SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient

Lam M. Nguyen [1] **Jie Liu** [1] **Katya Scheinberg** [1 2] **Martin Takáč** [1]

Abstract

In this paper, we propose a StochAstic Recursive grAdient algoritHm (SARAH), as well as its practical variant SARAH+, as a novel approach to the finite-sum minimization problems. Different from the vanilla SGD and other modern stochastic methods such as SVRG, S2GD, SAG and SAGA, SARAH admits a simple recursive framework for updating stochastic gradient estimates; when comparing to SAG/SAGA, SARAH does not require a storage of past gradients. The linear convergence rate of SARAH is proven under strong convexity assumption. We also prove a linear convergence rate (in the strongly convex case) for an inner loop of SARAH, the property that SVRG does not possess. Numerical experiments demonstrate the efficiency of our algorithm.

1. Introduction

We are interested in solving a problem of the form

$$\min_{w\in\mathbb{R}^d} \left\{ P(w) \stackrel{\text{def}}{=} \frac{1}{n}\sum_{i\in[n]} f_i(w) \right\}, \qquad (1)$$

where each f_i, $i \in [n] \stackrel{\text{def}}{=} \{1,\ldots,n\}$, is convex with a Lipschitz continuous gradient. Throughout the paper, we assume that there exists an optimal solution w^* of (1).

[1]Department of Industrial and Systems Engineering, Lehigh University, USA. [2]On leave at The University of Oxford, UK. All authors were supported by NSF Grant CCF-1618717. Katya Scheinberg was partially supported by NSF Grants DMS 13-19356, CCF-1320137 and CCF-1618717. Correspondence to: Lam M. Nguyen <lamnguyen.mltd@gmail.com>, Jie Liu <jie.liu.2018@gmail.com>, Katya Scheinberg <katyas@lehigh.edu>, Martin Takáč <Takac.MT@gmail.com>.

Proceedings of the 34th International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

Problems of this type arise frequently in supervised learning applications (Hastie et al., 2009). Given a training set $\{(x_i, y_i)\}_{i=1}^n$ with $x_i \in \mathbb{R}^d, y_i \in \mathbb{R}$, the least squares regression model, for example, is written as (1) with $f_i(w) \stackrel{\text{def}}{=} (x_i^T w - y_i)^2 + \frac{\lambda}{2}\|w\|^2$, where $\|\cdot\|$ denotes the ℓ_2-norm. The ℓ_2-regularized logistic regression for binary classification is written with $f_i(w) \stackrel{\text{def}}{=} \log(1+\exp(-y_i x_i^T w)) + \frac{\lambda}{2}\|w\|^2$ $(y_i \in \{-1,1\})$.

In recent years, many advanced optimization methods have been developed for problem (1). While the objective function is smooth and convex, the traditional optimization methods, such as gradient descent (GD) or Newton method are often impractical for this problem, when n – the number of training samples and hence the number of f_i's – is very large. In particular, GD updates iterates as follows

$$w_{t+1} = w_t - \eta_t \nabla P(w_t), \quad t = 0,1,2,\ldots.$$

Under strong convexity assumption on P and with appropriate choice of η_t, GD converges at a linear rate in terms of objective function values $P(w_t)$. However, when n is large, computing $\nabla P(w_t)$ at each iteration can be prohibitive.

As an alternative, stochastic gradient descent (SGD)[1], originating from the seminal work of Robbins and Monro in 1951 (Robbins & Monro, 1951), has become the method of choice for solving (1). At each step, SGD picks an index $i \in [n]$ uniformly at random, and updates the iterate as $w_{t+1} = w_t - \eta_t \nabla f_i(w_t)$, which is up-to n times cheaper than an iteration of a full gradient method. The convergence rate of SGD is slower than that of GD, in particular, it is sublinear in the strongly convex case. The tradeoff, however, is advantageous due to the tremendous per-iteration savings and the fact that low accuracy solutions are sufficient. This trade-off has been thoroughly analyzed in (Bottou, 1998). Unfortunately, in practice SGD method is often too slow and its performance is too sensitive to the variance in the sample gradients $\nabla f_i(w_t)$. Use of mini-batches (averaging multiple sample gradients $\nabla f_i(w_t)$) was used in (Shalev-Shwartz et al., 2007; Cotter et al., 2011; Takáč

[1]We mark here that even though stochastic gradient is referred to as SG in literature, the term stochastic gradient descent (SGD) has been widely used in many important works of large-scale learning, including SAG/SAGA, SDCA, SVRG and MISO.

Table 1: Comparisons between different algorithms for strongly convex functions. $\kappa = L/\mu$ is the condition number.

Method	Complexity	Fixed Learning Rate	Low Storage Cost
GD	$\mathcal{O}(n\kappa \log(1/\epsilon))$	✓	✓
SGD	$\mathcal{O}(1/\epsilon)$	✗	✓
SVRG	$\mathcal{O}((n+\kappa)\log(1/\epsilon))$	✓	✓
SAG/SAGA	$\mathcal{O}((n+\kappa)\log(1/\epsilon))$	✓	✗
SARAH	$\mathcal{O}((n+\kappa)\log(1/\epsilon))$	✓	✓

Table 2: Comparisons between different algorithms for convex functions.

Method	Complexity
GD	$\mathcal{O}(n/\epsilon)$
SGD	$\mathcal{O}(1/\epsilon^2)$
SVRG	$\mathcal{O}(n + (\sqrt{n}/\epsilon))$
SAGA	$\mathcal{O}(n + (n/\epsilon))$
SARAH	$\mathcal{O}((n + (1/\epsilon))\log(1/\epsilon))$
SARAH (one outer loop)	$\mathcal{O}(n + (1/\epsilon^2))$

et al., 2013) to reduce the variance and improve convergence rate by constant factors. Using diminishing sequence $\{\eta_t\}$ is used to control the variance (Shalev-Shwartz et al., 2011; Bottou et al., 2016), but the practical convergence of SGD is known to be very sensitive to the choice of this sequence, which needs to be hand-picked.

Recently, a class of more sophisticated algorithms have emerged, which use the specific finite-sum form of (1) and combine some deterministic and stochastic aspects to reduce variance of the steps. The examples of these methods are SAG/SAGA (Le Roux et al., 2012; Defazio et al., 2014), SDCA (Shalev-Shwartz & Zhang, 2013), SVRG (Johnson & Zhang, 2013; Xiao & Zhang, 2014), DIAG (Mokhtari et al., 2017), MISO (Mairal, 2013) and S2GD (Konečný & Richtárik, 2013), all of which enjoy faster convergence rate than that of SGD and use a fixed learning rate parameter η. In this paper we introduce a new method in this category, SARAH, which further improves several aspects of the existing methods. In Table 1 we summarize complexity and some other properties of the existing methods and SARAH when applied to strongly convex problems. Although SVRG and SARAH have the same convergence rate, we introduce a practical variant of SARAH that outperforms SVRG in our experiments.

In addition, theoretical results for complexity of the methods or their variants when applied to general convex functions have been derived (Schmidt et al., 2016; Defazio et al., 2014; Reddi et al., 2016; Allen-Zhu & Yuan, 2016; Allen-Zhu, 2017). In Table 2 we summarize the key complexity results, noting that convergence rate is now sublinear.

Our Contributions. In this paper, we propose a novel algorithm which combines some of the good properties of existing algorithms, such as SAGA and SVRG, while aiming to improve on both of these methods. In particular, our algorithm does not take steps along a stochastic gradient direction, but rather along an accumulated direction using past stochastic gradient information (as in SAGA) and occasional exact gradient information (as in SVRG). We summarize the key properties of the proposed algorithm below.

