A line search based proximal stochastic gradient algorithm with dynamical variance reduction

Giorgia Franchini, Federica Porta
Department of Physics, Informatics and Mathematics
University of Modena and Reggio Emilia
Modena, Italy
{giorgia.franchini, federica.porta}@unimore.it
Valeria Ruggiero
Department of Mathematics and Computer Science
University of Ferrara
Ferrara, Italy
valeria.ruggiero@unife,it
Ilaria Trombini
Department of Mathematical, Physical and Computer Sciences
University of Parma
Parma, Italy
ilaria.trombini@unife.it

August 25, 2022

Abstract

Many optimization problems arising from machine learning applications can be cast as the minimization of the sum of two functions: the first one typically represents the expected risk, and in practice it is replaced by the empirical risk, and the other one imposes a priori information on the solution. Since in general the first term is differentiable and the second one is convex, proximal gradient methods are very well suited to face such optimization problems. However, when dealing with large-scale machine learning issues, the computation of the full gradient of the differentiable term can be prohibitively expensive by making these algorithms unsuitable. For this reason, proximal stochastic gradient methods have been extensively studied in the optimization area in the last decades. In this paper we develop a proximal stochastic gradient algorithm which is based on two main ingredients. We indeed combine a proper technique to dynamically reduce the variance of the stochastic gradients along the iterative process with a descent condition in expectation for the objective function, aimed to fix the value for the steplength parameter at each iteration. For general objective functionals, the a.s. convergence of the limit points of the sequence generated by the proposed scheme to stationary points can be proved. For convex objective functionals, both the a.s. convergence of the whole sequence of the iterates to a minimum point and an $\mathcal{O}(1/k)$ convergence rate for the objective function values have been shown. The practical implementation of the proposed method does not need neither the computation of the exact gradient of the empirical risk during the iterations nor the tuning of an optimal value for the steplength. An extensive numerical experimentation highlights that the proposed approach appears robust with respect to the setting of the hyperparameters and competitive compared to state-of-the-art methods.

Keywords: First order stochastic methods, Stochastic proximal methods, Machine Learning, Green Artificial Intelligence

1 Introduction

In this paper we consider the following stochastic composite and possibly nonsmooth optimization problem

$$\min_{x\in\mathbb{R}^d} P(x) := \min_{x\in\mathbb{R}^d} F(x) + R(x) = \min_{x\in\mathbb{R}^d} \mathbb{E}[F(x,\xi)] + R(x), \tag{1}$$

where $F(x) = \mathbb{E}[F(x,\xi)]$ is the expectation of a stochastic function $F(x,\xi)$ depending on a multi-value random variable ξ in a given probability space $(\Theta, \mathcal{F}, \mathbb{P})$ and $R : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ is a proper, closed and convex function. $F(x)$ is a differentiable function, with Lipschitz-continuous gradient; if ξ is a uniformly random vector defined on a finite support set $\Theta = \{\xi_1, ..., \xi_N\}$, then problem (1) reduces to the following finite-sum minimization problem:

$$\min_{x\in\mathbb{R}^d} P(x) := \min_{x\in\mathbb{R}^d} F(x) + R(x) = \min_{x\in\mathbb{R}^d} \frac{1}{N}\sum_{i=1}^{N} f_i(x) + R(x), \tag{2}$$

with $f_i(x) \equiv F(x, \xi_i)$, $i = 1, ..., N$. In the machine learning framework, this problem is known as a regularized empirical risk minimization and N is the cardinality of a given training set.

The problem (1) covers a broad range of applications in signal and imaging processing as well as in the machine learning context. Forward-backward algorithms are effective tools to face minimization problems of this kind, since they consist of a forward step, which exploits the differentiability of F, and a backward step, which takes advantage of the convexity of R. Moreover, the class of forward-backward methods has been intensively investigated, developing several versions including inertial steps, inexact computation of the proximal operator, variable metric strategies and adaptive steplength selection rules (see for example [11, 10, 1, 14], the survey [6] and references therein). In the context of big data, when $F(x)$ has the role of the empirical risk, the cost required by one evaluation of the full gradient $\nabla F(x)$ is prohibitive and it is necessary to resort to its stochastic approximation, giving rise to the class of proximal stochastic gradient methods, tailored to address the problem (2). In particular, we recall the Proximal Stochastic Gradient (Prox-SG) method, where, at each iteration $k = 0, 1, 2, ...$, an index i_k is randomly drawn from $\{1, ..., N\}$ and the update iteration is

$$x^{(k+1)} = \text{prox}_{\alpha_k R}(x^{(k)} - \alpha_k \nabla f_{i_k}(x^{(k)})), \tag{3}$$

where $\alpha_k > 0$ is the steplength whose definition affects the convergence properties of Prox-SG, as discussed below. Let denote $x^* = \text{argmin}_{x\in\mathbb{R}^d} P(x)$. Under the assumption of strongly convexity for P and the selection of a suitable diminishing steplength $\alpha_k = \mathcal{O}(1/k)$, the expected value of the optimality gap $\mathbb{E}(P(x^{(k)}) - P(x^*))$ generated by Prox-SG method converges at a sublinear rate $\mathcal{O}(1/k)$ [12, 16]. Other very popular approaches to face (2) are the variance reduced methods, also known as hybrid methods, as Proximal SVRG (Prox-SVRG) [27], Proximal SARAH (Prox-SARAH) [22] or Proximal Spider Boost (Prox-Spider-boost) [26]. These schemes compute the full gradient of F at a given point along the iterations and use it for m successive inner iterations, where m is of the order of N. In the case of Prox-SVRG, for strongly convex functions F and R and suitable bounds on the fixed steplength, linear convergence results for the optimality gap and the sequence of the iterates can be obtained [27]. Under convexity assumption for both F and R, a sublinear ergodic convergence rate is proved in [23]. Further variance reduced schemes based on Spider or SARAH estimators and a fixed or adaptive steplength are Prox-Spider-boost or Prox-SARAH, well-suited also when the smooth term F is non convex. In [22, 26] an in-depth analysis of the features of the methods is reported, including the convergence properties and the setting of the parameters assuring an efficient performance. In particular, optimal values are estimated for the steplength, the number of iterations of the inner loop, and the mini batch size of the currently selected sample.

Nevertheless, these variance reduced methods have some drawbacks. More in detail, they exhibit a double loop where in the outer loop the full gradient (or a gradient based on a large mini batch sample) is required and the number of iterations of the inner loop and the mini batch size for the estimation of the current stochastic gradient are connected and dependent on both the Lipschitz constant of ∇F and the cardinality N of the training set. In the case of stochastic gradient methods, the control of the stochastic gradient variance without the cyclic computation of a full gradient is also obtained by dynamic sampling strategies, as suggested in [7, 9, 4, 13]. In this paper we study this technique in the framework of the

proximal gradient methods.
Contributions. We propose a proximal stochastic gradient method based on two main ingredients: a condition which assures the decrease of the objective function in expectation and a proper control of the variance of the current stochastic gradient. From the theoretical point of view, these ingredients enable to state that any limit point is a stationary point for (1), almost surely. Moreover, if F is a convex function, the almost surely convergence of the whole sequence generated by the scheme as well as the one of the optimality gap can be proved. From the practical point of view, in the case of the minimization of the regularized empirical risk, we show which strategies can be adopted to realize both the conditions mentioned above and required for the convergence results. In particular, a technique to dynamically increase the mini batch size of the sub-sample used to compute the current stochastic gradient drives the required progressive decrease of the variance of this direction; furthermore, the decrease of the objective function in expectation can be achieved by means of a line search procedure acting on the steplength parameters properly bounded. Differently from the Prox-SG, the Prox-SVRG the Prox-SARAH and the Prox-Spider-boost methods which need an optimal value for the steplength parameter to exhibit good convergence properties, the suggested proximal stochastic gradient algorithm has numerically proved to be free from the time consuming hand-tuning of the steplength, because of the presence of the line search. Techniques that avoid the setting of hyperparameters clearly addresses the challenges of Green Artificial Intelligence. Finally, we highlight that the scheme developed in this paper never requires the computation of the full gradient of F, in contrast to the variance reduced methods, as Prox-SVRG, Prox-SARAH and Prox-Spider-boost, which need to perform a full gradient evaluation over the data set per epoch, where with epoch we denote an entire view of the data set. However, such a technique is not employed in the practical deep learning applications and, more in general, in the big data framework, for its computational cost, the hardware memory constraints and the resulting highly inefficiency when training a deep neural network. Moreover, a proximal stochastic gradient algorithm which avoids the computation of the full gradient becomes necessary in the online learning scenarios, where data are not entirely available at the beginning of the training process as well as the full gradient [8, Sec. 2.2.2].
After the introduction, this paper is essentially structured in three sections. In Section 2, we develop a class of proximal stochastic gradient methods and we prove its theoretical convergence features. In Section 3, we consider the special case of minimizing a regularized empirical risk and we detail the techniques aimed to implement the scheme proposed in the previous section. Finally, in Section 4 we evaluate the robustness of the proposed method in training a binary classifier, using convex loss functions and ℓ_1-regularization term. A comparison with state-of-the-art methods is reported for several well known data sets. Finally conclusions and future perspectives are given.

2 A class of proximal stochastic gradient methods

In this section we develop a class of proximal stochastic gradient methods to face the problem (1).

Assumption 1. *Let assume that the functions involved in problem* (1) *have the following properties.*

(i) $R : \mathbb{R}^d \to \mathbb{R} \cup \{\infty\}$ *is a proper, convex and lower semicontinuous function, with a non empty and closed domain.*

(ii) $F : \mathbb{R}^d \longrightarrow \mathbb{R} \cup \{\infty\}$ *is a continuously differentiable function on an open subset* Y *of* $\mathbb{R}^d$ *containing* $dom(R)$*. Moreover* ∇F *is L-Lipschitz continuous.*

(iii) P *is bounded below on* $dom(R) \cap dom(F)$ *and* $X^* = \operatorname{argmin}_x P(x) \neq \emptyset$*.*

Before presenting the general scheme of the proposed algorithm and its properties, we briefly recall some useful definitions.

Definition 1. *Let* $R : \mathbb{R}^d \longrightarrow \mathbb{R} \cup \{+\infty\}$ *be a convex function. The proximal operator associated to* R *is defined as*

$$\operatorname{prox}_R(x) = \operatorname*{argmin}_{z \in \mathbb{R}^d} \; R(z) + \frac{1}{2}\|z - x\|^2.$$

As done in [5] to develop a deterministic proximal gradient algorithm, let us introduce the function $h_\alpha : \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R} \cup \{+\infty\}$, with $\alpha > 0$, defined as

$$h_\alpha(z;x) = \nabla F(x)^T(z-x) + \frac{1}{2\alpha}\|z-x\|^2 - R(x) + R(z) \tag{4}$$

and the operator $p_\alpha : Y \longrightarrow \mathbb{R}^d$ defined as

$$p_\alpha(x) = \mathrm{prox}_{\alpha R}(x - \alpha\nabla F(x)) = \underset{z\in\mathbb{R}^d}{\mathrm{argmin}}\, h_\alpha(z;x). \tag{5}$$

2.1 The algorithm

We suppose that for the problem (1) we can only compute

$$g(x) = \nabla F(x) + e_g(x) \tag{6}$$

with $e_g(x)$ being a random vector. The basic iteration of the proximal stochastic gradient methods we are interested in can be stated as follows:

$$\begin{aligned} g(x^{(k)}) &= \nabla F(x^{(k)}) + e_g(x^{(k)}) \\ x^{(k+1)} &= \mathrm{prox}_{\alpha_k R}(x^{(k)} - \alpha_k g(x^{(k)})) \end{aligned} \tag{7}$$

where $x^{(0)} \in \mathbb{R}^d$ and α_k is a positive steplength parameter. Hereafter, we denote $e_g^{(k)} = e_g(x^{(k)})$ and $g^{(k)} = g(x^{(k)})$. We make the following standard assumption.

Assumption 2. *Let $\mathcal{F}_k$ be the σ-algebra generated by $x^{(0)}, x^{(1)}, \dots, x^{(k)}$. The gradient estimator $g^{(k)}$ is unbiased, i.e.,* $\mathbb{E}(e_g^{(k)}|\mathcal{F}_k) = 0$.

Now, we observe that the iterate $x^{(k+1)}$ is related to the minimum point of the function $h_{\alpha_k}(x;x^{(k)})$. In particular, while $p_{\alpha_k}(x^{(k)}) = \mathrm{prox}_{\alpha_k R}(x^{(k)} - \alpha_k\nabla F(x^{(k)}))$ is the exact minimum point of $h_{\alpha_k}(x;x^{(k)})$, in view of the presence of $e_g^{(k)}$ in (7), $x^{(k+1)}$ can be written as

$$\begin{aligned} x^{(k+1)} &= \mathrm{prox}_{\alpha_k R}(x^{(k)} - \alpha_k g^{(k)}) \\ &= \underset{z\in\mathbb{R}^d}{\mathrm{argmin}}\; {g^{(k)}}^T(z - x^{(k)}) + \frac{1}{2\alpha_k}\|z - x^{(k)}\|^2 - R(x^{(k)}) + R(z) \\ &= \underset{z\in\mathbb{R}^d}{\mathrm{argmin}}\; (\nabla F(x^{(k)}) + e_g^{(k)})^T(z - x^{(k)}) + \frac{1}{2\alpha_k}\|z - x^{(k)}\|^2 \\ &\qquad\qquad - R(x^{(k)}) + R(z) \\ &= \underset{z\in\mathbb{R}^d}{\mathrm{argmin}}\, h_{\alpha_k}(z;x^{(k)}) + {e_g^{(k)}}^T(z - x^{(k)}). \end{aligned}$$

As a consequence, $\forall z \in \mathbb{R}^d$,

$$h_{\alpha_k}(x^{(k+1)};x^{(k)}) + {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)}) \le h_{\alpha_k}(z;x^{(k)}) + {e_g^{(k)}}^T(z - x^{(k)}).$$

By setting $z = x^{(k)}$, it holds that

$$h_{\alpha_k}(x^{(k+1)};x^{(k)}) + {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)}) \le 0. \tag{8}$$

The above inequality can be exploited to control the decrease of the objective function in expectation. For this reason we introduce the following condition

$$\begin{aligned} \mathbb{E}(P(x^{(k+1)})|\mathcal{F}_k) \le\; & \mathbb{E}(P(x^{(k)})|\mathcal{F}_k) + \\ & + \gamma\mathbb{E}(h_{\alpha_k}(x^{(k+1)};x^{(k)}) + {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k) + \eta_k, \end{aligned} \tag{9}$$

where $0 < \gamma \le 1$ and $\{\eta_k\}$ is a sequence of nonnegative random variable such that $\sum_{k=0}^\infty \eta_k < \infty$ almost surely. Condition (9) is crucial to prove convergence results for the iteration (7). In Section 3 we show that, when the objective function is defined as in (2), condition (9) can be practically implemented by means of a well defined line search procedure acting on the steplenght α_k properly bounded.

