\left\|g_j(x^k,\zeta) - \nabla f_j(x^*)\right\|^2\right] \\
&\quad +2\mathbf{E}\left[\left\|\nabla f_j(x^*) - g_j^k - \mathbf{E}\left[\nabla f_j(x^*) - g_j^k\right]\right\|^2\right] \\
&\overset{(78)}{\leq} 2\mathbf{E}\left[\left\|g_j(x^k,\zeta) - \nabla f_j(x^*)\right\|^2\right] + 2\mathbf{E}\left[\left\|\nabla f_j(x^*) - g_j^k\right\|^2\right] \\
&= 2\mathbf{E}\left[\left\|g_j(x^k,\zeta) - \nabla f_j(x^*)\right\|^2\right] + 2\underbrace{\frac{1}{n}\sum_{i=1}^n \left\|g_i^k - \nabla f_i(x^*)\right\|^2}_{\sigma_k^2} \\
&\overset{(78)}{\leq} 2\mathbf{E}\left[\left\|\nabla f_j(x^k) - \nabla f_j(x^*)\right\|^2\right] + 2\sigma^2 + 2\sigma_k^2 \\
&\overset{(37)}{\leq} 4LD_f(x^k, x^*) + 2\sigma_k^2 + 2\sigma^2
\end{aligned}$$

For the second inequality, we have

$$\begin{aligned}
\mathbf{E}\left[\sigma_{k+1}^2\right] &= \frac{1}{n}\sum_{i=1}^n \mathbf{E}\left[\left\|g_i^{k+1} - \nabla f_i(x^*)\right\|^2\right] \\
&= \frac{1}{n}\sum_{i=1}^n \left(\frac{n-1}{n}\left\|g_i^k - \nabla f_i(x^*)\right\|^2 + \frac{1}{n}\mathbf{E}\left[\left\|g_i(x^k,\zeta) - \nabla f_i(x^*)\right\|^2\right]\right) \\
&\leq \frac{1}{n}\sum_{i=1}^n \left(\frac{n-1}{n}\left\|g_i^k - \nabla f_i(x^*)\right\|^2 + \frac{1}{n}\left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2 + \frac{\sigma^2}{n}\right) \\
&\overset{(37)}{\leq} \left(1 - \frac{1}{n}\right)\sigma_k^2 + \frac{2L}{n}D_f(x^k, x^*) + \frac{\sigma^2}{n}.
\end{aligned}$$

A.7 SEGA

We show that the framework recovers the simplest version of SEGA (i.e., setup from Theorem D1 from (Hanzely et al., 2018)) in the proximal setting[10].

[10] General version for arbitrary gradient sketches instead of partial derivatives can be recovered as well, however, we omit it for simplicity

Algorithm 7 SEGA (Hanzely et al., 2018)

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$
  Set $h^0 = 0$
  for  $k = 0, 1, 2, \ldots$  do
      Sample $j \in [d]$ uniformly at random
      Set $h^{k+1} = h^k + e_i(\nabla_i f(x^k) - h_i^k)$
      $g^k = de_i(\nabla_i f(x^k) - h_i^k) + h^k$
      $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

Lemma A.8. (Consequence of Lemmas A.3., A.4. from (Hanzely et al., 2018)) We have

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*) \mid x^k\right\|^2\right] \leq 2d\left\|\nabla f\left(x^k\right) - \nabla f(x^*)\right\|^2 + 2d\sigma_k^2$$

and

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] = \left(1 - \frac{1}{d}\right)\sigma_k^2 + \frac{1}{d}\left\|\nabla f\left(x^k\right) - \nabla f(x^*)\right\|^2,$$

where $\sigma_k^2 \coloneqq \left\|h^k - \nabla f(x^*)\right\|^2$.

Given that we have from convexity and smoothness $\left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2 \leq 2LD_f(x^k, x^*)$, Assumption 4.1 holds the parameter choice as per Table 2. Setting further $M = 4d^2$, we get the next corollary.

Corollary A.7. SEGA with $\gamma = \frac{1}{6dL}$ satisfies

$$\mathbf{E}V^k \leq \left(1 - \frac{\mu}{6dL}\right)^k V^0.$$

A.8 N-SEGA

Algorithm 8 Noisy SEGA (N-SEGA)

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$
  Set $h^0 = 0$
  for  $k = 0, 1, 2, \ldots$  do
      Sample $i \in [d]$ uniformly at random and sample $\xi$
      Set $h^{k+1} = h^k + e_i(g_i(x, \xi) - h_i^k)$
      $g^k = de_i(g_i(x, \xi) - h_i^k) + h^k$
      $x^{k+1} = x^k - \gamma g^k$
  end for
```

Here we assume that $g_i(x, \zeta)$ is a noisy estimate of the partial derivative $\nabla_i f(x)$ such that $\mathbf{E}_\zeta g_i(x, \zeta) = \nabla_i f(x)$ and $\mathbf{E}_\zeta |g_i(x, \zeta) - \nabla_i f(x)|^2 \leq \frac{\sigma^2}{d}$.

Lemma A.9. The following inequalities hold:

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] \leq 4dLD_f(x^k, x^*) + 2d\sigma_k^2 + 2d\sigma^2,$$

$$\mathbf{E}\left[\sigma_{k+1}^2\right] \leq \left(1 - \frac{1}{d}\right)\sigma_k^2 + \frac{2L}{d}D_f(x^k, x^*) + \frac{\sigma^2}{d},$$

where $\sigma_k^2 = \left\|h^k - \nabla f(x^*)\right\|^2$.

Corollary A.8. Let $\gamma = \frac{1}{6Ld}$. Applying Theorem 4.1 with $M = 4d^2$, iterates of Algorithm 8 satisfy

$$\mathbf{E}V^k \leq \left(1 - \frac{\mu}{6dL}\right)^k V^0 + \frac{\sigma^2}{L\mu}.$$

Proof of Lemma A.9

Let all expectations be conditioned on x^k. For the first bound, we write

$$g^k - \nabla f(x^*) = \underbrace{h^k - \nabla f(x^*) - dh_i^k e_i + d\nabla_i f(x^*) e_i}_{a} + \underbrace{dg_i(x^k,\xi)e_i - d\nabla_i f(x^*)e_i}_{b}.$$

Let us bound the expectation of each term individually. The first term can be bounded as

$$\begin{aligned} \mathbf{E}\|a\|^2 &= \mathbf{E}\left\|\left(\mathbf{I} - de_ie_i^\top\right)\left(h^k - \nabla f(x^*)\right)\right\|_2^2 \\ &= (d-1)\left\|h^k - \nabla f(x^*)\right\|^2 \\ &\leq d\left\|h^k - \nabla f(x^*)\right\|^2. \end{aligned}$$

The second term can be bounded as

$$\begin{aligned} \mathbf{E}\|b\|^2 &= \mathbf{E}_i\mathbf{E}_\xi\|dg_i(x,\xi)e_i - d\nabla f_i(x^*)e_i\|^2 \\ &= \mathbf{E}_i\mathbf{E}_\xi\left\|dg_i(x^k,\xi)e_i - d\nabla_i f(x^k)e_i\right\|^2 + \mathbf{E}_i\left\|d\nabla_i f(x^k)e_i - d\nabla f_i(x^*)e_i\right\|^2 \\ &\leq d\sigma^2 + d\left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2 \\ &\leq d\sigma^2 + 2LdD_f(x^k,x^*), \end{aligned}$$

where in the last step we used L–smoothness of f. It remains to combine the two bounds.

For the second bound, we have

$$\begin{aligned} \mathbf{E}\left\|h^{k+1} - \nabla f(x^*)\right\|^2 &= \mathbf{E}\left\|h^k + g_i(x^k,\xi)e_i - h_i^k - \nabla f(x^*)\right\|^2 \\ &= \mathbf{E}\left\|\left(\mathbf{I} - e_ie_i^\top\right)h^k + g_i(x^k,\xi)e_i - \nabla f(x^*)\right\|^2 \\ &= \mathbf{E}\left\|\left(\mathbf{I} - e_ie_i^\top\right)\left(h^k - \nabla f(x^*)\right)\right\|^2 + \mathbf{E}\left\|g_i(x^k,\xi)e_i - \nabla_i f(x^*)e_i\right\|^2 \\ &= \left(1 - \frac{1}{d}\right)\left\|h^k - \nabla f(x^*)\right\|^2 + \mathbf{E}\left\|g_i(x^k,\xi)e_i - \nabla_i f(x^k)e_i\right\|^2 \\ &\qquad + \mathbf{E}\left\|\nabla_i f(x^k)e_i - \nabla_i f(x^*)e_i\right\|^2 \\ &= \left(1 - \frac{1}{d}\right)\left\|h^k - \nabla f(x^*)\right\|^2 + \frac{\sigma^2}{d} + \frac{1}{d}\left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2 \\ &\leq \left(1 - \frac{1}{d}\right)\left\|h^k - \nabla f(x^*)\right\|^2 + \frac{\sigma^2}{d} + \frac{2L}{d}D_f(x^k,x^*). \end{aligned}$$

A.9 SVRG

Algorithm 9 SVRG (Johnson and Zhang, 2013)

