2016 IEEE INTERNATIONAL WORKSHOP ON MACHINE LEARNING FOR SIGNAL PROCESSING, SEPT. 13–16, 2016, SALERNO, ITALY

STOCHASTIC GRADIENT DESCENT WITH FINITE SAMPLES SIZES

Kun Yuan *Bicheng Ying* *Stefan Vlaski* *Ali H. Sayed*

Department of Electrical Engineering, University of California, Los Angeles

ABSTRACT

The minimization of empirical risks over finite sample sizes is an important problem in large-scale machine learning. A variety of algorithms has been proposed in the literature to alleviate the computational burden per iteration at the expense of convergence speed and accuracy. Many of these approaches can be interpreted as stochastic gradient descent algorithms, where data is sampled from particular empirical distributions. In this work, we leverage this interpretation and draw from recent results in the field of online adaptation to derive new tight performance expressions for empirical implementations of stochastic gradient descent, mini-batch gradient descent, and importance sampling. The expressions are exact to first order in the step-size parameter and are tighter than existing bounds. We further quantify the performance gained from employing mini-batch solutions, and propose an optimal importance sampling algorithm to optimize performance.

Index Terms— Online learning, stochastic gradient descent, constant step-size, mini-batch technique, importance sampling.

1. INTRODUCTION

We consider minimizing an empirical risk function $J_e(w)$, which is the sample average over a possibly large, yet finite training set:

$$w^\star \triangleq \arg\min_{w\in\mathbb{R}^M} \quad J_e(w) \triangleq \frac{1}{N}\sum_{n=1}^{N} Q(w;x_n), \tag{1}$$

where the $\{x_n\}_{n=1}^N$ are training data samples. In this paper we assume the loss function $Q(w;x_n)$ is differentiable, and the empirical risk $J_e(w)$ is strongly convex. Problems of the form (1) are common in many areas of machine learning including linear regression, logistic regression and their regularized versions.

When the size of the dataset N is large, it is impractical to solve (1) directly with classical gradient descent. One simple, yet powerful, approach to remedy this difficulty is to employ the stochastic gradient method (SGD) [1–7]. In this method, at every iteration, rather than compute the full gradient $\nabla_w J_e(w)$ on the entire data set, the algorithm picks one index $\boldsymbol{n}_i$ at random, and employs $\nabla_w Q(w;x_{\boldsymbol{n}_i})$ to approximate $\nabla_w J_e(w)$. Specifically, at iteration i, the update for estimating the minimizer is of the form:

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu_i \nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}), \tag{2}$$

where μ_i is the step-size parameter. Note that we are using boldface notation to refer to random variables. Although uncommon in the literature, in this paper we refer to recursion (2) as the *empirical* stochastic gradient descent (E-SGD) iteration, mainly because we will be contrasting it with an *online* stochastic gradient descent (O-SGD) algorithm.

Stochastic gradient descent recursions of the form (2) have been studied extensively in the literature, primarily in the case when the step-size μ_i is diminishing [2–6, 8]. When $J_e(w)$ is strongly convex, these algorithms have been shown to converge to the minimizer $w^\star$ at a sublinear rate $O(1/i)$. In comparison, implementations with constant step-size $\mu_i = \mu$ do not converge to the exact minimizer $w^\star$, but rather to a small region around the minimizer in the order of $O(\mu)$ [7, 9–13]. However, convergence to this region occurs at an exponentially fast rate. Fast convergence to an approximate but close solution is very useful in the context of machine learning since, after all, empirical risks of the form (1) correspond to auxiliary problem formulations for the true, yet inaccessible problem of interest, namely,

$$w^o \triangleq \arg\min_{w\in\mathbb{R}^M} \; J(w) = \mathbb{E}[Q(w;\boldsymbol{x})], \tag{3}$$

where the letter $\mathbb{E}$ denotes expectation over the often unknown probability distribution of the data $\boldsymbol{x}$. While (1) is used for training, the actual performance on unseen data is measured through (3). The intrinsic bias between $w^\star$ and w^o removes the need for exact convergence to $w^\star$ [4, 5]. This line of reasoning, along with the fast exponential convergence rate and robustness to initialization, has motivated a tremendous interest in constant step-size implementations with a focus on practical solutions [5, 14–16].