The following theorems clarify which conditions are needed to guarantee (weaker or stronger) convergence properties for iteration (7). The convergence results are stated in the “almost sure (a.s.)” sense. To improve the flow of this section, the proofs of the theorems are provided in Appendix 5.

Theorem 1. *Under the Assumptions 1 and 2, let $\{x^{(k)}\}$ be the sequence generated by the method (7) with $\alpha_k \in [\alpha_{min}, \alpha_{max}]$, $\alpha_{min} > 0$. Let $0 < \gamma \leq 1$ and $\{\eta_k\}_{k\in\mathbb{N}}$ be a sequence of nonnegative random variables such that $\sum_{k=0}^{\infty} \eta_k < \infty$ a.s.*
If, for any $x^{(0)} \in dom(P)$, condition (9) holds, then, $P(x^{(k)}) - P^* \longrightarrow \overline{P}$ a.s., where $\overline{P} \geq 0$ is some random variable and P^* is such that $P(x) \geq P^*$, for $x \in dom(P)$. Furthermore, the following assertions hold:

i) $\sum_{k=0}^{\infty} \mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) | \mathcal{F}_k\right) < \infty$, *a.s.,*

ii) $h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) \to 0$ *a.s.*

In the following theorem, we state that, under suitable assumptions, any limit point of the sequence $\{x^{(k)}\}$ generated by the iteration (7) is a stationary point for the objective function $P(x)$ a.s. We remark that this result does not require the convexity of $F(x)$.

Theorem 2. *Under the Assumptions 1 and 2, let $\{x^{(k)}\}$ be the sequence generated by the iteration (7) with $\mathbb{E}(\|e_g^{(k)}\|^2|\mathcal{F}_k) \leq \varepsilon_k$ where $\{\varepsilon_k\}$ is a nonnegative sequence such that $\lim_{k\to\infty} \varepsilon_k = 0$ and $\alpha_k \in [\alpha_{min}, \alpha_{max}]$. Moreover, suppose that the condition (9) is satisfied for any $x^{(0)} \in dom(P)$. Then any limit point of the sequence $\{x^{(k)}\}$ is stationary for problem (1) a.s.*

Remark 1. Under the assumption that F is a convex function, when there exists a subsequence of $\{x^{(k)}\}$ converging a.s. to $\bar{x}$, since $\bar{x}$ is a minimum point, we can set $P(\bar{x}) = P^*$. Since $P(x^{(k)}) - P^*$ is convergent a.s. to $\bar{P}$ we can conclude that $\bar{P} = 0$ and that $P(x^{(k)})$ converges to P^* a.s.

If, in addition to the hypotheses of Theorem 2, the function F is supposed to be convex, then the a.s. convergence of the sequence generated by iteration (7) can be proved.

Theorem 3. *Under the Assumptions 1 and 2, let $\{x^{(k)}\}$ the sequence generated by the iteration (7) with $\mathbb{E}(\|e_g^{(k)}\|^2|\mathcal{F}_k) \leq \varepsilon_k$ where $\{\varepsilon_k\}$ is a nonnegative sequence such that $\lim_{k\to\infty} \varepsilon_k = 0$ and $\alpha_k \in [\alpha_{min}, \alpha_{max}]$. Moreover, assume that condition (9) holds and the function F is convex. Then the sequence $\{x^{(k)}\}$ converges to a solution of (1) a.s.*

The last theorem is concerning the convergence rate of the scheme.

Theorem 4. *Under the Assumptions 1 and 2, let $\{x^{(k)}\}$ be the sequence generated by the iteration (7) with $\mathbb{E}(\|e_g^{(k)}\|^2|\mathcal{F}_k) \leq \varepsilon_k$ where $\{\varepsilon_k\}$ is a nonnegative sequence such that $\lim_{k\to\infty} \varepsilon_k = 0$ and $\alpha_k \in [\alpha_{min}, \alpha_{max}]$. Moreover, assume that condition (9) holds, the function F is convex. Then, by denoting $\overline{x}^{(K)} = \frac{1}{K+1}\sum_{k=0}^{K} x^{(k)}$, we have*

$$\mathbb{E}(P(\overline{x}^{(K)}) - P(x^*)) = \mathcal{O}\left(\frac{1}{K}\right). \tag{10}$$

Furthermore, when $\mathbb{E}\left(\sum_{k=0}^{\infty} k\eta_k\right) < \infty$, we have

$$\mathbb{E}(P(x^{(k)}) - P(x^*)) = \mathcal{O}\left(\frac{1}{k}\right). \tag{11}$$

3 Practical implementation

The theoretical algorithm developed in the previous section to face problem (1) can be summarized as follows.

Given $x^{(0)} \in \mathbb{R}^d$ and $0 < \alpha_{min} < \alpha_{max}$, the general iteration reads as

$$\begin{cases} g^{(k)} = \nabla F(x^{(k)}) + e_g^{(k)} \\ x^{(k+1)} = \operatorname{prox}_{\alpha_k R}(x^{(k)} - \alpha_k g^{(k)}) \end{cases}$$

where $e_g^{(k)}$ and $\alpha_k \in [\alpha_{min}, \alpha_{max}]$ are such that

$$\mathbb{E}(\|e_g^{(k)}\|^2 | \mathcal{F}_k) \leq \varepsilon_k \quad \text{where} \quad \{\varepsilon_k\}_{k\in\mathbb{N}},\ \varepsilon_k \geq 0,\ \lim_{k\to\infty} \varepsilon_k = 0, \tag{12a}$$

$$\begin{aligned} &\mathbb{E}(P(x^{(k+1)})|\mathcal{F}_k) \leq \mathbb{E}(P(x^{(k)})|\mathcal{F}_k) + \\ &\qquad + \gamma\mathbb{E}(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + e_g^{(k)^T}(x^{(k+1)} - x^{(k)}|\mathcal{F}_k)) + \eta_k \\ &\qquad \text{with} \quad 0 < \gamma \leq 1,\ \sum_{k=0}^{\infty} \eta_k < \infty. \end{aligned} \tag{12b}$$

The practical implementation of this algorithm requires to specify how to select the stochastic gradient $g^{(k)}$ and the steplength α_k so that conditions (12a) and (12b) hold. In this section we detail how to practically apply such an algorithm to the particular class of finite-sum optimization problems (2). The hypotheses on the objective function are discussed in Section 3.1, while Section 3.2 is devoted to the description of the details of the practical implementation. The resulting scheme is a Proximal Line search based Stochastic first order Algorithm (Prox-LISA), where the stochastic gradient variance is dynamically reduced along the iterations.

3.1 Finite-sum optimization problems

We focus on optimization problems of the form (2) where the function F is a finite sum of a possibly large number of terms, namely

$$F(x) = \frac{1}{N} \sum_{i=1}^{N} f_i(x). \tag{13}$$

We suppose that each $f_i(x)$ is convex and has Lipschitz-continuous gradient with Lipschitz constant $\bar{L}_i$. Assumption 1 (ii) is clearly verified with $L \leq \frac{1}{N}\sum_{i=1}^{N} \bar{L}_i \leq \bar{L} = \max_{i=1,\ldots,N} \bar{L}_i$. In many machine learning applications, N is very large and hence the evaluation of F and its gradient can be prohibitively expensive. In addition to the computational convenience, it may happen that the objective function or its gradient are not available since the data set is not fully accessible at the beginning of the training process, as in the online learning scenario (see, for example, [8, Sec. 2.2.2]).

For these reasons, given at each iteration k a sample $\mathcal{N}_k$ of size $N_k << N$ randomly and uniformly chosen from $\mathcal{N} = \{1, \ldots, N\}$, we consider

$$f_{\mathcal{N}_k}(x) = \frac{1}{N_k} \sum_{i\in\mathcal{N}_k} f_i(x)$$

and, by taking into account the first derivatives of $f_{\mathcal{N}_k}$, we obtain the following sub-sampled gradient of F

$$g_{\mathcal{N}_k}(x) = \frac{1}{N_k} \sum_{i\in\mathcal{N}_k} \nabla f_i(x)$$

which is an unbiased estimator of the gradient of F. In this setting, we have

$$g_{\mathcal{N}_k}(x^{(k)}) = g^{(k)} = \nabla F(x^{(k)}) + e_g^{(k)}.$$

By assuming that all the f_i have Lipschitz-continuous gradients with Lipschitz constant $\bar{L}_i$, the gradient estimate $g_{\mathcal{N}_k}(x)$ is Lipschitz continuous with Lipschitz parameter $\bar{L}_{N_k} = \frac{1}{N_k}\sum_{i\in\mathcal{N}_k} \bar{L}_i$. As a consequence [2, Lemma 6.9.1], it holds that

$$f_{\mathcal{N}_k}(y) \leq f_{\mathcal{N}_k}(x) + g_{\mathcal{N}_k}(x)^T(y - x) + \frac{\bar{L}_{N_k}}{2}\|y - x\|^2, \qquad \forall x, y \in \mathbb{R}^d. \tag{14}$$

3.2 The Prox-LISA algorithm

The Prox-LISA method, detailed in Algorithm 1, is a version of (7) which takes into account the expression of F as in (13) and specifies how to practically satisfy the assumptions of the convergence theorems. In particular we have to act on the parameters N_k and α_k so that their choices force the validity of the requirements (12a) and (12b).

Algorithm 1 Prox-LISA

Given $x^{(0)} \in \mathbb{R}^d$, $0 < N_0 < N$, $\beta \in (0,1)$, $0 < \alpha_{min} < \alpha_{max}$ and a nonnegative sequence $\{\varepsilon_k\}_{k\in\mathbb{N}}$, $\lim_{k\to\infty} \varepsilon_k = 0$.

FOR $k = 0, 1, 2, ...$

STEP 1. Choose a sample $\mathcal{N}_k$ of size N_k and compute $g_{\mathcal{N}_k}(x^{(k)})$.
IF

$$V^{(k)} = \frac{1}{N_k(N_k-1)} \sum_{i\in\mathcal{N}_k} \|\nabla f_i(x^{(k)}) - g_{\mathcal{N}_k}(x^{(k)})\|^2 \leq \varepsilon_k \quad \text{OR} \quad N_k \geq N$$

THEN go to STEP 2.

ELSE set $N_k = \min\left\{N, \max\left\{\frac{N_k V^{(k)}}{\varepsilon_k}, N_k+1\right\}\right\}$ and go to STEP 1.

STEP 2. Compute $f_{\mathcal{N}_k}(x^{(k)})$ and set $\alpha_k \in [\alpha_{min}, \alpha_{max}]$.

STEP 3. Set $\bar{x}^{(k)} = \text{prox}_{\alpha_k R}(x^{(k)} - \alpha_k g_{\mathcal{N}_k}(x^{(k)}))$.
IF

$$f_{\mathcal{N}_k}(\bar{x}^{(k)}) \leq f_{\mathcal{N}_k}(x^{(k)}) + g_{\mathcal{N}_k}(x^{(k)})^T(\bar{x}^{(k)} - x^{(k)}) + \frac{1}{2\alpha_k}\|\bar{x}^{(k)} - x^{(k)}\|^2 \tag{15}$$

THEN go to STEP 4.

ELSE set $\alpha_k = \beta\alpha_k$ and go to STEP 3.

STEP 4. Set $x^{(k+1)} = \bar{x}^{(k)}$.

END FOR

The main features of Prox-LISA are described below.

3.2.1 STEP 1 – The selection of the sub-sampling $\mathcal{N}_k$

Condition (12a) requires that the conditional expected value of the variance of the stochastic gradients vanishes for $k \longrightarrow \infty$. Under the assumption that each stochastic gradient $\nabla f_i(x^{(k)})$ has an expectation equal to the full gradient $\nabla F(x^{(k)})$, the variance of each such stochastic gradient is equal and is bounded by a constant value $C \geq 0$; then, for an arbitrary $i \in \mathcal{N}_k$, we have that [15, p. 183]

$$\mathbb{E}(\|g_{\mathcal{N}_k}(x^{(k)}) - \nabla F(x^{(k)})\|^2|\mathcal{F}_k) \leq \frac{\mathbb{E}(\|\nabla f_i(x^{(k)}) - \nabla F(x^{(k)})\|^2|\mathcal{F}_k)}{N_k} \leq \frac{C}{N_k}.$$

This bound, when combined with a suitable rate of increase in N_k, enables to obtain that the condition on $\mathbb{E}(\|e_g^{(k)}\|^2|\mathcal{F}_k)$ can be satisfied. Indeed, it is sufficient that

$$\frac{\mathbb{E}(\|\nabla f_i(x^{(k)}) - \nabla F(x^{(k)})\|^2|\mathcal{F}_k)}{N_k} \leq \varepsilon_k \tag{16}$$

with $N_k = \frac{C}{\varepsilon_k}$. Following the strategy proposed in [4, 7, 9], the first term of the above condition can be approximated with the sample variance. More in detail, at the k-th iteration, the sample variance is defined as $\frac{1}{N_k-1}\sum_{i\in\mathcal{N}_k}\|\nabla f_i(x^{(k)})-g_{\mathcal{N}_k}(x^{(k)})\|^2$. Hence, as a practical counterpart of condition (12a), at each iteration we force that

$$V^{(k)} \equiv \frac{1}{N_k(N_k-1)}\sum_{i\in\mathcal{N}_k}\|\nabla f_i(x^{(k)})-g_{\mathcal{N}_k}(x^{(k)})\|^2 \leq \varepsilon_k, \tag{17}$$

where $\{\varepsilon_k\}_{k\in\mathbb{N}}$ is any nonnegative sequence such that $\lim_{k\to\infty}\varepsilon_k = 0$. An example of such a sequence, suitable to also guarantee that condition (12b) holds, has been provided at the end of Section 3.2.2. In view of inequality (17), the variance can be monitored by a proper increase of the sample size N_k: whenever condition (17) is not satisfied, the sample size N_k is increased. In particular, a tentative value for N_k can be the maximum between N_k+1 and, in view of (16), the ratio between the sample variance and ε_k; the obtained value has to be at most equal to the size N of the training set. Thus, N_k can be updated by the following rule:

$$N_k = \min\left\{N, \max\left\{\frac{N_k V^{(k)}}{\varepsilon_k}, N_k+1\right\}\right\}.$$

3.2.2 STEP 3 – The selection of the steplength α_k

In order to guarantee condition (12b), we impose the steplength α_k satisfies the line search procedure (15). Starting from a tentative value $\alpha_k\in[\alpha_{min},\alpha_{max}]$, the fulfillment of the condition (15) is checked and, if it does not hold, α_k is reduced until the condition holds.
First of all, we remark that, in view of (14), the line search procedure (15) is well defined. Indeed, if $\alpha_k \leq \dfrac{1}{\bar{L}_{N_k}}$ then inequality (15) is automatically satisfied.