```
Input: learning rate $\gamma > 0$, epoch length $m$, starting point $x^0 \in \mathbb{R}^d$
  $\phi = x^0$
  for $s = 0, 1, 2, \ldots$ do
      for $k = 0, 1, 2, \ldots, m-1$ do
          Sample $i \in \{1, \ldots, n\}$ uniformly at random
          $g^k = \nabla f_i(x^k) - \nabla f_i(\phi) + \nabla f(\phi)$
          $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
      end for
      $\phi = x^0 = \frac{1}{m}\sum_{k=1}^m x^k$
  end for
```

Let $\sigma_k^2 \coloneqq \frac{1}{n}\sum\limits_{i=1}^n \|\nabla f_i(\phi) - \nabla f_i(x^*)\|^2$. We will show that Lemma 4.1 recovers per-epoch analysis of SVRG in a special case.

Lemma A.10. For $k \mod m \neq 0$ we have

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right] \leq 4LD_f(x^k, x^*) + 2\sigma_k^2 \tag{40}$$

and

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] = \sigma_{k+1}^2 = \sigma_k^2. \tag{41}$$

Proof. The proof of (40) is identical to the proof of (34). Next, (41) holds since σ_k does not depend on k. □

Thus, Assumption 4.1 holds with parameter choice as per Table 2 and Lemma 4.1 implies the next corollary.

Corollary A.9.

$$\mathbf{E}\left\|x^{k+1} - x^*\right\|^2 + \gamma(1 - 2\gamma L)\mathbf{E}D_f(x^k, x^*) \leq (1 - \gamma\mu)\mathbf{E}\left\|x^k - x^*\right\|^2 + 2\gamma^2\mathbf{E}\sigma_k^2. \tag{42}$$

Recovering SVRG rate

Summing (42) for $k = 0, \ldots, m-1$ using $\sigma_k = \sigma_0$ we arrive at

$$\begin{aligned}\mathbf{E}\left\|x^m - x^*\right\|^2 + \sum_{k=1}^{m}\gamma(1 - 2\gamma L)\mathbf{E}D_f(x^k, x^*) &\leq (1 - \gamma\mu)\mathbf{E}\left\|x^0 - x^*\right\|^2 + 2m\gamma^2\mathbf{E}\sigma_0^2 \\ &\leq 2\left(\mu^{-1} + 2m\gamma^2 L\right) D_f(x^0, x^*) .\end{aligned}$$

Since D_f is convex in the first argument, we have

$$m\gamma(1 - 2\gamma L)D_f\left(\frac{1}{m}\sum_{k=1}^{m} x^k, x^*\right) \leq \left\|x^m - x^*\right\|^2 + \sum_{k=1}^{m}\gamma(1 - 2\gamma L)D_f(x^k, x^*)$$

and thus

$$D_f\left(\frac{1}{m}\sum_{k=1}^{m} x^k, x^*\right) \leq \frac{2\left(\mu^{-1} + 2m\gamma^2 L\right)}{m\gamma(1 - 2\gamma L)} D_f(x^0, x^*),$$

which recovers rate from Theorem 1 in (Johnson and Zhang, 2013).

A.10 L-SVRG

In this section we show that our approach also covers L-SVRG analysis from (Hofmann et al., 2015; Kovalev et al., 2019) (see Algorithm 10) with a minor extension – it allows for proximable regularizer R. Consider the finite-sum minimization problem

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) + R(x), \tag{43}$$

where each f_i convex and L-smooth for each i and f is μ-strongly convex.

Algorithm 10 L-SVRG (Hofmann et al. (2015); Kovalev et al. (2019))

```
Input: learning rate $\gamma > 0$, probability $p \in (0, 1]$, starting point $x^0 \in \mathbb{R}^d$
  $w^0 = x^0$
  for $k = 0, 1, 2, \ldots$ do
      Sample $i \in \{1, \ldots, n\}$ uniformly at random
      $g^k = \nabla f_i(x^k) - \nabla f_i(w^k) + \nabla f(w^k)$
      $x^{k+1} = x^k - \gamma g^k$
      $w^{k+1} = \begin{cases} x^k & \text{with probability } p \\ w^k & \text{with probability } 1 - p \end{cases}$
  end for
```

Note that the gradient estimator is again unbiased, i.e. $\mathbf{E}\left[g^k \mid x^k\right] = \nabla f(x^k)$. Next, Lemma A.11 provides with the remaining constants for Assumption 4.1. The corresponding choice is stated in Table 2.

Lemma A.11 (Lemma 4.2 and Lemma 4.3 from (Kovalev et al., 2019) extended to prox setup)**.** We have

$$\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right] \leq 4LD_f(x^k, x^*) + 2\sigma_k^2 \tag{44}$$

and

$$\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] \leq (1-p)\sigma_k^2 + 2LpD_f(x^k, x^*), \tag{45}$$

where $\sigma_k^2 \coloneqq \frac{1}{n}\sum\limits_{i=1}^n \left\|\nabla f_i(w^k) - \nabla f_i(x^*)\right\|^2$.

Next, applying Theorem 4.1 on Algorithm 10 with $M = \frac{4}{p}$ we get Corollary A.10.

Corollary A.10. L-SVRG with $\gamma = \frac{1}{6L}$ satisfies

$$\mathbf{E}V^k \leq \left(1 - \min\left\{\frac{\mu}{6L}, \frac{p}{2}\right\}\right)^k V^0. \tag{46}$$

Proof of Lemma A.11

Let all expectations be conditioned on x^k. Using definition of g^k

$$\begin{aligned}
\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] &\overset{\text{Alg. 10}}{=} \mathbf{E}\left[\left\|\nabla f_i(x^k) - \nabla f_i(x^*) + \nabla f_i(x^*) - \nabla f_i(w^k) + \nabla f(w^k) - \nabla f(x^*)\right\|^2\right] \\
&\overset{(76)}{\leq} 2\mathbf{E}\left[\left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2\right] \\
&\quad +2\mathbf{E}\left[\left\|\nabla f_i(x^*) - \nabla f_i(w^k) - \mathbf{E}\left[\nabla f_i(x^*) - \nabla f_i(w^k) \mid x^k\right]\right\|^2\right] \\
&\overset{(37),(78)}{\leq} 4LD_f(x^k, x^*) + 2\mathbf{E}\left[\left\|\nabla f_i(w^k) - \nabla f_i(x^*)\right\|^2\right] \\
&= 4LD_f(x^k, x^*) + 2\sigma_k^2.
\end{aligned}$$

For the second bound, we shall have

$$\begin{aligned}
\mathbf{E}\left[\sigma_{k+1}^2\right] &\overset{\text{Alg. 10}}{=} (1-p)\sigma_k^2 + \frac{p}{n}\sum_{i=1}^n \left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2 \\
&\overset{(37)}{\leq} (1-p)\sigma_k^2 + 2LpD_f(x^k, x^*).
\end{aligned}$$

A.11 DIANA

In this section we consider a distributed setup where each function f_i from (3) is owned by i-th machine (thus, we have all together n machines).

We show that our approach covers the analysis of DIANA from (Mishchenko et al., 2019a; Horváth et al., 2019). DIANA is a specific algorithm for distributed optimization with *quantization* – lossy compression of gradient updates, which reduces the communication between the server and workers[11].

In particular, DIANA quantizes gradient differences instead of the actual gradients. This trick allows for the linear convergence to the optimum once the full gradients are evaluated on each machine, unlike other popular quantization methods such as QSGD (Alistarh et al., 2017) or TernGrad (Wen et al., 2017). In this case, DIANA behaves as variance reduced method – it reduces a variance that was injected due to the quantization. However, DIANA also allows for evaluation of stochastic gradients on each machine, as we shall further see.

First of all, we introduce the notion of quantization operator.

Definition A.1 (Quantization)**.** We say that $\hat{\Delta}$ is a *quantization* of vector $\Delta \in \mathbb{R}^d$ and write $\hat{\Delta} \sim \mathrm{Q}(\Delta)$ if

$$\mathbf{E}\hat{\Delta} = \Delta, \qquad \mathbf{E}\left\|\hat{\Delta} - \Delta\right\|^2 \leq \omega \left\|\Delta\right\|^2 \tag{47}$$

for some $\omega > 0$.

Algorithm 11 `DIANA` (Mishchenko et al., 2019a; Horváth et al., 2019)