A fundamental question that arises when employing a constant step-size is how to choose μ in order to ensure a desired tolerance on the excess risk (ER) or mean-square-deviation (MSD) that persist after convergence. Non-asymptotic bounds have been given in [7, 12, 13, 17–19], which are useful in revealing worst-case performance guarantees, but do not predict exact performance. Recent advances in the field of online adaptation, on the other hand, have yielded insights into the related problem of learning from streaming data [10, 11, 16]. In particular, MSD and ER expressions, which are accurate to first order in the step-size, are derived in [11, 16] for a broad class of risk functions beyond the traditional quadratic measure.

In this work, we exploit these results to reveal an interesting connection between the two classes of empirical (E-SGD) and online (O-SGD) constructions. First, we show that the stochastic gradient descent algorithm for learning empirical risks (E-SGD) is a special case of online stochastic gradient descent algorithms (O-SGD) studied in [11, 16]. This connection helps establish a powerful unification for learning from finite datasets and learning from streaming data. Once this connection is established, we then leverage this insight to great effect to derive first-order expressions for both the MSD and ER of empirical (E-SGD) implementations. The resulting

This work was supported in part by NSF grants CCF-1524250 and ECCS-1407712, and by DARPA project N66001-14-2-4029. Emails:{kunyuan,ybc,svlaski,sayed}@ucla.edu

978-1-5090-0746-2/16/$31.00

expressions appear to be the tightest in comparison to available results in the literature, such as [7, 12, 13, 19] and other similar works. We further extend the analysis to include mini-batch gradient descent [20, 21] and importance sampling methods [19, 22], and also derive the corresponding MSD and ER expressions for both algorithms. In particular, we show that the MSD and ER of mini-batch are inversely proportional to the batch size.

Another important contribution in this work is that we use the performance expressions to optimize the probability with which the data samples are selected during the empirical implementation. Different from previous works [19, 22], which assume knowledge of Lipschitz constants and use them to design the sampling probability, we start from the uniform distribution and devise a procedure that automatically learns the optimal sampling distribution and attains the optimal ER performance.

2. EMPIRICAL STOCHASTIC GRADIENT DESCENT

In this section we derive the steady-state performance of E-SGD implementations by showing how they can be viewed as special cases of O-SGD implementations. The analysis will build on results from [11], which considered stochastic optimization problems of the form (3). In online implementations, data $\boldsymbol{x}_i$ keep streaming in and the gradient vector of $J(w)$ is approximated by $\nabla_w Q(w;\boldsymbol{x}_i)$. In this way, the successive iterates are computed by means of the following so-called O-SGD recursion:

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu\nabla_w Q(\boldsymbol{w}_{i-1};\boldsymbol{x}_i). \tag{4}$$

Since the data stream in continuously, it is observed in recursion (4) that the data $\boldsymbol{x}_i$ has the same index i as the iteration number. In comparison, in the empirical (E-SGD) implementation (2), the data $x_{\boldsymbol{n}_i}$ is indexed by a randomly selected index, $\boldsymbol{n}_i$, from the finite sample-size range $1 \le n \le N$.

2.1. Relating both formulations

Given a finite number of data samples $\{x_1, x_2, \ldots, x_N\}$, we introduce a discrete random variable $\boldsymbol{x}_e$ having these samples as realizations and a uniform probability mass function (pmf) defined by

$$p(\boldsymbol{x}_e) = \begin{cases} \frac{1}{N}, & \text{if } \boldsymbol{x}_e = x_1, \\ \vdots & \vdots \\ \frac{1}{N}, & \text{if } \boldsymbol{x}_e = x_N. \end{cases} \tag{5}$$

As a result, the empirical problem (1) can be rewritten as

$$\min_{w\in\mathbb{R}^M} \quad J_e(w) = \mathbb{E}[Q(w;\boldsymbol{x}_e)] = \frac{1}{N}\sum_{n=1}^{N} Q(w;x_n), \tag{6}$$

which has the same form as (4) with the random data $\boldsymbol{x}$ replaced by $\boldsymbol{x}_e$. Therefore, we can apply the O-SGD algorithm (4) to solve (6), namely,