- Similarly to SVRG, SARAH's iterations are divided into the outer loop where a full gradient is computed and the inner loop where only stochastic gradient is computed. Unlike the case of SVRG, the steps of the inner loop of SARAH are based on accumulated stochastic information.
- Like SAG/SAGA and SVRG, SARAH has a sublinear rate of convergence for general convex functions, and a linear rate of convergence for strongly convex functions.
- SARAH uses a constant learning rate, whose size is larger than that of SVRG. We analyze and discuss the optimal choice of the learning rate and the number of inner loop steps. However, unlike SAG/SAGA but similar to SVRG, SARAH does not require a storage of n past stochastic gradients.
- We also prove a linear convergence rate (in the strongly convex case) for the inner loop of SARAH, the property that SVRG does not possess. We show that the variance of the steps inside the inner loop goes to zero, thus SARAH is theoretically more stable and reliable than SVRG.
- We provide a practical variant of SARAH based on the convergence properties of the inner loop, where the simple stable stopping criterion for the inner loop is used (see Section 4 for more details). This variant shows how SARAH can be made more stable than SVRG in practice.

2. Stochastic Recursive Gradient Algorithm

Now we are ready to present our SARAH (Algorithm 1).

The key step of the algorithm is a recursive update of the stochastic gradient estimate *(SARAH update)*

$$v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}, \tag{2}$$

followed by the iterate update:

$$w_{t+1} = w_t - \eta v_t. \tag{3}$$

For comparison, SVRG update can be written in a similar way as

$$v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_0) + v_0. \tag{4}$$

Algorithm 1 SARAH

Parameters: the learning rate $\eta > 0$ and the inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \ldots$ **do**
  $w_0 = \tilde{w}_{s-1}$
  $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 = w_0 - \eta v_0$
  Iterate:
  for $t = 1, \ldots, m-1$ **do**
    Sample i_t uniformly at random from $[n]$
    $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} = w_t - \eta v_t$
  end for
  Set $\tilde{w}_s = w_t$ with t chosen uniformly at random from $\{0, 1, \ldots, m\}$
end for

Observe that in SVRG, v_t is an unbiased estimator of the gradient, while it is not true for SARAH. Specifically, [2]

$$\mathbb{E}[v_t|\mathcal{F}_t] = \nabla P(w_t) - \nabla P(w_{t-1}) + v_{t-1} \neq \nabla P(w_t), \quad (5)$$

where [3] $\mathcal{F}_t = \sigma(w_0, i_1, i_2, \ldots, i_{t-1})$ is the σ-algebra generated by $w_0, i_1, i_2, \ldots, i_{t-1}$; $\mathcal{F}_0 = \mathcal{F}_1 = \sigma(w_0)$. Hence, SARAH is different from SGD and SVRG type of methods, however, the following total expectation holds, $\mathbb{E}[v_t] = \mathbb{E}[\nabla P(w_t)]$, differentiating SARAH from SAG/SAGA.

SARAH is similar to SVRG since they both contain outer loops which require one full gradient evaluation per outer iteration followed by one full gradient descent step with a given learning rate. The difference lies in the inner loop, where SARAH updates the stochastic step direction v_t recursively by adding and subtracting component gradients to and from the previous v_{t-1} ($t \geq 1$) in (2). Each inner iteration evaluates 2 stochastic gradients and hence the total work per outer iteration is $\mathcal{O}(n+m)$ in terms of the number of gradient evaluations. Note that due to its nature, without running the inner loop, i.e., $m = 1$, SARAH reduces to the GD algorithm.

3. Theoretical Analysis

To proceed with the analysis of the proposed algorithm, we will make the following common assumptions.

Assumption 1 (L-smooth)**.** *Each $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is L-smooth, i.e., there exists a constant $L > 0$ such that*

$$\|\nabla f_i(w) - \nabla f_i(w')\| \leq L\|w - w'\|, \ \forall w, w' \in \mathbb{R}^d.$$

[2] $\mathbb{E}[\cdot|\mathcal{F}_t] = \mathbb{E}_{i_t}[\cdot]$, which is expectation with respect to the random choice of index i_t (conditioned on $w_0, i_1, i_2, \ldots, i_{t-1}$).

[3] $\mathcal{F}_t$ also contains all the information of $w_0, \ldots, w_t$ as well as $v_0, \ldots, v_{t-1}$.

Note that this assumption implies that $P(w) = \frac{1}{n}\sum_{i=1}^{n} f_i(w)$ is also *L-smooth*. The following strong convexity assumption will be made for the appropriate parts of the analysis, otherwise, it would be dropped.

Assumption 2a (μ-strongly convex)**.** *The function $P : \mathbb{R}^d \to \mathbb{R}$, is μ-strongly convex, i.e., there exists a constant $\mu > 0$ such that $\forall w, w' \in \mathbb{R}^d$,*

$$P(w) \geq P(w') + \nabla P(w')^T(w - w') + \tfrac{\mu}{2}\|w - w'\|^2.$$

Another, stronger, assumption of μ-strong convexity for (1) will also be imposed when required in our analysis. Note that Assumption 2b implies Assumption 2a but not vice versa.

Assumption 2b. *Each function $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is strongly convex with $\mu > 0$.*

Under Assumption 2a, let us define the (unique) optimal solution of (1) as w^*, Then strong convexity of P implies that

$$2\mu[P(w) - P(w^*)] \leq \|\nabla P(w)\|^2, \ \forall w \in \mathbb{R}^d. \quad (6)$$

We note here, for future use, that for strongly convex functions of the form (1), arising in machine learning applications, the condition number is defined as $\kappa \stackrel{\text{def}}{=} L/\mu$. Furthermore, we should also notice that Assumptions 2a and 2b both cover a wide range of problems, e.g. l_2-regularized empirical risk minimization problems with convex losses.

Finally, as a special case of the strong convexity of all f_i's with $\mu = 0$, we state the general convexity assumption, which we will use for convergence analysis.

Assumption 3. *Each function $f_i : \mathbb{R}^d \to \mathbb{R}$, $i \in [n]$, is convex, i.e.,*

$$f_i(w) \geq f_i(w') + \nabla f_i(w')^T(w - w'), \quad \forall i \in [n].$$

Again, we note that Assumption 2b implies Assumption 3, but Assumption 2a does not. Hence in our analysis, depending on the result we aim at, we will require Assumption 3 to hold by itself, or Assumption 2a and Assumption 3 to hold together, or Assumption 2b to hold by itself. We will always use Assumption 1.

Our iteration complexity analysis aims to bound the number of outer iterations $\mathcal{T}$ (or total number of stochastic gradient evaluations) which is needed to guarantee that $\|\nabla P(w_{\mathcal{T}})\|^2 \leq \epsilon$. In this case we will say that $w_{\mathcal{T}}$ is an ϵ-accurate solution. However, as is common practice for stochastic gradient algorithms, we aim to obtain the bound on the number of iterations, which is required to guarantee the bound on the expected squared norm of a gradient, i.e.,

$$\mathbb{E}[\|\nabla P(w_{\mathcal{T}})\|^2] \leq \epsilon. \quad (7)$$

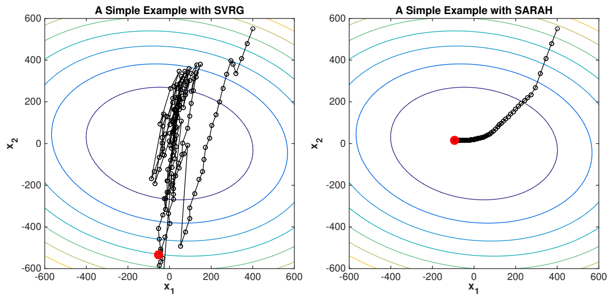

Figure 1: A two-dimensional example of $\min_w P(w)$ with $n = 5$ for SVRG (left) and SARAH (right).

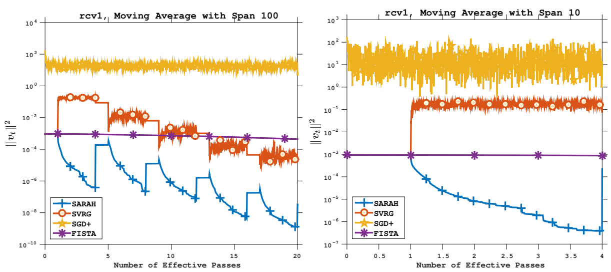

Figure 2: An example of ℓ_2-regularized logistic regression on *rcv1* training dataset for SARAH, SVRG, SGD+ and FISTA with multiple outer iterations (left) and a single outer iteration (right).