```
Input: learning rates $\alpha > 0$ and $\gamma > 0$, initial vectors $x^0, h_1^0, \ldots, h_n^0 \in \mathbb{R}^d$ and $h^0 = \frac{1}{n}\sum_{i=1}^n h_i^0$
for $k = 0, 1, \ldots$ do
    Broadcast $x^k$ to all workers
    for $i = 1, \ldots, n$ in parallel do
        Sample $g_i^k$ such that $\mathbf{E}[g_i^k \mid x^k] = \nabla f_i(x^k)$
        $\Delta_i^k = g_i^k - h_i^k$
        Sample $\hat{\Delta}_i^k \sim \mathrm{Q}(\Delta_i^k)$
        $h_i^{k+1} = h_i^k + \alpha\hat{\Delta}_i^k$
        $\hat{g}_i^k = h_i^k + \hat{\Delta}_i^k$
    end for
    $\hat{\Delta}^k = \frac{1}{n}\sum_{i=1}^n \hat{\Delta}_i^k$
    $g^k = \frac{1}{n}\sum_{i=1}^n \hat{g}_i^k = h^k + \hat{\Delta}^k$
    $x^{k+1} = \mathrm{prox}_{\gamma R}\left(x^k - \gamma g^k\right)$
    $h^{k+1} = \frac{1}{n}\sum_{i=1}^n h_i^{k+1} = h^k + \alpha\hat{\Delta}^k$
end for
```

The aforementioned method is applied to solve problem (1)+(3) where each f_i is convex and L-smooth and f is μ-strongly convex.

Lemma A.12 (Lemma 1 and consequence of Lemma 2 from (Horváth et al., 2019))**.** Suppose that $\alpha \le \frac{1}{1+\omega}$. For all iterations $k \ge 0$ of Algorithm 11 it holds

$$\begin{aligned}
\mathbf{E}\left[g^k \mid x^k\right] &= \nabla f(x^k), &(48)\\
\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right] &\le \left(1 + \frac{2\omega}{n}\right)\frac{1}{n}\sum_{i=1}^n \left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2 \\
&\quad + \frac{2\omega\sigma_k^2}{n} + \frac{(1+\omega)\sigma^2}{n}, &(49)\\
\mathbf{E}\left[\sigma_{k+1}^2 \mid x^k\right] &\le (1-\alpha)\sigma_k^2 + \frac{\alpha}{n}\sum_{i=1}^n \left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2 + \alpha\sigma^2. &(50)
\end{aligned}$$

where $\sigma_k^2 = \frac{1}{n}\sum\limits_{i=1}^n \left\|h_i^k - \nabla f_i(x^*)\right\|^2$ and σ^2 is such that $\frac{1}{n}\sum\limits_{i=1}^n \mathbf{E}\left[\left\|g_i^k - \nabla f_i(x^k)\right\|^2 \mid x^k\right] \le \sigma^2$.

Bounding further $\frac{1}{n}\sum_{i=1}^n \left\|\nabla f_i(x^k) - \nabla f_i(x^*)\right\|^2 \le 2LD_f(x^k, x^*)$ in the above Lemma, we see that Assumption 4.1 as per Table 2 is valid. Thus, as a special case of Theorem 4.1, we obtain the following corollary.

Corollary A.11. Assume that f_i is convex and L-smooth for all $i \in [n]$ and f is μ strongly convex, $\alpha \le \frac{1}{\omega+1}$, $\gamma \le \frac{1}{\left(1+\frac{2\omega}{n}\right)L + ML\alpha}$ where $M > \frac{2\omega}{n\alpha}$. Then the iterates of `DIANA` satisfy

$$\mathbf{E}\left[V^k\right] \le \max\left\{(1-\gamma\mu)^k, \left(1 + \frac{2\omega}{nM} - \alpha\right)^k\right\} V^0 + \frac{\left(\frac{1+\omega}{n} + M\alpha\right)\sigma^2\gamma^2}{\min\left\{\gamma\mu, \alpha - \frac{2\omega}{nM}\right\}}, \tag{51}$$

where the Lyapunov function V^k is defined by $V^k := \left\|x^k - x^*\right\|^2 + M\gamma^2\sigma_k^2$. For the particular choice $\alpha = \frac{1}{\omega+1}$, $M = \frac{4\omega(\omega+1)}{n}$, $\gamma = \frac{1}{\left(1+\frac{6\omega}{n}\right)L}$, then `DIANA` converges to a solution neighborhood and the leading iteration complexity term is

$$\max\left\{\frac{1}{\gamma\mu}, \frac{1}{\alpha - \frac{2\omega}{nM}}\right\} = \max\left\{\kappa + \kappa\frac{6\omega}{n}, 2(\omega+1)\right\}, \tag{52}$$

where $\kappa = \frac{L}{\mu}$.

[11] It is a well-known problem in distributed optimization that the communication between machines often takes more time than actual computation.

Algorithm 12 DIANA: 1 node & exact gradients (Mishchenko et al., 2019a; Horváth et al., 2019)

```
Input: learning rates $\alpha > 0$ and $\gamma > 0$, initial vectors $x^0, h^0 \in \mathbb{R}^d$
for $k = 0, 1, \dots$ do
    $\Delta^k = \nabla f(x^k) - h^k$
    Sample $\hat{\Delta}^k \sim \mathrm{Q}(\Delta^k)$
    $h^{k+1} = h^k + \alpha\hat{\Delta}^k$
    $g^k = h^k + \hat{\Delta}^k$
    $x^{k+1} = \mathrm{prox}_{\gamma R}\left(x^k - \gamma g^k\right)$
end for
```

As mentioned, once the full (deterministic) gradients are evaluated on each machine, DIANA converges linearly to the exact optimum. In particular, in such case we have $\sigma^2 = 0$. Corollary A.12 states the result in the case when $n = 1$, i.e. there is only a single node [12]. For completeness, we present the mentioned simple case of DIANA as Algorithm 12.

Corollary A.12. Assume that f_i is μ-strongly convex and L-smooth for all $i \in [n]$, $\alpha \leq \frac{1}{\omega+1}$, $\gamma \leq \frac{1}{(1+2\omega)L+ML\alpha}$ where $M > \frac{2\omega}{\alpha}$. Then the stochastic gradient $\hat{g}^k$ and the objective function f satisfy Assumption 4.1 with $A = (1+2\omega)L, B = 2\omega, \sigma_k^2 = \left\|h^k - h^*\right\|^2, \rho = \alpha, C = L\alpha, D_1 = 0, D_2 = 0$ and

$$\mathbf{E}\left[V^k\right] \leq \max\left\{(1-\gamma\mu)^k, \left(1 + \frac{2\omega}{M} - \alpha\right)^k\right\} V^0, \tag{53}$$

where the Lyapunov function V^k is defined by $V^k := \left\|x^k - x^*\right\|^2 + M\gamma^2\sigma_k^2$. For the particular choice $\alpha = \frac{1}{\omega+1}$, $M = 4\omega(\omega+1)$, $\gamma = \frac{1}{(1+6\omega)L}$ the leading term in the iteration complexity bound is

$$\max\left\{\frac{1}{\gamma\mu}, \frac{1}{\alpha - \frac{2\omega}{M}}\right\} = \max\left\{\kappa + 6\kappa\omega, 2(\omega+1)\right\}, \tag{54}$$

where $\kappa = \frac{L}{\mu}$.

A.12 Q-SGD-SR

In this section, we consider a quantized version of SGD-SR.

Algorithm 13 Q-SGD-SR

```
Input: learning rate $\gamma > 0$, starting point $x^0 \in \mathbb{R}^d$, distribution $\mathcal{D}$ over $\xi \in \mathbb{R}^n$ such that $\mathbf{E}_{\mathcal{D}}[\xi]$ is vector of ones
  for $k = 0, 1, 2, \dots$ do
      Sample $\xi \sim \mathcal{D}$
      $g^k \sim \mathrm{Q}(\nabla f_\xi(x^k))$
      $x^{k+1} = \mathrm{prox}_{\gamma R}(x^k - \gamma g^k)$
  end for
```

Lemma A.13 (Generalization of Lemma 2.4, (Gower et al., 2019))**.** If $(f, \mathcal{D}) \sim ES(\mathcal{L})$, then

$$\mathbf{E}_{\mathcal{D}}\left[\left\|g^k - \nabla f(x^*)\right\|^2\right] \leq 4\mathcal{L}(1+\omega)D_f(x^k, x^*) + 2\sigma^2(1+\omega). \tag{55}$$

where $\sigma^2 := \mathbf{E}_{\mathcal{D}}\left[\|\nabla f_\xi(x^*)\|^2\right]$.

A direct consequence of Theorem 4.1 in this setup is Corollary A.13.

Corollary A.13. Assume that $f(x)$ is μ-strongly quasi-convex and $(f, \mathcal{D}) \sim ES(\mathcal{L})$. Then Q-SGD-SR with $\gamma^k \equiv \gamma \leq \frac{1}{2(1+\omega)\mathcal{L}}$ satisfies

$$\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] \leq (1-\gamma\mu)^k \left\|x^0 - x^*\right\|^2 + \frac{2\gamma(1+\omega)\sigma^2}{\mu}. \tag{56}$$

[12] node = machine

Proof of Lemma A.13

In this proof all expectations are conditioned on x^k. First of all, from Lemma A.2 we have

$$\mathbf{E}_{\mathcal{D}}\left[\left\|\nabla f_\xi(x^k) - \nabla f(x^*)\right\|^2\right] \le 4\mathcal{L}D_f(x^k, x^*) + 2\sigma^2.$$

The remaining step is to understand how quantization of $\nabla f_\xi(x^k)$ changes the above inequality if we put $g^k \sim \mathrm{Q}(\nabla f_\xi(x^k))$ instead of $\nabla f_\xi(x^k)$. Let us denote mathematical expectation with respect randomness coming from quantization by $\mathbf{E}_Q\left[\cdot\right]$. Using tower property of mathematical expectation we get

$$\begin{aligned}
\mathbf{E}\left[\|g^k - \nabla f(x^*)\|^2\right] &= \mathbf{E}_{\mathcal{D}}\left[\mathbf{E}_Q\|g^k - \nabla f(x^*)\|^2\right] \\
&\overset{(78)}{=} \mathbf{E}\left[\|g^k - \nabla f_\xi(x^k)\|^2\right] + \mathbf{E}\left[\|\nabla f_\xi(x^k) - \nabla f(x^*)\|^2\right] \\
&\overset{(55)}{\le} \mathbf{E}\left[\|g^k - \nabla f_\xi(x^k)\|^2\right] + 4\mathcal{L}D_f(x^k, x^*) + 2\sigma^2.
\end{aligned}$$

Next, we estimate the first term in the last row of the previous inequality

$$\begin{aligned}
\mathbf{E}\left[\|g^k - \nabla f_\xi(x^k)\|^2\right] &\overset{(47)}{\le} \omega\mathbf{E}\left[\|\nabla f_\xi(x^k)\|^2\right] \\
&\overset{(76)}{\le} 2\omega\mathbf{E}\left[\|\nabla f_\xi(x^k) - \nabla f_\xi(x^*)\|^2\right] + 2\omega\mathbf{E}\left[\|\nabla f_\xi(x^*)\|^2\right] \\
&\le 4\omega\mathcal{L}D_f(x^k, x^*) + 2\omega\sigma^2.
\end{aligned}$$

Putting all together we get the result.

A.13 VR-DIANA

Corollary A.11 shows that once each machine evaluates a stochastic gradient instead of the full gradient, DIANA converges linearly only to a certain neighborhood. In contrast, VR-DIANA (Horváth et al., 2019) uses a variance reduction trick within each machine, which enables linear convergence to the exact solution. In this section, we show that our approach recovers VR-DIANA as well.

The aforementioned method is applied to solve problem (1)+(3) where each f_i is also of a finite sum structure, as in (4), with each $f_{ij}(x)$ being convex and L-smooth, and $f_i(x)$ being μ-strongly convex. Note that $\nabla f(x^*) = 0$ and, in particular, $D_f(x, x^*) = f(x) - f(x^*)$ since the problem is considered without regularization.

Lemma A.14 (Lemmas 3, 5, 6 and 7 from (Horváth et al., 2019))**.** Let $\alpha \le \frac{1}{\omega+1}$. Then for all iterates $k \ge 0$ of Algorithm 14 the following inequalities hold:

$$\mathbf{E}\left[g^k \mid x^k\right] = \nabla f(x^k), \tag{57}$$

$$\mathbf{E}\left[H^{k+1} \mid x^k\right] \le (1-\alpha)H^k + \frac{2\alpha}{m}D^k + 8\alpha Ln\left(f(x^k) - f(x^*)\right), \tag{58}$$

$$\mathbf{E}\left[D^{k+1} \mid x^k\right] \le \left(1 - \frac{1}{m}\right)D^k + 2Ln\left(f(x^k) - f(x^*)\right), \tag{59}$$

$$\mathbf{E}\left[\left\|g^k\right\|^2 \mid x^k\right] \le 2L\left(1 + \frac{4\omega+2}{n}\right)\left(f(x^k) - f(x^*)\right) + \frac{2\omega}{n^2}\frac{D^k}{m} + \frac{2(\omega+1)}{n^2}H^k, \tag{60}$$

where $H^k = \sum\limits_{i=1}^n \left\|h_i^k - \nabla f_i(x^*)\right\|^2$ and $D^k = \sum\limits_{i=1}^n\sum\limits_{j=1}^m \left\|\nabla f_{ij}(w_{ij}^k) - \nabla f_{ij}(x^*)\right\|^2$.

Corollary A.14. Let $\alpha \le \min\left\{\frac{1}{3m}, \frac{1}{\omega+1}\right\}$. Then stochastic gradient $\hat{g}^k$ (Algorithm 14) and the objective function f satisfy Assumption 4.1 with $A = \left(1 + \frac{4\omega+2}{n}\right)L, B = \frac{2(\omega+1)}{n}, \rho = \alpha, C = L\left(\frac{1}{m} + 4\alpha\right), D_1 = 0, D_2 = 0$ and

$$\sigma_k^2 = \frac{H^k}{n} + \frac{D^k}{nm} = \frac{1}{n}\sum_{i=1}^n \left\|h_i^k - \nabla f_i(x^*)\right\|^2 + \frac{1}{nm}\sum_{i=1}^n\sum_{j=1}^m \left\|\nabla f_{ij}(w_{ij}^k) - \nabla f_{ij}(x^*)\right\|^2.$$

Algorithm 14 VR-DIANA based on L-SVRG (Variant 1), SAGA (Variant 2), (Horváth et al., 2019)