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu\nabla Q(\boldsymbol{w}_{i-1};\boldsymbol{x}_{e,i}), \tag{7}$$

where the notation $\boldsymbol{x}_{e,i}$ represents the realization of $\boldsymbol{x}_e$ that streams in at iteration i. Since $\boldsymbol{x}_{e,i}$ is selected from $\{x_1, x_2, \cdots, x_N\}$ at iteration i according to the pmf (5), we can rewrite $\boldsymbol{x}_{e,i}$ as $x_{\boldsymbol{n}_i}$ and replace (7) by

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu\nabla Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i}). \tag{8}$$

Here, the variable $\boldsymbol{n}_i$ is a uniform discrete random variable indicating the index of the sample that is picked at iteration i. Recursion (8) is the E-SGD algorithm (2). We therefore conclude that the E-SGD recursion is an O-SGD recursion applied to the solution of the stochastic optimization problem (6). This interpretation is useful because we can now call upon results from [11] for O-SGD and apply them to characterize the performance of E-SGD. This step is not as straightforward as it appears. This is because the results in [11], as is common in studies on stochastic optimization, rely on certain regularity conditions on the risk function and the gradient noise process. In order to be able to appeal to the earlier results from stochastic optimization theory, we need to verify first that problem (6) satisfies these regularity conditions. In preparation for the main results, we list two typical conditions on the empirical loss function.

Assumption 1 (CONDITION ON LOSS FUNCTION)**.** *It is assumed that $Q(w;x_n)$ is differentiable and has a δ_n-Lipschitz continuous gradient, i.e., for every $n = 1, \ldots, N$ and any $w_1, w_2 \in \mathbb{R}^M$:*

$$\|\nabla_w Q(w_1;x_n) - \nabla_w Q(w_2;x_n)\| \le \delta_n\|w_1 - w_2\|. \tag{9}$$

We also assume $J_e(w)$ is ν-strongly convex. ■

If we introduce $\delta = \max\{\delta_1, \delta_2, \cdots, \delta_N\}$, then each $\nabla_w Q(w;x_n)$ is also δ-Lipschitz continuous.

Assumption 2 (SMOOTHNESS CONDITION)**.** *It is assumed that $J_e(w)$ is twice differentiable and that the Hessian matrix of $J_e(w)$ is locally Lipschitz continuous in a small neighborhood around $w^\star$:*

$$\|\nabla_w^2 J_e(w^\star + \Delta w) - \nabla_w^2 J_e(w^\star)\| \le \kappa_e\|\Delta w\|, \tag{10}$$

where $\|\Delta w\| \le \epsilon$ and constant $\kappa_e \ge 0$. ■

2.2. Gradient noise and its moments

For the E-SGD algorithm (8), the gradient noise is given by

$$\boldsymbol{s}_i(\boldsymbol{w}_{i-1}) \triangleq \nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i}) - \nabla_w J_e(\boldsymbol{w}_{i-1}). \tag{11}$$

Let $\boldsymbol{\mathcal{F}}_{i-1}$ refer to the collection of all past iterates $\{\boldsymbol{w}_j, j < i\}$ and define

$$\begin{aligned} R_s &\triangleq \lim_{i\to\infty} \mathbb{E}[\boldsymbol{s}_i(w^\star)\boldsymbol{s}_i^{\mathsf{T}}(w^\star)|\boldsymbol{\mathcal{F}}_{i-1}] \\ &\overset{(a)}{=} \frac{1}{N}\sum_{n=1}^{N}[\nabla_w Q(w^\star;x_n)\nabla_w Q(w^\star;x_n)^{\mathsf{T}}], \end{aligned} \tag{12}$$

where (a) holds because $\nabla_w J_e(w^\star) = 0$. Based on this definition,

$$\mathrm{Tr}(R_s) = \frac{1}{N}\sum_{n=1}^{N}\|\nabla_w Q(w^\star;x_n)\|^2. \tag{13}$$

We now verify that the gradient noise process (11) has zero mean and its second-order moment improves as the iterate gets closer to the desired minimizer, $w^\star$. These are among the regularity conditions required in [11]. Here we show that this is not an assumption anymore for E-SGD but that it does actually hold.