3.1. Linearly Diminishing Step-Size in a Single Inner Loop

The most important property of the SVRG algorithm is the variance reduction of the steps. This property holds as the number of outer iteration grows, but it does not hold, if only the number of inner iterations increases. In other words, if we simply run the inner loop for many iterations (without executing additional outer loops), the variance of the steps does not reduce in the case of SVRG, while it goes to zero in the case of SARAH. To illustrate this effect, let us take a look at Figures 1 and 2.

In Figure 1, we applied one outer loop of SVRG and SARAH to a sum of 5 quadratic functions in a two-dimensional space, where the optimal solution is at the origin, the black lines and black dots indicate the trajectory of each algorithm and the red point indicates the final iterate. Initially, both SVRG and SARAH take steps along stochastic gradient directions towards the optimal solution. However, later iterations of SVRG wander randomly around the origin with large deviation from it, while SARAH follows a much more stable convergent trajectory, with a final iterate falling in a small neighborhood of the optimal solution.

In Figure 2, the x-axis denotes the *number of effective passes* which is equivalent to the number of passes through all of the data in the dataset, the cost of each pass being equal to the cost of one full gradient evaluation; and y-axis represents $\|v_t\|^2$. Figure 2 shows the evolution of $\|v_t\|^2$ for SARAH, SVRG, SGD+ (SGD with decreasing learning rate) and FISTA (an accelerated version of GD (Beck & Teboulle, 2009)) with $m = 4n$, where the left plot shows the trend over multiple outer iterations and the right plot shows a single outer iteration[4]. We can see that for SVRG, $\|v_t\|^2$ decreases over the outer iterations, while it has an increasing trend or oscillating trend for each inner loop. In contrast, SARAH enjoys decreasing trends both in the outer and the inner loop iterations.

We will now show that the stochastic steps computed by SARAH converge linearly in the inner loop. We present two linear convergence results based on our two different assumptions of *μ-strong convexity*. These results substantiate our conclusion that SARAH uses more stable stochastic gradient estimates than SVRG. The following theorem is our first result to demonstrate the linear convergence of our stochastic recursive step v_t.

Theorem 1a. *Suppose that Assumptions 1, 2a and 3 hold. Consider v_t defined by (2) in SARAH (Algorithm 1) with $\eta < 2/L$. Then, for any $t \geq 1$,*

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]\mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \mathbb{E}[\|\nabla P(w_0)\|^2].\end{aligned}$$

This result implies that by choosing $\eta = \mathcal{O}(1/L)$, we obtain the linear convergence of $\|v_t\|^2$ in expectation with the rate $(1 - 1/\kappa^2)$. Below we show that a better convergence rate can be obtained under a stronger convexity assumption.

Theorem 1b. *Suppose that Assumptions 1 and 2b hold. Consider v_t defined by (2) in SARAH (Algorithm 1) with $\eta \leq 2/(\mu + L)$. Then the following bound holds, $\forall\, t \geq 1$,*

$$\begin{aligned}\mathbb{E}[\|v_t\|^2] &\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right)\mathbb{E}[\|v_{t-1}\|^2] \\ &\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right)^t \mathbb{E}[\|\nabla P(w_0)\|^2].\end{aligned}$$

Again, by setting $\eta = \mathcal{O}(1/L)$, we derive the linear convergence with the rate of $(1 - 1/\kappa)$, which is a significant improvement over the result of Theorem 1a, when the problem is severely ill-conditioned.

3.2. Convergence Analysis

In this section, we derive the general convergence rate results for Algorithm 1. First, we present two important Lemmas as the foundation of our theory. Then, we proceed to prove sublinear convergence rate of a single outer iteration when applied to general convex functions. In the end, we

[4] In the plots of Figure 2, since the data for SVRG is noisy, we smooth it by using moving average filters with spans 100 for the left plot and 10 for the right one.

prove that the algorithm with multiple outer iterations has linear convergence rate in the strongly convex case.

We begin with proving two useful lemmas that do not require any convexity assumption. The first Lemma 1 bounds the sum of expected values of $\|\nabla P(w_t)\|^2$. The second, Lemma 2, bounds $\mathbb{E}[\|\nabla P(w_t) - v_t\|^2]$.

Lemma 1. *Suppose that Assumption 1 holds. Consider SARAH (Algorithm 1). Then, we have*

$$\sum_{t=0}^{m} \mathbb{E}[\|\nabla P(w_t)\|^2] \leq \frac{2}{\eta}\mathbb{E}[P(w_0) - P(w^*)] \qquad (8)$$
$$+ \sum_{t=0}^{m} \mathbb{E}[\|\nabla P(w_t) - v_t\|^2] - (1 - L\eta)\sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2].$$

Lemma 2. *Suppose that Assumption 1 holds. Consider v_t defined by (2) in SARAH (Algorithm 1). Then for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] = \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2]$$
$$- \sum_{j=1}^{t} \mathbb{E}[\|\nabla P(w_j) - \nabla P(w_{j-1})\|^2].$$

Now we are ready to provide our main theoretical results.

3.2.1. General Convex Case

Following from Lemma 2, we can obtain the following upper bound for $\mathbb{E}[\|\nabla P(w_t) - v_t\|^2]$ for convex functions $f_i, i \in [n]$.

Lemma 3. *Suppose that Assumptions 1 and 3 hold. Consider v_t defined as (2) in SARAH (Algorithm 1) with $\eta < 2/L$. Then we have that for any $t \geq 1$,*

$$\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] \leq \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2]\Big]$$
$$\leq \frac{\eta L}{2 - \eta L}\mathbb{E}[\|v_0\|^2]. \qquad (9)$$

Using the above lemmas, we can state and prove one of our core theorems as follows.

Theorem 2. *Suppose that Assumptions 1 and 3 hold. Consider SARAH (Algorithm 1) with $\eta \leq 1/L$. Then for any $s \geq 1$, we have*

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] \leq \frac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*)]$$
$$+ \frac{\eta L}{2 - \eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \qquad (10)$$

Proof. Since $v_0 = \nabla P(w_0)$ implies $\|\nabla P(w_0) - v_0\|^2 = 0$ then by Lemma 3, we can write

$$\textstyle\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] \leq \frac{m\eta L}{2-\eta L}\mathbb{E}[\|v_0\|^2]. \qquad (11)$$

Hence, by Lemma 1 with $\eta \leq 1/L$, we have

$$\textstyle\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t)\|^2]$$
$$\leq \tfrac{2}{\eta}\mathbb{E}[P(w_0) - P(w^*)] + \textstyle\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t) - v_t\|^2]$$
$$\overset{(11)}{\leq} \tfrac{2}{\eta}\mathbb{E}[P(w_0) - P(w^*)] + \tfrac{m\eta L}{2-\eta L}\mathbb{E}[\|v_0\|^2]. \qquad (12)$$

Since we are considering one outer iteration, with $s \geq 1$, then we have $v_0 = \nabla P(w_0) = \nabla P(\tilde{w}_{s-1})$ (since $w_0 = \tilde{w}_{s-1}$), and $\tilde{w}_s = w_t$, where t is picked uniformly at random from $\{0, 1, \dots, m\}$. Therefore, the following holds,

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] = \tfrac{1}{m+1}\textstyle\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t)\|^2]$$
$$\overset{(12)}{\leq} \tfrac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*)]$$
$$+ \tfrac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \qquad \square$$

Theorem 2, in the case when $\eta \leq 1/L$ implies that

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] \leq \tfrac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*)]$$
$$+ \eta L\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2].$$

By choosing the learning rate $\eta = \sqrt{\frac{2}{L(m+1)}}$ (with m such that $\sqrt{\frac{2}{L(m+1)}} \leq 1/L$) we can derive the following convergence result,

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2]$$
$$\leq \sqrt{\tfrac{2L}{m+1}}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*) + \|\nabla P(\tilde{w}_{s-1})\|^2].$$

Clearly, this result shows a sublinear convergence rate for SARAH under general convexity assumption within a single inner loop, with increasing m, and consequently, we have the following result for complexity bound.