In the following we show that the condition (12b) can be assured by the line search strategy (15) combined with a proper bound on α_{max} and the summability of the sequence $\{\varepsilon_k\}$. We start by noting that

$$h_{\alpha_k}(z;x^{(k)}) + {e_g^{(k)}}^T(z-x^{(k)}) = g_{\mathcal{N}_k}(x^{(k)})^T(z-x^{(k)}) + \frac{1}{2\alpha_k}\|z-x^{(k)}\|^2 - R(x^{(k)}) + R(z).$$

Hence to require that $x^{(k+1)}$ satisfies the line search inequality (15), namely,

$$f_{\mathcal{N}_k}(x^{(k+1)}) \leq f_{\mathcal{N}_k}(x^{(k)}) + g_{\mathcal{N}_k}(x^{(k)})^T(x^{(k+1)}-x^{(k)}) + \frac{1}{2\alpha_k}\|x^{(k+1)}-x^{(k)}\|^2,$$

is equivalent to force that

$$f_{\mathcal{N}_k}(x^{(k+1)}) - f_{\mathcal{N}_k}(x^{(k)}) + R(x^{(k+1)}) - R(x^{(k)}) \leq h_{\alpha_k}(x^{(k+1)};x^{(k)}) + {e_g^{(k)}}^T(x^{(k+1)}-x^{(k)}).$$

Therefore, taking the conditional expectation with respect to the σ-algebra $\mathcal{F}_k$, we get

$$\begin{aligned}\mathbb{E}\left(f_{\mathcal{N}_k}(x^{(k+1)}) - f_{\mathcal{N}_k}(x^{(k)}) + R(x^{(k+1)}) - R(x^{(k)})|\mathcal{F}_k\right) \leq \\ \mathbb{E}\left(h_{\alpha_k}(x^{(k+1)};x^{(k)}) + {e_g^{(k)}}^T(x^{(k+1)}-x^{(k)})|\mathcal{F}_k\right).\end{aligned} \tag{18}$$

Now we observe that, while $\mathbb{E}(f_{\mathcal{N}_k}(x^{(k)})|\mathcal{F}_k) = \mathbb{E}(F(x^{(k)})|\mathcal{F}_k)$, $\mathbb{E}\left(f_{\mathcal{N}_k}(x^{(k+1)})|\mathcal{F}_k\right)$ is different from $\mathbb{E}\left(F(x^{(k+1)})\,|\mathcal{F}_k\right)$. Nevertheless, by assuming the convexity of $f_{\mathcal{N}_k}$ and the Lipschitz-continuity of ∇F, we can state that

$$\begin{aligned}f_{\mathcal{N}_k}(x^{(k)}) &\leq f_{\mathcal{N}_k}(x^{(k+1)}) + g_{\mathcal{N}_k}(x^{(k)})^T(x^{(k)}-x^{(k+1)})\\ F(x^{(k+1)}) &\leq F(x^{(k)}) + \nabla F(x^{(k)})^T(x^{(k+1)}-x^{(k)}) + \frac{L}{2}\|x^{(k+1)}-x^{(k)}\|^2.\end{aligned}$$

Combining the previous two inequalities, it holds that

$$\begin{aligned}
F(x^{(k+1)}) - f_{\mathcal{N}_k}(x^{(k+1)}) &\leq F(x^{(k)}) + \nabla F(x^{(k)})^T(x^{(k+1)} - x^{(k)}) + \\
&+ \frac{L}{2}\|x^{(k+1)} - x^{(k)}\|^2 - f_{\mathcal{N}_k}(x^{(k)}) + \\
&- g_{\mathcal{N}_k}(x^{(k)})^T(x^{(k+1)} - x^{(k)}) \\
&= F(x^{(k)}) - f_{\mathcal{N}_k}(x^{(k)}) + \\
&+ (\nabla F(x^{(k)}) - g_{\mathcal{N}_k}(x^{(k)}))^T(x^{(k+1)} - x^{(k)}) + \\
&+ \frac{L}{2}\|x^{(k+1)} - x^{(k)}\|^2 \\
&= F(x^{(k)}) - f_{\mathcal{N}_k}(x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)}) + \\
&+ \frac{L}{2}\|x^{(k+1)} - x^{(k)}\|^2 \\
&\leq F(x^{(k)}) - f_{\mathcal{N}_k}(x^{(k)}) + \frac{\theta L}{2}\|x^{(k+1)} - x^{(k)}\|^2 + \\
&+ \frac{1}{2\theta L}\|e_g^{(k)}\|^2 + \frac{L}{2}\|x^{(k+1)} - x^{(k)}\|^2 \\
&= F(x^{(k)}) - f_{\mathcal{N}_k}(x^{(k)}) + \frac{1}{2\theta L}\|e_g^{(k)}\|^2 + \\
&+ \frac{L}{2}(1+\theta)\|x^{(k+1)} - x^{(k)}\|^2,
\end{aligned} \tag{19}$$

where $\theta > 0$. First of all we observe that, from (32) with $y = x^{(k)}$, the following inequality is true

$$\begin{aligned}
\|x^{(k+1)} - x^{(k)}\|^2 \leq \alpha_k &\left(R(x^{(k)}) - R(x^{(k+1)}) - {g^{(k)}}^T(x^{(k+1)} - x^{(k)})\right) + \\
&\pm \frac{1}{2}\|x^{(k+1)} - x^{(k)}\|^2
\end{aligned}$$

and, hence,

$$\begin{aligned}
\frac{1}{2}\|x^{(k+1)} - x^{(k)}\|^2 &\leq \alpha_k \left(R(x^{(k)}) - R(x^{(k+1)}) - {g^{(k)}}^T(x^{(k+1)} - x^{(k)}) + \right. \\
&\left. - \frac{1}{2\alpha_k}\|x^{(k+1)} - x^{(k)}\|^2\right) \\
&= \alpha_k \left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})\right) \\
&\leq \alpha_{max} \left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})\right).
\end{aligned} \tag{20}$$

By taking the conditional expected value with respect to $\mathcal{F}_k$ in (19) and considering (20), we have

$$\begin{aligned}
&\mathbb{E}\left(F(x^{(k+1)}) - f_{\mathcal{N}_k}(x^{(k+1)})|\mathcal{F}_k\right) \leq \mathbb{E}\left(F(x^{(k)}) - f_{\mathcal{N}_k}(x^{(k)})|\mathcal{F}_k\right) + \frac{1}{2L\theta}\mathbb{E}\left(\|e_g^{(k)}\|^2|\mathcal{F}_k\right) + \\
&\quad + L(\theta+1)\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) \\
&= \frac{1}{2L\theta}\mathbb{E}\left(\|e_g^{(k)}\|^2|\mathcal{F}_k\right) + L(\theta+1)\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) \\
&\leq \frac{\varepsilon_k}{2L\theta} + L(\theta+1)\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right),
\end{aligned} \tag{21}$$

where the equality follows from the fact that $\mathbb{E}\left(F(x^{(k)})|\mathcal{F}_k\right) = \mathbb{E}\left(f_{\mathcal{N}_k}(x^{(k)})|\mathcal{F}_k\right)$. Consequently, combining (18) and (21), we can write

$$
\begin{aligned}
\mathbb{E}\left(P(x^{(k+1)}) - P(x^{(k)})|\mathcal{F}_k\right) &= \mathbb{E}\left(F(x^{(k+1)}) + R(x^{(k+1)})|\mathcal{F}_k\right) + \\
&+ \mathbb{E}\left(-f_{\mathcal{N}_k}(x^{(k)}) - R(x^{(k)})|\mathcal{F}_k\right) \pm \mathbb{E}\left(f_{\mathcal{N}_k}(x^{(k+1)})|\mathcal{F}_k\right) \\
&\leq \mathbb{E}\left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) + \\
&+ \mathbb{E}\left(F(x^{(k+1)}) - f_{\mathcal{N}_k}(x^{(k+1)})|\mathcal{F}_k\right) \\
&\leq \mathbb{E}\left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) + \frac{\varepsilon_k}{2L\theta} + \\
&+ L(\theta+1)\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) \\
&= (1 - L(\theta+1)\alpha_{max})\mathbb{E}\left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) + \\
&+ \frac{\varepsilon_k}{2L\theta}.
\end{aligned}
\tag{22}
$$

Then, sufficient conditions to be able to implement the condition (12b) are $\alpha_{max} < \dfrac{1}{L(\theta+1)}$, $\eta_k = \mathcal{O}\left(\dfrac{1}{2L\theta}\varepsilon_k\right)$ and $N_k = \dfrac{C}{\varepsilon_k}$ with $\{\varepsilon_k\}$ such that $\sum_{k=1}^{\infty} \varepsilon_k < \infty$.

We observe that the sequence $\{\varepsilon_k\} = \{Ca^k\}$, with $0 < a < 1$ and $C > 0$ is such that $\sum_{k=0}^{\infty} \eta_k < \infty$ and $\sum_{k=0}^{\infty} k\eta_k < \infty$. Finally, we remark that, even if the value of the Lipschitz constant L is not always known, the line search strategy (15) can also be practically employed as an iterative technique to approximate it. Such an approach has been also exploited in [4, 24].

4 Numerical experiments

In order to evaluate the effectiveness of the proposed method, we consider the minimization problem arising in training binary classifier, having the form

$$
\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} f_i(x) + \delta\|x\|_1 \equiv \min_{x \in \mathbb{R}^d} F(x) + \delta\|x\|_1, \tag{23}
$$

where $\delta > 0$ is the regularization parameter. For all the experiments we set $\delta = \frac{1}{N}$ where N is the number of samples in the training set.

4.1 Performance evaluation for medium-large data sets

In this section, we consider seven data sets for three different convex loss functions. Table 1 shows the details of the seven data sets and the cardinality of the training and the testing sets. The data sets *w8a*, *GISETTE*, *IJCNN1*, *RCV1* are downloadable from `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/`, whereas *MNIST* is available at `https://yann.lecun.com/exdb/mnist/`, *CIFAR10* at `https://www.cs.toronto.edu/~kriz/cifar.html` and *CHINA0* at `https://www.causality.inf.ethz.ch/home.php`.

In particular, we adapt *MNIST* and *CIFAR10* for the binary case: for the *MNIST* data set, the two classes are the even and odd digits; for the *CIFAR10* data set, the two classes are the even and odd class positions.

We built linear classifiers corresponding to three different convex loss functions. By denoting as $a_i \in \mathbb{R}^d$ and $b_i \in \{1, -1\}$ the feature vector and the class label of the i-th example, respectively, the loss function $F(x)$ assumes one of the following form:

- logistic regression (LR) loss:

$$
F(x) = \frac{1}{N} \sum_{i=1}^{N} \log\left[1 + e^{-b_i a_i^T x}\right];
$$

Data set	d	#training set (N)	#testing set
MNIST	784	60000	10000
w8a	300	44774	4975
CIFAR10	3072	50000	10000
CHINA0	132	16033	1604
GISETTE	5000	6000	1000
IJCNN1	22	49990	91701
RCV1	47236	20242	10000

Table 1: Data set characteristics.

- square loss (SL):

$$F(x) = \frac{1}{N}\sum_{i=1}^{N}(1 - b_i a_i^T x)^2;$$

- smooth hinge (SH) loss:

$$F(x) = \frac{1}{N}\sum_{i=1}^{N}\begin{cases} \frac{1}{2} - b_i a_i^T x, & \text{if } b_i a_i^T x \leq 0 \\ \frac{1}{2}(1 - b_i a_i^T x)^2, & \text{if } 0 < b_i a_i^T x < 1 \\ 0, & \text{if } b_i a_i^T x \geq 1 \end{cases}. \tag{24}$$

We compare the behaviour of Algorithm 1, hereafter denoted by Prox-LISA, with Prox-SG, Prox-SVRG, Prox-SARAH and Prox-Spider-boost. In the following section we detail how we fix the hyperparameters involved in the definition of the considered methods.

4.1.1 Hyperparameter setting

For the Prox-SG method, we consider a fixed size $\overline{N}$ for the mini batch and a decreasing sequence of steplengths. In particular, for all the test problems, we set $\overline{N} = 50$ and $\alpha_j = \frac{100\overline{\alpha}_0}{100+j}$, $j \geq 0$, where $\overline{\alpha}_0$ is the initial steplength and j denotes the counter of the epochs. We select the initial steplength as $\overline{\alpha}_0 = \alpha_{opt} \cdot \overline{N}$, where α_{opt} is the best tuned value for the initial steplength found for Prox-SG with $\overline{N} = 1$. We remark that the value for α_{opt} has been obtained after a time and resource consuming procedure of repeated trials. In Table 2 we report the best tuned values of α_{opt} for all the considered test problems. For the

	MNIST	*w8a*	*CIFAR10*	*CHINA0*	*GISETTE*	*IJCNN1*	*RCV1*
LR	1e-3	1e-1	5e-5	1e-2	1e-3	1e-2	1e-1
SL	1e-4	1e-3	1e-6	1e-3	1e-4	1e-3	1e-1
SH	1e-3	5e-2	1e-5	1e-2	1e-3	1e-2	1e-1

Table 2: Best tuned values of α_{opt} for the considered test problems.

Prox-SVRG, Prox-SARAH and Prox-Spider-boost methods, we followed the selection strategy described in [27], [22] and [26]. The settings that have been used are presented in Appendix 6. For Prox-LISA the initial mini batch size is $N_0 = 3$ and the line search hyperparameter at STEP 3 of Algorithm 1 is $\beta = \frac{1}{2}$ in all the experiments. The attempt value for the steplength at STEP 2 is, in general, $\alpha_0 = 1$ for the first iteration, and $\alpha_k = \min(\alpha_0, \alpha_{k-1}\frac{1}{\beta})$ for the following iterations. This initialization technique for α_k enables to avoid too much expensive line search reductions. Furthermore, the rule $\varepsilon_k = 100 \cdot 0.999^k$, guarantees the theoretical requirements.

