Signal Processing 144 (2018) 253–264

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Performance limits of stochastic sub-gradient learning, part II: Multi-agent case

Bicheng Ying[a,*], Ali H. Sayed[b,1]

[a] Department of Electrical Engineering, University of California, Los Angeles, United States
[b] School of Engineering, Ecole Polytechnique Federale de Lausanne, Switzerland

ARTICLE INFO

Article history:
Received 14 April 2017
Revised 23 September 2017
Accepted 4 October 2017
Available online 16 October 2017

Keywords:
Sub-gradient algorithm
Affine-Lipschitz
Linear rate
Diffusion strategy
Networked agents
SVM
LASSO

ABSTRACT

The analysis in Part I [1] revealed interesting properties for subgradient learning algorithms in the context of *stochastic* optimization. These algorithms are used when the risk functions are non-smooth or involve non-differentiable components. They have been long recognized as being slow converging methods. However, it was revealed in Part I [1] that the rate of convergence becomes linear for *stochastic* optimization problems, with the error iterate converging at an exponential rate α^i to within an $O(\mu)$−neighborhood of the optimizer, for some $\alpha \in (0, 1)$ and small step-size μ. The conclusion was established under weaker assumptions than the prior literature and, moreover, several important problems were shown to satisfy these weaker assumptions automatically. These results revealed that sub-gradient learning methods have more favorable behavior than originally thought. The results of Part I [1] were exclusive to single-agent adaptation. The purpose of current Part II is to examine the implications of these discoveries when a collection of networked agents employs subgradient learning as their cooperative mechanism. The analysis will show that, despite the coupled dynamics that arises in a networked scenario, the agents are still able to attain linear convergence in the stochastic case; they are also able to reach agreement within $O(\mu)$ of the optimizer.

1. Introduction and review of [1]

We review briefly the notation and findings from Part I [1] in preparation for examining the challenges that arise in the multi-agent scenario. In Part I [1], we considered an optimization problem of the form:

$$w^\star = \arg\min_{w\in\mathbb{R}^M} J(w) \tag{1}$$

where the possibly *non-differentiable* but strongly-convex risk function $J(w)$ was expressed as the expectation of some convex but also possibly non-differentiable loss function $Q(\cdot)$, namely,

$$J(w) \triangleq \mathbb{E}\, Q(w; \boldsymbol{x}) \tag{2}$$

Here, the letter $\boldsymbol{x}$ represents the random data and the expectation operation is over the distribution of this data. The following sub-gradient algorithm was introduced and studied in Part I [1] for seeking $w^\star$:

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu\widehat{g}(\boldsymbol{w}_{i-1}) \tag{3}$$

$$S_i = \kappa S_{i-1} + 1 \tag{4}$$

$$\bar{\boldsymbol{w}}_i = \left(1 - \frac{1}{S_i}\right)\bar{\boldsymbol{w}}_{i-1} + \frac{1}{S_i}\boldsymbol{w}_i \tag{5}$$

with initial conditions $S_0 = 1$, $\boldsymbol{w}_0 = 0$, and $\bar{\boldsymbol{w}}_0 = 0$. Boldface notation is used for $\boldsymbol{w}_i$ to highlight its stochastic nature since the successive iterates are generated by relying on streaming data realizations for $\boldsymbol{x}$. Moreover, the scalar $\kappa \in [\alpha, 1)$, where $\alpha = 1 - O(\mu)$ is a number close to one. The term $\widehat{g}(\boldsymbol{w}_{i-1})$ in [3] is an approximate sub-gradient at location $\boldsymbol{w}_{i-1}$; it is computed from the data available at time i and approximates a true sub-gradient denoted by $g(\boldsymbol{w}_{i-1})$. This true sub-gradient is unavailable since $J(w)$ itself is unavailable in the stochastic context. This is because the distribution of the data $\boldsymbol{x}$ is unknown beforehand, which means that the expected loss function cannot be evaluated. The difference between a true sub-gradient vector and its approximation is gradient noise and is denoted by

$$\boldsymbol{s}_i(\boldsymbol{w}_{i-1}) \triangleq \widehat{g}(\boldsymbol{w}_{i-1}) - g(\boldsymbol{w}_{i-1}) \tag{6}$$

* Corresponding author.
E-mail addresses: ybc@ucla.edu (B. Ying), ali.sayed@epfl.ch (A.H. Sayed).

