A Proximal Stochastic Gradient Method with Progressive Variance Reduction

Lin Xiao[*] Tong Zhang[†]

March 18, 2014

Abstract

We consider the problem of minimizing the sum of two convex functions: one is the average of a large number of smooth component functions, and the other is a general convex function that admits a simple proximal mapping. We assume the whole objective function is strongly convex. Such problems often arise in machine learning, known as regularized empirical risk minimization. We propose and analyze a new proximal stochastic gradient method, which uses a multi-stage scheme to progressively reduce the variance of the stochastic gradient. While each iteration of this algorithm has similar cost as the classical stochastic gradient method (or incremental gradient method), we show that the expected objective value converges to the optimum at a geometric rate. The overall complexity of this method is much lower than both the proximal full gradient method and the standard proximal stochastic gradient method.

1 Introduction

We consider the problem of minimizing the sum of two convex functions:

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \{P(x) \overset{\text{def}}{=} F(x) + R(x)\}, \tag{1}$$

where $F(x)$ is the average of many smooth component functions $f_i(x)$, i.e.,

$$F(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x), \tag{2}$$

and $R(x)$ is relative simple but can be non-differentiable. We are especially interested in the case where the number of components n is very large, and it can be advantageous to use incremental methods (such as stochastic gradient method) that operate on a single component f_i at each iteration, rather than on the entire cost function.

[*] Machine Learning Group, Microsoft Research, Redmond, WA 98052. Email: lin.xiao@microsoft.com.

[†] Department of Statistics, Rutgers University, Piscataway, NJ 08854; and Baidu Inc., Beijing 100085. Email: tzhang@stat.rutgers.edu.

Problems of this form often arise in machine learning and statistics, known as *regularized empirical risk minimization*; see, e.g., [HTF09]. In such problems, we are given a collection of training examples $(a_1, b_1), \ldots, (a_n, b_n)$, where each $a_i \in \mathbb{R}^d$ is a feature vector and $b_i \in \mathbb{R}$ is the desired response. For least-squares regression, the component loss functions are $f_i(x) = (1/2)(a_i^T x - b_i)^2$, and popular choices of the regularization term include $R(x) = \lambda_1 \|x\|_1$ (the Lasso), $R(x) = (\lambda_2/2)\|x\|_2^2$ (ridge regression), or $R(x) = \lambda_1\|x\|_1 + (\lambda_2/2)\|x\|_2^2$ (elastic net), where λ_1 and λ_2 are nonnegative regularization parameters. For binary classification problems, each $b_i \in \{+1, -1\}$ is the desired class label, and a popular loss function is the logistic loss $f_i(x) = \log(1 + \exp(-b_i a_i^T x))$, which can be combined with any of the regularization terms mentioned above.

The function $R(x)$ can also be used to model convex constraints. Given a closed convex set $C \subseteq \mathbb{R}^d$, the constrained problem

$$\underset{x \in C}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

can be formulated as (1) by setting $R(x)$ to be the indicator function of C, i.e., $R(x) = 0$ if $x \in C$ and $R(x) = \infty$ otherwise. Mixtures of the "soft" regularizations (such as ℓ_1 or ℓ_2 penalties) and "hard" constraints are also possible.

The results presented in this paper are based on the following assumptions.

Assumption 1. *The function $R(x)$ is lower semi-continuous and convex, and its effective domain,* $\mathrm{dom}(R) := \{x \in \mathbb{R}^d \,|\, R(x) < +\infty\}$, *is closed. Each $f_i(x)$, for $i = 1, \ldots, n$, is differentiable on an open set that contains* $\mathrm{dom}(R)$, *and their gradients are Lipschitz continuous. That is, there exist $L_i > 0$ such that for all $x, y \in \mathrm{dom}(R)$,*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L_i \|x - y\|. \tag{3}$$

Assumption 1 implies that the gradient of the average function $F(x)$ is also Lipschitz continuous, i.e., there is an $L > 0$ such that for all $x, y \in \mathrm{dom}(R)$,

$$\|\nabla F(x) - \nabla F(y)\| \leq L\|x - y\|.$$

Moreover, we have $L \leq (1/n)\sum_{i=1}^n L_i$.

Assumption 2. *The overall cost function $P(x)$ is strongly convex, i.e., there exist $\mu > 0$ such that for all $x \in \mathrm{dom}(R)$ and $y \in \mathbb{R}^d$,*

$$P(y) \geq P(x) + \xi^T(y - x) + \frac{\mu}{2}\|y - x\|^2, \quad \forall\, \xi \in \partial P(x). \tag{4}$$

The *convexity parameter* of a function is the largest μ such that the above condition holds. The strong convexity of $P(x)$ may come from either $F(x)$ or $R(x)$ or both. More precisely, let $F(x)$ and $R(x)$ have convexity parameters μ_F and μ_R respectively, then $\mu \geq \mu_F + \mu_R$. We note that it is possible to have $\mu > L$ although we must have $\mu_F \leq L$.

1.1 Proximal gradient and stochastic gradient methods

A standard method for solving problem (1) is the *proximal gradient method.* Given an initial point $x_0 \in \mathbb{R}^d$, the proximal gradient method uses the following update rule for $k = 1, 2, \ldots$

$$x_k = \arg\min_{x\in\mathbb{R}^d} \left\{ \nabla F(x_{k-1})^T x + \frac{1}{2\eta_k}\|x - x_{k-1}\|^2 + R(x) \right\},$$

where η_k is the step size at the k-th iteration. Throughout this paper, we use $\|\cdot\|$ to denote the usual Euclidean norm, i.e., $\|\cdot\|_2$, unless otherwise specified. With the definition of *proximal mapping*

$$\mathrm{prox}_R(y) = \arg\min_{x\in\mathbb{R}^d} \left\{ \frac{1}{2}\|x - y\|^2 + R(x) \right\},$$

the proximal gradient method can be written more compactly as

$$x_k = \mathrm{prox}_{\eta_k R}\big(x_{k-1} - \eta_k \nabla F(x_{k-1})\big). \tag{5}$$

This method can be viewed as a special case of *splitting* algorithms [LM79, CR97, Tse00], and its accelerated variants have been proposed and analyzed in [BT09, Nes13].

When the number of components n is very large, each iteration of (5) can be very expensive since it requires computing the gradients for all the n component functions f_i, and also their average. For this reason, we refer to (5) as the proximal *full* gradient (Prox-FG) method. An effective alternative is the *proximal stochastic gradient* (Prox-SG) method: at each iteration $k = 1, 2, \ldots$, we draw i_k randomly from $\{1, \ldots, n\}$ and take the update

$$x_k = \mathrm{prox}_{\eta_k R}\big(x_{k-1} - \eta_k \nabla f_{i_k}(x_{k-1})\big). \tag{6}$$

Clearly we have $\mathbb{E}\nabla f_{i_k}(x_{k-1}) = \nabla F(x_{k-1})$. The advantage of the Prox-SG method is that at each iteration, it only evaluates gradient of a single component function, thus the computational cost per iteration is only $1/n$ that of the Prox-FG method. However, due to the variance introduced by random sampling, the Prox-SG method converges much more slowly than the Prox-FG method. To have a fair comparison of their overall computational cost, we need to combine their cost per iteration and iteration complexity.

Let $x^\star = \arg\min_x P(x)$. Under the Assumptions 1 and 2, the Prox-FG method with a constant step size $\eta_k = 1/L$ generates iterates that satisfy

$$P(x_k) - P(x_\star) \le O\left(\left(\frac{L - \mu_F}{L + \mu_R} \right)^k \right). \tag{7}$$

(See Appendix B for a proof of this result.) The most interesting case for large-scale applications is when $\mu \ll L$, and the ratio L/μ is often called the *condition number* of the problem (1). In this case, the Prox-FG method needs $O\left((L/\mu)\log(1/\epsilon)\right)$ iterations to ensure $P(x_k) - P(x_\star) \le \epsilon$. Thus the overall complexity of Prox-FG, in terms of the total number of component gradients evaluated to find an ϵ-accurate solution, is $O\left(n(L/\mu)\log(1/\epsilon)\right)$. The accelerated Prox-FG methods in [BT09, Nes13] reduce the complexity to $O\big(n\sqrt{L/\mu}\log(1/\epsilon)\big)$.