Lemma 1 (GRADIENT NOISE PROPERTIES)**.** *The first, second and fourth-order moments of the gradient noise $\boldsymbol{s}_i(\boldsymbol{w}_{i-1})$ satisfy:*

$$\mathbb{E}[\boldsymbol{s}_i(\boldsymbol{w}_{i-1})|\boldsymbol{\mathcal{F}}_{i-1}] = 0, \tag{14}$$

$$\mathbb{E}[\|\boldsymbol{s}_i(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \le \beta_e^2\|\widetilde{w}_{i-1}\|^2 + \sigma_e^2, \tag{15}$$

$$\mathbb{E}[\|\boldsymbol{s}_i(\boldsymbol{w}_{i-1})\|^4|\boldsymbol{\mathcal{F}}_{i-1}] \le \beta_{e4}^4\|\widetilde{w}_{i-1}\|^2 + \sigma_{e4}^4, \tag{16}$$

where $\widetilde{w}_{i-1} = w^\star - w_{i-1}$ *and*

$$\beta_e^2 \triangleq 2\delta^2, \quad \sigma_e^2 \triangleq \frac{2}{N}\sum_{n=1}^{N}\|\nabla_w Q(w^\star; x_n)\|^2. \tag{17}$$

$$\beta_{e4}^4 \triangleq 128\delta^4, \quad \sigma_{e4}^4 \triangleq \frac{8}{N}\sum_{n=1}^{N}\|\nabla_w Q(w^\star; x_n)\|^4. \tag{18}$$

Proof. We first prove (14). Since $\boldsymbol{w}_{i-1} \in \boldsymbol{\mathcal{F}}_{i-1}$ and $\boldsymbol{n}_i$ is selected uniformly, it holds that

$$\mathbb{E}[\boldsymbol{s}_i(\boldsymbol{w}_{i-1})|\boldsymbol{\mathcal{F}}_{i-1}] = \frac{1}{N}\sum_{n=1}^{N}\nabla_w Q(w_{i-1}; x_n) - \nabla_w J_e(w_{i-1}) = 0. \tag{19}$$

Next we establish (15). Using Jensen's inequality:

$$\begin{aligned}
&\mathbb{E}[\|\boldsymbol{s}_i(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \\
=&\mathbb{E}[\|\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) - \nabla_w J_e(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \\
\le& 2\mathbb{E}[\|\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) - \nabla_w Q(w^\star; x_{\boldsymbol{n}_i}) - \nabla_w J_e(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \\
&+ 2\mathbb{E}[\|\nabla_w Q(w^\star; x_{\boldsymbol{n}_i})\|^2|\boldsymbol{\mathcal{F}}_{i-1}].
\end{aligned} \tag{20}$$

Noting that for any random variable $\boldsymbol{x}$:

$$\mathbb{E}\|\boldsymbol{x} - \mathbb{E}\boldsymbol{x}\|^2 = \mathbb{E}\|\boldsymbol{x}\|^2 - \|\mathbb{E}\boldsymbol{x}\|^2 \le \mathbb{E}\|\boldsymbol{x}\|^2, \tag{21}$$

and using $\nabla_w J_e(w^\star) = 0$, we have

$$\begin{aligned}
&\mathbb{E}[\|\underbrace{\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) - \nabla_w Q(w^\star; x_{\boldsymbol{n}_i})}_{\boldsymbol{x}} - \underbrace{\nabla_w J_e(\boldsymbol{w}_{i-1})}_{\mathbb{E}\boldsymbol{x}}\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \\
&\le \mathbb{E}[\|\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) - \nabla_w Q(w^\star; x_{\boldsymbol{n}_i})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \\
&\overset{(a)}{\le} \delta^2\|w_{i-1} - w^\star\|^2 = \delta^2\|\widetilde{w}_{i-1}\|^2,
\end{aligned} \tag{22}$$

where (a) holds because of Assumption 1. Substituting (22) into (20), we obtain (15). A similar argument can be used to establish the fourth-order moment property (16), which we omit for brevity. □

2.3. Mean-square stability and performance of E-SGD

We can now appeal directly to Lemma 3.1 from [11] to conclude the mean-square stability of E-SGD.