```
Input: learning rates $\alpha > 0$ and $\gamma > 0$, initial vectors $x^0, h_1^0, \dots, h_n^0$, $h^0 = \frac{1}{n}\sum_{i=1}^n h_i^0$
for $k = 0, 1, \dots$ do
    Sample random $u^k = \begin{cases} 1, & \text{with probability } \frac{1}{m} \\ 0, & \text{with probability } 1 - \frac{1}{m} \end{cases}$        ▷ only for Variant 1
    Broadcast $x^k$, $u^k$ to all workers
    for $i = 1, \dots, n$ in parallel do        ▷ Worker side
        Pick random $j_i^k \sim_{\text{u.a.r.}} [m]$
        $\mu_i^k = \frac{1}{m}\sum_{j=1}^{m} \nabla f_{ij}(w_{ij}^k)$
        $g_i^k = \nabla f_{ij_i^k}(x^k) - \nabla f_{ij_i^k}(w_{ij_i^k}^k) + \mu_i^k$
        $\hat{\Delta}_i^k = Q(g_i^k - h_i^k)$
        $h_i^{k+1} = h_i^k + \alpha \hat{\Delta}_i^k$
        for $j = 1, \dots, m$ do
            $w_{ij}^{k+1} = \begin{cases} x^k, & \text{if } u^k = 1 \\ w_{ij}^k, & \text{if } u^k = 0 \end{cases}$        ▷ Variant 1 (L-SVRG): update epoch gradient if $u^k = 1$
            $w_{ij}^{k+1} = \begin{cases} x^k, & j = j_i^k \\ w_{ij}^k, & j \neq j_i^k \end{cases}$        ▷ Variant 2 (SAGA): update gradient table
        end for
    end for
    $h^{k+1} = h^k + \frac{\alpha}{n}\sum_{i=1}^{n} \hat{\Delta}_i^k$        ▷ Gather quantized updates
    $g^k = \frac{1}{n}\sum_{i=1}^{n} (\hat{\Delta}_i^k + h_i^k)$
    $x^{k+1} = x^k - \gamma g^k$
end for
```

Proof. Indeed, (7) holds due to (57). Inequality (8) follows from (60) with $A = \left(1 + \frac{4\omega+2}{n}\right)L, B = \frac{2(\omega+1)}{n}, D_1 = 0, \sigma_k^2 = \frac{H^k}{n} + \frac{D^k}{nm}$ if we take into account that $\frac{2\omega}{n^2}\frac{D^k}{m} + \frac{2(\omega+1)}{n^2}H^k \le \frac{2(\omega+1)}{n}\left(\frac{D^k}{nm} + \frac{H^k}{n}\right)$. Finally, summing inequalities (58) and (59) and using $\alpha \le \frac{1}{3m}$

$$\begin{aligned} \mathbf{E}\left[\sigma_k^2 \mid x^k\right] &= \frac{1}{n}\mathbf{E}\left[H^{k+1} \mid x^k\right] + \frac{1}{nm}\mathbf{E}\left[D^{k+1} \mid x^k\right] \\ &\overset{(58)+(59)}{\le} (1-\alpha)\frac{H^k}{n} + \left(1 + 2\alpha - \frac{1}{m}\right)\frac{D^k}{nm} + 2L\left(\frac{1}{m} + 4\alpha\right)\left(f(x^k) - f(x^*)\right) \\ &\le (1-\alpha)\sigma_k^2 + 2L\left(\frac{1}{m} + 4\alpha\right)\left(f(x^k) - f(x^*)\right) \end{aligned}$$

we get (9) with $\rho = \alpha, C = L\left(\frac{1}{m} + 4\alpha\right), D_2 = 0.$ □

Corollary A.15. Assume that f_i is μ-strongly convex and f_{ij} is convex and L-smooth for all $i \in [n], j \in [m]$, $\alpha \le \min\left\{\frac{1}{3m}, \frac{1}{\omega+1}\right\}$, $\gamma \le \frac{1}{\left(1+\frac{4\omega+2}{n}\right)L + ML\left(\frac{1}{m}+4\alpha\right)}$ where $M > \frac{2(\omega+1)}{n\alpha}$. Then the iterates of VR-DIANA satisfy

$$\mathbf{E}\left[V^k\right] \le \max\left\{(1-\gamma\mu)^k, \left(1 + \frac{2(\omega+1)}{nM} - \alpha\right)^k\right\} V^0, \tag{61}$$

where the Lyapunov function V^k is defined by $V^k := \left\|x^k - x^*\right\|^2 + M\gamma^2\sigma_k^2$. Further, if we set $\alpha = \min\left\{\frac{1}{3m}, \frac{1}{\omega+1}\right\}$, $M = \frac{4(\omega+1)}{n\alpha}$, $\gamma = \frac{1}{\left(1+\frac{20\omega+18}{n}+\frac{4\omega+4}{n\alpha m}\right)L}$, then to achieve precision $\mathbf{E}\left[\left\|x^k - x^*\right\|^2\right] \le \varepsilon V^0$ VR-DIANA needs $\mathcal{O}\left(\max\left\{\kappa + \kappa\frac{\omega+1}{n} + \kappa\frac{(\omega+1)\max\{m,\omega+1\}}{nm}, m, \omega+1\right\}\log\frac{1}{\varepsilon}\right)$ iterations, where $\kappa = \frac{L}{\mu}$.

Proof. Using Corollary A.14 we apply Theorem 4.1 and get the result. □

Remark A.3. `VR-DIANA` can be easily extended to the proximal setup in our framework.

A.14 `JacSketch`

In this section, we show that our approach covers the analysis of `JacSketch` from (Gower et al., 2018). `JacSketch` is a generalization of `SAGA` in the following manner. `SAGA` observes every iteration $\nabla f_i(x)$ for random index i and uses it to build both stochastic gradient as well as the control variates on the stochastic gradient in order to progressively decrease variance. In contrast, `JacSketch` observes every iteration the random sketch of the Jacobian, which is again used to build both stochastic gradient as well as the control variates on the stochastic gradient.

For simplicity, we do not consider proximal setup, since (Gower et al., 2018) does not either.

We first introduce the necessary notation (same as in (Gower et al., 2018)). Denote first the Jacobian the objective

$$\nabla \mathbf{F}(x) \coloneqq [\nabla f_1(x), \ldots, \nabla f_n(x)] \in \mathbb{R}^{d\times n}. \tag{62}$$

Every iteration of the method, a random sketch of Jacobian $\nabla F(x^k)\mathbf{S}$ (where $\mathbf{S} \sim \mathcal{D}$) is observed. Then, the method builds a variable $\mathbf{J}^k$, which is the current Jacobian estimate, updated using so-called sketch and project iteration (Gower and Richtárik, 2015):

$$\mathbf{J}^{k+1} = \mathbf{J}^k(\mathbf{I} - \mathbf{\Pi}_{\mathbf{S}_k}) + \nabla \mathbf{F}(x^k)\mathbf{\Pi}_{\mathbf{S}_k},$$

where $\mathbf{\Pi}_{\mathbf{S}}$ is a projection under $\mathbf{W}$ norm[13] ($\mathbf{W} \in \mathbb{R}^{n\times n}$ is some positive definite weight matrix) defined as $\mathbf{\Pi}_{\mathbf{S}} \coloneqq \mathbf{S}(\mathbf{S}^\top \mathbf{W}\mathbf{S})^\dagger \mathbf{S}^\top \mathbf{W}$[14].

Further, in order to construct unbiased stochastic gradient, an access to the random scalar $\theta_{\mathbf{S}}$ such that

$$\mathbf{E}_{\mathcal{D}}\left[\theta_{\mathbf{S}}\mathbf{\Pi}_{\mathbf{S}}\right] e = e, \tag{63}$$

where e is the vector of all ones.

Next, the simplest option for the choice of the stochastic gradient is $\nabla f_{\mathbf{S}}(x)$ – an unbiased estimate of ∇f directly constructed using $\mathbf{S}, \theta_{\mathbf{S}}$:

$$\nabla f_{\mathbf{S}}(x) = \frac{\theta_{\mathbf{S}}}{n}\nabla \mathbf{F}(x)\mathbf{\Pi}_{\mathbf{S}} e. \tag{64}$$

However, one can build a smarter estimate $\nabla f_{\mathbf{S},\mathbf{J}}(x)$ via control variates constructed from $\mathbf{J}$:

$$\nabla f_{\mathbf{S},\mathbf{J}}(x) = \frac{\theta_{\mathbf{S}}}{n}(\nabla \mathbf{F}(x) - \mathbf{J})\mathbf{\Pi}_{\mathbf{S}} e + \frac{1}{n}\mathbf{J}e. \tag{65}$$

The resulting algorithm is stated as Algorithm 15.

Algorithm 15 `JacSketch` (Gower et al., 2018)