On the other hand, with a diminishing step size $\eta_k = 1/(\mu k)$, the Prox-SG method converges at a sublinear rate ([DS09, LLZ09]):

$$\mathbb{E}P(x_k) - P(x_\star) \leq O\left(1/\mu k\right). \tag{8}$$

Consequently, the total number of component gradient evaluations required by the Prox-SG method to find an ϵ-accurate solution (in expectation) is $O(1/\mu\epsilon)$. This complexity scales poorly in ϵ compared with $\log(1/\epsilon)$, but it is independent of n. Therefore, when n is very large, the Prox-SG method can be more efficient, especially to obtain low-precision solutions.

There is also a vast literature on *incremental gradient methods* for minimizing the sum of a large number of component functions. The Prox-SG method can be viewed as a variant of the randomized incremental proximal algorithms proposed in [Ber11]. Asymptotic convergence of such methods typically requires diminishing step sizes and only have sublinear convergence rates. A comprehensive survey on this topic can be found in [Ber10].

1.2 Recent progresses and our contributions

Both the Prox-FG and Prox-SG methods do not fully exploit the problem structure defined by (1) and (2). In particular, Prox-FG ignores the fact that the smooth part $F(x)$ is the average of n component functions. On the other hand, Prox-SG can be applied for more general stochastic optimization problems, and it does not exploit the fact that the objective function in (1) is actually a deterministic function. Such inefficiencies in exploiting problem structure leave much room for further improvements.

Several recent work considered various special cases of (1) and (2), and developed algorithms that enjoy the complexity (total number of component gradient evaluations)

$$O\big((n + L_{\max}/\mu)\log(1/\epsilon)\big), \tag{9}$$

where $L_{\max} = \max\{L_1, \ldots, L_n\}$. If $L_{\max}$ is not significantly larger than L, this complexity is far superior than that of both the Prox-FG and Prox-SG methods. In particular, Shalev-Shwartz and Zhang [SSZ13, SSZ12] considered the case where the component functions have the form $f_i(x) = \phi_i(a_i^T x)$ and the Fenchel conjugate functions of ϕ_i and R can be computed efficiently. With the additional assumption that $R(x)$ itself is μ-strongly convex, they showed that a proximal stochastic dual coordinate ascent (Prox-SDCA) method achieves the complexity in (9).

Le Roux et al. [RSB12] considered the case where $R(x) \equiv 0$, and proposed a *stochastic average gradient* (SAG) method which has complexity $O\big(\max\{n, L_{\max}/\mu\}\log(1/\epsilon)\big)$. Apparently this is on the same order as (9). The SAG method is a randomized variant of the *incremental aggregated gradient* method of Blatt et al. [BHG07], and needs to store the most recent gradient for each component function f_i, which is $O(nd)$. While this storage requirement can be prohibitive for large-scale problems, it can be reduced to $O(n)$ for problems with more favorable structure, such as linear prediction problems in machine learning.

More recently, Johnson and Zhang [JZ13] developed another algorithm for the case $R(x) \equiv 0$, called *stochastic variance-reduced gradient* (SVRG). The SVRG method employs a multi-stage scheme to progressively reduce the variance of the stochastic gradient, and achieves the same low complexity in (9). Moreover, it avoids storage of past gradients for the component functions, and its convergence analysis is considerably simpler than that of SAG. A very similar algorithm was proposed by Zhang et al. [ZMJ13], but with a worse convergence rate analysis. Another recent effort to extend the SVRG method is [KR13].

In this paper, we extend the variance reduction technique of SVRG to develop a proximal SVRG (Prox-SVRG) method for solving the more general class of problems defined in (1) and (2). We show that with uniform sampling of the component functions, the Prox-SVRG method achieves the same complexity in (9). Moreover, our method incorporates a weighted sampling strategy. When the sampling probabilities for f_i are proportional to their Lipschitz constants L_i, the Prox-SVRG method has complexity

$$O\big((n + L_{\mathrm{avg}}/\mu)\log(1/\epsilon)\big), \tag{10}$$

where $L_{\mathrm{avg}} = (1/n)\sum_{i=1}^n L_i$. This bound improves upon the one in (9), especially for applications where the component functions vary substantially in smoothness.

2 The Prox-SVRG method

Recall that in the Prox-SG method (6), with uniform sampling of i_k, we have unbiased estimate of the full gradient at each iteration. In order to ensure asymptotic convergence, the step size η_k has to decay to zero to mitigate the effect of variance introduced by random sampling, which leads to slow convergence. However, if we can gradually reduce the variance in estimating the full gradient, then it is possible to use much larger (even constant) step sizes and obtain much faster convergence rate. Several recent work (e.g., [FS12, BCNW12, FG13]) have explored this idea by using mini-batches with exponentially growing sizes, but their overall computational cost is still on the same order as full gradient methods.

Instead of increasing the batch size gradually, we use the variance reduction technique of SVRG [JZ13], which computes the full batch periodically. More specifically, we maintain an estimate $\tilde{x}$ of the optimal point $x_\star$, which is updated periodically, say after every m Prox-SG iterations. Whenever $\tilde{x}$ is updated, we also computes the full gradient

$$\nabla F(\tilde{x}) = \frac{1}{n}\sum_{i=1}^n \nabla f_i(\tilde{x}),$$

and use it to modify the next m stochastic gradient directions. Suppose the next m iterations are initialized with $x_0 = \tilde{x}$ and indexed by $k = 1, \ldots, m$. For each $k \geq 1$, we first randomly pick $i_k \in \{1, \ldots, n\}$ and compute

$$v_k = \nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x}) + \nabla F(\tilde{x}),$$

then we replace $\nabla f_{i_k}(x_{k-1})$ in the Prox-SG method (6) with v_k, i.e.,

$$x_k = \mathrm{prox}_{\eta_k R}\big(x_{k-1} - \eta_k v_k\big). \tag{11}$$

```
Algorithm: Prox-SVRG($\tilde{x}_0, \eta, m$)
iterate: for $s = 1, 2, \ldots$
    $\tilde{x} = \tilde{x}_{s-1}$
    $\tilde{v} = \nabla F(\tilde{x})$
    $x_0 = \tilde{x}$
    probability $Q = \{q_1, \ldots, q_n\}$ on $\{1, \ldots, n\}$
    iterate: for $k = 1, 2, \ldots, m$
        pick $i_k \in \{1, \ldots, n\}$ randomly according to $Q$
        $v_k = (\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x}))/(q_{i_k} n) + \tilde{v}$
        $x_k = \mathrm{prox}_{\eta R}(x_{k-1} - \eta v_k)$
    end
    set $\tilde{x}_s = \frac{1}{m}\sum_{k=1}^{m} x_k$
end
```

Figure 1: The Prox-SVRG method.

Conditioned on x_{k-1}, we can take expectation with respect to i_k and obtain

$$\begin{aligned}\mathbb{E}v_k &= \mathbb{E}\nabla f_{i_k}(x_{k-1}) - \mathbb{E}\nabla f_{i_k}(\tilde{x}) + \nabla F(\tilde{x}) \\ &= \nabla F(x_{k-1}) - \nabla F(\tilde{x}) + \nabla F(\tilde{x}) \\ &= \nabla F(x_{k-1}).\end{aligned}$$

Hence, just like $\nabla f_{i_k}(x_{k-1})$, the modified direction v_k is also a stochastic gradient of F at x_{k-1}. However, the variance $\mathbb{E}\|v_k - \nabla F(x_{k-1})\|^2$ can be much smaller than $\mathbb{E}\|\nabla f_{i_k}(x_{k-1}) - \nabla F(x_{k-1})\|^2$. In fact we will show in Section 3.1 that the following inequality holds:

$$\mathbb{E}\|v_k - \nabla F(x_{k-1})\|^2 \leq 4L_{\max}\big[P(x_{k-1}) - P(x_\star) + P(\tilde{x}) - P(x_\star)\big]. \tag{12}$$

Therefore, when both x_{k-1} and $\tilde{x}$ converge to $x_\star$, the variance of v_k also converges to zero. As a result, we can use a constant step size and obtain much faster convergence.