Theorem 1 (MEAN-SQUARE-ERROR STABILITY OF E-SGD)**.** *Under Assumption 1 and any step-size satisfying* $\mu < 2\nu/(\delta^2 + \beta_e^2) = 2\nu/3\delta^2$*, it holds that*

$$\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|^2 \le \alpha^i\,\mathbb{E}\|w_0 - w^\star\|^2 + O(\mu). \tag{23}$$

where $\alpha = 1 - 2\nu\mu + 3\delta^2\mu^2 \in (0, 1)$. ■

Theorem 1 states that E-SGD converges exponentially fast to a small neighborhood around $w^\star$ of size O(μ). From (23), it is also observed that E-SGD is not sensitive to the initial staring point w_0 because $\|w_0 - w^\star\|^2$ will diminish exponentially fast. Theorem 1 does not provide an accurate expression for the steady-state performance of E-SGD. More is needed to arrive at this expression. Here we appeal to Theorem 4.7 from [11] to derive the expression for the steady-state performance of E-SGD. By steady-state we mean the algorithm is applied repeatedly in random passes over the finite training data. We denote the Hessian of the empirical risk (1) at $w^\star$ by

$$H \triangleq \nabla_w^2 J_e(w^\star). \tag{24}$$

Theorem 2 (STEADY-STATE PERFORMANCE)**.** *Assume the conditions under Assumptions 1 and 2 hold. When the step-size is sufficiently small, the MSD and ER metrics to first-order in* μ *for the E-SGD algorithm* (2) *are given by the following expressions:*

$$\text{MSD} \triangleq \limsup_{i\to\infty}\mathbb{E}[\|\boldsymbol{w}_i - \boldsymbol{w}^\star\|^2] = \frac{\mu}{2}\text{Tr}(H^{-1}R_s), \tag{25}$$

$$\text{ER} \triangleq \limsup_{i\to\infty}\mathbb{E}[J_e(\boldsymbol{w}_i) - J_e(w^\star)] = \frac{\mu}{4}\text{Tr}(R_s), \tag{26}$$

where H *is defined in* (24) *and* R_s *is defined in* (12)*.* ■

The MSD expression (25) is tighter than the bound given in [19], which is written as

$$\limsup_{i\to\infty}\mathbb{E}\|\widetilde{\boldsymbol{w}}_i\|^2 \le \frac{\mu\text{Tr}(R_s)}{\nu(1 - \mu\max_n\{\delta_n\})} = \frac{\mu\text{Tr}(R_s)}{\nu(1-\mu\delta)}. \tag{27}$$

Since $J_e(w)$ is ν-strongly convex, we have $H \ge \nu I$. Therefore,

$$\frac{\mu}{2}\text{Tr}(H^{-1}R_s) \le \frac{\mu}{2\nu}\text{Tr}(R_s) < \frac{\mu\text{Tr}(R_s)}{\nu(1-\mu\delta)}, \tag{28}$$

where the last inequality holds because $1 - \mu\delta < 2$. Relation (28) shows that our MSD expression (25) is tighter. As a result, expressions (25) and (26) are more helpful to determine a proper step-size when a certain accuracy ϵ is required for the MSD or ER performance.

3. MINI-BATCH GRADIENT DESCENT

In this section, we derive the MSD and ER performance expressions for mini-batch gradient descent. Following arguments similar to Section 2, we will also interpret mini-batch gradient descent as a special case of O-SGD, and then refer to the theoretical results in [11].