Corollary 1. *Suppose that Assumptions 1 and 3 hold. Consider SARAH (Algorithm 1) within a single outer iteration with the learning rate $\eta = \sqrt{\frac{2}{L(m+1)}}$ where $m \geq 2L - 1$ is the total number of iterations, then $\|\nabla P(w_t)\|^2$ converges sublinearly in expectation with a rate of $\sqrt{\frac{2L}{m+1}}$, and therefore, the total complexity to achieve an ϵ-accurate solution defined in (7) is $\mathcal{O}(n + 1/\epsilon^2)$.*

We now turn to estimating convergence of SARAH with multiple outer steps. Simply using Theorem 2 for each of the outer steps we have the following result.

Theorem 3. *Suppose that Assumptions 1 and 3 hold. Consider SARAH (Algorithm 1) and define*

$$\delta_k = \tfrac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_k) - P(w^*)],\ k = 0, 1, \dots, s-1,$$

and $\delta = \max_{0 \leq k \leq s-1} \delta_k$. Then we have

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] - \Delta \leq \alpha^s(\|\nabla P(\tilde{w}_0)\|^2 - \Delta), \qquad (13)$$

where $\Delta = \delta\left(1 + \frac{\eta L}{2(1-\eta L)}\right)$, and $\alpha = \frac{\eta L}{2-\eta L}$.

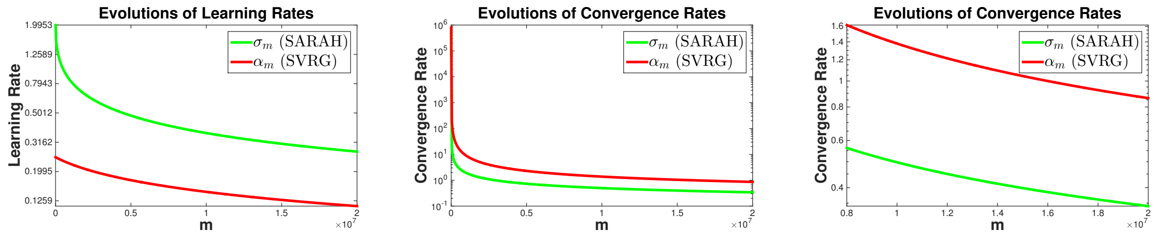

Figure 3: Theoretical comparisons of learning rates (left) and convergence rates (middle and right) with $n = 1,000,000$ for SVRG and SARAH in one inner loop.

Based on Theorem 3, we have the following total complexity for SARAH in the general convex case.

Corollary 2. *Let us choose $\Delta = \epsilon/4$, $\alpha = 1/2$ (with $\eta = 2/(3L)$), and $m = \mathcal{O}(1/\epsilon)$ in Theorem 3. Then, the total complexity to achieve an ϵ-accuracy solution defined in (7) is $\mathcal{O}((n + (1/\epsilon))\log(1/\epsilon))$.*

3.2.2. Strongly Convex Case

We now turn to the discussion of the linear convergence rate of SARAH under the strong convexity assumption on P. From Theorem 2, for any $s \geq 1$, using property (6) of the μ-*strongly convex* P, we have

$$\begin{aligned}\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] &\leq \tfrac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*)] \\ &\quad + \tfrac{\eta L}{2-\eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2] \\ &\overset{(6)}{\leq} \left(\tfrac{1}{\mu\eta(m+1)} + \tfrac{\eta L}{2-\eta L}\right)\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2],\end{aligned}$$

and equivalently,

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] \leq \sigma_m \, \mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2]. \tag{14}$$

Let us define $\sigma_m \overset{\text{def}}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L}$. Then by choosing η and m such that $\sigma_m < 1$, and applying (14) recursively, we are able to reach the following convergence result.

Theorem 4. *Suppose that Assumptions 1, 2a and 3 hold. Consider SARAH (Algorithm 1) with the choice of η and m such that*

$$\sigma_m \overset{def}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} < 1. \tag{15}$$

Then, we have

$$\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] \leq (\sigma_m)^s \|\nabla P(\tilde{w}_0)\|^2.$$

Remark 1. *Theorem 4 implies that any $\eta < 1/L$ will work for SARAH. Let us compare our convergence rate to that of SVRG. The linear rate of SVRG, as presented in (Johnson & Zhang, 2013), is given by*

$$\alpha_m = \tfrac{1}{\mu\eta(1-2L\eta)m} + \tfrac{2\eta L}{1-2\eta L} < 1.$$

We observe that it implies that the learning rate has to satisfy $\eta < 1/(4L)$, which is a tighter restriction than $\eta < 1/L$ required by SARAH. In addition, with the same values of m and η, the rate or convergence of (the outer iterations) of SARAH is always smaller than that of SVRG.

$$\begin{aligned}\sigma_m &= \tfrac{1}{\mu\eta(m+1)} + \tfrac{\eta L}{2-\eta L} = \tfrac{1}{\mu\eta(m+1)} + \tfrac{1}{2/(\eta L)-1} \\ &< \tfrac{1}{\mu\eta(1-2L\eta)m} + \tfrac{1}{0.5/(\eta L)-1} = \alpha_m.\end{aligned}$$

Remark 2. *To further demonstrate the better convergence properties of SARAH, let us consider following optimization problem*

$$\min_{0<\eta<1/L} \sigma_m, \qquad \min_{0<\eta<1/4L} \alpha_m,$$

which can be interpreted as the best convergence rates for different values of m, for both SARAH and SVRG. After simple calculations, we plot both learning rates and the corresponding theoretical rates of convergence, as shown in Figure 3, where the right plot is a zoom-in on a part of the middle plot. The left plot shows that the optimal learning rate for SARAH is significantly larger than that of SVRG, while the other two plots show significant improvement upon outer iteration convergence rates for SARAH over SVRG.

Based on Theorem 4, we are able to derive the following total complexity for SARAH in the strongly convex case.

Corollary 3. *Fix $\epsilon \in (0,1)$, and let us run SARAH with $\eta = 1/(2L)$ and $m = 4.5\kappa$ for $\mathcal{T}$ iterations where $\mathcal{T} = \lceil \log(\|\nabla P(\tilde{w}_0)\|^2/\epsilon)/\log(9/7)\rceil$, then we can derive an ϵ-accuracy solution defined in (7). Furthermore, we can obtain the total complexity of SARAH, to achieve the ϵ-accuracy solution, as $\mathcal{O}\left((n+\kappa)\log(1/\epsilon)\right)$.*

4. A Practical Variant

While SVRG is an efficient variance-reducing stochastic gradient method, one of its main drawbacks is the sensitivity of the practical performance with respect to the choice of m. It is know that m should be around $\mathcal{O}(\kappa)$,[5] while it still remains unknown that what the exact best choice is. In this section, we propose a practical variant of SARAH as

[5] In practice, when n is large, $P(w)$ is often considered as a regularized Empirical Loss Minimization problem with regularization parameter $\lambda = \frac{1}{n}$, then $\kappa \sim \mathcal{O}(n)$.

SARAH+ (Algorithm 2), which provides an automatic and adaptive choice of the inner loop size m. Guided by the linear convergence of the steps in the inner loop, demonstrated in Figure 2, we introduce a stopping criterion based on the values of $\|v_t\|^2$ while upper-bounding the total number of steps by a large enough m for robustness. The other modification compared to SARAH (Algorithm 1) is the more practical choice $\tilde{w}_s = w_t$, where t is the last index of the particular inner loop, instead of randomly selected intermediate index.