Figure 1 gives the full description of the Prox-SVRG method with a constant step size η. It allows random sampling from a general distribution $\{q_1, \ldots, q_n\}$, thus is more flexible than the uniform sampling scheme described above. It is not hard to verify that the modified stochastic gradient,

$$v_k = (\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x}))/(q_{i_k} n) + \nabla F(\tilde{x}), \tag{13}$$

still satisfies $\mathbb{E}v_k = \nabla F(x_{k-1})$. In addition, its variance can be bounded similarly as in (12) (see Corollary 3 in Section 3.1).

The Prox-SVRG method uses a multi-stage scheme to progressively reduce the variance of the modified stochastic gradient v_k as both $\tilde{x}$ and x_{k-1} converges to $x_\star$. Each stage s requires $n + 2m$ component gradient evaluations: n for the full gradient at the beginning of each stage, and two for each of the m proximal stochastic gradient steps. For some problems such as linear prediction in machine learning, the cost per stage can be further reduced to only $n + m$ gradient evaluations. In practical implementations, we can also set $\tilde{x}_s$ to be the last iterate x_m, instead of $(1/m)\sum_{k=1}^m x_k$, of the previous stage. This simplifies the computation and we did not observe much difference in the convergence speed.

3 Convergence analysis

Theorem 1. *Suppose Assumptions 1 and 2 hold, and let* $x_\star = \arg\min_x P(x)$ *and* $L_Q = \max_i L_i/(q_i n)$. *In addition, assume that* $0 < \eta < 1/(4L_Q)$ *and* m *is sufficiently large so that*

$$\rho = \frac{1}{\mu\eta(1-4L_Q\eta)m} + \frac{4L_Q\eta(m+1)}{(1-4L_Q\eta)m} < 1. \tag{14}$$

Then the Prox-SVRG method in Figure 1 has geometric convergence in expectation:

$$\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq \rho^s[P(\tilde{x}_0) - P(x_\star)].$$

We have the following remarks regarding the above result:

- The ratio L_Q/μ can be viewed as a "weighted" condition number of $P(x)$. Theorem 1 implies that setting m to be on the same order as L_Q/μ is sufficient to have geometric convergence. To see this, let $\eta = \theta/L_Q$ with $0 < \theta < 1/4$. When $m \gg 1$, we have

  $$\rho \approx \frac{L_Q/\mu}{\theta(1-4\theta)m} + \frac{4\theta}{1-4\theta}.$$

  As a result, choosing $\theta = 0.1$ and $m = 100(L_Q/\mu)$ results in $\rho \approx 5/6$.

- In order to satisfy $\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq \epsilon$, the number of stages s needs to satisfy

  $$s \geq \log\rho^{-1}\log\frac{P(\tilde{x}_0) - P(x_\star)}{\epsilon}.$$

  Since each stage requires $n + 2m$ component gradient evaluations, and it is sufficient to set $m = \Theta(L_Q/\mu)$, the overall complexity is

  $$O\big((n + L_Q/\mu)\log(1/\epsilon)\big).$$

- For uniform sampling, $q_i = 1/n$ for all $i = 1, \ldots, n$, so we have $L_Q = \max_i L_i$ and the above complexity bound becomes (9).

  The smallest possible value for L_Q is $L_Q = (1/n)\sum_{i=1}^n L_i$, achieved at $q_i = L_i/\sum_{j=1}^n L_j$, i.e., when the sampling probabilities for the component functions are proportional to their Lipschitz constants. In this case, the above complexity bound becomes (10).

Since $P(\tilde{x}_s) - P(x_\star) \geq 0$, Markov's inequality and Theorem 1 imply that for any $\epsilon > 0$,

$$\operatorname{Prob}\Big(P(\tilde{x}_s) - P(x_\star) \geq \epsilon\Big) \;\leq\; \frac{\mathbb{E}[P(\tilde{x}_s) - P(x_\star)]}{\epsilon} \;\leq\; \frac{\rho^s[P(\tilde{x}_0) - P(x_\star)]}{\epsilon}.$$

Thus we have the following high-probability bound.

Corollary 1. *Suppose the assumptions in Theorem 1 hold. Then for any $\epsilon > 0$ and $\delta \in (0,1)$, we have*

$$\operatorname{Prob}\big(P(\tilde{x}_s) - P(x_\star) \leq \epsilon\big) \geq 1 - \delta$$

provided that the number of stages s satisfies

$$s \geq \log\left(\frac{[P(\tilde{x}_0) - P(x_\star)]}{\delta\epsilon}\right) \Big/ \log\left(\frac{1}{\rho}\right).$$

If $P(x)$ is convex but not strongly convex, then for any $\epsilon > 0$, we can define

$$P_\epsilon(x) = F(x) + R_\epsilon(x), \qquad R_\epsilon(x) = \frac{\epsilon}{2}\|x\|^2 + R(x).$$

It follows that $P_\epsilon(x)$ is ϵ-strongly convex. We can apply the Prox-SVRG method in Figure 1 to $P_\epsilon(x)$, which replaces the update formula for x_k by the following update rule:

$$x_k = \operatorname{prox}_{\eta R_\epsilon}(x_{k-1} - \eta v_k) = \operatorname*{arg\,min}_{x\in\mathbb{R}^d}\left\{\frac{1}{2}\left\|x - \frac{1}{1+\eta\epsilon}(x_{k-1} - \eta v_k)\right\|^2 + \frac{\eta}{1+\eta\epsilon}R(x)\right\}.$$

Theorem 1 implies the following result.

Corollary 2. *Suppose Assumption 1 holds and let $L_Q = \max_i L_i/(q_i n)$. In addition, assume that $0 < \eta < 1/(4L_Q)$ and m is sufficiently large so that*

$$\rho = \frac{1}{\epsilon\eta(1 - 4L_Q\eta)m} + \frac{4L_Q\eta(m+1)}{(1 - 4L_Q\eta)m} < 1.$$

Then the Prox-SVRG method in Figure 1, applied to $P_\epsilon(x)$, achieves

$$\mathbb{E}P(\tilde{x}_s) \leq \min_x [P(x) + (\epsilon/2)\|x\|^2] + \rho^s[P(\tilde{x}_0) + (\epsilon/2)\|\tilde{x}_0\|^2].$$

If $P(x)$ has a minimum and it is achieved by some $x_\star \in \operatorname{dom}(R)$, then Corollary 2 implies

$$\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq (\epsilon/2)\|x_\star\|^2 + \rho^s[P(\tilde{x}_0) + (\epsilon/2)\|\tilde{x}_0\|^2].$$

This result means that if we take $m = O(L_Q/\epsilon)$ and $s \geq \log(1/\epsilon)/\log(1/\rho)$, then

$$\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq \epsilon\,[P(\tilde{x}_0) + (1/2)\|x_\star\|^2 + (\epsilon/2)\|\tilde{x}_0\|^2]$$

The overall complexity (in terms of the number of component gradient evaluations) is

$$O\big((n + L_Q/\epsilon)\log(1/\epsilon)\big).$$

Similar results for the case of $R(x) \equiv 0$ have been obtained in [RSB12, MZJ13, KR13]. We can also derive a high-probability bound based on Corollary 1, but omit the details here.

3.1 Bounding the variance

Our bound on the variance of the modified stochastic gradient v_k is a corollary of the following lemma.