[1] This work was supported in part by NSF grants CCF-1524250, ECCS-1407712, and DARPA N66001-14-2-4029. A short conference version appears in [2].

https://doi.org/10.1016/j.sigpro.2017.10.007
0165-1684/

1.1. Datamodel and assumptions

The following three assumptions were motivated in Part I [1]:

1. $J(w)$ is $\eta-$strongly-convex so that $w^\star$ is unique. The strong convexity of $J(w)$ means that

$$J(\theta w_1 + (1-\theta)w_2) \le \theta J(w_1) + (1-\theta)J(w_2) - \frac{\eta}{2}\theta(1-\theta)\|w_1 - w_2\|^2, \quad (7)$$

for any $\theta \in [0, 1]$, w_1, and w_2. The above condition is equivalent to requiring [3]:

$$J(w_1) \ge J(w_2) + g(w_2)^{\mathsf{T}}(w_1 - w_2) + \frac{\eta}{2}\|w_1 - w_2\|^2. \quad (8)$$

2. The subgradient is affine Lipschitz, meaning that there exist constants $c \ge 0$ and $d \ge 0$ such that

$$\|g(w_1) - g'(w_2)\| \le c\|w_1 - w_2\| + d, \quad \forall w_1, w_2 \quad (9)$$

and for any $g'(\cdot) \in \partial J(\cdot)$. Here, the notation $\partial J(w)$ denotes the differential at location w (i.e., the set of all possible subgradient vectors at w). It was explained in Part I [1] how this affine Lipschitz condition is weaker than conditions used before in the literature and how important cases of interest (such as SVM, LASSO, Total Variation) satisfy it automatically (but do not satisfy the previous conditions). For later use, it is easy to verify (as was done in (50) in Part I [1]) that condition (9) implies that

$$\|g(w_1) - g'(w_2)\|^2 \le e^2\|w_1 - w_2\|^2 + f^2, \quad \forall w_1, w_2, \quad (10)$$

for any $g'(\cdot) \in \partial J(\cdot)$ and some constants $e^2 \ge 0$ and $f^2 \ge 0$.

3. The first and second-order moments of the gradient noise process satisfy the conditions:

$$\mathbb{E}[\, s_i(\boldsymbol{w}_{i-1}) \,|\, \mathcal{F}_{i-1}\,] = 0, \quad (11)$$

$$\mathbb{E}[\, \|s_i(\boldsymbol{w}_{i-1})\|^2 \,|\, \mathcal{F}_{i-1}\,] \le \beta^2\|w^\star - \boldsymbol{w}_{i-1}\|^2 + \sigma^2, \quad (12)$$

for some constants $\beta^2 \ge 0$ and $\sigma^2 \ge 0$, and where the notation $\mathcal{F}_{i-1}$ denotes the filtration (collection) corresponding to all past iterates:

$$\mathcal{F}_{i-1} = \text{filtration by } \{\boldsymbol{w}_j,\ j \le i-1\}. \quad (13)$$

It was again shown in Part I [1] how the gradient noise process in important applications (e.g., SVM,LASSO) satisfy (11) and (12) directly.