4.1.2 Steplength selection

The selection of the best tuned steplength is a time and resource consuming task. Without a suitable value for the steplength, the Prox-SG and Prox-SVRG methods could have poor performance also at the initial epochs, as highlighted in Figure 1. Indeed, Figure 1 shows the behaviour of the average optimality gap $|F(x) - F^*|$ with respect to the epochs, obtained on 10 runs with the same initial setup. Here F^* is an estimate of the optimal objective value on the training set, obtained by a huge number of iterations of the Prox-SVRG method. On the other hand, the Prox-LISA method performs well for any initial value α_0 for the steplength, thanks to the line search procedure that automatically adjusts its value. For this reason, Prox-LISA demonstrates robustness with respect to the choice of the steplength, by avoiding the expensive tuning of an optimal value.

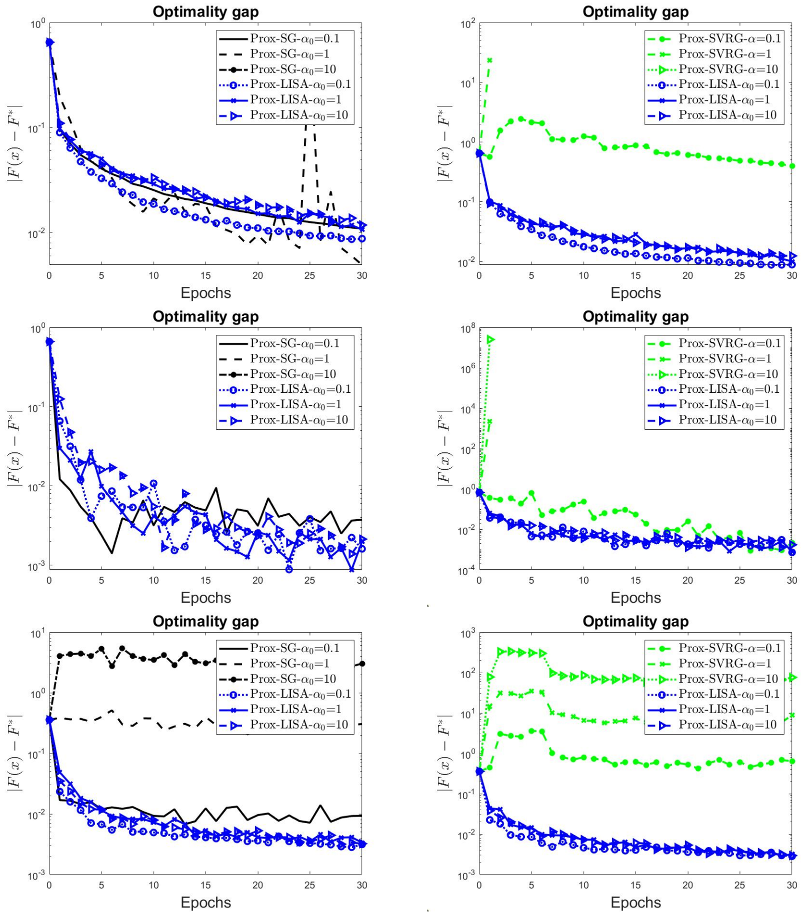

Figure 1: Behaviour of the average optimality gap for different initial steplength values in the cases of the *GISETTE* data set with the LR loss (first row), the *CHINA0* data set with the SL loss (second row) and the *MNIST* data set with the SH loss (third row) generated by either Prox-SG and Prox-LISA (first column) or Prox-SVRG and Prox-LISA (second column).

4.1.3 Results

For all the numerical experiments, carried out in Matlab® on 1.8 GHz Intel Core i7 processor, we perform 10 runs with the same hyperparameters, but leaving the possibility to the random number generator to vary. In fact, due to the stochastic nature of the methods, the average values and STandard Deviations (STD) of the results obtained in ten different simulations provide more reliable comments. Specifically, for each numerical test, we report the following results:

- the average and the STD values of the optimality gap $|F(\overline{x}) - F^*|$ on the training set, where $\overline{x}$ is the iterate at the end of 30 epochs;
- the average and STD for the accuracy $A(\overline{x})$ at the end of the 30 epochs, evaluated on the testing set, i.e., the percentage of well-classified examples. This choice is to keep overfitting under control;
- the average execution time required by any method to perform the iterations corresponding to 30 epochs.

Method		*MNIST*	*w8a*	*CIFAR10*	*CHINA0*	*GISETTE*	*IJCNN1*	*RCV1*
Prox-SG								
	$F(\bar{x})-F^*$	0.0058	0.0028	0.0148	0.0070	0.0188	0.0002	0.0130
	$\pm STD$	±0.0005	±0.0009	±0.0017	±0.0002	±0.0003	$\pm 5.63e^{-5}$	$\pm 6.50e^{-5}$
	$A(\bar{x})$	0.8988	0.9063	0.6651	0.9207	0.9806	0.9196	0.9559
	$\pm STD$	±0.0010	±0.0012	±0.0024	±0.0018	±0.0010	±0.0004	±0.0004
	Time (s)	8.8739	3.3963	44.6086	0.4643	9.0889	3.3025	6.7344
Prox-LISA								
	$F(\bar{x})-F^*$	0.0047	0.0076	0.0172	0.0044	0.0099	$7.72e^{-5}$	0.0384
	$\pm STD$	±0.0014	±0.0002	±0.0007	±0.0008	±0.0011	$\pm 3.09e^{-5}$	±0.0002
	$A(\bar{x})$	0.8978	0.9045	0.6647	0.9220	0.9828	0.9195	0.9511
	$\pm STD$	±0.0014	±0.0004	±0.0016	±0.0016	±0.0014	±0.0003	±0.0005
	Time (s)	13.0627	3.5072	45.3131	0.7572	6.6877	2.4240	32.6620
Prox-SVRG								
	$F(\bar{x})-F^*$	0.0007	0.0050	0.0043	$4.45e^{-5}$	0.0142	$3.28e^{-12}$	0.0003
	$\pm STD$	$\pm 1.33e^{-5}$	$\pm 5.08e^{-5}$	$\pm 5.96e^{-6}$	$\pm 1.28e^{-5}$	±0.0002	$\pm 3.09e^{-12}$	$\pm 1.36e^{-5}$
	$A(\bar{x})$	0.8993	0.9060	0.6665	0.9215	0.9788	0.9200	0.9563
	$\pm STD$	±0.0003	±0.0004	±0.0004	±0.0002	±0.0017	$\pm 7.36e^{-6}$	±0.0001
	Time (s)	33.1527	19.3612	76.6365	6.2881	28.8811	21.3319	79.7286
Prox-SARAH								
	$F(\bar{x})-F^*$	0.0095	0.0267	0.0183	0.0101	0.0370	0.0013	0.0240
	$\pm STD$	±0.0001	0.0014	$\pm 1.47e^{-5}$	±0.0002	±0.0003	±0.0001	$\pm 6.63e^{-5}$
	$A(\bar{x})$	0.8980	0.8968	0.6649	0.9195	0.9756	0.9175	0.9552
	$\pm STD$	±0.0004	0.0015	±0.0003	±0.0007	±0.0022	$\pm 7.01e^{-5}$	±0.0004
	Time (s)	32.9768	19.6675	75.4867	6.4740	38.3911	20.9983	120.0760
Prox-Spider-boost								
	$F(\bar{x})-F^*$	0.0094	0.0714	0.0180	0.0100	0.0366	0.0012	0.0231
	$\pm STD$	±0.0002	0.0002	$\pm 1.09e^{-5}$	±0.0006	±0.0004	±0.0001	$\pm 5.78e^{-5}$
	$A(\bar{x})$	0.8980	0.8947	0.6652	0.9193	0.9762	0.9175	0.9552
	$\pm STD$	±0.0006	0.0005	±0.0003	±0.0007	±0.0014	$\pm 6.87e^{-5}$	±0.0003
	Time (s)	33.1968	3.3072	74.0672	6.4511	37.1445	21.0805	95.1606

Table 3: Results for the LR loss function.

Method		*MNIST*	*w8a*	*CIFAR10*	*CHINA0*	*GISETTE*	*IJCNN1*	*RCV1*
Prox-SG								
	$F(\bar{x})-F^*$	0.0040	0.0034	0.0475	0.0012	0.0264	0.0003	0.0131
	$\pm STD$	±0.0019	±0.0012	±0.0006	±0.0009	±0.0014	±0.0001	±0.0006
	$A(\bar{x})$	0.8929	0.8902	0.6560	0.9189	0.9797	0.9109	0.9624
	$\pm STD$	±0.0010	±0.0022	±0.0009	±0.0019	±0.0012	±0.0004	±0.0008
	Time (s)	9.8683	3.3704	46.4524	0.5530	8.3145	1.2225	7.1736
Prox-LISA								
	$F(\bar{x})-F^*$	0.0055	0.0090	0.0423	0.0007	0.0338	0.0008	0.0096
	$\pm STD$	±0.0034	±0.0077	±0.0015	±0.0005	±0.0052	±0.0002	±0.0001
	$A(\bar{x})$	0.8915	0.8917	0.6593	0.9191	0.9782	0.9111	0.9642
	$\pm STD$	±0.0022	±0.0030	±0.0017	±0.0008	±0.0023	±0.0008	±0.0006
	Time (s)	16.4417	3.7876	51.6428	0.9155	7.2947	1.6522	52.2726
Prox-SVRG								
	$F(\bar{x})-F^*$	0.0003	0.0004	0.0112	$9.92e^{-5}$	0.0174	$7.25e^{-12}$	0.0049
	$\pm STD$	$\pm 9.61e^{-6}$	$\pm 5.96e^{-6}$	$\pm 1.38e^{-5}$	$\pm 1.73e^{-7}$	$\pm 9.71e^{-5}$	$\pm 5.39e^{-12}$	±0.0011
	$A(\bar{x})$	0.8941	0.8918	0.6632	0.9208	0.9786	0.9107	0.9633
	$\pm STD$	±0.0002	±0.0003	±0.0005	±0.0000	±0.0008	±0.0000	±0.0009
	Time (s)	75.4468	18.7737	85.5064	6.3724	17.2537	19.1054	118.4410
Prox-SARAH								
	$F(\bar{x})-F^*$	0.0032	0.0035	0.0609	0.0004	0.0469	$1.11e^{-13}$	0.0066
	$\pm STD$	$\pm 3.22e^{-5}$	±0.0002	$\pm 1.25e^{-5}$	$\pm 6.86e^{-5}$	±0.0005	$\pm 5.92e^{-14}$	0.0008
	$A(\bar{x})$	0.8938	0.8909	0.6472	0.9207	0.9788	0.9107	0.9634
	$\pm STD$	±0.0003	±0.0005	±0.0001	±0.0008	±0.0011	±0.0000	±0.0006
	Time (s)	37.1714	18.6542	83.2887	6.8066	18.7305	20.5479	148.3810
Prox-Spider-boost								
	$F(\bar{x})-F^*$	0.0031	0.0034	0.0450	0.0004	0.0455	$1.05e^{-15}$	0.0068
	$\pm STD$	$\pm 5.81e^{-5}$	0.0003	$\pm 1.03e^{-5}$	$\pm 6.80e^{-5}$	±0.0004	$\pm 2.38e^{-16}$	±0.0008
	$A(\bar{x})$	0.8940	0.8908	0.6576	0.9206	0.9791	0.9107	0.9634
	$\pm STD$	±0.0005	0.0004	±0.0001	±0.0010	±0.0023	±0.0000	±0.0006
	Time (s)	31.6962	18.7820	33.6751	6.7120	19.2858	0.5254	125.2840

Table 4: Results for the SL loss function.

Tables 3-4-5 report the results obtained by the considered methods in all the data sets for the LR, SL, and SH loss functions, respectively. As highlighted by these tables, the behaviour of Prox-LISA seems promising and competitive for all the test problems. Indeed Prox-LISA achieves a comparable performance with respect to the other methods, by avoiding both the expensive tuning of the steplength and the computation of the full gradient of F. These two benefits are crucial in a big data and a green Artificial Intelligence contexts. More specific remarks on Tables 3-4-5 are needed.

i) The values of the accuracy obtained for *CIFAR10* are less than the values of the accuracies corresponding to all the other data sets. Even if the number of the epochs is increased, the same accuracy is reached. However, this is in line with the results given in [25] where the authors state

Method		*MNIST*	*w8a*	*CIFAR10*	*CHINA0*	*GISETTE*	*IJCNN1*	*RCV1*
Prox-SG								
	$F(\bar{x}) - F^*$	0.0048	0.0022	0.0158	0.0029	0.0031	0.0003	0.0043
	$\pm STD$	±0.0039	±0.0008	±0.0009	±0.0018	±8.87e^{-5}	±0.0001	±7.01e^{-5}
	$A(\bar{x})$	0.8988	0.9071	0.6645	0.9213	0.9789	0.9224	0.9609
	$\pm STD$	±0.0017	±0.0012	±0.0011	±0.0017	±0.0012	±0.0005	±0.0009
	Time (s)	14.5694	5.0630	175.4150	0.7968	10.5035	5.1354	47.2714
Prox-LISA								
	$F(\bar{x}) - F^*$	0.0029	0.0023	0.0233	0.0018	0.0058	9.47e^{-5}	0.0090
	$\pm STD$	±0.0003	±0.0001	±0.0021	±0.0011	±0.0008	±3.49e^{-5}	±0.0001
	$A(\bar{x})$	0.8999	0.9069	0.6569	0.9218	0.9796	0.9223	0.9600
	$\pm STD$	±0.0009	±0.0006	±0.0047	±0.0008	±0.0020	±0.0006	±0.0007
	Time (s)	11.1626	3.0313	43.9968	0.9352	5.6401	4.2684	6.65674
Prox-SVRG								
	$F(\bar{x}) - F^*$	0.0012	0.0022	0.0064	0.0004	0.0136	2.60e^{-8}	0.0011
	$\pm STD$	±1.11e^{-5}	±0.0007	±7.85e^{-6}	±6.01e^{-5}	±0.0002	±2.13e^{-8}	±3.11e^{-5}
	$A(\bar{x})$	0.9011	0.9078	0.6674	0.9201	0.9786	0.9217	0.9615
	$\pm STD$	±0.0002	±0.0005	±0.0004	±0.0002	±0.0022	±1.49e^{-5}	±0.0003
	Time (s)	41.3201	22.6324	114.3400	8.6883	17.3228	26.1497	87.1046
Prox-SARAH								
	$F(\bar{x}) - F^*$	0.0057	0.0220	0.0206	0.0040	0.1891	1.04e^{-5}	0.0072
	$\pm STD$	±0.0012	0.0003	±1.12e^{-5}	±0.0023	±0.0764	±1.87e^{-5}	±0.0004
	$A(\bar{x})$	0.8992	0.8988	0.6594	0.9221	0.9466	0.9216	0.9601
	$\pm STD$	±0.0009	0.0006	±0.0002	±0.0017	±0.0034	±0.0005	±0.0008
	Time (s)	40.3616	26.5678	114.4630	8.8509	17.3031	262.0627	85.0363
Prox-Spider-boost								
	$F(\bar{x}) - F^*$	0.0072	0.0260	0.0205	0.0068	0.0704	4.22e^{-6}	0.0074
	$\pm STD$	±0.0001	0.0002	±7.71e^{-6}	±0.0001	±2.75e^{-5}	±3.94e^{-6}	±0.0005
	$A(\bar{x})$	0.8984	0.8978	0.6596	0.9166	0.9619	0.9216	0.9598
	$\pm STD$	±0.0005	0.0005	±0.0003	±0.0011	±0.0007	±0.0003	±0.0009
	Time (s)	20.4269	4.3881	114.4560	0.9238	24.7770	57.5633	80.2073

Table 5: Results for the SH loss function.

that the accuracy of *CIFAR10* in general cannot exceed 70% with a binary classification by convex loss functions.

ii) In general, the STD values for Prox-SG and Prox-LISA are higher than the STD values related to the results provided by Prox-SARAH, Prox-SVRG and Prox-Spider-boost. This fact is not surprising. Indeed, Prox-SARAH, Prox-SVRG and Prox-Spider-boost periodically perform a full gradient computation and this results in a greater stability of their performance over the ten runs.

iii) The computational time taken by the hybrid methods is significantly higher than the one of Prox-SG and Prox-LISA due to the cyclic calculation of a full gradient. As later shown in Figure 6, the behaviour of the optimality gap provided by the considered methods with respect to the computational time is greatly different from that obtained with respect to the epochs (Figures 2-5).