Lemma 1. *Consider $P(x)$ as defined in (1) and (2). Suppose Assumption 1 holds, and let $x_\star = \arg\min_x P(x)$ and $L_Q = \max_i L_i/(q_i n)$. Then*

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{nq_i}\|\nabla f_i(x)-\nabla f_i(x_\star)\|^2 \leq 2L_Q\left[P(x)-P(x_\star)\right].$$

Proof. Given any $i \in \{1,\ldots,n\}$, consider the function

$$\phi_i(x) = f_i(x) - f_i(x_\star) - \nabla f_i(x_\star)^T(x-x_\star).$$

It is straightforward to check that $\nabla\phi_i(x_\star)=0$, hence $\min_x \phi_i(x)=\phi_i(x_\star)=0$. Since $\nabla\phi_i(x)$ is Lipschitz continuous with constant L_i, we have (see, e.g., [Nes04, Theorem 2.1.5])

$$\frac{1}{2L_i}\|\nabla\phi_i(x)\|^2 \leq \phi_i(x) - \min_y \phi_i(y) = \phi_i(x)-\phi_i(x_\star) = \phi_i(x).$$

This implies

$$\|\nabla f_i(x)-\nabla f_i(x_\star)\|^2 \leq 2L_i\left[f_i(x)-f_i(x_\star)-\nabla f_i(x_\star)^T(x-x_\star)\right].$$

By dividing the above inequality by $1/(n^2 q_i)$, and summing over $i=1,\ldots,n$, we obtain

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{nq_i}\|\nabla f_i(x)-\nabla f_i(x_\star)\|^2 \leq 2L_Q\left[F(x)-F(x_\star)-\nabla F(x_\star)(x-x_\star)\right].$$

By the optimality of $x_\star$, i.e.,

$$x_\star = \arg\min_x P(x) = \arg\min_x \left\{F(x)+R(x)\right\},$$

there exist $\xi_\star \in \partial R(x_\star)$ such that $\nabla F(x_\star)+\xi_\star = 0$. Therefore

$$\begin{aligned} F(x)-F(x_\star)-\nabla F(x_\star)(x-x_\star) &= F(x)-F(x_\star)+\xi_\star(x-x_\star) \\ &\leq F(x)-F(x_\star)+R(x)-R(x_\star) \\ &= P(x)-P(x_\star), \end{aligned}$$

where in the last inequality, we used convexity of $R(x)$. This proves the desired result. □

Corollary 3. *Consider v_k defined in (13). Conditioned on x_{k-1}, we have $\mathbb{E}v_k = \nabla F(x_{k-1})$ and*

$$\mathbb{E}\|v_k-\nabla F(x_{k-1})\|^2 \leq 4L_Q\big[P(x_{k-1})-P(x_\star)+P(\tilde{x})-P(x_\star)\big].$$

Proof. Conditioned on x_{k-1}, we take expectation with respect to i_k to obtain

$$\mathbb{E}\left[\frac{1}{nq_{i_k}}\nabla f_{i_k}(x_{k-1})\right] = \sum_{i=1}^{n}\frac{q_i}{nq_i}\nabla f_i(x_{k-1}) = \sum_{i=1}^{n}\frac{1}{n}\nabla f_i(x_{k-1}) = \nabla F(x_{k-1}).$$

Similarly we have $\mathbb{E}\left[(1/(nq_{i_k}))\nabla f_{i_k}(\tilde{x})\right] = \nabla F(\tilde{x})$, and therefore

$$\mathbb{E}v_k = \mathbb{E}\left[\frac{1}{nq_{i_k}}\big(\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x})\big) + \nabla F(\tilde{x})\right] = \nabla F(x_{k-1}).$$

To bound the variance, we have

$$\begin{aligned}
\mathbb{E}\|v_k - \nabla F(x_{k-1})\|^2 &= \mathbb{E}\left\|\frac{1}{nq_{i_k}}\big(\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x})\big) + \nabla F(\tilde{x}) - \nabla F(x_{k-1})\right\|^2 \\
&= \mathbb{E}\frac{1}{(nq_{i_k})^2}\|\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x})\|^2 - \|\nabla F(x_{k-1}) - \nabla F(\tilde{x})\|^2 \\
&\leq \mathbb{E}\frac{1}{(nq_{i_k})^2}\|\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(\tilde{x})\|^2 \\
&\leq \mathbb{E}\frac{2}{(nq_{i_k})^2}\|\nabla f_{i_k}(x_{k-1}) - \nabla f_{i_k}(x_\star)\|^2 + \mathbb{E}\frac{2}{(nq_{i_k})^2}\|\nabla f_{i_k}(\tilde{x}) - \nabla f_{i_k}(x_\star)\|^2 \\
&= \frac{2}{n}\sum_{i=1}^{n}\frac{1}{nq_i}\|\nabla f_i(x_{k-1}) - \nabla f_i(x_\star)\|^2 + \frac{2}{n}\sum_{i=1}^{n}\frac{1}{nq_i}\|\nabla f_i(\tilde{x}) - \nabla f_i(x_\star)\|^2 \\
&\leq 4L_Q\big[P(x_{k-1}) - P(x_\star) + P(\tilde{x}) - P(x_\star)\big].
\end{aligned}$$

In the second equality above, we used the fact that for any random vector $\zeta \in \mathbb{R}^d$, it holds that $\mathbb{E}\|\zeta - \mathbb{E}\zeta\|^2 = \mathbb{E}\|\zeta\|^2 - \|\mathbb{E}\zeta\|^2$. In the second inequality, we used $\|a+b\|^2 \leq 2\|a\|^2 + 2\|b\|^2$. In the last inequality, we applied Lemma 1 twice. □

3.2 Proof of Theorem 1

For convenience, we define the *stochastic gradient mapping*

$$g_k = \frac{1}{\eta}(x_{k-1} - x_k) = \frac{1}{\eta}\left(x_{k-1} - \mathrm{prox}_{\eta R}(x_{k-1} - \eta v_k)\right),$$

so that the proximal gradient step (11) can be written as

$$x_k = x_{k-1} - \eta g_k. \tag{15}$$

We need the following lemmas in the convergence analysis. The first one is on the *non-expansiveness* of proximal mapping, which is well known (see, e.g., [Roc70, Section 31]).

Lemma 2. *Let R be a closed convex function on $\mathbb{R}^d$ and $x, y \in \mathrm{dom}(R)$. Then*

$$\big\|\mathrm{prox}_R(x) - \mathrm{prox}_R(y)\big\| \leq \|x - y\|.$$

The next lemma provides a lower bound of the function $P(x)$ using stochastic gradient mapping. It is a slight generalization of [HKP09, Lemma 3], and we give the proof in Appendix A for completeness.

Lemma 3. *Let $P(x) = F(x)+R(x)$, where $\nabla F(x)$ is Lipschitz continuous with parameter L, and $F(x)$ and $R(x)$ has strong convexity parameters μ_F and μ_R respectively. For any $x \in \mathrm{dom}(R)$ and arbitrary $v \in \mathbb{R}^d$, define*

$$\begin{aligned} x^+ &= \mathrm{prox}_{\eta R}(x - \eta v) \\ g &= \frac{1}{\eta}(x - x^+) \\ \Delta &= v - \nabla F(x), \end{aligned}$$

where η is a step size satisfying $0 < \eta \le 1/L$. Then we have for any $y \in \mathbb{R}^d$,

$$P(y) \ge P(x^+) + g^T(y - x) + \frac{\eta}{2}\|g\|^2 + \frac{\mu_F}{2}\|y - x\|^2 + \frac{\mu_R}{2}\|y - x^+\|^2 + \Delta^T(x^+ - y).$$