```
Input: $(\mathcal{D}, \mathbf{W}, \theta_{\mathbf{S}})$, $x^0 \in \mathbb{R}^d$, Jacobian estimate $\mathbf{J}^0 \in \mathbb{R}^{d\times n}$, stepsize $\gamma > 0$
for $k = 0, 1, 2, \ldots$ do
    Sample a fresh copy $\mathbf{S}_k \sim \mathcal{D}$
    $\mathbf{J}^{k+1} = \mathbf{J}^k(\mathbf{I} - \mathbf{\Pi}_{\mathbf{S}_k}) + \nabla \mathbf{F}(x^k)\mathbf{\Pi}_{\mathbf{S}_k}$
    $g^k = \nabla f_{\mathbf{S}_k, \mathbf{J}^k}(x^k)$
    $x^{k+1} = x^k - \gamma g^k$
end for
```

Next we present Lemma A.15 which directly justifies the parameter choice from Table 1.

[13] Weighted Frobenius norm of matrix $\mathbf{X} \in \mathbb{R}^{n\times n}$ with a positive definite weight matrix $\mathbf{W} \in \mathbb{R}^{n\times n}$ is defined as $\|\mathbf{X}\|_{\mathbf{W}^{-1}} \coloneqq \sqrt{\operatorname{Tr}\left(\mathbf{X}\mathbf{W}^{-1}\mathbf{X}^\top\right)}$.

[14] Symbol † stands for Moore-Penrose pseudoinverse.

Lemma A.15 (Lemmas 2.5, 3.9 and 3.10 from (Gower et al., 2018))**.** Suppose that there are constants $\mathcal{L}_1, \mathcal{L}_2 > 0$ such that

$$\begin{aligned}\mathbf{E}_{\mathcal{D}}\left[\|\nabla f_{\mathbf{S}}(x) - \nabla f_{\mathbf{S}}(x^*)\|_2^2\right] &\leq 2\mathcal{L}_1(f(x) - f(x^*)), \qquad \forall x \in \mathbb{R}^d \\ \mathbf{E}_{\mathcal{D}}\left[\|(\nabla \mathbf{F}(x) - \nabla \mathbf{F}(x^*))\mathbf{\Pi}_{\mathbf{S}}\|_{\mathbf{W}^{-1}}^2\right] &\leq 2\mathcal{L}_2(f(x) - f(x^*)), \qquad \forall x \in \mathbb{R}^d,\end{aligned}$$

Then

$$\mathbf{E}_{\mathcal{D}}\left[\left\|\mathbf{J}^{k+1} - \nabla \mathbf{F}(x^*)\right\|_{\mathbf{W}^{-1}}^2\right] \leq (1 - \lambda_{\min})\left\|\mathbf{J}^k - \nabla \mathbf{F}(x^*)\right\|_{\mathbf{W}^{-1}}^2 + 2\mathcal{L}_2(f(x^k) - f(x^*)), \tag{66}$$

$$\mathbf{E}_{\mathcal{D}}\left[\left\|g^k\right\|_2^2\right] \leq 4\mathcal{L}_1(f(x^k) - f(x^*)) + 2\frac{\lambda_{\max}}{n^2}\left\|\mathbf{J}^k - \nabla \mathbf{F}(x^*)\right\|_{\mathbf{W}^{-1}}^2, \tag{67}$$

where $\lambda_{\min} = \lambda_{\min}\left(\mathbf{E}_{\mathcal{D}}\left[\mathbf{\Pi}_{\mathbf{S}}\right]\right)$ and $\lambda_{\max} = \lambda_{\max}\left(\mathbf{W}^{1/2}\left(\mathbf{E}_{\mathcal{D}}\left[\theta_{\mathbf{S}}^2 \mathbf{\Pi}_{\mathbf{S}} ee^\top \mathbf{\Pi}_{\mathbf{S}}\right] - ee^\top\right)\mathbf{W}^{1/2}\right)$. Further, $\mathbf{E}_{\mathcal{D}}\left[\nabla f_{\mathbf{S},\mathbf{J}}(x)\right] = \nabla f(x)$.

Thus, as a direct consequence of Theorem 4.1, we obtain the next corollary.

Corollary A.16. Consider the setup from Lemma A.15. Suppose that f is μ-strongly convex and choose $\gamma \leq \min\left\{\frac{1}{\mu}, \frac{1}{2\mathcal{L}_1 + M\frac{\mathcal{L}_2}{n}}\right\}$ where $M > \frac{2\lambda_{\max}}{n\lambda_{\min}}$. Then the iterates of `JacSketch` satisfy

$$\mathbf{E}\left[V^k\right] \leq \max\left\{(1 - \gamma\mu)^k, \left(1 + \frac{2\lambda_{\max}}{nM} - \lambda_{\min}\right)^k\right\} V^0. \tag{68}$$

Remark A.4. We shall note that concurrently with this work, a more general version of JacSketch with refined analysis was proposed in (Hanzely and Richtárik, 2019b), obtaining many new methods in special case (such as `LSVRG`, `SEGA` and several new ones), with best known rate in each special case. As mentioned in the main body of the paper, the rates from (Hanzely and Richtárik, 2019b) for methods that have randomness in partial derivatives and non-uniform smoothness are better to what can Theorem 4.1 achieve. On the other hand, (Hanzely and Richtárik, 2019b) only focuses on variance reduced methods, while this paper analyzes also methods with extra noise.

A.15 Interpolation between methods

Given that a set of stochastic gradients satisfy Assumption 4.1, we show that an any convex combination of the mentioned stochastic gradients satisfy Assumption 4.1 as well.

Lemma A.16. Assume that sequences of stochastic gradients $\{g_1^k\}_{k\geq 0}, \ldots, \{g_m^k\}_{k\geq 0}$ at the common iterates $\{x^k\}_{k\geq 0}$ satisfy the Assumption 4.1 with parameters $A(j), B(j), \{\sigma_k^2(j)\}_{k\geq 0}, C(j), \rho(j), D_1(j), D_2(j),\ j \in [m]$ respectively. Then for any vector $\tau = (\tau_1, \ldots, \tau_m)^\top$ such as $\sum\limits_{j=1}^m \tau_j = 1$ and $\tau_j \geq 0, j \in [m]$ stochastic gradient $g_\tau^k = \sum\limits_{j=1}^m \tau_j g_j^k$ satisfies the Assumption 4.1 with parameters:

$$\begin{aligned}&A_\tau = \sum_{j=1}^m \tau_j A(j), \quad B_\tau = 1, \quad \sigma_{\tau,k}^2 = \sum_{j=1}^m B(j)\tau_j \sigma_k^2(j), \quad \rho_\tau = \min_{j\in[m]} \rho(j), \\ &C_\tau = \sum_{j=1}^m \tau_j C(j)B(j), \quad D_{\tau,1} = \sum_{j=1}^m \tau_j D_1(j), \quad D_{\tau,2} = \sum_{j=1}^m \tau_j D_2(j)B(j).\end{aligned} \tag{69}$$

Furthermore, if stochastic gradients $g_1^k, \ldots, g_m^k$ are independent for all k, Assumption 4.1 is satisfied with parameters

$$\begin{aligned}&A_\tau = L + \sum_{j=1}^m \tau_j^2 A(j), \quad B_\tau = 1, \quad \sigma_{\tau,k}^2 = \sum_{j=1}^m B(j)\tau_j^2 \sigma_k^2(j), \quad \rho_\tau = \min_{j\in[m]} \rho(j), \\ &C_\tau = \sum_{j=1}^m \tau_j^2 C(j)B(j), \quad D_{\tau,1} = \sum_{j=1}^m \tau_j^2 D_1(j), \quad D_{\tau,2} = \sum_{j=1}^m \tau_j^2 D_2(j)B(j).\end{aligned} \tag{70}$$

What is more, instead of taking convex combination one can choose stochastic gradient at random. Lemma A.17 provides the result.

Lemma A.17. Assume that sequences of stochastic gradients $\{g_1^k\}_{k\geq 0}, \ldots, \{g_m^k\}_{k\geq 0}$ at the common iterates $\{x^k\}_{k\geq 0}$ satisfy the Assumption 4.1 with parameters $A(j), B(j), \{\sigma_k^2(j)\}_{k\geq 0}, C(j), \rho(j), D_1(j), D_2(j),\ j \in [m]$ respectively. Then for any vector $\tau = (\tau_1, \ldots, \tau_m)^\top$ such as $\sum\limits_{j=1}^m \tau_j = 1$ and $\tau_j \geq 0, j \in [m]$ stochastic gradient g_τ^k which equals g_j^k with probability τ_j satisfies the Assumption 4.1 with parameters:

$$\begin{aligned} A_\tau &= \sum_{j=1}^m \tau_j A(j), \quad B_\tau = 1, \quad \sigma_{\tau,k}^2 = \sum_{j=1}^m \tau_j B(j) \sigma_k^2(j), \quad \rho_\tau = \min_{j\in[m]} \rho(j), \\ C_\tau &= \sum_{j=1}^m \tau_j B(j) C(j), \quad D_{\tau,1} = \sum_{j=1}^m \tau_j D_1(j), \quad D_{\tau,2} = \sum_{j=1}^m B(j) \tau_j D_2(j). \end{aligned} \tag{71}$$

Furthermore, if stochastic gradients $g_1^k, \ldots, g_m^k$ are independent for all k, Assumption 4.1 is satisfied with parameters

$$\begin{aligned} A_\tau &= L + \sum_{j=1}^m \tau_j^2 A(j), \quad B_\tau = 1, \quad \sigma_{\tau,k}^2 = \sum_{j=1}^m B(j) \tau_j^2 \sigma_k^2(j), \quad \rho_\tau = \min_{j\in[m]} \rho(j), \\ C_\tau &= \sum_{j=1}^m \tau_j^2 C(j) B(j), \quad D_{\tau,1} = \sum_{j=1}^m \tau_j^2 D_1(j), \quad D_{\tau,2} = \sum_{j=1}^m \tau_j^2 D_2(j) B(j). \end{aligned} \tag{72}$$

Example A.1 (τ-L-SVRG)**.** Consider the following method — τ-L-SVRG — which interpolates between vanilla SGD and L-SVRG. When $\tau = 0$ the Algorithm 16 becomes L-SVRG and when $\tau = 1$ it is just SGD with uniform sampling. Notice that Lemmas A.11 and A.2 still hold as they does not depend on the update rule for x^{k+1}.

Algorithm 16 τ-L-SVRG