In Figures 2-5 we show the behaviour of the considered schemes for some of the test problems. We report three types of plots in all figures. All the plots are obtained over 10 runs with the same hyperparameters. The first row of each figure shows the average optimality gap achieved by the methods with respect to the first 30 epochs. The cyan vertical line (corresponding to 10 epochs) highlights the ability of the methods in reducing the optimality gap in fewer epochs. In the second row of Figures 2-5, the average accuracy obtained on the testing set by the methods is reported. The third row presents the increase of the mini batch size for the Prox-LISA method with respect to the iterations. The red circles indicate when the steplength is reduced by the line search. We can observe that the Prox-LISA performance is at least comparable to the one of the other methods and in several cases it is the best, especially considering the performance in the first 10 epochs. On the other hand, it is worth to remark that, almost in all the examples, the line search procedure is performed a large number of times, as well as the increase of the sample size which is fundamental to monitor the variance. However, we observe that, in general, the final mini batch size is less than about two orders of magnitude of the size N of the training set. In Figure 6 the decrease of the optimality gap provided by the different methods is reported for two different test problems. It is quite evident how the computation of the full gradient badly affects the performance of the hybrid methods, especially for large data sets.

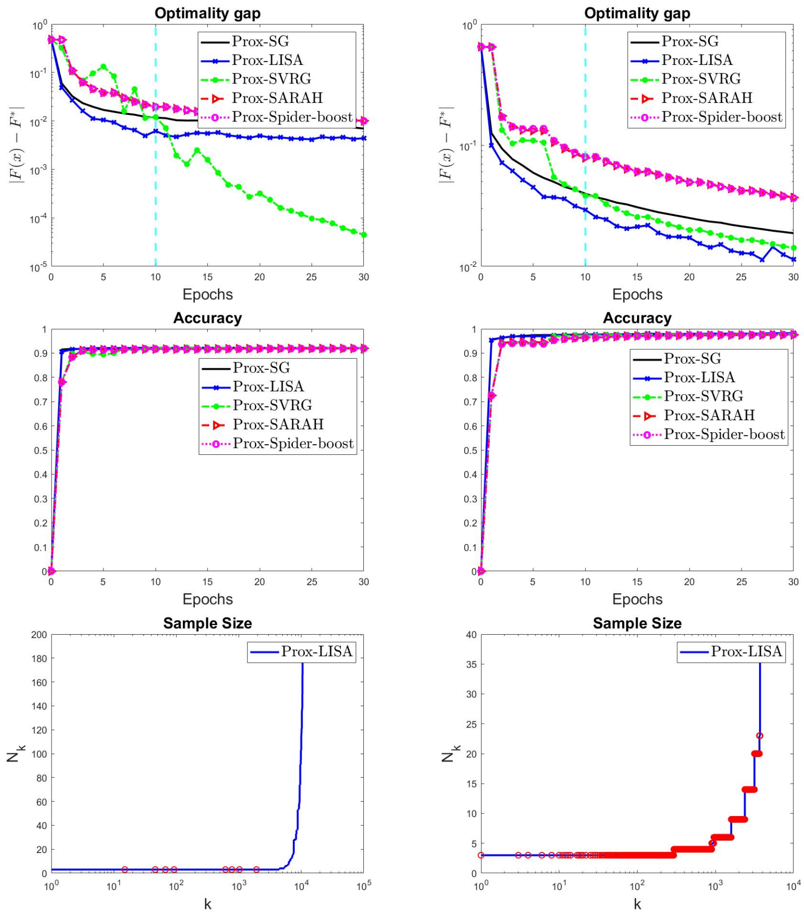

Figure 2: Optimality gap (first row) and accuracy (second row) given by all the methods and increase of sample size in Prox-LISA (third row) for the *CHINA0* data set (left column) and the *GISETTE* data set (right column) with LR loss.

4.2 Performance evaluation for a very large data set

As a final experiment, we consider the *AVAZU-APP* data set, available at `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/`, that is several orders of magnitude greater than the data sets discussed above. Indeed the number d of features of each example is 999990, the cardinality N of the training set is 14596137, whereas the number of examples of the testing set is 1459614. In addition to the trouble of the size of the training set, binary classification in this case is also particularly challenging because of the large imbalance in the data set itself. Specifically, the proportion between the two classes is about $88\% - 12\%$. The huge size of this data set requires some preliminary evaluations and consequent adaptations of the methods. In particular, the time to execute the iterations corresponding to one epoch dramatically increases whatever is the method. This last consideration combined with the fact that the number d of features in the data set is near one million leads to evaluate the numerical behavior of the methods only for 10 epochs and in the case of the LR loss function, limiting ourselves to a single run. Furthermore, the setup for Prox-LISA is slightly modified, by setting the initial mini batch size as $N_0 = 50$, the initial steplegth as $\alpha_0 = 10$; furthermore the law governing the variance control is set as $\varepsilon_k = 100 \cdot 0.999^{\frac{k}{10^2}}$.

The latter choice was made since we need to avoid that ε_k reaches values less than the machine precision in a single epoch. Turning now to more specific time considerations, we observed that the execution time of a single epoch of a hybrid method is near 10 times greater than the one of Prox-SG. For this reason, and from a GreenAI perspective, determining effective setup values for the Prox-SARAH

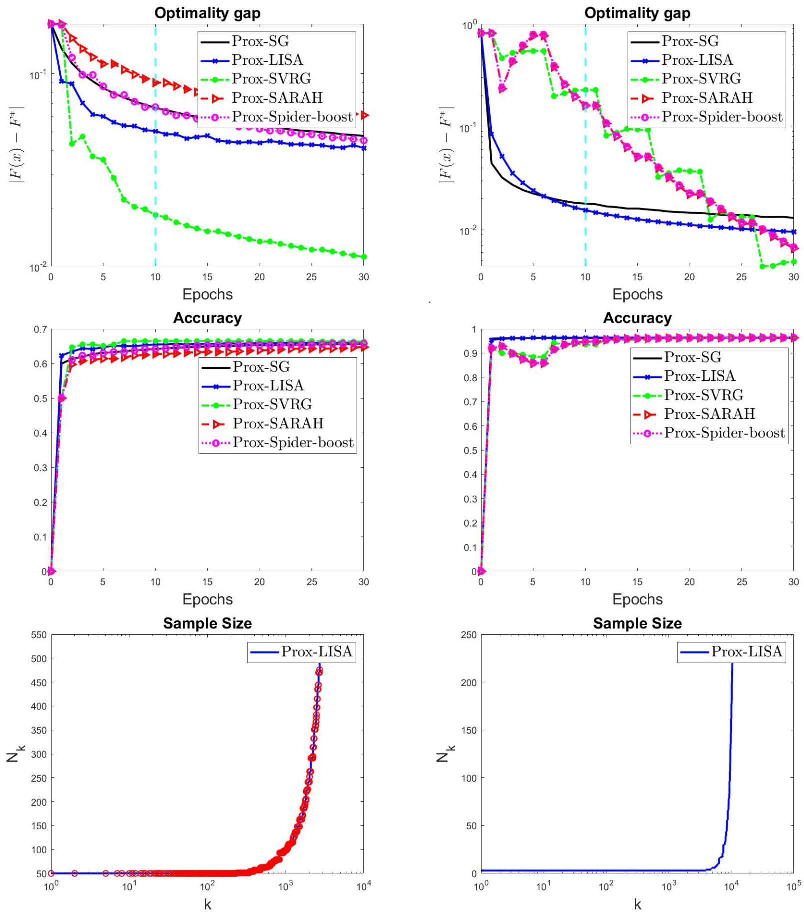

Figure 3: Optimality gap (first row) and accuracy (second row) given by all the methods and increase of sample size in Prox-LISA (third row) for the *CIFAR10* data set (left column) and the *RCV1* data set (right column) with SL loss.

and Prox-Spider-boost methods would become prohibitively expensive.
However, in order to have at least a comparison with a hybrid method, we considered the Prox-SVRG algorithm. To summarize, the binary classification of the *AVAZU-APP* data set combined with the LR loss function is obtained by comparing Prox-SG, Prox-LISA and Prox-SVRG. Table 6 shows the best tuned values of the steplength for Prox-SG and Prox-SVRG. We remember that, for LR, $\hat{L} = max_i(\|a_i\|^2))/4$ [28].

	Prox-SG	Prox-SVRG
LR	$\alpha_0 = 1e-2$	$\alpha_0 = 0.1/\hat{L}$

Table 6: Best tuned values for the steplength in the considered test problem.

4.2.1 Results

The following numerical results are obtained in Matlab® environment on a AMD Ryzen 7 3750H with Radeon Vega MobileGfx2.30GHz, 16GB RAM; in view of the huge size of the considered data set, we also report memory information since memory traffic also matters and greatly impacts the computational time. In Table 7 we report the numerical results obtained in a single trial. In particular, the optimality gap, the accuracy and the execution time in seconds at the end of 10 epochs are given. As already mentioned, the execution time of Prox-SVRG method is an order of magnitude higher with respect the other non-hybrid methods. This large difference in time can be traced not only to the different computational cost, but

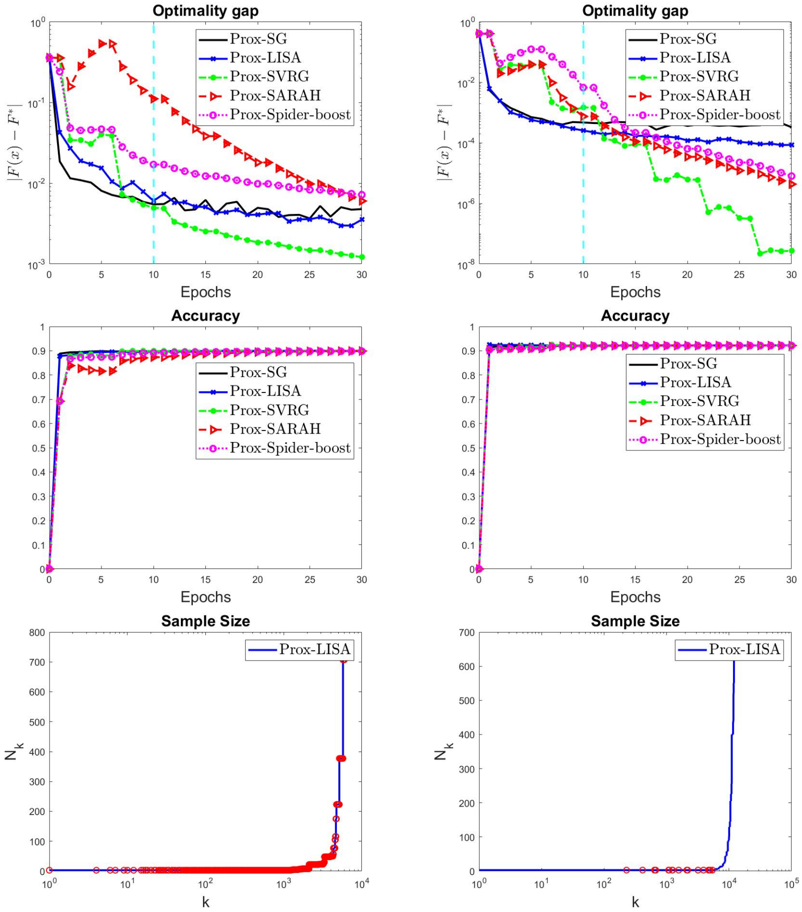

Figure 4: Optimality gap (first row) and accuracy (second row) given by all the methods and increase of sample size in Prox-LISA (third row) for the *MNIST* data set (left column) and the *IJCNN1* data set (right column) with SH loss.

also to the memory traffic.
On the other hand, Figure 7 shows the behaviour of the optimality gap and the accuracy provided by the considered methods with respect to both the epochs and the execution times in seconds. The Prox-SVRG scheme achieves the best performance in terms of optimality gap and accuracy, even if the results of both Prox-SG and Prox-LISA are comparable. Nevertheless, the usage of the computational resources is very different. Indeed, it is evident from the second column of Figure 7 that the purely stochastic methods perform better.We also observe that the execution time of Prox-LISA is slightly higher than that of Prox-SG, while being of the same order of magnitude. The comparison, however, does not take into account the different attempts that were made for Prox-SG to find the best steplength. As a final statistic, we observe that the maximum mini batch size for Prox-LISA reaches the value 228.