Now we proceed to prove Theorem 1. We start by analyzing how the distance between x_k and $x_\star$ changes in each iteration. Using the update rule (15), we have

$$\begin{aligned} \|x_k - x_\star\|^2 &= \|x_{k-1} - \eta g_k - x_\star\|^2 \\ &= \|x_{k-1} - x_\star\|^2 - 2\eta g_k^T(x_{k-1} - x_\star) + \eta^2\|g_k\|^2. \end{aligned}$$

Applying Lemma 3 with $x = x_{k-1}$, $v = v_k$, $x^+ = x_k$, $g = g_k$ and $y = x_\star$, we have

$$-g_k^T(x_{k-1} - x_\star) + \frac{\eta}{2}\|g_k\|^2 \le P(x_\star) - P(x_k) - \frac{\mu_F}{2}\|x_{k-1} - x_\star\|^2 - \frac{\mu_R}{2}\|x_k - x_\star\|^2 - \Delta_k^T(x_k - x_\star),$$

where $\Delta_k = v_k - \nabla F(x_{k-1})$. Note that the assumption in Theorem 1 implies $\eta < 1/(4L_Q) < 1/L$ because $L_Q \ge (1/n)\sum_{i=1}^n L_i \ge L$. Therefore,

$$\begin{aligned} \|x_k - x_\star\|^2 &\le \|x_{k-1} - x_\star\|^2 - \eta\mu_F\|x_{k-1} - x_\star\|^2 - \eta\mu_R\|x_k - x_\star\|^2 \\ &\quad -2\eta[P(x_k) - P(x_\star)] - 2\eta\Delta_k^T(x_k - x_\star) \\ &\le \|x_{k-1} - x_\star\|^2 - 2\eta[P(x_k) - P(x_\star)] - 2\eta\Delta_k^T(x_k - x_\star) \qquad (16) \end{aligned}$$

Next we upper bound the quantity $-2\eta\Delta_k^T(x_k - x_\star)$. Although not used in the Prox-SVRG algorithm, we can still define the proximal full gradient update as

$$\bar{x}_k = \mathrm{prox}_{\eta R}(x_{k-1} - \eta\nabla F(x_{k-1})),$$

which is independent of the random variable i_k. Then,

$$\begin{aligned} -2\eta\Delta_k^T(x_k - x_\star) &= -2\eta\Delta_k^T(x_k - \bar{x}_k) - 2\eta\Delta_k^T(\bar{x}_k - x_\star) \\ &\le 2\eta\|\Delta_k\|\|x_k - \bar{x}_k\| - 2\eta\Delta_k^T(\bar{x}_k - x_\star) \\ &\le 2\eta\|\Delta_k\|\,\big\|(x_{k-1} - \eta v_k) - \big(x_{k-1} - \eta\nabla F(x_{k-1})\big)\big\| - 2\eta\Delta_k^T(\bar{x}_k - x_\star) \\ &= 2\eta^2\|\Delta_k\|^2 - 2\eta\Delta_k^T(\bar{x}_k - x_\star), \end{aligned}$$

where in the first inequality we used the Cauchy-Schwarz inequality, and in the second inequality we used Lemma 2. Combining with (16), we get

$$\|x_k - x_\star\|^2 \leq \|x_{k-1} - x_\star\|^2 - 2\eta[P(x_k) - P(x_\star)] + 2\eta^2\|\Delta_k\|^2 - 2\eta\Delta_k^T(\bar{x}_k - x_\star).$$

Now we take expectation on both sides of the above inequality with respect to i_k to obtain

$$\mathbb{E}\|x_k - x_\star\|^2 \leq \|x_{k-1} - x_\star\|^2 - 2\eta[\,\mathbb{E}P(x_k) - P(x_\star)] + 2\eta^2\,\mathbb{E}\|\Delta_k\|^2 - 2\eta\,\mathbb{E}[\Delta_k^T(\bar{x}_k - x_\star)].$$

We note that both $\bar{x}_k$ and $x_\star$ are independent of the random variable i_k and $\mathbb{E}\Delta_k = 0$, so

$$\mathbb{E}[\Delta_k^T(\bar{x}_k - x_\star)] = (\mathbb{E}\Delta_k)^T(\bar{x}_k - x_\star) = 0.$$

In addition, we can bound the term $\mathbb{E}\|\Delta_k\|^2$ using Corollary 3 to obtain

$$\mathbb{E}\|x_k - x_\star\|^2 \leq \|x_{k-1} - x_\star\|^2 - 2\eta[\,\mathbb{E}P(x_k) - P(x_\star)] + 8L_Q\eta^2[P(x_{k-1}) - P(x_\star) + P(\tilde{x}) - P(x_\star)].$$

We consider a fixed stage s, so that $x_0 = \tilde{x} = \tilde{x}_{s-1}$ and $\tilde{x}_s = \frac{1}{m}\sum_{k=1}^m x_k$. By summing the previous inequality over $k = 1, \ldots, m$ and taking expectation with respect to the history of random variables $i_1, \ldots, i_m$, we obtain

$$\begin{aligned} & \mathbb{E}\|x_m - x_\star\|^2 + 2\eta[\,\mathbb{E}P(x_m) - P(x_\star)] + 2\eta(1 - 4L_Q\eta)\sum_{k=1}^{m-1}[\,\mathbb{E}P(x_k) - P(x_\star)] \\ \leq\; & \|x_0 - x_\star\|^2 + 8L_Q\eta^2\big[P(x_0) - P(x_\star) + m(P(\tilde{x}) - P(x_\star))\big]. \end{aligned}$$

Notice that $2\eta(1 - 4L_Q\eta) < 2\eta$ and $x_0 = \tilde{x}$, so we have

$$2\eta(1 - 4L_Q\eta)\sum_{k=1}^{m}[\,\mathbb{E}P(x_k) - P(x_\star)] \leq \|\tilde{x} - x_\star\|^2 + 8L_Q\eta^2(m+1)[P(\tilde{x}) - P(x_\star)].$$

By convexity of P and definition of $\tilde{x}_s$, we have $P(\tilde{x}_s) \leq \frac{1}{m}\sum_{t=1}^m P(x_k)$. Moreover, strong convexity of P implies $\|\tilde{x} - x_\star\|^2 \leq \frac{2}{\mu}[P(\tilde{x}) - P(\star)]$. Therefore, we have

$$2\eta(1 - 4L_Q\eta)m[\,\mathbb{E}P(\tilde{x}_s) - P(x_\star)] \leq \left(\frac{2}{\mu} + 8L_Q\eta^2(m+1)\right)[P(\tilde{x}_{s-1}) - P(x_\star)].$$

Divide both sides of the above inequality by $2\eta(1 - 4L_Q\eta)m$, we arrive at

$$\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq \left(\frac{1}{\mu\eta(1 - 4L_Q\eta)m} + \frac{4L_Q\eta(m+1)}{(1 - 4L_Q\eta)m}\right)[P(\tilde{x}_{s-1}) - P(x_\star)].$$

Finally using the definition of ρ in (14), and applying the above inequality recursively, we obtain

$$\mathbb{E}P(\tilde{x}_s) - P(x_\star) \leq \rho^s[P(\tilde{x}_0) - P(x_\star)],$$

which is the desired result.

data sets	n	d	source	λ_2	λ_1
`rcv1`	20,242	47,236	[LYRL04]	10^{-4}	10^{-5}
`covertype`	581,012	54	[BDA13]	10^{-5}	10^{-4}
`sido0`	12,678	4,932	[Guy08]	10^{-4}	10^{-4}

Table 1: Summary of data sets and regularization parameters used in our experiments.

4 Numerical experiments

In this section we present results of several numerical experiments to illustrate the properties of the Prox-SVRG method, and compare its performance with several related algorithms.