```
Input: learning rate $\gamma > 0$, probability $p \in (0,1]$, starting point $x^0 \in \mathbb{R}^d$, convex combination parameter $\tau \in [0,1]$
  $w^0 = x^0$
  for $k = 0, 1, 2, \ldots$ do
      Sample $i \in \{1, \ldots, n\}$ uniformly at random
      $g_{L-SVRG}^k = \nabla f_i(x^k) - \nabla f_i(w^k) + \nabla f(w^k)$
      Sample $j \in \{1, \ldots, n\}$ uniformly at random
      $g_{SGD}^k = \nabla f_j(x^k)$
      $g^k = \tau g_{SGD}^k + (1-\tau) g_{L-SVRG}^k$
      $x^{k+1} = x^k - \gamma g^k$
      $w^{k+1} = \begin{cases} x^k & \text{with probability } p \\ w^k & \text{with probability } 1-p \end{cases}$
  end for
```

Thus, sequences $\{g_{SGD}^k\}_{k\geq 0}$ and $\{g_{L-SVRG}^k\}_{k\geq 0}$ satisfy the conditions of Lemma A.16 and, as a consequence, stochastic gradient g^k from τ-L-SVRG meets the Assumption 4.1 with the following parameters:

$$A_\tau = L + 2\tau^2 \mathcal{L} + 2(1-\tau)^2 L, \quad B_\tau = 1, \quad \sigma_{\tau,k}^2 = 2\frac{(1-\tau)^2}{n} \sum_{i=1}^n \left\|\nabla f_i(w^k) - \nabla f_i(x^*)\right\|^2,$$

$$\rho_\tau = p, \quad C_\tau = 2(1-\tau)^2 L p, \quad D_{\tau,1} = 2\tau^2 \sigma^2, \quad D_{\tau,2} = 0.$$

Remark A.5. Similar interpolation with the analogous analysis can be considered between SGD and SAGA, or SGD and SVRG.

Proof of Lemma A.16

Indeed, (7) holds due to linearity of mathematical expectation. Next, summing inequalities (8) for $g_1^k, \ldots, g_m^k$ and using convexity of $\|\cdot\|^2$ we get

$$\begin{aligned}\mathbf{E}\left[\left\|g_\tau^k - \nabla f(x^*)\right\|^2 \mid x^k\right] &\leq \sum_{j=1}^m \tau_j \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right]\\ &\overset{(8)}{\leq} 2\sum_{j=1}^m \tau_j A(j) D_f(x^k, x^*) + \sum_{j=1}^m B(j)\tau_j \sigma_k^2(j) + \sum_{j=1}^m \tau_j D_1(j),\end{aligned}$$

which implies (8) for g_τ^k with $A_\tau = \sum\limits_{j=1}^m \tau_j A(j), B_\tau = 1, \sigma_{\tau,k}^2 = \sum\limits_{j=1}^m \tau_j B(j)\sigma_k^2(j), D_{\tau,1} = \sum\limits_{j=1}^m \tau_j D_1(j)$. Finally, summing (9) for $g_1^k, \ldots, g_m^k$ gives us

$$\mathbf{E}\left[\sigma_{\tau,k+1}^2 \mid \sigma_{\tau,k}^2\right] \overset{(9)}{\leq} \left(1 - \min_{j\in[m]} \rho(j)\right)\sigma_{\tau,k}^2 + 2\sum_{j=1}^m \tau_j B(j) C(j) D_f(x^k, x^*) + \sum_{j=1}^m \tau_j B(j) D_2(j),$$

which is exactly (9) for $\sigma_{\tau,k}^2$ with $\rho = \min\limits_{j\in[m]} \rho(j), C_\tau = \sum\limits_{j=1}^m \tau_j C(j), D_{\tau,2} = \sum\limits_{j=1}^m \tau_j D_2(j)$.

Next, for independent gradients we have

$$\begin{aligned}\mathbf{E}\left[\left\|g_\tau^k - \nabla f(x^*)\right\|^2 \mid x^k\right] &= \sum_{j=1}^m \tau_j^2 \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right] + 2\sum_{i<j} \tau_i\tau_j \mathbf{E}\left\langle g_j^k - \nabla f(x^*), g_i^k - \nabla f(x^*)\right\rangle\\ &= \sum_{j=1}^m \tau_j^2 \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right] + 2\sum_{i<j} \tau_i\tau_j \left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2\\ &\leq \sum_{j=1}^m \tau_j^2 \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right] + \left(\sum_{j=1}^m \tau_j\right)^2 \left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2\\ &= \sum_{j=1}^m \tau_j^2 \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right] + \left\|\nabla f(x^k) - \nabla f(x^*)\right\|^2\\ &\leq \sum_{j=1}^m \tau_j^2 \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right] + 2LD_f(x^k, x^*). \qquad (73)\end{aligned}$$

and further the bounds follow.

Proof of Lemma A.17

Indeed, (7) holds due to linearity and tower property of mathematical expectation. Next, using tower property of mathematical expectation and inequalities (8) for $g_1^k, \ldots, g_m^k$ we get

$$\begin{aligned}\mathbf{E}\left[\left\|g_\tau^k - \nabla f(x^*)\right\|^2 \mid x^k\right] &= \mathbf{E}\left[\mathbf{E}_\tau\left[\left\|g_\tau^k - \nabla f(x^*)\right\|^2\right] \mid x^k\right] = \sum_{j=1}^m \tau_j \mathbf{E}\left[\left\|g_j^k - \nabla f(x^*)\right\|^2 \mid x^k\right]\\ &\overset{(8)}{\leq} 2\sum_{j=1}^m \tau_j A(j) D_f(x^k, x^*) + \sum_{j=1}^m B(j)\tau_j \sigma_k^2(j) + \sum_{j=1}^m \tau_j D_1(j),\end{aligned}$$

which implies (8) for g_τ^k with $A_\tau = \sum\limits_{j=1}^m \tau_j A(j), B_\tau = 1, \sigma_{\tau,k}^2 = \sum\limits_{j=1}^m \tau_j B(j)\sigma_k^2(j), D_{\tau,1} = \sum\limits_{j=1}^m \tau_j D_1(j)$. Finally, summing (9) for $g_1^k, \ldots, g_m^k$ gives us