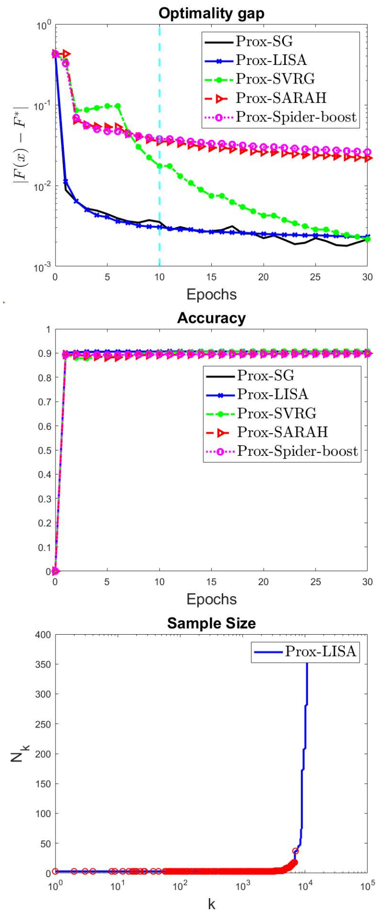

Figure 5: Optimality gap (first panel) and accuracy (second panel) given by all the methods and increase of sample size in Prox-LISA (third panel) for the *w8a* data set with SH loss.

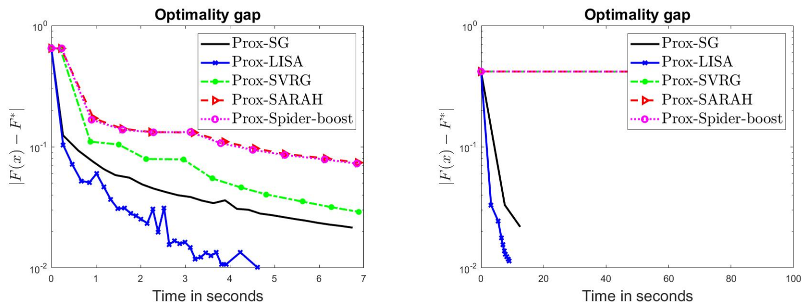

Figure 6: Optimality gap with respect to the computational time given by all the methods for the *GISETTE* data set with LR loss (on the left) and *RCV1* data set with SH loss (on the right).

Conclusions and future works

In this paper we presented and analysed a proximal stochastic gradient algorithm based on two conditions: one requires the decrease of the objective function values in expectation and the other forces the control of the variance of the stochastic directions. We performed the convergence analysis of the method, obtaining results in both the nonconvex and convex cases and providing also a convergence rate estimate in the latter one.

In the case of the minimization of the regularized empirical risk, proper strategies can be adopted to practically realize both the above mentioned conditions. The resulting algorithm does not need the computation of the full gradient of the empirical risk and its numerical performance does not depend on

Method		LR
Prox-SG		
	$F(\bar{x}) - F^*$	0.0017
	$A(\bar{x})$	0.874
	Time	5 hrs 10 mins.
Prox-LISA		
	$F(\bar{x}) - F^*$	0.0007
	$A(\bar{x})$	0.8742
	Time	6 hrs 10 mins.
Prox-SVRG		
	$F(\bar{x}) - F^*$	0.0002
	$A(\bar{x})$	0.8743
	Time	1 day 17 hrs 24 mins.

Table 7: A single run results for *AVAZU-APP* data set.

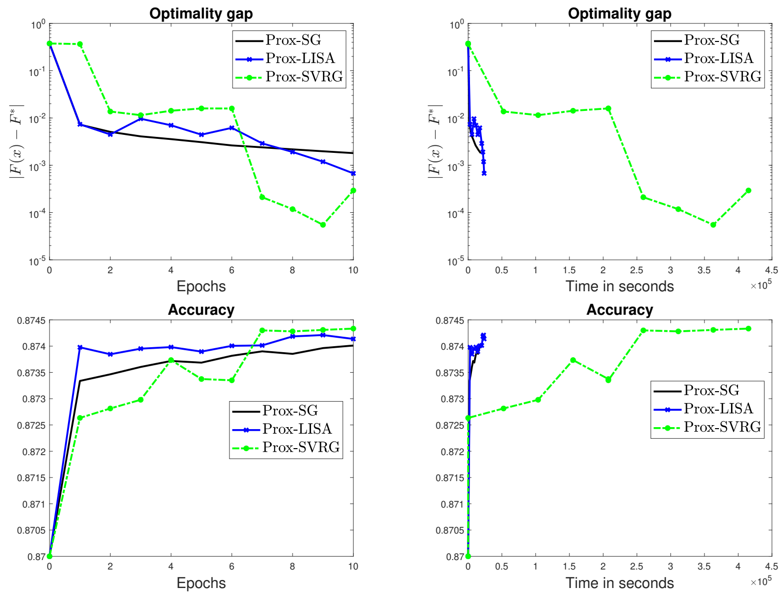

Figure 7: *AVAZU-APP* data set with LR loss. First row: optimality gap versus epochs (left) and execution times in seconds (right). Second row: accuracy versus epochs (left) and execution time in seconds (right). The results are related to a single run.

an optimal value for the steplength parameter. The numerical results on large-scale machine learning problems show that the performances of the suggested method are promising and comparable with those of state-of-the-art methods. Future work will be addressed especially to deepen the theoretical and numerical analysis in the nonconvex case. More general results about line search based schemes could be obtained by using Empirical Process Theory [17] or the approaches developed in [18, 3].

This work has been partially supported by the INdAM research group GNCS.

Appendices

5 Proofs of theorems for Section 2

To prove Theorems 1, 2, 3 and 4, Lemmas 1, 2 and 3 are needed. Lemma 1 recalls well known results on the proximal operator (for the proof, see [10, 5] and references therein) while Lemma 3 is a classical result from stochastic analysis.

Lemma 1. *Let $\alpha > 0$, $x \in dom(P)$, $\hat{y} = p_\alpha(x)$. The following statements hold true.*

a. $\hat{y} = \mathrm{prox}_{\alpha R}(x - \alpha u)$ if and only if $0 \in \partial h_\alpha(\hat{y}; x)$; equivalently, $\frac{1}{\alpha}(x - \hat{y}) - u = w$, $w \in \partial R(\hat{y})$.

b. The function h_α is strongly convex with modulus of convexity $\dfrac{1}{\alpha}$.

c. $h_\alpha(x; x) = 0$.

d. $h_\alpha(\hat{y}; x) \leq 0$ and $h_\alpha(\hat{y}; x) = 0$ if and only if $\hat{y} = x$.

e. x is a stationary point for problem (1) if and only if $x = \hat{y}$ if and only if $h_\alpha(\hat{y}; x) = 0$.

Lemma 2. *Under the Assumption 1 (i), let us consider the sequence $\{x^{(k)}\}$ generated by the iteration (7). If $\alpha_k > 0$, the following inequality holds:*

$$h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) \leq \frac{\alpha_k}{2} \|e_g^{(k)}\|^2. \tag{25}$$

Proof. In view of (4), we have

$$\begin{aligned}
h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) &= \frac{1}{2\alpha_k} \|x^{(k+1)} - x^{(k)}\|^2 + \\
&+ (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}) + R(x^{(k+1)}) - R(x^{(k)}) \\
&= \nabla F(x^{(k)})^T (x^{(k+1)} - p_{\alpha_k}(x^{(k)})) + \nabla F(x^{(k)})^T (p_{\alpha_k}(x^{(k)}) - x^{(k)}) + \\
&+ {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) + \frac{1}{2\alpha_k} \|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2 + \\
&+ \frac{1}{2\alpha_k} \|p_{\alpha_k}(x^{(k)}) - x^{(k)}\|^2 + \frac{1}{\alpha_k} (x^{(k+1)} - p_{\alpha_k}(x^{(k)}))^T (p_{\alpha_k}(x^{(k)}) - x^{(k)}) + \\
&+ R(x^{(k+1)}) - R(x^{(k)}) + R(p_{\alpha_k}(x^{(k)})) - R(p_{\alpha_k}(x^{(k)})) \\
&= h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) + \nabla F(x^{(k)})^T (x^{(k+1)} - p_{\alpha_k}(x^{(k)})) + \\
&+ {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) + \frac{1}{2\alpha_k} \|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2 + \\
&+ \frac{1}{\alpha_k} (x^{(k+1)} - p_{\alpha_k}(x^{(k)}))^T (p_{\alpha_k}(x^{(k)}) - x^{(k)}) + R(x^{(k+1)}) - R(p_{\alpha_k}(x^{(k)})).
\end{aligned} \tag{26}$$

Now, from the convexity of R at $x^{(k+1)}$ and $\frac{x^{(k)} - x^{(k+1)}}{\alpha_k} - (\nabla F(x^{(k)}) + e_g^{(k)}) \in \partial R(x^{(k+1)})$ (Lemma 1 a), we obtain

$$\begin{aligned}
R(x^{(k+1)}) - R(p_{\alpha_k}(x^{(k)})) &\leq \frac{1}{\alpha_k} (x^{(k)} - x^{(k+1)})^T (x^{(k+1)} - p_{\alpha_k}(x^{(k)})) + \\
&- (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - p_{\alpha_k}(x^{(k)})).
\end{aligned}$$

Including the above inequality in (26), we obtain

$$\begin{aligned}
h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) &\leq h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) + \\
&+ {e_g^{(k)}}^T (p_{\alpha_k}(x^{(k)}) - x^{(k)}) - \frac{1}{2\alpha_k} \|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2.
\end{aligned} \tag{27}$$

Then, we have

$$h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) \le {e_g^{(k)}}^T (p_{\alpha_k}(x^{(k)}) - x^{(k+1)}) +$$
$$-\frac{1}{2\alpha_k}\|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2.$$

By adding and subtracting $\frac{\alpha_k}{2}\|e_g^{(k)}\|^2$, we obtain

$$h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) \le$$
$$\le -\frac{1}{2\alpha_k}\| - \alpha_k e_g^{(k)} - x^{(k+1)} + p_{\alpha_k}(x^{(k)})\|^2 + \frac{\alpha_k}{2}\|e_g^{(k)}\|^2 \le \frac{\alpha_k}{2}\|e_g^{(k)}\|^2.$$

□

Lemma 3. *[19, Lemma 11] Let ν_k, u_k, α_k, β_k be nonnegative random variables and let*

$$\mathbb{E}(\nu_{k+1}|\mathcal{F}_k) \le (1+\alpha_k)\nu_k - u_k + \beta_k \qquad a.s.$$
$$\sum_{k=0}^{\infty} \alpha_k < \infty \quad a.s., \qquad \sum_{k=0}^{\infty} \beta_k < \infty \quad a.s.,$$

where $\mathbb{E}(\nu_{k+1}|\mathcal{F}_k)$ denotes the conditional expectation for the given $\nu_0, \dots, \nu_k$, $u_0, \dots, u_k$, $\alpha_0, \dots, \alpha_k$, $\beta_0, \dots, \beta_k$. Then

$$\nu_k \longrightarrow \nu \quad a.s, \qquad \sum_{k=0}^{\infty} u_k < \infty \quad a.s,$$

where $\nu \ge 0$ is some random variable.

Proof of Theorem 1. In view of Assumption 1 (iii), $P(x^{(k)}) - P^*$ is a nonnegative random variable and, from (9), we obtain:

$$\mathbb{E}(P(x^{(k+1)}) - P^*|\mathcal{F}_k) \le \quad (P(x^{(k)}) - P^*) +$$
$$-\gamma \mathbb{E}(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k) + \eta_k.$$

In view of (8) and Lemma 3, we obtain that $P(x^{(k+1)}) - P^* \longrightarrow \overline{P}$ a.s. and

$$\sum_{k=0}^{\infty} \mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) < \infty \qquad \text{a.s.}$$

In order to conclude the proof, we follow a strategy similar to the one employed in the proof of [23, Theorem 2.1]. Define a new random variable $w_j = \sum_{k \ge j} \mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right)$. The sequence $\{w_j\}$ is non increasing and converges to 0 as $j \to +\infty$. As a consequence, from the monotone convergence theorem, it holds that

$$\begin{aligned} 0 &= \mathbb{E}(\lim_{j\to+\infty} w_j) = \lim_{j\to+\infty} \mathbb{E}(w_j) \\ &= \lim_{j\to+\infty} \sum_{k\ge j} \mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})\right) \\ &= \lim_{j\to+\infty} \mathbb{E}\left(\sum_{k\ge j} -h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})\right) \end{aligned}$$

which implies

$$\mathbb{E}\left(\sum_{k\ge j} -h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})\right) < +\infty$$

and, hence,

$$\sum_k -h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) < +\infty \quad \text{a.s.}$$

Then $h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) \to 0$ a.s.

□

Proof of Theorem 2. We suppose that there exists a subsequence of $\{x^{(k)}\}$ that converges a.s. to $\bar{x}$, namely there exists $\mathcal{K} \subseteq \mathbb{N}$ such that

$$\lim_{k\to\infty,\, k\in\mathcal{K}} x^{(k)} = \bar{x} \text{ a.s.}$$

We observe that, since h_{α_k} is strongly convex with modulus of convexity $\dfrac{1}{\alpha_{max}}$ and $p_{\alpha_k}(x^{(k)})$ is its minimum point, we have

$$\frac{1}{2\alpha_{max}}\|z - p_{\alpha_k}(x^{(k)})\|^2 \leq h_{\alpha_k}(z; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}), \quad \forall z. \tag{28}$$

Setting $z = x^{(k+1)}$ in the previous inequality gives

$$\frac{1}{2\alpha_{max}}\|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2 \leq h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}). \tag{29}$$

From the last inequality and Lemma 2, we have

$$0 \leq h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) \leq \frac{\alpha_{max}}{2}\|e_g^{(k)}\|^2$$

and, consequently, by considering the conditional expectation in both members, we have

$$\begin{aligned} 0 \leq \mathbb{E}\left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)}) | \mathcal{F}_k\right) &\leq \frac{\alpha_{max}}{2}\mathbb{E}\left(\|e_g^{(k)}\|^2 | \mathcal{F}_k\right) \\ &\leq \frac{\alpha_{max}\varepsilon_k}{2}. \end{aligned}$$

In view of the law of total expectation and the hypothesis on the sequence $\{\varepsilon_k\}$, the above inequality allows to state that

$$\lim_{k\to\infty} \mathbb{E}(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - h_{\alpha_k}(p_{\alpha_k}(x^{(k)}); x^{(k)})) = 0. \tag{30}$$

From (29) and (30) we can conclude that

$$\lim_{k\to\infty} \mathbb{E}(\|x^{(k+1)} - p_{\alpha_k}(x^{(k)})\|^2) = 0. \tag{31}$$

Then there exists $\mathcal{K}' \subseteq \mathcal{K}$ such that $\lim_{k\to\infty, k\in\mathcal{K}'}(x^{(k+1)} - p_{\alpha_k}(x^{(k)})) = 0$ a.s. By continuity of the operator $p_{\alpha_k}(\cdot)$ with respect to all its arguments, since $\{x^{(k)}\}_{k\in\mathcal{K}}$ is bounded a.s., $\{p_{\alpha_k}(x^{(k)})\}_{k\in\mathcal{K}'}$ is bounded a.s. as well. Thus $\{x^{(k+1)}\}_{k\in\mathcal{K}'}$ is also bounded a.s. and there exists a limit point $\bar{\bar{x}}$ of $\{x^{(k+1)}\}_{k\in\mathcal{K}'}$. We define $\mathcal{K}'' \subseteq \mathcal{K}'$ such that $\lim_{k\to\infty,\, k\in\mathcal{K}''} x^{(k+1)} = \bar{\bar{x}}$ a.s. By continuity of the operator $p_{\alpha_k}(\cdot)$, (31) implies that $\bar{\bar{x}} = p_{\alpha_k}(\bar{x})$ a.s.