We focus on the regularized logistic regression problem for binary classification: given a set of training examples $(a_1, b_1), \ldots, (a_n, b_n)$ where $a_i \in \mathbb{R}^d$ and $b_i \in \{+1, -1\}$, we find the optimal predictor $x \in \mathbb{R}^d$ by solving

$$\underset{x\in\mathbb{R}^d}{\text{minimize}} \quad \frac{1}{n}\sum_{i=1}^{n} \log\bigl(1+\exp(-b_i a_i^T x)\bigr) + \frac{\lambda_2}{2}\|x\|_2^2 + \lambda_1\|x\|_1,$$

where λ_2 and λ_1 are two regularization parameters. The ℓ_1 regularization is added to promote sparse solutions. In terms of the model (1) and (2), we can have either

$$f_i(x) = \log(1+\exp(-b_i a_i^T x)) + (\lambda_2/2)\|x\|_2^2, \qquad R(x) = \lambda_1\|x\|_1, \tag{17}$$

or

$$f_i(x) = \log(1+\exp(-b_i a_i^T x)), \qquad R(x) = (\lambda_2/2)\|x\|_2^2 + \lambda_1\|x\|_1, \tag{18}$$

depending on the algorithm used.

We used three publicly available data sets. Their sizes n, dimensions d as well as sources as listed in Table 1. For `rcv1` and `covertype`, we used the processed data for binary classification from [FL11]. The table also listed the values of λ_2 and λ_1 that were used in our experiments. These choices are typical in machine learning benchmarks to obtain good classification performance.

4.1 Properties of Prox-SVRG

We first illustrate the numerical characteristics of Prox-SVRG on the `rcv1` dataset. Each example in this dataset has been normalized so that $\|a_i\|_2 = 1$ for all $i = 1, \ldots, n$, which leads to the same upper bound on the Lipschitz constants $L = L_i = \|a_i\|_2^2/4$. In our implementation, we used the splitting in (17) and uniform sampling of the component functions. We choose the number of stochastic gradient steps m between full gradient evaluations as a small multiple of n.

Figure 2 shows the behavior of Prox-SVRG with $m = 2n$ when we used three different step sizes. The horizontal axis is the number of effective passes over the data, where each effective pass evaluates n component gradients. Each full gradient evaluation counts as one

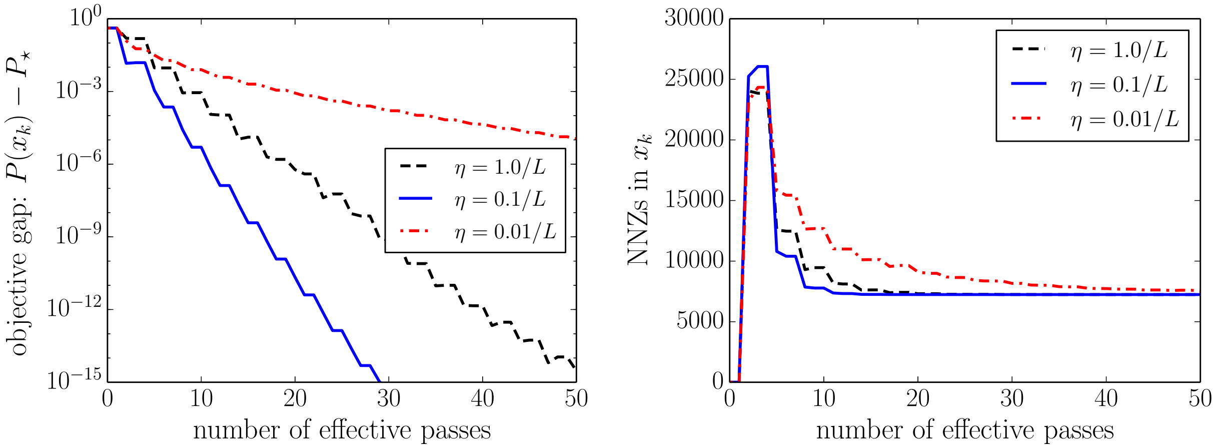

Figure 2: Prox-SVRG on the `rcv1` dataset: varying the step size η with $m = 2n$.

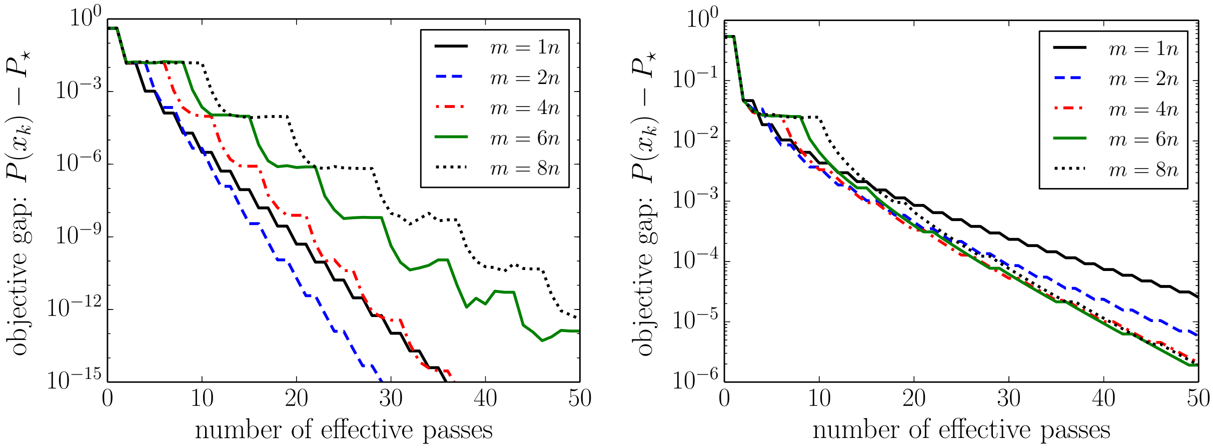

Figure 3: Prox-SVRG on the `rcv1` dataset with step size $\eta = 0.1/L$: varying the period m between full gradient evaluations, with $\lambda_2 = 10^{-4}$ on the left and $\lambda_2 = 10^{-5}$ on the right.

effective pass, and appears as a small flat segment of length 1 on the curves. It can be seen that the convergence of Prox-SVRG becomes slow if the step size is either too big or too small. The best choice of $\eta = 0.1/L$ matches our theoretical analysis (see the first remark after Theorem 1). The number of non-zeros (NNZs) in the iterates x_k converges quickly to 7237 after about 10 passes over the data.

Figure 3 shows how the objective gap $P(x_k) - P_\star$ decreases when we vary the period m of evaluating full gradients. For $\lambda_2 = 10^{-4}$, the fastest convergence per stage is achieved by $m = 1$, but the frequent evaluation of full gradients makes its overall performance slightly worse than $m = 2$. Longer periods leads to slower convergence, due to the lack of effective variance reduction. For $\lambda_2 = 10^{-5}$, the condition number is much larger, thus longer period m is required to have sufficient reduction during each stage.

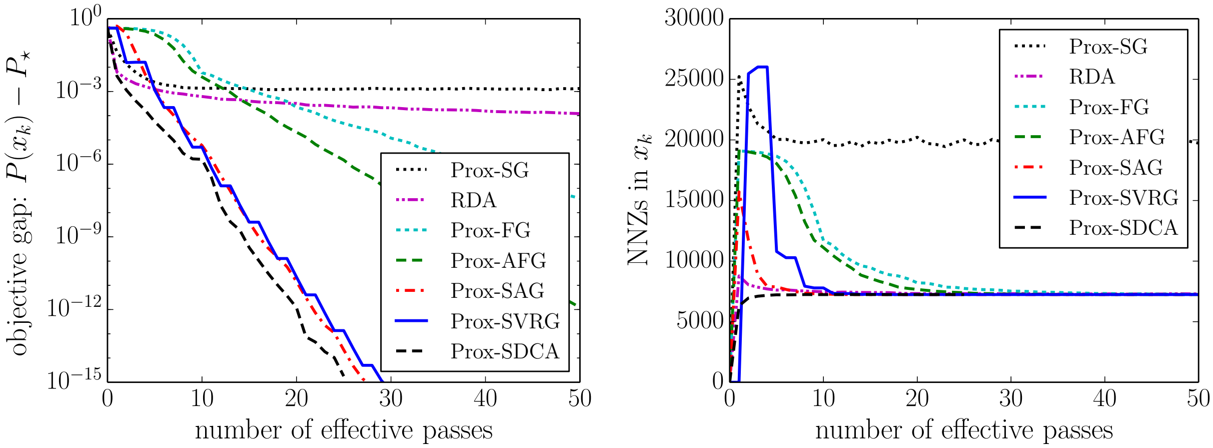

Figure 4: Comparison of different methods on the `rcv1` dataset.