Algorithm 2 SARAH+

Parameters: the learning rate $\eta > 0$, $0 < \gamma \le 1$ and the maximum inner loop size m.
Initialize: $\tilde{w}_0$
Iterate:
for $s = 1, 2, \dots$ **do**
  $w_0 = \tilde{w}_{s-1}$
  $v_0 = \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(w_0)$
  $w_1 = w_0 - \eta v_0$
  $t = 1$
  while $\|v_{t-1}\|^2 > \gamma \|v_0\|^2$ **and** $t < m$ **do**
    Sample i_t uniformly at random from $[n]$
    $v_t = \nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1}) + v_{t-1}$
    $w_{t+1} = w_t - \eta v_t$
    $t = t + 1$
  end while
  Set $\tilde{w}_s = w_t$
end for

Different from SARAH, SARAH+ provides a possibility of earlier termination and unnecessary careful choices of m, and it also covers the classical gradient descent when we set $\gamma = 1$ (since the while loop does not proceed). In Figure 4 we present the numerical performance of SARAH+ with different γs on *rcv1* and *news20* datasets. The size of the inner loop provides a trade-off between the fast sub-linear convergence in the inner loop and linear convergence in the outer loop. From the results, it appears that $\gamma = 1/8$ is the optimal choice. With a larger γ, i.e. $\gamma > 1/8$, the iterates in the inner loop do not provide sufficient reduction, before another full gradient computation is required, while with $\gamma < 1/8$ an unnecessary number of inner steps is performed without gaining substantial progress. Clearly γ is another parameter that requires tuning, however, in our experiments, the performance of SARAH+ has been very robust with respect to the choices of γ and did not vary much from one data set to another.

Similarly to SVRG, $\|v_t\|^2$ decreases in the outer iterations of SARAH+. However, unlike SVRG, SARAH+ also inherits from SARAH the consistent decrease of $\|v_t\|^2$ in expectation in the inner loops. It is not possible to apply the same idea of adaptively terminating the inner loop of

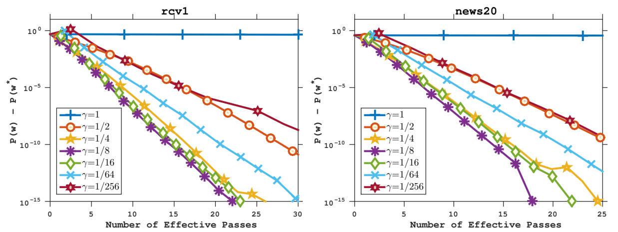

Figure 4: An example of ℓ_2-regularized logistic regression on *rcv1* (left) and *news20* (right) training datasets for SARAH+ with different γs on loss residuals $P(w) - P(w^*)$.

Table 3: Summary of datasets used for experiments.

Dataset	d	n (train)	Sparsity	n (test)	L
covtype	54	406,709	22.12%	174,303	1.90396
ijcnn1	22	91, 701	59.09%	49, 990	1.77662
news20	1,355,191	13, 997	0.03375%	5, 999	0.2500
rcv1	47,236	677,399	0.1549%	20,242	0.2500

SVRG based on the reduction in $\|v_t\|^2$, as $\|v_t\|^2$ may have side fluctuations as shown in Figure 2.

5. Numerical Experiments

To support the theoretical analyses and insights, we present our empirical experiments, comparing SARAH and SARAH+ with the state-of-the-art first-order methods for ℓ_2-regularized logistic regression problems with

$$f_i(w) = \log(1 + \exp(-y_i x_i^T w)) + \tfrac{\lambda}{2}\|w\|^2,$$

on datasets *covtype, ijcnn1, news20* and *rcv1* [6]. For *ijcnn1* and *rcv1* we use the predefined testing and training sets, while *covtype* and *news20* do not have test data, hence we randomly split the datasets with 70% for training and 30% for testing. Some statistics of the datasets are summarized in Table 3.

The penalty parameter λ is set to $1/n$ as is common practice (Le Roux et al., 2012). Note that like SVRG/S2GD and SAG/SAGA, SARAH also allows an efficient sparse implementation named "lazy updates" (Konečný et al., 2016). We conduct and compare numerical results of SARAH with SVRG, SAG, SGD+ and FISTA. SVRG (Johnson & Zhang, 2013) and SAG (Le Roux et al., 2012) are classic modern stochastic methods. SGD+ is SGD with decreasing learning rate $\eta = \eta_0/(k+1)$ where k is the number of effective passes and η_0 is some initial constant learning rate. FISTA (Beck & Teboulle, 2009) is the Fast Iterative Shrinkage-Thresholding Algorithm, well-known as an efficient accelerated version of the gradient descent. Even though for each method, there is a theoretical safe learning rate, we compare the results for the best learning rates in hindsight.

Figure 5 shows numerical results in terms of loss residuals

[6]All datasets are available at http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.

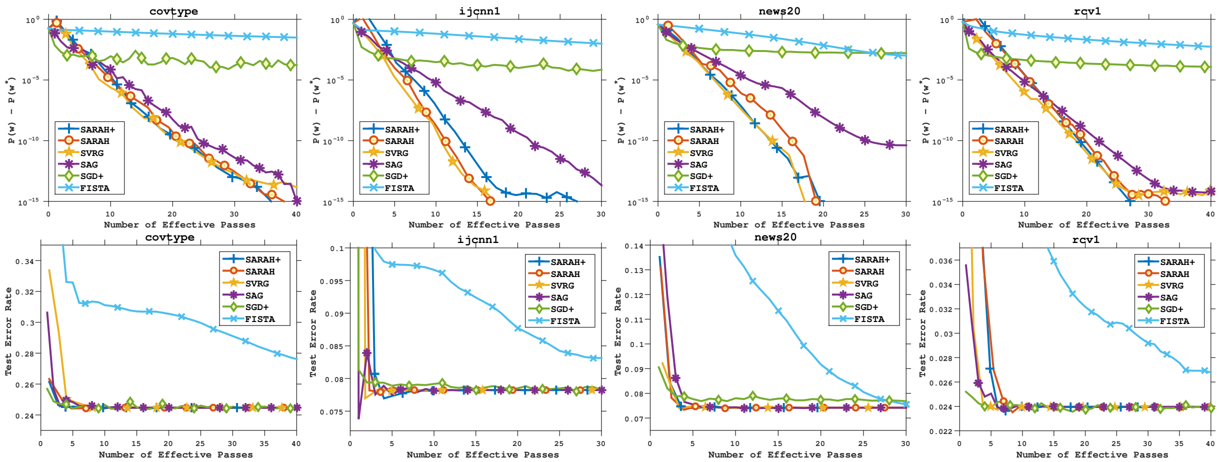

Figure 5: Comparisons of loss residuals $P(w) - P(w^*)$ (top) and test errors (bottom) from different modern stochastic methods on *covtype, ijcnn1, news20* and *rcv1*.

Table 4: Summary of best parameters for all the algorithms on different datasets.

Dataset	SARAH (m^*, η^*)	SVRG (m^*, η^*)	SAG (η^*)	SGD+ (η^*)	FISTA (η^*)
covtype	(2n, 0.9/L)	(n, 0.8/L)	0.3/L	0.06/L	50/L
ijcnn1	(0.5n, 0.8/L)	(n, 0.5/L)	0.7/L	0.1/L	90/L
news20	(0.5n, 0.9/L)	(n, 0.5/L)	0.1/L	0.2/L	30/L
rcv1	(0.7n, 0.7/L)	(0.5n, 0.9/L)	0.1/L	0.1/L	120/L

(top) and test errors (bottom) on the four datasets, SARAH is sometimes comparable or a little worse than other methods at the beginning. However, it quickly catches up to or surpasses all other methods, demonstrating a faster rate of decrease across all experiments. We observe that on *covtype* and *rcv1*, SARAH, SVRG and SAG are comparable with some advantage of SARAH on *covtype*. On *ijcnn1* and *news20*, SARAH and SVRG consistently surpass the other methods.

In particular, to validate the efficiency of our practical variant SARAH+, we provide an insight into how important the choices of m and η are for SVRG and SARAH in Table 4 and Figure 6. Table 4 presents the optimal choices of m and η for each of the algorithm, while Figure 6 shows the behaviors of SVRG and SARAH with different choices of m for *covtype* and *ijcnn1*, where m^*s denote the best choices. In Table 4, the optimal learning rates of SARAH vary less among different datasets compared to all the other methods and they approximate the theoretical upper bound for SARAH ($1/L$); on the contrary, for the other methods the empirical optimal rates can exceed their theoretical limits (SVRG with $1/(4L)$, SAG with $1/(16L)$, FISTA with $1/L$). This empirical studies suggest that it is much easier to tune and find the ideal learning rate for SARAH. As observed in Figure 6, the behaviors of both SARAH and SVRG are quite sensitive to the choices of m. With improper choices of m, the loss residuals can be increased considerably from 10^{-15} to 10^{-3} on both *covtype* in 40 effective passes and *ijcnn1* in 17 effective passes for SARAH/SVRG.