$$\mathbf{E}\left[\sigma_{\tau,k+1}^2 \mid \sigma_{\tau,k}^2\right] \overset{(9)}{\leq} \left(1 - \min_{j\in[m]} \rho(j)\right)\sigma_{\tau,k}^2 + 2\sum_{j=1}^m \tau_j B(j) C(j) D_f(x^k, x^*) + \sum_{j=1}^m \tau_j B(j) D_2(j),$$

which is exactly (9) for $\sigma^2_{\tau,k}$ with $\rho = \min\limits_{j\in[m]} \rho(j), C_\tau = \sum\limits_{j=1}^m \tau_j B(j) C(j), D_{\tau,2} = \sum\limits_{j=1}^m \tau_j B(j) D_2(j).$ To show (72), it suffices to combine above bounds with the trick (73).

Remark A.6. Recently, (Tran-Dinh et al., 2019) demonstrated in that the convex combination of `SGD` and `SARAH` (Nguyen et al., 2017) performs very well on non-convex problems.

B Extra Experiments

B.1 SGD-MB: remaining experiments and exact problem setup.

As already described in Section 6, we demonstrate that SGD-MB have indistinguishable iteration complexity to independent SGD. The considered problem is logistic regression with Tikhonov regularization of order λ:

$$\frac{1}{n}\sum_{i=1}^{n}\log\left(1+\exp\left(a_i^\top x \cdot b_i\right)\right)+\frac{\lambda}{2}\left\|x\right\|^2, \tag{74}$$

where $a_i \in \mathbb{R}^n$, $b_i \in \{-1,1\}$ is i-th data-label pair is a vector of labels and $\lambda \geq 0$ is the regularization parameter. The data and labels were obtained from LibSVM datasets a1a, a9a, w1a, w8a, gisette, madelon, phishing and mushrooms. Further, the data were rescaled by a random variable cu_i^2 where u_i is random integer from $1, 2, \dots, 1000$ and c is such that the mean norm of a_i is 1.

Note that we have now an infinite array of possibilities on how to write (74) as (3). For simplicity, distribute $l2$ term evenly among the finite sum.

The full results can be found in Figure 2.

Note that plots which are not included in the main body (due to space limitations) only support claims from Section 6.

B.2 Experiments on SGD-star

In this section, we study SGD-star and numerically verify claims from Section A.4. In particular, Corollary A.4 shows that SGD-star enjoys linear convergence rate which is constant times better to the rate of SAGA (given that problem condition number is high enough). We compare 3 methods – SGD-star, SGD and SAGA. We consider simple and well-understood least squares problem $\min_x \frac{1}{2}\|\mathbf{A}x - b\|^2$ where elements of $\mathbf{A}, b$ were generated (independently) from standard normal distribution. Further, rows of $\mathbf{A}$ were normalized so that $\|\mathbf{A}_{i:}\| = 1$. Thus, denoting $f_i(x) = \frac{1}{2}(\mathbf{A}_{i:}^\top x - b_i)^2$, f_i is 1-smooth. For simplicity, we consider SGD-star with uniform serial sampling, i.e. $\mathcal{L} = 1$.

Next, for both SGD-star and SGD we use stepsize $\gamma = \frac{1}{2}$ (theory supported stepsize for SGD-star), while for SAGA we set $\gamma = \frac{1}{5}$ (almost theory supported stepsize). Figure 3 shows the results.

Note that, as theory predicts, SGD-star is always faster to SAGA, although only constant times. Further, in the cases where $d \geq n$, performance of SGD seems identical to the performance of SGD-shift. This is due to a simple reason: if $d \geq n$, we must have $\nabla f_i(x^*) = 0$ for all i, and thus SGD and SGD-shift are in fact identical algorithms.

B.3 Experiments on N-SEGA

In this experiment we study the effect of noise on N-SEGA. We consider unit ball constrained least squares problem: $\min_{\|x\|\leq 1} f(x)$ where $f(x) = \|\mathbf{A}x - b\|^2$. and we suppose that there is an oracle providing us with noised partial derivative $g_i(x, \zeta) = \nabla_i f(x) + \zeta$, where $\zeta \sim N(0, \sigma^2)$. For each problem instance (i.e. pair $\mathbf{A}, b$), we compare performance of N-SEGA under various noise magnitudes σ^2.

The specific problem instances are presented in Table 3. Figure 4 shows the results.

Type	$\mathbf{A}$	b
1	$\mathbf{A}_{ij} \sim N(0,1)$ (independently)	vector of ones
2	Same as 1, but scaled so that $\lambda_{\max}(A^\top A) = 1$	vector of ones
3	$\mathbf{A}_{ij} = \varrho_{ij}\varpi_j \ \forall i, j : \varrho_{ij}, \varpi_j \sim N(0,1)$ (independently)	vector of ones
4	Same as 3, but scaled so that $\lambda_{\max}(A^\top A) = 1$	vector of ones

Table 3: Four types of least squares.

We shall mention that this experiment serves to support and give a better intuition about the results from Section A.8 and is by no means practical. The results show, as predicted by theory, linear convergence to a