4.2 Comparison with related algorithms

We implemented the following algorithms to compare with Prox-SVRG:

- Prox-SG: the proximal stochastic gradient method given in (6). We used a constant step size that gave the best performance among all powers of 10.
- RDA: the regularized dual averaging method in [Xia10]. The step size parameter γ in RDA is also chosen as the one that gave best performance among all powers of 10.
- Prox-FG: the proximal full gradient method given in (5), with an adaptive line search scheme proposed in [Nes13].
- Prox-AFG: an accelerated version of the Prox-FG method that is very similar to FISTA [BT09], also with an adaptive line search scheme.
- Prox-SAG: a proximal version of the stochastic average gradient (SAG) method [SRB13, Section 6]. We note that the convergence of this Prox-SAG method has not been established for the general model considered in this paper. Nevertheless it demonstrates good performance in practice.
- Prox-SDCA: the proximal stochastic dual coordinate ascent method [SSZ12]. In order to obtain the complexity $O\left((n + L/\mu)\log(1/\epsilon)\right)$, it needs to use the splitting (18).

Figure 4 shows the comparison of Prox-SVRG ($m = 2n$ and $\eta = 0.1/L$) with different methods described above on the `rcv1` dataset. For the Prox-SAG method, we used the same step size $\eta = 0.1/L$ as for Prox-SVRG. We can see that the three methods that performed best are Prox-SAG, Prox-SVRG and Prox-SDCA. The superior performance of Prox-SVRG and Prox-SDCA are predicted by their low complexity analysis. While the complexity of Prox-SAG has not been formally established, its performance is among the best. In terms of obtaining sparse iterates under the ℓ_1-regularization, RDA, Prox-SDCA and Prox-SAG

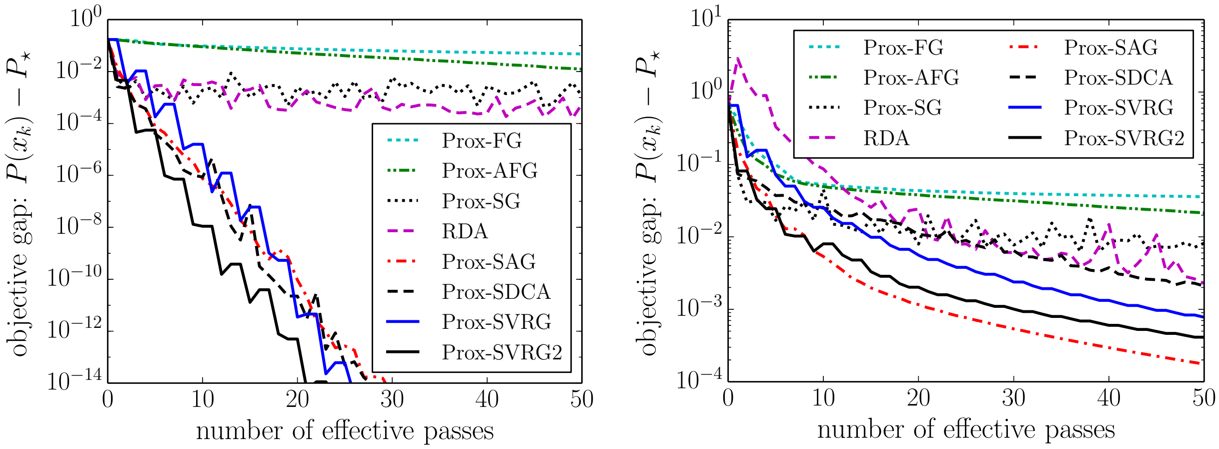

Figure 5: Comparison of different methods on `covertype` (left) and `sido0` (right).

converged to the correct NNZs quickly, followed by Prox-SVRG and the two full gradient methods. The Prox-SG method didn't converge to the correct NNZs.

Figure 5 shows the comparison of different methods on two other data sets listed in Table 1. Here we also included comparison with Prox-SVRG2, which is a hybrid method by performing Prox-SG for one pass over the data and then switch to Prox-SVRG. This hybrid scheme was suggested in [JZ13], and it often improves the performance of Prox-SVRG substantially. Similar hybrid schemes also exist for SDCA [SSZ12] and SAG [SRB13].

The behaviors of the stochastic gradient type of algorithms on `covertype` (Figure 5, left) are similar to those on `rcv1`, but the two full gradient methods Prox-FG and Prox-AFG perform worse because of the smaller regularization parameter λ_2 and hence worse condition number. The `sido0` data set turns out to be more difficult to optimize, and much slower convergence are observed in Figure 5 (right). The Prox-SAG method performs best on this data set, followed by Prox-SVRG2 and Prox-SVRG.

5 Conclusions

We developed a new proximal stochastic gradient method, called Prox-SVRG, for minimizing the sum of two convex functions: one is the average of a large number of smooth component functions, and the other is a general convex function that admits a simple proximal mapping. This method exploits the finite average structure of the smooth part by extending the variance reduction technique of SVRG [JZ13], which computes the full gradient periodically to modify the stochastic gradients in order to reduce their variance.

The Prox-SVRG method enjoys the same low complexity as that of SDCA [SSZ13, SSZ12] and SAG [RSB12, SRB13], but applies to a more general class of problems, and does not require the storage of the most recent gradient for each component function. In addition, our method incorporates a weighted sampling scheme, which achieves an improved complexity result for problems where the component functions vary substantially in smoothness.

A Proof of Lemma 3

We can write the proximal update $x^+ = \mathrm{prox}_{\eta R}(x - \eta v)$ more explicitly as

$$x^+ = \arg\min_y \left\{ \frac{1}{2}\|y - (x - \eta v)\|^2 + \eta R(y) \right\}.$$

The associated optimality condition states that there is a $\xi \in \partial R(x^+)$ such that

$$x^+ - (x - \eta v) + \eta \xi = 0.$$

Combining with the definition of $g = (x - x^+)/\eta$, we have $\xi = g - v$.

By strong convexity of F and R, we have for any $x \in \mathrm{dom}(R)$ and $y \in \mathbb{R}^d$,

$$\begin{aligned} P(y) &= F(y) + R(y) \\ &\geq F(x) + \nabla F(x)^T(y - x) + \frac{\mu_F}{2}\|y - x\|^2 + R(x^+) + \xi^T(y - x^+) + \frac{\mu_R}{2}\|y - x^+\|^2. \end{aligned}$$