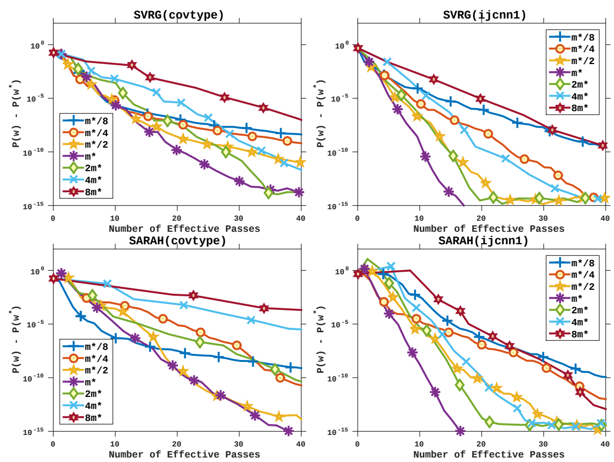

Figure 6: Comparisons of loss residuals $P(w) - P(w^*)$ for different inner loop sizes with SVRG (top) and SARAH (bottom) on *covtype* and *ijcnn1*.

6. Conclusion

We propose a new variance reducing stochastic recursive gradient algorithm SARAH, which combines some of the properties of well known existing algorithms, such as SAGA and SVRG. For smooth convex functions, we show a sublinear convergence rate, while for strongly convex cases, we prove the linear convergence rate and the computational complexity as those of SVRG and SAG. However, compared to SVRG, SARAH's convergence rate constant is smaller and the algorithms is more stable both theoretically and numerically. Additionally, we prove the linear convergence for inner loops of SARAH which support the claim of stability. Based on this convergence we derive a practical version of SARAH, with a simple stopping criterion for the inner loops.

Acknowledgements

The authors would like to thank the reviewers for useful suggestions which helped to improve the exposition in the paper.

References

Allen-Zhu, Zeyuan. Katyusha: The First Direct Acceleration of Stochastic Gradient Methods. *Proceedings of the 49th Annual ACM on Symposium on Theory of Computing (to appear)*, 2017.

Allen-Zhu, Zeyuan and Yuan, Yang. Improved SVRG for Non-Strongly-Convex or Sum-of-Non-Convex Objectives. In *ICML*, pp. 1080–1089, 2016.

Beck, Amir and Teboulle, Marc. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sciences*, 2(1):183–202, 2009.

Bottou, Léon. Online learning and stochastic approximations. In Saad, David (ed.), *Online Learning in Neural Networks*, pp. 9–42. Cambridge University Press, New York, NY, USA, 1998. ISBN 0-521-65263-4.

Bottou, Léon, Curtis, Frank E, and Nocedal, Jorge. Optimization methods for large-scale machine learning. *arXiv:1606.04838*, 2016.

Cotter, Andrew, Shamir, Ohad, Srebro, Nati, and Sridharan, Karthik. Better mini-batch algorithms via accelerated gradient methods. In *NIPS*, pp. 1647–1655, 2011.

Defazio, Aaron, Bach, Francis, and Lacoste-Julien, Simon. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NIPS*, pp. 1646–1654, 2014.

Hastie, Trevor, Tibshirani, Robert, and Friedman, Jerome. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics, 2nd edition, 2009.

Johnson, Rie and Zhang, Tong. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, pp. 315–323, 2013.

Konečný, Jakub, Liu, Jie, Richtárik, Peter, and Takáč, Martin. Mini-batch semi-stochastic gradient descent in the proximal setting. *IEEE Journal of Selected Topics in Signal Processing*, 10:242–255, 2016.

Konečný, Jakub and Richtárik, Peter. Semi-stochastic gradient descent methods. *arXiv:1312.1666*, 2013.

Le Roux, Nicolas, Schmidt, Mark, and Bach, Francis. A stochastic gradient method with an exponential convergence rate for finite training sets. In *NIPS*, pp. 2663–2671, 2012.

Mairal, Julien. Optimization with first-order surrogate functions. In *ICML*, pp. 783–791, 2013.

Mokhtari, Aryan, Gürbüzbalaban, Mert, and Ribeiro, Alejandro. A double incremental aggregated gradient method with linear convergence rate for large-scale optimization. *Proceedings of IEEE International Conference on Acoustic, Speech and Signal Processing (to appear)*, 2017.

Nesterov, Yurii. *Introductory lectures on convex optimization : a basic course*. Applied optimization. Kluwer Academic Publ., Boston, Dordrecht, London, 2004. ISBN 1-4020-7553-7.

Reddi, Sashank J., Hefny, Ahmed, Sra, Suvrit, Póczos, Barnabás, and Smola, Alexander J. Stochastic variance reduction for nonconvex optimization. In *ICML*, pp. 314–323, 2016.

Robbins, Herbert and Monro, Sutton. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

Schmidt, Mark, Le Roux, Nicolas, and Bach, Francis. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, pp. 1–30, 2016.

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

Shalev-Shwartz, Shai, Singer, Yoram, and Srebro, Nathan. Pegasos: Primal estimated sub-gradient solver for SVM. In *ICML*, pp. 807–814, 2007.

Shalev-Shwartz, Shai, Singer, Yoram, Srebro, Nathan, and Cotter, Andrew. Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical Programming*, 127(1): 3–30, 2011.

Takáč, Martin, Bijral, Avleen Singh, Richtárik, Peter, and Srebro, Nathan. Mini-batch primal and dual methods for SVMs. In *ICML*, pp. 1022–1030, 2013.

Xiao, Lin and Zhang, Tong. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

SARAH: A Novel Method for Machine Learning Problems Using Stochastic Recursive Gradient Supplementary Material, ICML 2017

A. Technical Results

Lemma 4 (Theorem 2.1.5 in (Nesterov, 2004))**.** *Suppose that f is convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$f(w) \le f(w') + \nabla f(w')^T(w - w') + \frac{L}{2}\|w - w'\|^2, \tag{16}$$

$$f(w) \ge f(w') + \nabla f(w')^T(w - w') + \frac{1}{2L}\|\nabla f(w) - \nabla f(w')\|^2, \tag{17}$$

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \ge \frac{1}{L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{18}$$

Note that (16) does not require the convexity of f.

Lemma 5 (Theorem 2.1.11 in (Nesterov, 2004))**.** *Suppose that f is μ-strongly convex and L-smooth. Then, for any w, $w' \in \mathbb{R}^d$,*

$$(\nabla f(w) - \nabla f(w'))^T(w - w') \ge \frac{\mu L}{\mu + L}\|w - w'\|^2 + \frac{1}{\mu + L}\|\nabla f(w) - \nabla f(w')\|^2. \tag{19}$$

Lemma 6 (Choices of m and η)**.** *Consider the rate of convergence σ_m in Theorem 4. If we choose $\eta = 1/(\theta L)$ with $\theta > 1$ and fix σ_m, then the best choice of m is*

$$m^* = \frac{1}{2}(2\theta^* - 1)^2\kappa - 1,$$

where $\kappa \overset{def}{=} L/\mu$, with θ^* calculated as:

$$\theta^* = \frac{\sigma_m + 1 + \sqrt{\sigma_m + 1}}{2\sigma_m}.$$

Furthermore, we require $\theta^* > 1 + \sqrt{2}/2$ for $\sigma_m < 1$.

B. Proofs

B.1. Proof of Lemma 1

By Assumption 1 and $w_{t+1} = w_t - \eta v_t$, we have

$$\begin{aligned}
\mathbb{E}[P(w_{t+1})] &\overset{(16)}{\le} \mathbb{E}[P(w_t)] - \eta\mathbb{E}[\nabla P(w_t)^T v_t] + \frac{L\eta^2}{2}\mathbb{E}[\|v_t\|^2] \\
&= \mathbb{E}[P(w_t)] - \frac{\eta}{2}\mathbb{E}[\|\nabla P(w_t)\|^2] + \frac{\eta}{2}\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\mathbb{E}[\|v_t\|^2],
\end{aligned}$$

where the last equality follows from the fact $a^T b = \frac{1}{2}\left[\|a\|^2 + \|b\|^2 - \|a - b\|^2\right]$.