Under the three conditions (1)–(3), which are automatically satisfied for important cases of interest, the following important conclusion was proven in Part I [1] for the stochastic subgradient algorithm (3)–(5) above. At every iteration i, the average risk value converges to a small $O(\mu)-$neighborhood around $J(w^\star)$, namely,

$$\lim_{i\to\infty} \mathbb{E}J(\tilde{\boldsymbol{w}}_i) - J(w^\star) \le \mu(f^2 + \sigma^2)/2 \quad (14)$$

where the convergence of $\mathbb{E}J(\tilde{\boldsymbol{w}}_i)$ towards this neighborhood occurs at an exponential rate $O(\alpha^i)$ where $\alpha = 1 - \mu\eta + O(\mu^2)$.

1.2. Interpretation of result

For the benefit of the reader, we repeat here the interpretation that was given in Sec. IV.D of Part I [1] for the key results (14); these remarks will be relevant in the networked case and are therefore useful to highlight again:

1. First, it has been observed in the optimization literature [3–5] that sub-gradient descent iterations can perform poorly in deterministic problems (where $J(w)$ is known). Their convergence rate is $O(1/\sqrt{i})$ under convexity and $O(1/i)$ under strong-convexity when decaying step-sizes, $\mu(i) = 1/i$, are used to ensure convergence [5]. Result (14) shows that the situation is different in the context of stochastic optimization when true subgradients are approximated from streaming data due to different requirements. By using *constant* step-sizes to enable continuous learning and adaptation, the sub-gradient iteration is now able to achieve exponential convergence at the rate of $O(\alpha^i)$ to steady-state.
2. Second, this substantial improvement in convergence rate comes at a cost, but one that is acceptable and controllable. Specifically, we cannot guarantee convergence of the algorithm to the global minimum value, $J(w^\star)$, anymore but can instead approach this optimal value with high accuracy in the order of $O(\mu)$, where the size of μ is under the designer's control and can be selected as small as desired.
3. Third, this performance level is sufficient in most cases of interest because, in practice, one rarely has an infinite amount of data and, moreover, the data is often subject to distortions not captured by any assumed models. It is increasingly recognized in the literature that it is not always necessary to ensure exact convergence towards the optimal solution, $w^\star$, or the minimum value, $J(w^\star)$, because these optimal values may not reflect accurately the true state due to modeling errors. For example, it is explained in the works [3,6–8] that it is generally unnecessary to reduce the error measures below the statistical error level that is present in the data.

1.3. This work

The purpose of this work is to examine how these properties reveal themselves in the networked case when a multitude of interconnected agents cooperate to minimize an aggregate cost function that is not generally smooth. In this case, it is necessary to examine closely the effect of the coupled dynamics and whether agents will still be able to agree fast enough under non-differentiability.

Distributed learning under non-smooth risk functions is common in many applications including distributed estimation and distributed machine learning. For example, ℓ_1-regularization or hinge-loss functions (as in SVM implementations) lead to non-smooth risks. Several useful techniques have been developed in the literature for the solution of such distributed optimization problems, including the use of consensus strategies [9–11] and diffusion strategies [12–15]. In this paper, we will focus on the Adapt-then-Combine (ATC) diffusion strategy mainly because diffusion strategies have been shown to have superior mean-square-error and stability performance in adaptive scenarios where agents are expected to continually learn from streaming data [15]. In particular, we shall examine the performance and stability behavior of networked diffusion learning under weaker conditions than previously considered in the literature. It is true that there have been several useful studies that employed sub-gradient constructions in the distributed setting before, most notably [9,16,17]. However, these earlier works generally assume bounded subgradients. As was already explained in Part I [1], this is a serious limitation (which does not hold even for quadratic risks where the gradient vector is linear in w and grows unbounded). Instead, we shall consider the weaker affine Lipschitz condition (9), which was shown in Part I [1] to be satisfied automatically by important risk functions such as those arising in popular quadratic, SVM, and LASSO formulations.

Notation: We use lowercase letters to denote vectors, uppercase letters for matrices, plain letters for deterministic variables, and boldface letters for random variables. We also use $(\cdot)^{\mathsf{T}}$ to denote transposition, $(\cdot)^{-1}$ for matrix inversion, $\mathrm{Tr}(\cdot)$ for the trace of a