By smoothness of F, we can further lower bound $F(x)$ by

$$F(x) \geq F(x^+) - \nabla F(x)^T(x^+ - x) - \frac{L}{2}\|x^+ - x\|^2.$$

Therefore,

$$\begin{aligned} P(y) &\geq F(x^+) - \nabla F(x)^T(x^+ - x) - \frac{L}{2}\|x^+ - x\|^2 \\ &\quad + \nabla F(x)^T(y - x) + \frac{\mu_F}{2}\|y - x\|^2 + R(x^+) + \xi^T(y - x^+) + \frac{\mu_R}{2}\|y - x^+\|^2 \\ &= P(x^+) - \nabla F(x)^T(x^+ - x) - \frac{L\eta^2}{2}\|g\|^2 \\ &\quad + \nabla F(x)^T(y - x) + \frac{\mu_F}{2}\|y - x\|^2 + \xi^T(y - x^+) + \frac{\mu_R}{2}\|y - x^+\|^2, \end{aligned}$$

where in the last equality we used $P(x^+) = F(x^+) + R(x^+)$ and $x^+ - x = -\eta g$. Collecting all inner products on the right-hand side, we have

$$\begin{aligned} & -\nabla F(x)^T(x^+ - x) + \nabla F(x)^T(y - x) + \xi^T(y - x^+) \\ =\; & \nabla F(x)^T(y - x^+) + (g - v)^T(y - x^+) \\ =\; & g^T(y - x^+) + (v - \nabla F(x))^T(x^+ - y) \\ =\; & g^T(y - x + x - x^+) + \Delta^T(x^+ - y) \\ =\; & g^T(y - x) + \eta\|g\|^2 + \Delta^T(x^+ - y), \end{aligned}$$

where in the first equality we used $\xi = g - v$, in the third equality we used $\Delta = v - \nabla F(x)$, and in the last equality we used $x - x^+ = \eta g$. Putting everything together, we obtain

$$P(y) \geq P(x^+) + g^T(y - x) + \frac{\eta}{2}(2 - L\eta)\|g\|^2 + \frac{\mu_F}{2}\|y - x\|^2 + \frac{\mu_R}{2}\|y - x^+\|^2 + \Delta^T(x^+ - y).$$

Finally using the assumption $0 < \eta \leq 1/L$, we arrive at the desired result.

B Convergence analysis of the Prox-FG method

Here we prove the convergence rate in (7) for the Prox-FG method (5). First we define the full gradient mapping $G_k = (x_k - x_{k-1})/\eta$ and use it to obtain

$$\begin{aligned} \|x_k - x_\star\|^2 &= \|x_{k-1} - x_\star - \eta G_k\|^2 \\ &= \|x_{k-1} - x_\star\|^2 - 2\eta G_k^T(x_{k-1} - x_\star) + \eta^2\|G_k\|^2. \end{aligned}$$

Applying Lemma 3 with $x = x_{k-1}$, $v = \nabla F(x_{k-1})$, $x^+ = x_k$, $g = G_k$ and $y = x_\star$, we have $\Delta = 0$ and

$$-G_k^T(x_{k-1} - x_\star) + \frac{\eta}{2}\|G_k\|^2 \le P(x_\star) - P(x_k) - \frac{\mu_F}{2}\|x_{k-1} - x_\star\|^2 - \frac{\mu_R}{2}\|x_k - x_\star\|^2.$$

Therefore,

$$\|x_k - x_\star\|^2 \le \|x_{k-1} - x_\star\|^2 + 2\eta\left(F(x_\star) - F(x_k) - \frac{\mu_F}{2}\|x_{k-1} - x_\star\|^2 - \frac{\mu_R}{2}\|x_k - x_\star\|^2\right).$$

Rearranging terms in the above inequality yields

$$2\eta\big(F(x_k) - F(x_\star)\big) + (1 + \eta\mu_R)\|x_k - x_\star\|^2 \le (1 - \eta\mu_F)\|x_{k-1} - x_\star\|^2. \tag{19}$$

Dropping the nonnegative term $2\eta\big(F(x_k) - F(x_\star)\big)$ on the left-hand side results in

$$\|x_k - x_\star\|^2 \le \frac{1 - \eta\mu_F}{1 + \eta\mu_R}\|x_{k-1} - x_\star\|^2,$$

which leads to

$$\|x_k - x_\star\|^2 \le \left(\frac{1 - \eta\mu_F}{1 + \eta\mu_R}\right)^k \|x_0 - x_\star\|^2.$$

Dropping the nonnegative term $(1 + \eta\mu_R)\|x_k - x_\star\|^2$ on the left-hand side of (19) yields

$$F(x_k) - F(x_\star) \le \frac{1 - \eta\mu_F}{2\eta}\|x_{k-1} - x_\star\|^2 \le \frac{1 + \eta\mu_R}{2\eta}\left(\frac{1 - \eta\mu_F}{1 + \eta\mu_R}\right)^k \|x_0 - x_\star\|^2.$$

Setting $\eta = 1/L$, the above inequality is equivalent to (7).

References

[BCNW12] R. H. Byrd, G. M. Chin, J. Nocedal, and Y. Wu. Sample size selection in optimization methods for machine learning. *Mathematical Programming, Ser. B*, 134:127–155, 2012.

[BDA13] J. A. Blackard, D. J. Dean, and C. W. Anderson. Covertype data set. In K. Bache and M. Lichman, editors, *UCI Machine Learning Repository*, URL: http://archive.ics.uci.edu/ml, 2013. University of California, Irvine, School of Information and Computer Sciences.

[Ber10] D. P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: a survey. Report LIDS-P-2848, Laboratory for Information and Decision Systems, MIT, Cambridge, MA, 2010.

[Ber11] D. P. Bertsekas. Incremental proximal methods for large scale convex optimization. *Mathematical Programming, Ser. B*, 129:163–195, 2011.

[BHG07] D. Blatt, A. O. Hero, and H. Gauchman. A convergent incremental gradient method with a constant step size. *SIAM Journal on Optimization*, 18(1):29–51, 2007.

[BT09] A. Beck and M. Teboulle. A fast iterative shrinkage-threshold algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[CR97] G. H.-G. Chen and R. T. Rockafellar. Convergence rates in forward-backward splitting. *SIAM Journal on Optimization*, 7(2):421–444, 1997.

[DS09] J. Duchi and Y. Singer. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2873–2898, 2009.

[FG13] M. P. Friedlander and G. Goh. Tail bounds for stochastic approximation. arXiv:1304.5586, April 2013.

[FL11] R.-E. Fan and C.-J. Lin. LIBSVM data: Classification, regression and multi-label. URL: http://www.csie.ntu.edu.tw/˜cjlin/libsvmtools/datasets, 2011.

[FS12] M. P. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing*, 34(3):13801405, 2012.

[Guy08] I. Guyon. Sido: A phamacology dataset. URL: http://www.causality.inf.ethz.ch/data/SIDO.html, 2008.

[HKP09] C. Hu, J. T. Kwok, and W. Pan. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems 22*, pages 781–789. 2009.

[HTF09] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, New York, 2nd edition, 2009.

[JZ13] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems 26*, pages 315–323. 2013.

[KR13] J. Konečný and P. Richtárik. Semi-stochastic gradient descent methods. arXiv:1312.1666, 2013.

[LLZ09] J. Langford, L. Li, and T. Zhang. Sparse online learning via truncated gradient. *Journal of Machine Learning Research*, 10:777–801, 2009.

[LM79] P.-L. Lions and B. Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16:964–979, 1979.

[LYRL04] D. D. Lewis, Y. Yang, T. Rose, and F. Li. RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, 5:361–397, 2004.

[MZJ13] M. Mahdavi, L. Zhang, and R. Jin. Mixed optimization for smooth functions. In *Advances in Neural Information Processing Systems 26*, pages 674–682. 2013.

[Nes04] Yu. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer, Boston, 2004.

[Nes13] Yu. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming, Ser. B*, 140:125–161, 2013.

[Roc70] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[RSB12] N. Le Roux, M. Schmidt, and F. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Advances in Neural Information Processing Systems 25*, pages 2672–2680. 2012.

[SRB13] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. Technical Report HAL 00860051, INRIA, Paris, France, 2013.

[SSZ12] S. Shalev-Shwartz and T. Zhang. Proximal stochatic dual coordinate ascent. arXiv:1211.2772, November 2012.

[SSZ13] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14:567–599, 2013.

[Tse00] P. Tseng. A modified forward-backward splitting method for maximal monotone mappings. *SIAM Journal on Control and Optimization*, 38(2):431–446, 2000.

[Xia10] L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2534–2596, 2010.

[ZMJ13] L. Zhang, M. Mahdavi, and R. Jin. Linear convergence with condition number independent access of full gradients. In *Advances in Neural Information Processing Systems 26*, pages 980–988. 2013.