By summing over $t = 0, \ldots, m$, we have

$$\mathbb{E}[P(w_{m+1})] \le \mathbb{E}[P(w_0)] - \frac{\eta}{2}\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t)\|^2] + \frac{\eta}{2}\sum_{t=0}^{m}\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] - \left(\frac{\eta}{2} - \frac{L\eta^2}{2}\right)\sum_{t=0}^{m}\mathbb{E}[\|v_t\|^2],$$

which is equivalent to ($\eta > 0$):

$$\begin{aligned}\sum_{t=0}^{m} \mathbb{E}[\|\nabla P(w_t)\|^2] &\le \frac{2}{\eta}\mathbb{E}[P(w_0) - P(w_{m+1})] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla P(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2] \\ &\le \frac{2}{\eta}\mathbb{E}[P(w_0) - P(w^*)] + \sum_{t=0}^{m} \mathbb{E}[\|\nabla P(w_t) - v_t\|^2] - (1 - L\eta) \sum_{t=0}^{m} \mathbb{E}[\|v_t\|^2],\end{aligned}$$

where the last inequality follows since w^* is a global minimizer of P.

B.2. Proof of Lemma 2

Note that $\mathcal{F}_j$ contains all the information of $w_0, \dots, w_j$ as well as $v_0, \dots, v_{j-1}$. For $j \ge 1$, we have

$$\begin{aligned}\mathbb{E}[\|\nabla P(w_j) - v_j\|^2 | \mathcal{F}_j] &= \mathbb{E}[\|[\nabla P(w_{j-1}) - v_{j-1}] + [\nabla P(w_j) - \nabla P(w_{j-1})] - [v_j - v_{j-1}]\|^2 | \mathcal{F}_j] \\ &= \|\nabla P(w_{j-1}) - v_{j-1}\|^2 + \|\nabla P(w_j) - \nabla P(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j] \\ &\quad + 2(\nabla P(w_{j-1}) - v_{j-1})^T (\nabla P(w_j) - \nabla P(w_{j-1})) \\ &\quad - 2(\nabla P(w_{j-1}) - v_{j-1})^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\ &\quad - 2(\nabla P(w_j) - \nabla P(w_{j-1}))^T \mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \\ &= \|\nabla P(w_{j-1}) - v_{j-1}\|^2 - \|\nabla P(w_j) - \nabla P(w_{j-1})\|^2 + \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],\end{aligned}$$

where the last equality follows from

$$\mathbb{E}[v_j - v_{j-1} | \mathcal{F}_j] \overset{(2)}{=} \mathbb{E}[\nabla f_{i_j}(w_j) - \nabla f_{i_j}(w_{j-1}) | \mathcal{F}_j] = \nabla P(w_j) - \nabla P(w_{j-1}).$$

By taking expectation for the above equation, we have

$$\mathbb{E}[\|\nabla P(w_j) - v_j\|^2] = \mathbb{E}[\|\nabla P(w_{j-1}) - v_{j-1}\|^2] - \mathbb{E}[\|\nabla P(w_j) - \nabla P(w_{j-1})\|^2] + \mathbb{E}[\|v_j - v_{j-1}\|^2].$$

Note that $\|\nabla P(w_0) - v_0\|^2 = 0$. By summing over $j = 1, \dots, t$ ($t \ge 1$), we have

$$\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] = \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] - \sum_{j=1}^{t} \mathbb{E}[\|\nabla P(w_j) - \nabla P(w_{j-1})\|^2].$$

B.3. Proof of Lemma 3

For $j \ge 1$, we have

$$\begin{aligned}\mathbb{E}[\|v_j\|^2 | \mathcal{F}_j] &= \mathbb{E}[\|v_{j-1} - (\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j))\|^2 | \mathcal{F}_j] \\ &= \|v_{j-1}\|^2 + \mathbb{E}\Big[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 - \tfrac{2}{\eta}(\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j))^T (w_{j-1} - w_j) | \mathcal{F}_j\Big] \\ &\overset{(18)}{\le} \|v_{j-1}\|^2 + \mathbb{E}\Big[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 - \tfrac{2}{L\eta}\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 | \mathcal{F}_j\Big] \\ &= \|v_{j-1}\|^2 + \Big(1 - \tfrac{2}{\eta L}\Big) \mathbb{E}[\|\nabla f_{i_j}(w_{j-1}) - \nabla f_{i_j}(w_j)\|^2 | \mathcal{F}_j] \\ &\overset{(2)}{=} \|v_{j-1}\|^2 + \Big(1 - \tfrac{2}{\eta L}\Big) \mathbb{E}[\|v_j - v_{j-1}\|^2 | \mathcal{F}_j],\end{aligned}$$

which, if we take expectation, implies that

$$\mathbb{E}[\|v_j - v_{j-1}\|^2] \le \frac{\eta L}{2 - \eta L}\Big[\mathbb{E}[\|v_{j-1}\|^2] - \mathbb{E}[\|v_j\|^2]\Big],$$

when $\eta < 2/L$.

By summing the above inequality over $j = 1, \ldots, t$ $(t \geq 1)$, we have

$$\sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \leq \frac{\eta L}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2] \Big]. \tag{20}$$

By Lemma 2, we have

$$\mathbb{E}[\|\nabla P(w_t) - v_t\|^2] \leq \sum_{j=1}^{t} \mathbb{E}[\|v_j - v_{j-1}\|^2] \overset{(20)}{\leq} \frac{\eta L}{2 - \eta L} \Big[\mathbb{E}[\|v_0\|^2] - \mathbb{E}[\|v_t\|^2] \Big].$$

B.4. Proof of Lemma 6

With $\eta = 1/(\theta L)$ and $\kappa = L/\mu$, the rate of convergence α_m can be written as

$$\sigma_m \overset{(15)}{=} \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2 - \eta L} = \frac{\theta L}{\mu(m+1)} + \frac{1/\theta}{2 - 1/\theta} = \left(\frac{\kappa}{m+1} \right) \theta + \frac{1}{2\theta - 1},$$

which is equivalent to

$$m(\theta) \overset{\text{def}}{=} m = \frac{\theta(2\theta - 1)}{\sigma_m(2\theta - 1) - 1} \kappa - 1.$$

Since σ_m is considered fixed, then the optimal choice of m in terms of θ can be solved from $\min_\theta m(\theta)$, or equivalently, $0 = (\partial m)/(\partial \theta) = m'(\theta)$, and therefore we have the equation with the optimal θ satisfying

$$\sigma_m = (4\theta^* - 1)/(2\theta^* - 1)^2, \tag{21}$$

and by plugging it into $m(\theta)$ we conclude the optimal m:

$$m^* = m(K^*) = \frac{1}{2}(2K^* - 1)^2 \kappa - 1,$$

while by solving for θ^* in (21) and taking into account that $\theta > 1$, we have the optimal choice of θ:

$$\theta^* = \frac{\sigma_m + 1 + \sqrt{\sigma_m + 1}}{2\sigma_m}.$$

Obviously, for $\sigma_m < 1$, we require $\theta^* > 1 + \sqrt{2}/2$.