Since $h_{\alpha_k}(x; x^{(k)}) + {e_g^{(k)}}^T (x - x^{(k)})$ is strongly convex with modulus of convexity $\frac{1}{\alpha_{max}}$ as well and $x^{(k+1)}$ is its minimum point, we have

$$\begin{aligned} \frac{1}{\alpha_{max}}\|z - x^{(k+1)}\|^2 \leq h_{\alpha_k}(z; x^{(k)}) &+ {e_g^{(k)}}^T (z - x^{(k)}) - h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + \\ &- {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}). \end{aligned}$$

By setting $z = x^{(k)}$ in the previous inequality, we obtain

$$\frac{1}{\alpha_{max}}\|x^{(k)} - x^{(k+1)}\|^2 \leq -h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}).$$

In view of Theorem 1 (ii), we can state that

$$\|x^{(k)} - x^{(k+1)}\|^2 \longrightarrow 0 \quad \text{a.s.}$$

Thus we proved that $\bar{x} = \bar{\bar{x}} = p_{\alpha_k}(\bar{x})$ a.s. and by Lemma 1 e., we have that $\bar{x}$ is a stationary point a.s.

□

Proof of Theorem 3. Let $x^* \in X^*$. Since $\dfrac{x^{(k)} - x^{(k+1)}}{\alpha_k} - g^{(k)} \in \partial R(x^{(k+1)})$, it holds that

$$R(y) \geq R(x^{(k+1)}) + \frac{1}{\alpha_k}(x^{(k)} - x^{(k+1)} - \alpha_k g^{(k)})^T (y - x^{(k+1)}), \quad \forall y \in \mathbb{R}^d.$$

It follows that, $\forall y \in \mathbb{R}^d$,

$$\begin{aligned}\alpha_k R(y) &\geq \alpha_k R(x^{(k+1)}) + (x^{(k)} - x^{(k+1)} - \alpha_k g^{(k)})^T (y - x^{(k+1)}) \\ &= \alpha_k R(x^{(k+1)}) + (x^{(k)} - x^{(k+1)})^T (y - x^{(k+1)}) - \alpha_k {g^{(k)}}^T (y - x^{(k+1)}),\end{aligned}$$

and, hence, the following inequality holds

$$(x^{(k+1)} - x^{(k)})^T (y - x^{(k+1)}) \geq \alpha_k \left(R(x^{(k+1)}) - R(y) + {g^{(k)}}^T (x^{(k+1)} - y) \right). \tag{32}$$

For $y = x^*$ the previous inequality gives

$$\begin{aligned}&(x^{(k+1)} - x^{(k)})^T (x^* - x^{(k)} + x^{(k)} - x^{(k+1)}) \geq \\ \geq &\alpha_k \left(R(x^{(k+1)}) - R(x^*) + {g^{(k)}}^T (x^{(k+1)} - x^{(k)} + x^{(k)} - x^*) \right).\end{aligned}$$

As a consequence, we obtain the following relations:

$$\begin{aligned}
&(x^{(k+1)} - x^{(k)})^T (x^* - x^{(k)}) \geq \alpha_k \left(R(x^{(k+1)}) - R(x^*) + {g^{(k)}}^T (x^{(k)} - x^*) \right) + \\
&\quad - (x^{(k+1)} - x^{(k)})^T (x^{(k)} - x^{(k+1)}) + \alpha_k {g^{(k)}}^T (x^{(k+1)} - x^{(k)}) \\
&= \alpha_k \left(R(x^{(k+1)}) - R(x^*) + (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k)} - x^*) \right) + \\
&\quad + (x^{(k+1)} - x^{(k)})^T (x^{(k+1)} - x^{(k)}) + \alpha_k (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}) \\
&\geq \alpha_k \left(R(x^{(k+1)}) - R(x^*) + F(x^{(k)}) - F(x^*) \right) + \alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) + \\
&\quad + \|x^{(k+1)} - x^{(k)}\|^2 + \alpha_k (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}) \\
&= \alpha_k \left(R(x^{(k+1)}) + R(x^{(k)}) - R(x^{(k)}) + F(x^{(k)}) - P(x^*) \right) + \\
&\quad + \|x^{(k+1)} - x^{(k)}\|^2 + \alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) + \\
&\quad + \alpha_k (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}) \\
&= \alpha_k \left(R(x^{(k+1)}) - R(x^{(k)}) + P(x^{(k)}) - P(x^*) \right) + \|x^{(k+1)} - x^{(k)}\|^2 + \\
&\quad + \alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) + \alpha_k (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}) \\
&\geq \alpha_k \left(R(x^{(k+1)}) - R(x^{(k)}) \right) + \|x^{(k+1)} - x^{(k)}\|^2 + \alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) + \\
&\quad + \alpha_k (\nabla F(x^{(k)}) + e_g^{(k)})^T (x^{(k+1)} - x^{(k)}),
\end{aligned} \tag{33}$$

where the second inequality follows from the convexity of F and the last inequality follows from the fact that $P(x^{(k)}) - P(x^*) \geq 0$. From a basic property of the Euclidean norm[1] we can write

$$\begin{aligned}
\|x^{(k+1)} - x^*\|^2 &= \|x^{(k+1)} - x^{(k)}\|^2 + \|x^{(k)} - x^*\|^2 - 2(x^{(k+1)} - x^{(k)})^T(x^* - x^{(k)}) \\
&\overset{(33)}{\leq} \|x^{(k+1)} - x^{(k)}\|^2 + \|x^{(k)} - x^*\|^2 - 2\alpha_k \left(R(x^{(k+1)}) - R(x^{(k)})\right) + \\
&\quad - 2\|x^{(k+1)} - x^{(k)}\|^2 + \\
&\quad - 2\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) - 2\alpha_k(\nabla F(x^{(k)}) + e_g^{(k)})^T(x^{(k+1)} - x^{(k)}) \\
&= \|x^{(k)} - x^*\|^2 - \|x^{(k+1)} - x^{(k)}\|^2 - 2\alpha_k \left(R(x^{(k+1)}) - R(x^{(k)})\right) + \\
&\quad - 2\alpha_k \nabla F(x^{(k)})^T(x^{(k+1)} - x^{(k)}) - 2\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) + \\
&\quad - 2\alpha_k {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) \\
&= \|x^{(k)} - x^*\|^2 - 2\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) - 2\alpha_k {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) + \\
&\quad - 2\alpha_k \left(R(x^{(k+1)}) - R(x^{(k)}) + \nabla F(x^{(k)})^T(x^{(k+1)} - x^{(k)}) + \right. \\
&\quad \left. + \frac{1}{2\alpha_k}\|x^{(k+1)} - x^{(k)}\|^2\right) \\
&= \|x^{(k)} - x^*\|^2 - 2\alpha_k \left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})\right) + \\
&\quad - 2\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) \\
&\leq \|x^{(k)} - x^*\|^2 - 2\alpha_{max} \left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) + {e_g^{(k)}}^T (x^{(k+1)} - x^{(k)})\right) + \\
&\quad - 2\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*).
\end{aligned}$$

Taking the conditional expectation with respect to the σ-algebra $\mathcal{F}_k$, we obtain

$$\begin{aligned}
\mathbb{E}\left(\|x^{(k+1)} - x^*\|^2 | \mathcal{F}_k\right) \leq \|x^{(k)} - x^*\|^2 &- 2\alpha_{max}\mathbb{E}\left(h_{\alpha_k}(x^{(k+1)}; x^{(k)}) | \mathcal{F}_k\right) + \\
&- 2\alpha_{max}\mathbb{E}\left({e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) | \mathcal{F}_k\right) + \\
&- 2\mathbb{E}\left(\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) | \mathcal{F}_k\right).
\end{aligned} \tag{34}$$

Since $\alpha_k \in \mathcal{F}_{k+1}$ where $\mathcal{F}_k \subset \mathcal{F}_{k+1}$, in view of the tower property we obtain $\mathbb{E}\left(\alpha_k {e_g^{(k)}}^T (x^{(k)} - x^*) | \mathcal{F}_k\right) = 0$ and we rewrite (34) as

$$\begin{aligned}
\mathbb{E}\left(\|x^{(k+1)} - x^*\|^2 | \mathcal{F}_k\right) \leq \|x^{(k)} - x^*\|^2 &+ 2\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) | \mathcal{F}_k\right) + \\
&+ 2\alpha_{max}\mathbb{E}\left(-{e_g^{(k)}}^T (x^{(k+1)} - x^{(k)}) | \mathcal{F}_k\right).
\end{aligned} \tag{35}$$

By combining (35) and part *i)* of Theorem 1 together with Lemma 3, we can state that the sequence $\{\|x^{(k)} - x^*\|\}_{k\in\mathbb{N}}$ converges a.s.

Next we prove the almost sure convergence of the sequence $\{x^{(k)}\}$ by following a strategy similar to the one employed in [20, Theorem 2.1]. Let $\{x_i^*\}_i$ be a countable subset of the relative interior $\mathrm{ri}(X^*)$ that is dense in X^*. From the almost sure convergence of $\|x^{(k)} - x^*\|$, $x^* \in X^*$, we have that for each i, the probability $\mathrm{Prob}(\{\|x^{(k)} - x_i^*\|\}$ is not convergent$) = 0$. Therefore, we observe that

$$\begin{aligned}
&\mathrm{Prob}(\forall i\ \exists b_i \text{ s.t. } \lim_{k\to+\infty} \|x^{(k)} - x_i^*\| = b_i) = 1 - \mathrm{Prob}(\{\|x^{(k)} - x_i^*\|\} \text{ is not convergent}) \\
&\geq 1 - \sum_i \mathrm{Prob}(\{\|x^{(k)} - x_i^*\|\} \text{ is not convergent}) = 1,
\end{aligned}$$

[1] $\|a-b\|^2 + \|b-c\|^2 - \|a-c\|^2 = 2(a-b)^T(c-b), \quad \forall a, b, c \in \mathbb{R}^d$.

where the inequality follows from the union bound, i.e. for each i, $\{\|x^{(k)} - x_i^*\|\}$ is a convergent sequence a.s. For a contradiction, suppose that there are convergent subsequences $\{u_{k_j}\}_{k_j}$ and $\{v_{k_j}\}_{k_j}$ of $\{x^{(k)}\}$ which converge to their limiting points u^* and v^* respectively, with $\|u^* - v^*\| = r > 0$. By Theorem 2, u^* and v^* are stationary; in particular, since P is convex, they are minimum points, i.e. $u^*, v^* \in X^*$. Since $\{x_i^*\}_i$ is dense in X^*, we may assume that for all $\epsilon > 0$, we have $x_{i_1}^*$ and $x_{i_2}^*$ are such that $\|x_{i_1}^* - u^*\| < \epsilon$ and $\|x_{i_2}^* - v^*\| < \epsilon$. Therefore, for all k_j sufficiently large,

$$\|u_{k_j} - x_{i_1}^*\| \leq \|u_{k_j} - u^*\| + \|u^* - x_{i_1}^*\| < \|u_{k_j} - u^*\| + \epsilon.$$

On the other hand, for sufficiently large j, we have

$$\|v_{k_j} - x_{i_1}^*\| \geq \|v^* - u^*\| - \|u^* - x_{i_1}^*\| - \|v_{k_j} - v^*\| > r - \epsilon - \|v_{k_j} - v^*\| > r - 2\epsilon.$$

This contradicts with the fact that $x^{(k)} - x_{i_1}^*$ is convergent. Therefore, we must have $u^* = v^*$, hence there exists $\bar{x} \in X^*$ such that $x^{(k)} \longrightarrow \bar{x}$.