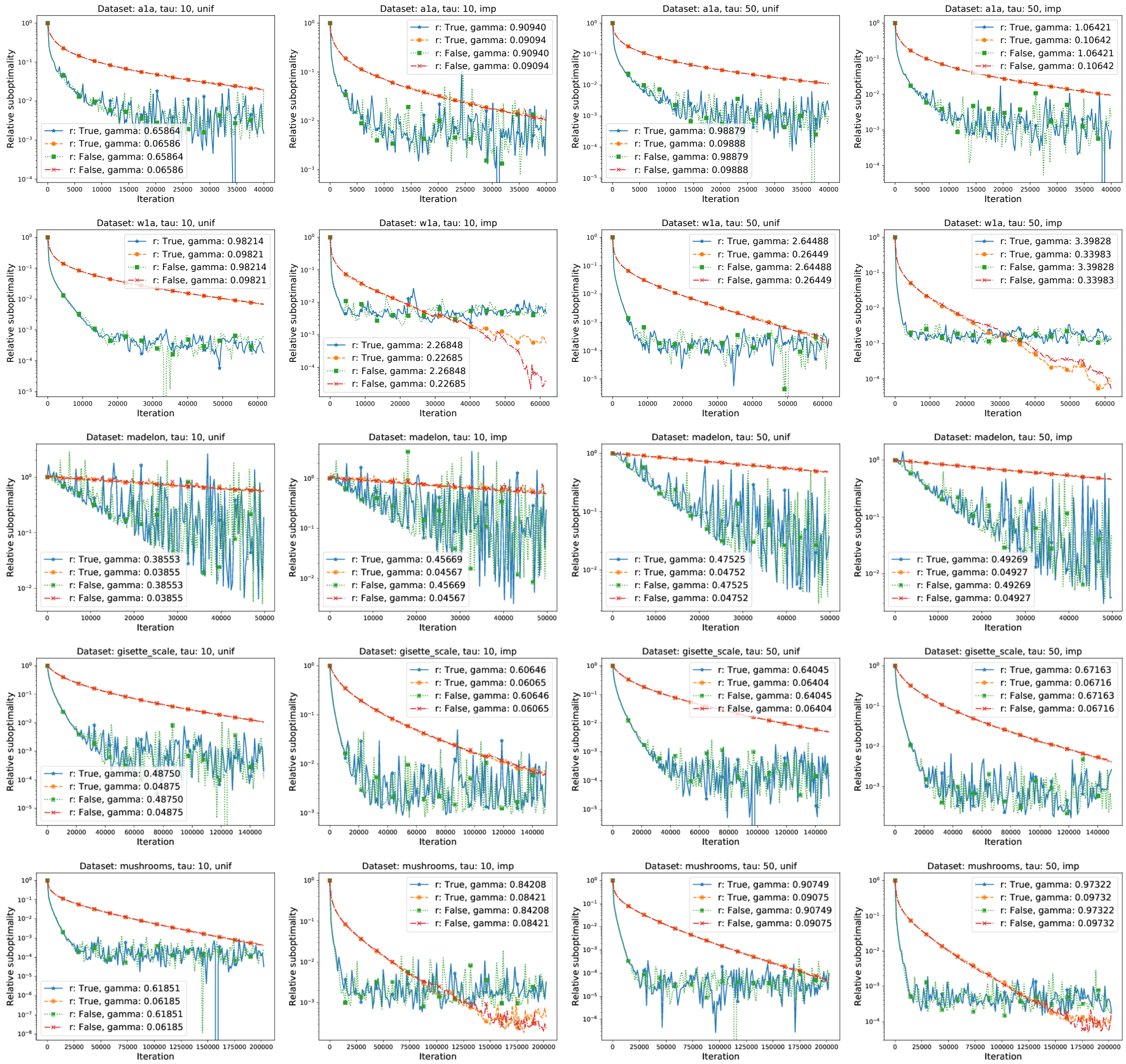

Figure 2: SGD-MB and independent SGD applied on LIBSVM (Chang and Lin, 2011) datasets with regularization parameter $\lambda = 10^{-5}$. Axis y stands for relative suboptimality, i.e. $\frac{f(x^k)-f(x^*)}{f(x^k)-f(x^0)}$. Title label "unif" corresponds to probabilities chosen by (i) while label "imp" corresponds to probabilities chosen by (ii). Lastly, legend label "r" corresponds to "replacement" with value "True" for SGD-MB and value "False" for independent SGD.

specific neighborhood of the objective. The effect of the noise varies, however, as a general rule, the larger strong convexity μ is (i.e. problems 1,3 where scaling was not applied), the smaller the effect of noise is.

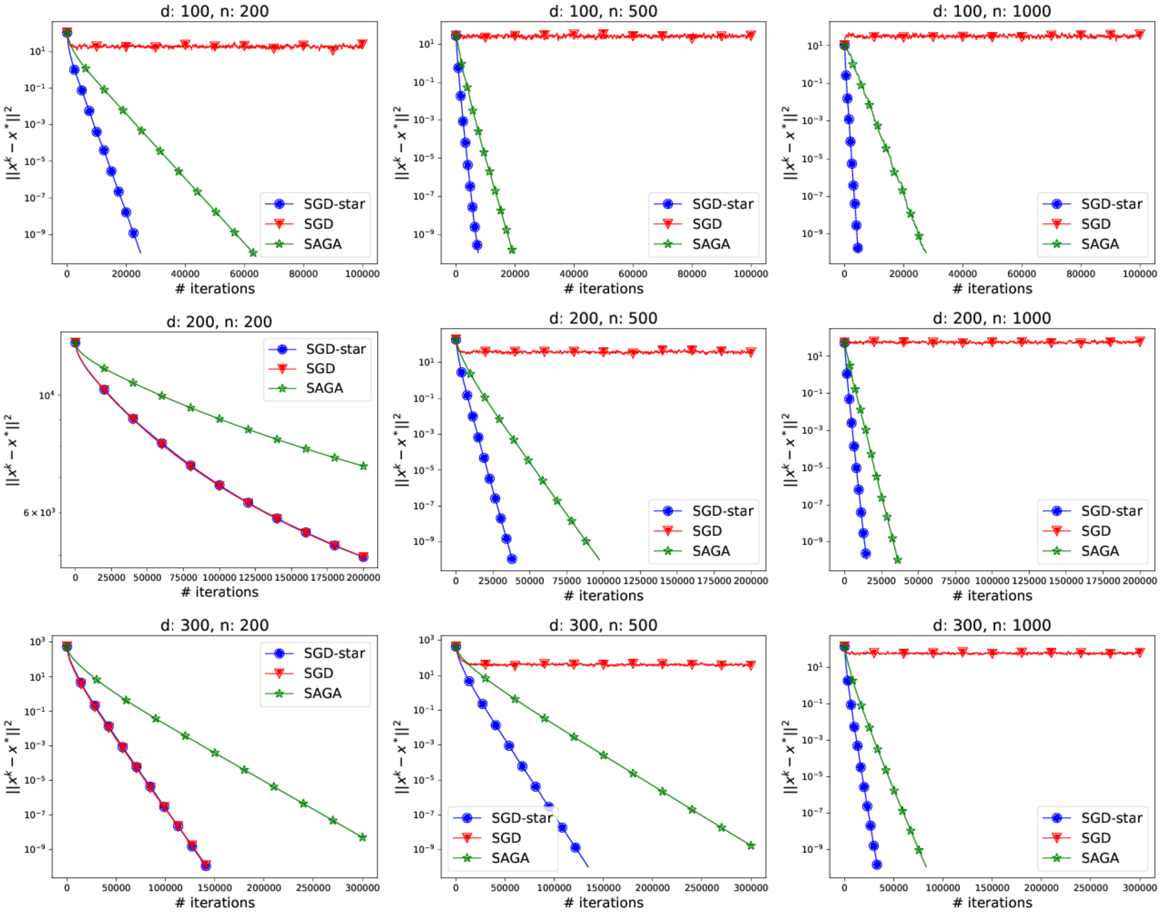

Figure 3: Comparison of SGD-star, SGD and SAGA on least squares problem.

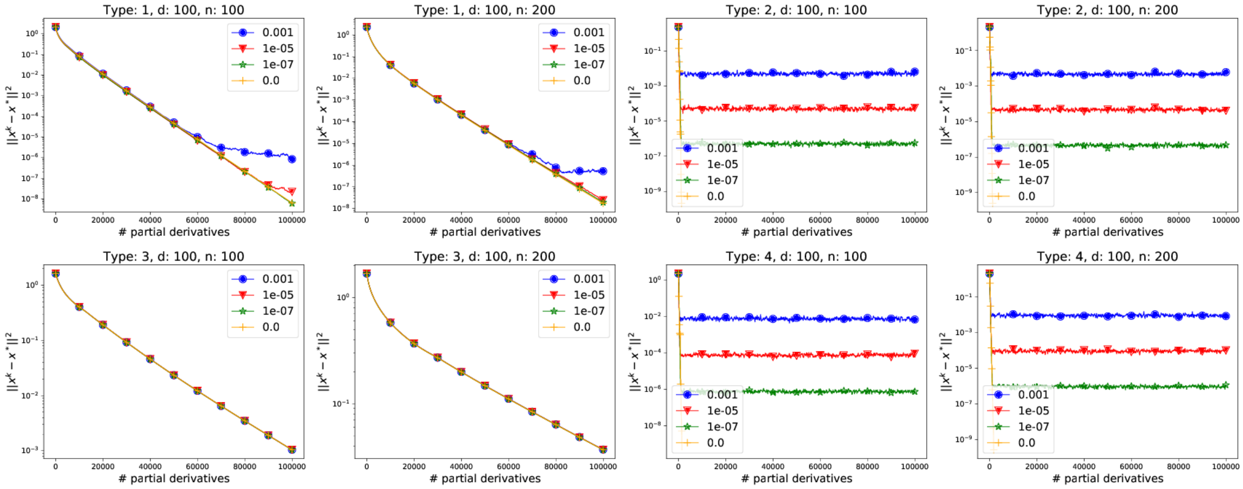

Figure 4: N-SEGA applied on constrained least squares problem with noised partial derivative oracle. Legend labels stand for the magnitude σ^2 of the oracle noise.

C Proofs for Section 4

C.1 Basic Facts and Inequalities

For all $a, b \in \mathbb{R}^d$ and $\xi > 0$ the following inequalities holds:

$$\langle a, b\rangle \le \frac{\|a\|^2}{2\xi} + \frac{\xi\|b\|^2}{2}, \tag{75}$$

$$\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2, \tag{76}$$

and

$$\frac{1}{2}\|a\|^2 - \|b\|^2 \le \|a+b\|^2. \tag{77}$$

For a random vector $\xi \in \mathbb{R}^d$ and any $x \in \mathbb{R}^d$ the variance can be decomposed as

$$\mathbf{E}\left[\|\xi - \mathbf{E}\xi\|^2\right] = \mathbf{E}\left[\|\xi - x\|^2\right] - \|\mathbf{E}\xi - x\|^2 \ . \tag{78}$$

C.2 Proof of Lemma 4.1

We start with estimating the first term of the Lyapunov function. Let $r^k = x^k - x^*$. Then

$$\begin{aligned}\left\|r^{k+1}\right\|^2 &= \left\|\mathrm{prox}_{\gamma R}(x^k - \gamma g^k) - \mathrm{prox}_{\gamma R}(x^* - \gamma\nabla f(x^*))\right\|^2\\ &\le \left\|x^k - x^* - \gamma(g^k - \nabla f(x^*))\right\|^2\\ &= \left\|r^k\right\|^2 - 2\gamma\langle r^k, g^k - \nabla f(x^*)\rangle + \gamma^2\left\|g^k - \nabla f(x^*)\right\|^2.\end{aligned}$$

Taking expectation conditioned on x^k we get

$$\begin{aligned}\mathbf{E}\left[\left\|r^{k+1}\right\|^2 \mid x^k\right] &= \left\|r^k\right\|^2 - 2\gamma\langle r^k, \nabla f(x^k) - \nabla f(x^*)\rangle + \gamma^2\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right]\\ &\overset{(12)}{\le} (1-\gamma\mu)\left\|r^k\right\|^2 - 2\gamma D_f(x^k, x^*) + \gamma^2\mathbf{E}\left[\left\|g^k - \nabla f(x^*)\right\|^2 \mid x^k\right]\\ &\overset{(7)+(8)}{\le} (1-\gamma\mu)\left\|r^k\right\|^2 + 2\gamma\left(A\gamma - 1\right)D_f(x^k, x^*) + B\gamma^2\sigma_k^2 + \gamma^2 D_1.\end{aligned}$$

Using this we estimate the full expectation of V^{k+1} in the following way:

$$\begin{aligned}&\mathbf{E}\left\|x^{k+1} - x^*\right\|^2 + M\gamma^2\mathbf{E}\sigma_{k+1}^2\\ \overset{(9)}{\le}\ & (1-\gamma\mu)\mathbf{E}\left\|x^k - x^*\right\|^2 + 2\gamma\left(A\gamma - 1\right)\mathbf{E}\left[D_f(x^k, x^*)\right] + B\gamma^2\mathbf{E}\sigma_k^2\\ &\quad + (1-\rho)M\gamma^2\mathbf{E}\sigma_k^2 + 2CM\gamma^2\mathbf{E}\left[D_f(x^k, x^*)\right] + (D_1 + MD_2)\gamma^2\\ =\ & (1-\gamma\mu)\mathbf{E}\left\|x^k - x^*\right\|^2 + \left(1 + \frac{B}{M} - \rho\right)M\gamma^2\mathbf{E}\sigma_k^2\\ &\quad + 2\gamma\left(\gamma(A + CM) - 1\right)\mathbf{E}\left[D_f(x^k, x^*)\right] + (D_1 + MD_2)\gamma^2.\end{aligned}$$

It remains to rearrange the terms.

C.3 Proof of Theorem 4.1

Note first that due to (13) we have $2\gamma\left(1 - \gamma(A + CM)\right)\mathbf{E}D_f(x^k, x^*) > 0$, thus we can omit the term.

Unrolling the recurrence from Lemma 4.1 and using the Lyapunov function notation gives us

$$
\begin{aligned}
\mathbf{E}V^k &\leq \max\left\{(1-\gamma\mu)^k, \left(1+\frac{B}{M}-\rho\right)^k\right\}V^0 \\
&\quad +(D_1+MD_2)\gamma^2\sum_{l=0}^{k-1}\max\left\{(1-\gamma\mu)^l, \left(1+\frac{B}{M}-\rho\right)^l\right\} \\
&\leq \max\left\{(1-\gamma\mu)^k, \left(1+\frac{B}{M}-\rho\right)^k\right\}V^0 \\
&\quad +(D_1+MD_2)\gamma^2\sum_{l=0}^{\infty}\max\left\{(1-\gamma\mu)^l, \left(1+\frac{B}{M}-\rho\right)^l\right\} \\
&\leq \max\left\{(1-\gamma\mu)^k, \left(1+\frac{B}{M}-\rho\right)^k\right\}V^0 + \frac{(D_1+MD_2)\gamma^2}{\min\left\{\gamma\mu, \rho-\frac{B}{M}\right\}}.
\end{aligned}
$$