B.5. Proof of Theorem 1a

For $t \geq 1$, we have

$$\begin{aligned}
\|\nabla P(w_t) - \nabla P(w_{t-1})\|^2 &= \Big\| \frac{1}{n} \sum_{i=1}^{n} [\nabla f_i(w_t) - \nabla f_i(w_{t-1})] \Big\|^2 \\
&\leq \frac{1}{n} \sum_{i=1}^{n} \|\nabla f_i(w_t) - \nabla f_i(w_{t-1})\|^2 \\
&= \mathbb{E}[\|\nabla f_{i_t}(w_t) - \nabla f_{i_t}(w_{t-1})\|^2 | \mathcal{F}_t].
\end{aligned} \tag{22}$$

Using the proof of Lemma 3, for $t \geq 1$, we have

$$\begin{aligned}
\mathbb{E}[\|v_t\|^2 | \mathcal{F}_t] &\leq \|v_{t-1}\|^2 + \left(1 - \tfrac{2}{\eta L}\right) \mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2 | \mathcal{F}_t] \\
&\overset{(22)}{\leq} \|v_{t-1}\|^2 + \left(1 - \tfrac{2}{\eta L}\right) \|\nabla P(w_t) - \nabla P(w_{t-1})\|^2 \\
&\leq \|v_{t-1}\|^2 + \left(1 - \tfrac{2}{\eta L}\right) \mu^2 \eta^2 \|v_{t-1}\|^2.
\end{aligned}$$

Note that $1 - \frac{2}{\eta L} < 0$ since $\eta < 2/L$. The last inequality follows by the strong convexity of P, that is, $\mu\|w_t - w_{t-1}\| \leq \|\nabla P(w_t) - \nabla P(w_{t-1})\|$ and the fact that $w_t = w_{t-1} - \eta v_{t-1}$. By taking the expectation and applying recursively, we have

$$
\begin{aligned}
\mathbb{E}[\|v_t\|^2] &\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right] \mathbb{E}[\|v_{t-1}\|^2] \\
&\leq \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \mathbb{E}[\|v_0\|^2] \\
&= \left[1 - \left(\tfrac{2}{\eta L} - 1\right)\mu^2\eta^2\right]^t \mathbb{E}[\|\nabla P(w_0)\|^2].
\end{aligned}
$$

B.6. Proof of Theorem 1b

We obviously have $\mathbb{E}[\|v_0\|^2|\mathcal{F}_0] = \|\nabla P(w_0)\|^2$. For $t \geq 1$, we have

$$
\begin{aligned}
\mathbb{E}[\|v_t\|^2|\mathcal{F}_t] &\overset{(2)}{=} \mathbb{E}[\|v_{t-1} - (\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t))\|^2|\mathcal{F}_t] \\
&\overset{(3)}{=} \|v_{t-1}\|^2 + \mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2 - \tfrac{2}{\eta}(\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t))^T(w_{t-1} - w_t)|\mathcal{F}_t] \\
&\overset{(19)}{\leq} \|v_{t-1}\|^2 - \tfrac{2\mu L\eta}{\mu+L}\|v_{t-1}\|^2 + \mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t] - \tfrac{2}{\eta}\cdot\tfrac{1}{\mu+L}\mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t] \\
&= (1 - \tfrac{2\mu L\eta}{\mu+L})\|v_{t-1}\|^2 + (1 - \tfrac{2}{\eta}\cdot\tfrac{1}{\mu+L})\mathbb{E}[\|\nabla f_{i_t}(w_{t-1}) - \nabla f_{i_t}(w_t)\|^2|\mathcal{F}_t] \\
&\leq \left(1 - \tfrac{2\mu L\eta}{\mu+L}\right)\|v_{t-1}\|^2, \qquad (23)
\end{aligned}
$$

where in last inequality we have used that $\eta \leq 2/(\mu + L)$. By taking the expectation and applying recursively, the desired result is achieved.

B.7. Proof of Theorem 3

By Theorem 2, we have

$$
\begin{aligned}
\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] &\leq \frac{2}{\eta(m+1)}\mathbb{E}[P(\tilde{w}_{s-1}) - P(w^*)] + \frac{\eta L}{2 - \eta L}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2] \\
&= \delta_{s-1} + \alpha\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2] \\
&\leq \delta_{s-1} + \alpha\delta_{s-2} + \cdots + \alpha^{s-1}\delta_0 + \alpha^s\|\nabla P(\tilde{w}_0)\|^2 \\
&\leq \delta + \alpha\delta + \cdots + \alpha^{s-1}\delta + \alpha^s\|\nabla P(\tilde{w}_0)\|^2 \\
&\leq \delta\frac{1 - \alpha^s}{1 - \alpha} + \alpha^s\|\nabla P(\tilde{w}_0)\|^2 \\
&= \Delta(1 - \alpha^s) + \alpha^s\|\nabla P(\tilde{w}_0)\|^2 \\
&= \Delta + \alpha^s(\|\nabla P(\tilde{w}_0)\|^2 - \Delta),
\end{aligned}
$$

where the second last equality follows since

$$
\frac{\delta}{1 - \alpha} = \delta\left(\frac{2 - \eta L}{2 - 2\eta L}\right) = \delta\left(1 + \frac{\eta L}{2(1 - \eta L)}\right) = \Delta.
$$

Hence, the desired result is achieved.

B.8. Proof of Corollary 2

Based on Theorem 3, if we would aim for an ϵ-accuracy solution, we can choose $\Delta = \epsilon/4$ and $\alpha = 1/2$ (with $\eta = 2/(3L)$). To obtain the convergence to an ϵ-accuracy solution, we need to have $\delta = \mathcal{O}(\epsilon)$, or equivalently, $m = \mathcal{O}(1/\epsilon)$. Then we have

$$\begin{aligned}
\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] &\overset{(13)}{\leq} \frac{\Delta}{2} + \frac{1}{2}\mathbb{E}[\|\nabla P(\tilde{w}_{s-1})\|^2] \\
&\leq \frac{\Delta}{2} + \frac{\Delta}{2^2} + \frac{1}{2^2}\mathbb{E}[\|\nabla P(\tilde{w}_{s-2})\|^2] \\
&\leq \Delta\left(\frac{1}{2} + \frac{1}{2^2} + \cdots + \frac{1}{2^s}\right) + \frac{1}{2^s}\|\nabla P(\tilde{w}_0)\|^2 \\
&\leq \Delta + \frac{1}{2^s}\|\nabla P(\tilde{w}_0)\|^2.
\end{aligned}$$

To guarantee that $\mathbb{E}[\|\nabla P(\tilde{w}_s)\|^2] \leq \epsilon$, it is sufficient to make $\frac{1}{2^s}\|\nabla P(\tilde{w}_0)\|^2 \leq \frac{3}{4}\epsilon$, or $s = \mathcal{O}(\log(1/\epsilon))$. This implies the total complexity to achieve an ϵ-accuracy solution is $(n+2m)s = \mathcal{O}((n+(1/\epsilon))\log(1/\epsilon))$.

B.9. Proof of Corollary 3

Based on Lemma 6 and Theorem 4, let us pick $\theta^* = 2$, i.e, then we have $m^* = 4.5\kappa - 1$. So let us run SARAH with $\eta = 1/(2L)$ and $m = 4.5\kappa$, then we can calculate σ_m in (15) as

$$\sigma_m = \frac{1}{\mu\eta(m+1)} + \frac{\eta L}{2-\eta L} = \frac{1}{[\mu/(2L)](4.5\kappa+1)} + \frac{1/2}{2-1/2} < \frac{4}{9} + \frac{1}{3} = \frac{7}{9}.$$

According to Theorem 4, if we run SARAH for $\mathcal{T}$ iterations where

$$\mathcal{T} = \lceil \log(\|\nabla P(\tilde{w}_0)\|^2/\epsilon)/\log(9/7)\rceil = \lceil \log_{7/9}(\epsilon/\|\nabla P(\tilde{w}_0)\|^2)\rceil \geq \log_{7/9}(\epsilon/\|\nabla P(\tilde{w}_0)\|^2),$$

then we have

$$\mathbb{E}[\|\nabla P(\tilde{w}_{\mathcal{T}})\|^2] \leq (\sigma_m)^{\mathcal{T}}\|\nabla P(\tilde{w}_0)\|^2 < (7/9)^{\mathcal{T}}\|\nabla P(\tilde{w}_0)\|^2 \leq (7/9)^{\log_{7/9}(\epsilon/\|\nabla P(\tilde{w}_0)\|^2)}\|\nabla P(\tilde{w}_0)\|^2 = \epsilon,$$

thus we can derive (7). If we consider the number of gradient evaluations as the main computational complexity, then the total complexity can be obtained as

$$(n+2m)\mathcal{T} = \mathcal{O}\left((n+\kappa)\log(1/\epsilon)\right).$$