□

Proof of Theorem 4. If we do not neglect the term $P(x^{(k)}) - P(x^*)$ in (33) and in all the subsequent inequalities, instead of (35) we obtain

$$\begin{aligned}
\mathbb{E}\left(\|x^{(k+1)} - x^*\|^2|\mathcal{F}_k\right) \leq{} & \|x^{(k)} - x^*\|^2 + \\
& + 2\alpha_{max}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) + \\
& - 2\alpha_{min}\mathbb{E}\left(P(x^{(k)}) - P(x^*)|\mathcal{F}_k\right).
\end{aligned} \tag{36}$$

Summing the previous inequality from 0 to K and taking the total expectation, we obtain

$$\begin{aligned}
\sum_{k=0}^{K}\mathbb{E}\left(P(x^{(k)}) - P(x^*)\right) \leq{} & \frac{1}{2\alpha_{min}}\left(\|x^{(0)} - x^*\|^2 - \mathbb{E}(\|x^{(K+1)} - x^*\|^2)\right) + \\
& + \frac{\alpha_{max}}{\alpha_{min}}\mathbb{E}\left(\sum_{k=0}^{K}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right)\right).
\end{aligned}$$

By neglecting the term $-\mathbb{E}(\|x^{(K+1)} - x^*\|^2)$ and bounding by S the second term (Theorem 1 *i)*)

$$\sum_{k=0}^{K}\mathbb{E}\left(-h_{\alpha_k}(x^{(k+1)}; x^{(k)}) - {e_g^{(k)}}^T(x^{(k+1)} - x^{(k)})|\mathcal{F}_k\right) \leq S,$$

we obtain

$$\sum_{k=0}^{K}\mathbb{E}\left(P(x^{(k)}) - P(x^*)\right) \leq \frac{1}{2\alpha_{min}}\|x^{(0)} - x^*\|^2 + \frac{\alpha_{max}}{\alpha_{min}}S. \tag{37}$$

Setting $\overline{x}^{(K)} = \frac{1}{K+1}\sum_{k=0}^{K} x^{(k)}$, from the Jensen's inequality, we observe that $\mathbb{E}(P(\overline{x}^{(K)})) \leq \frac{1}{K+1}\sum_{k=0}^{K}\mathbb{E}(P(x^{(k)}))$. Thus, by dividing (37) by $K+1$, we can write

$$\mathbb{E}\left(P(\overline{x}^{(K)}) - P(x^*)\right) \leq \frac{1}{K+1}\left(\frac{1}{2\alpha_{min}}\|x^{(0)} - x^*\|^2 + \frac{\alpha_{max}}{\alpha_{min}}S\right). \tag{38}$$

Thus, we obtain the $\mathcal{O}(1/K)$ ergodic convergence rate of $\mathbb{E}\left(P(\overline{x}^{(K)}) - P(x^*)\right)$.
Now, we assume $\sum_{k=0}^{\infty} k\eta_k = \Sigma$. In (37) the term $\sum_{k=0}^{K}\mathbb{E}\left(P(x^{(k)}) - P(x^*)\right)$ is equal to $\mathbb{E}\left(\sum_{k=0}^{K} P(x^{(k)})\right) - (K+1)P(x^*)$. We observe that, since $0 \leq P(x^{(0)}) - P(x^*)$, we can write

$$\begin{aligned}
\mathbb{E}\left(\sum_{k=1}^{K} P(x^{(k)})\right) - KP(x^*) & \leq \mathbb{E}\left(\sum_{k=0}^{K} P(x^{(k)})\right) - (K+1)P(x^*) \\
& \leq \frac{1}{2\alpha_{min}}\|x^{(0)} - x^*\|^2 + \frac{\alpha_{max}}{\alpha_{min}}S.
\end{aligned}$$

Now we determine a lower bound for $\mathbb{E}\left(\sum_{k=1}^{K} P(x^{(k)})\right)$. From the inequality (8), we have that $\mathbb{E}\left(P(x^{(k)}) - P(x^{(k+1)})|\mathcal{F}_k\right) +$ $\eta_k \geq 0$ and, hence, by considering the total expectation we obtain $\mathbb{E}\left(P(x^{(k)}) - P(x^{(k+1)})\right) + \mathbb{E}(\eta_k) \geq 0$. Thus, we have

$$\begin{aligned} 0 &\leq \sum_{k=1}^{K} k\mathbb{E}\left(P(x^{(k)}) - P(x^{(k+1)})\right) + \sum_{k=1}^{K} k\mathbb{E}(\eta_k) \\ &= \sum_{k=1}^{K} \mathbb{E}(P(x^{(k)})) - K\mathbb{E}(P(x^{(K+1)})) + \mathbb{E}\left(\sum_{k=1}^{K} k\eta_k\right). \end{aligned} \tag{39}$$

Then, we can write

$$K\mathbb{E}(P(x^{(K+1)})) - \Sigma \leq \sum_{k=1}^{K} \mathbb{E}\left(P(x^{(k)})\right). \tag{40}$$

Consequently, we can conclude that

$$\mathbb{E}(P(x^{(K+1)}) - P(x^*)) \leq \frac{1}{K}\left(\frac{1}{2\alpha_{min}}\|x^{(0)} - x^*\|^2 + \frac{\alpha_{max}}{\alpha_{min}} S + \Sigma\right).$$

□

6 Hyperparameter settings for hybrid methods.

For Prox-SVRG method we use the hyperparameter setting proposed in [27], i.e., $\overline{N} = 1$, $m = 2N$, where m is the number of Prox-SVRG inner iterations. This means that a full gradient has to be computed every two epochs. As for the fixed steplength $\overline{\alpha}$, we tried the suggestions reported in the experimental part of [27], i.e., $\alpha = \{\frac{1}{\hat{L}}, \frac{0.1}{\hat{L}}, \frac{0.01}{\hat{L}}\}$, where $\hat{L}$ is an approximation of the Lipschitz constant L of ∇F. In Table 8 we report the best obtained steplength values for all the test problems. For the Prox-SARAH

	MNIST	*w8a*	*CIFAR10*	*CHINA0*	*GISETTE*	*IJCNN1*	*RCV1*
LR	$1/\hat{L}$	$1/\hat{L}$	$1/\hat{L}$	$1/\hat{L}$	$1/\hat{L}$	$0.1/\hat{L}$	$1/\hat{L}$
SL	$0.1/\hat{L}$	$0.1/\hat{L}$	$0.1/\hat{L}$	$0.01/\hat{L}$	$0.1/\hat{L}$	$0.01/\hat{L}$	$0.1/\hat{L}$
SH	$0.1/\hat{L}$	$1/\hat{L}$	$0.1/\hat{L}$	$0.1/\hat{L}$	$1/\hat{L}$	$0.1/\hat{L}$	$0.1/\hat{L}$

Table 8: Best tuned values of the steplength for Prox-SVRG for the considered test problems.

method we use the hyperparameter setting specified in [22] where, by borrowing the notation of the referred paper, $q = 2 + 0.01 + (\frac{1}{100})$, $C = \frac{q^2}{(q^2+8)\hat{L}^2\gamma^2}$ and the values for the other hyperparameters are shown in Table 9.

For the Prox-Spider-boost method we use the hyperparameter setting specified in [26] and the values for the hyperparameters are shown in Table 10.

References

[1] Attouch, H. and Bolte, J. and Svaiter, B. F., Convergence of descent methods for semi-algebraic and tame problems: proximal algorithms, forward-backward splitting, and regularized Gauss–Seidel methods, Math. Program., Ser. A, 137(1) (2013) 91-129.

[2] Bertsekas, D., Convex Optimization Theory, chapter 6 on Convex Optimization Algorithms, Athena Scientific, Belmont, Massachussets (2009) 251-489.

[3] Berahas, A.S. and Cao, L. and Scheinberg K., Global convergence rate analysis of a generic line search algorithm with noise, Siam J. Optim. 31(2) (2021) 1489-1518.

Method	Loss	γ	α	$\overline{N}$	m
MNIST	LR	0.99	α_{opt}	1	2N
MNIST	SL	0.99	α_{opt}	1	2N
MNIST	SH	0.95	α_{opt}	1	2N
w8a	LR	0.99	$\frac{0.1}{\hat{L}}$	1	2N
w8a	SL	0.95	α_{opt}	1	2N
w8a	SH	0.95	$\frac{0.001}{\hat{L}}$	1	2N
CIFAR10	LR	0.95	α_{opt}	1	2N
CIFAR10	SL	0.99	α_{opt}	1	2N
CIFAR10	SH	0.99	α_{opt}	1	2N
CHINA0	LR	0.99	α_{opt}	1	2N
CHINA0	SL	0.99	α_{opt}	1	2N
CHINA0	SH	0.99	α_{opt}	1	2N
GISETTE	LR	0.95	α_{opt}	1	2N
GISETTE	SL	0.95	α_{opt}	1	2N
GISETTE	SH	0.95	α_{opt}	1	2N
IJCNN1	LR	0.99	α_{opt}	1	2N
IJCNN1	SL	0.99	$\frac{2}{q+\gamma\hat{L}}$	$\frac{N^{1/3}}{C}$	$N^{1/3}$
IJCNN1	SH	0.95	α_{opt}	1	2N
RCV1	LR	0.95	α_{opt}	1	2N
RCV1	SL	0.99	α_{opt}	1	2N
RCV1	SH	0.95	α_{opt}	1	2N

Table 9: Settings of Prox-SARAH [22].

Method	Loss	α	$\overline{N}$	m
MNIST	LR	α_{opt}	1	2N
MNIST	SL	α_{opt}	1	2N
MNIST	SH	0.05	256	$\frac{2N}{256}$
w8a	LR	0.05	256	$\frac{2N}{256}$
w8a	SL	α_{opt}	1	2N
w8a	SH	0.05	256	$\frac{2N}{256}$
CIFAR10	LR	α_{opt}	1	2N
CIFAR10	SL	$\frac{1}{2\hat{L}}$	$\sqrt{N}$	$\frac{N}{\sqrt{N}}$
CIFAR10	SH	α_{opt}	1	2N
CHINA0	LR	α_{opt}	1	2N
CHINA0	SL	α_{opt}	1	2N
CHINA0	SH	$\frac{1}{2\hat{L}}$	$\sqrt{N}$	$\frac{N}{\sqrt{N}}$
GISETTE	LR	α_{opt}	1	2N
GISETTE	SL	α_{opt}	1	2N
GISETTE	SH	$\frac{1}{2\hat{L}}$	$\sqrt{N}$	$\frac{N}{\sqrt{N}}$
IJCNN1	LR	α_{opt}	1	2N
IJCNN1	SL	$\frac{1}{2\hat{L}}$	$\sqrt{N}$	$\frac{N}{\sqrt{N}}$
IJCNN1	SH	α_{opt}	1	2N
RCV1	LR	α_{opt}	1	2N
RCV1	SL	α_{opt}	1	2N
RCV1	SH	α_{opt}	1	2N

Table 10: Settings of Prox-Spider-boost [26].

[4] Bollapragada, R. and Byrd, R. and Nocedal, J., Adaptive sampling strategies for stochastic optimization, SIAM Journal on Optimization, 28(4) (2018) 3312-3343.

[5] Bonettini, S. and Loris, I. and Porta, F. and Prato, M., Variable metric inexact line-search based methods for nonsmooth optimization, SIAM Journal on Optimization, 26 (2016) 891-921.

[6] Bonettini, S. and Porta, F. and Prato, M. and Rebegoldi, S. and Ruggiero, V. and Zanni, L., Recent advances in variable metric first-order methods, M. Donatelli, S. Serra-Capizzano, Computational Methods for Inverse Problems in Imaging, Springer INDAM Series, 36 (2019) 1-31.

[7] Bottou, L. and Curtis, F. E. and Nocedal, J., Optimization Methods for Large-Scale Machine Learning, SIAM Review, 60(2) (2018) 223-311.

[8] Bottou, L., Online Algorithms and Stochastic Approximations, in Online Learning and Neural Networks, D. Saad ed., Cambridge University Press, Cambridge, UK, 1998., Revised 2018, .

[9] Byrd, R. H. and Chin, G. M. and Nocedal, J. and Wu, Y., Sample size selection in optimization methods for machine learning, Mathematical Programming, 134(1) (2012) 128-155.

[10] Combettes, P.L. and Pesquet, J.-C., Proximal splitting methods in signal processing, Bauschke, H.H. and Burachik, R.S. and Combettes, P.L. and Elser, V. and Luke, D.R. and Wolkowicz, H., Fixed-point algorithms for inverse problems in science and engineering, Springer Optimization and Its Applications, Springer, New York, NY, (2011) 185-212.

[11] Combettes, P. L. and Wajs, V. R., Signal recovery by proximal forward-backward splitting, SIAM Multiscale Model. Simul., 4 (2005) 1168-1200.

[12] Duchi, J. and Singer, Y., Efficient online and batch learning using forward backward splitting, J. Mach. Learn. Res., 10 (2009) 2873-2898.

[13] Franchini, G. and Ruggiero, V. and Zanni, L., Ritz-like values in steplength selections for stochastic gradient methods, Soft Computing, 24 (2020) 17573-17588.

[14] Franchini, G. and Ruggiero, V. and Trombini, I., Automatic steplength selection in Stochastic Gradient Methods, Machine Learning, Optimization, and Data Science, LOD 2021, (2021), 4124-4132.

[15] Freund, J. E., Mathematical Statistics, Prentice-Hall., Englewood Cliffs, NJ, USA, (1962).

[16] Ghadimi, S. and Lan, G., Stochastic First- and Zeroth-Order Methods for Nonconvex Stochastic Programming, SIAM J. Optim., 23(4) (2013) 2341-2368.

[17] Iusem, A. N., and Jofrè, A. and Oliveira, R. I. and Thompson, P., Variance-based extragradient methods with line search for stochastic variational Inequalities, SIAM J. Optim., 29 (1) (2019) 175-206.

[18] Paquette, C. and Scheinberg, K., A stochastic line search method with expected complexity analysis, SIAM J. Optim., 30 (1) (2020) 349-376.

[19] Polyak, B. T., Introduction to Optimization, Optimization Software, New York, (1987).

[20] Poon, C. and Liang, J. and Schoenlieb, C., Local Convergence Properties of SAGA/Prox-SVRG and Acceleration, PMLR, Proceedings of the 35th International Conference on Machine Learning, 80 (2018) 4124-4132.

[21] Rockafellar, R.T. and Wets, R.J.-B. and Wets, M., Variational Analysis, Springer, Grundlehren der Mathematischen Wissenschaften, Berlin, 317 (1998).

[22] Phamy, N. H. and Nguyen, L. M. and Phan, D. T. and Tran–Dinh, Q., ProxSARAH: An Efficient Algorithmic Framework for Stochastic Composite Nonconvex Optimization, Journal of Machine Learning Research, 21 (2020) 1-48.

[23] Poon, C. and Liang, J. and Schoenlieb, C., Local Convergence Properties of SAGA/Prox-SVRG and Acceleration, Dy, J and Krause, A, Proceedings of the 35th International Conference on Machine Learning, PMLR, Proceedings of Machine Learning Research, 80 (2018) 4124-4132.

[24] Schmidt, M. and Le Roux, N. and Bach, F., Minimizing finite sums with the stochastic average gradient, Mathematical Programming, 162(1) (2017), 83-112.

[25] Le, T. V., and Gopee, N., Classifying CIFAR-10 Images Using Unsupervised Feature & Ensemble Learning, `https://trucvietle.me/files/601-report.pdf`

[26] Wang, Z. and Ji, K. and Zhou, Y. and Liang, Y. and Tarokh, V., SpiderBoost and Momentum: Faster Stochastic Variance Reduction Algorithms, Proceedings of the 33rd International Conference on Neural Information Processing Systems, Curran Associates Inc., 216 (2019) 2406-2416

[27] Xiao, L. and Zhang, T., A proximal Stochastic Gradient Method with Progressive Variance Reduction, SIAM J. Optim., 24(4) (2014) 2057-2075.

[28] Yang, Z., Wang, C., Zang , Y. and Li, J., Mini-batch algorithms with Barzilai-Borwein update step, Neurocomputing, 314 (2018) 2177-185.