Suppose we have B independent discrete random variables $\boldsymbol{x}_e^{(1)}, \boldsymbol{x}_e^{(2)}, \cdots, \boldsymbol{x}_e^{(B)}$, each with the same distribution as $\boldsymbol{x}_e$ that is defined from Section 2.1. Using these variables, we define

$$\mathcal{Q}(w; \{\boldsymbol{x}_e^{(1)}, \boldsymbol{x}_e^{(2)}, \cdots, \boldsymbol{x}_e^{(B)}\}) \triangleq \frac{1}{B}\sum_{j=1}^{B} Q(w; \boldsymbol{x}_e^{(j)}), \tag{29}$$

and note that

$$\begin{aligned}
&\mathbb{E}[\mathcal{Q}(w; \{\boldsymbol{x}_e^{(1)}, \boldsymbol{x}_e^{(2)}, \cdots, \boldsymbol{x}_e^{(B)}\})] \\
&\overset{(29)}{=} \frac{1}{B}\sum_{j=1}^{B}\mathbb{E}[Q(w; \boldsymbol{x}_e^{(j)})] \overset{(6)}{=} \frac{1}{B}\sum_{j=1}^{B} J_e(w) = J_e(w).
\end{aligned} \tag{30}$$

It follows that the empirical problem (1) can also be rewritten as:

$$\min_{w\in\mathbb{R}^M} J_e(w) = \mathbb{E}[\mathcal{Q}(w; \{\boldsymbol{x}_e^{(1)}, \boldsymbol{x}_e^{(2)}, \cdots, \boldsymbol{x}_e^{(B)}\})], \tag{31}$$

which is a stochastic optimization problem. We can therefore apply the O-SGD algorithm to seek its minimizer, which leads to the mini-batch gradient descent algorithm:

$$\begin{aligned}
\boldsymbol{w}_i &= \boldsymbol{w}_{i-1} - \mu\nabla_w\mathcal{Q}(\boldsymbol{w}_{i-1}; \{\boldsymbol{x}_{e,i}^{(1)}, \boldsymbol{x}_{e,i}^{(2)}, \cdots, \boldsymbol{x}_{e,i}^{(B)}\}) \\
&= \boldsymbol{w}_{i-1} - \frac{\mu}{B}\sum_{j=1}^{B}\nabla_w Q(\boldsymbol{w}_{i-1}; \boldsymbol{x}_{e,i}^{(j)}) \\
&\overset{(a)}{=} \boldsymbol{w}_{i-1} - \frac{\mu}{B}\sum_{j=1}^{B}\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i(j)}),
\end{aligned} \tag{32}$$

where $\boldsymbol{x}_{e,i}^{(j)}$ is the instantaneous realization of the random variable

$\boldsymbol{x}_e^{(j)}$ at iteration i. Equality (a) holds because, similarly to (7) and (8), we redefine $\boldsymbol{x}_{e,i}^{(j)}$ as $x_{\boldsymbol{n}_i(j)}$ where the random variables $\{\boldsymbol{n}_i(j)\}_{j=1}^B$ are mutually independent with the same pmf as $\boldsymbol{n}_i$. During the implementation of recursion (32), we will sample B data with replacement at each iteration, and then compute their average.

3.1. Gradient noise and its moments

According to the mini-batch recursion (32), the gradient noise is

$$\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})=\frac{1}{B}\sum_{j=1}^{B}\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(j)})-\nabla_w J_e(\boldsymbol{w}_{i-1}). \quad (33)$$

The following result extends Lemma 1 to the mini-batch method.

Lemma 2 (GRADIENT NOISE PROPERTIES)**.** *The first and second-order moments of the gradient noise $\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})$ defined in* (33) *satisfy:*

$$\mathbb{E}[\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})|\boldsymbol{\mathcal{F}}_{i-1}]=0, \quad (34)$$

$$\mathbb{E}[\|\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}]\leq\beta_b^2\|\widetilde{w}_{i-1}\|^2+\sigma_b^2, \quad (35)$$

$$\mathbb{E}[\|\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})\|^4|\boldsymbol{\mathcal{F}}_{i-1}]\leq\beta_{b4}^4\|\widetilde{w}_{i-1}\|^2+\sigma_{b4}^4, \quad (36)$$

where $\widetilde{w}_{i-1}=w^\star-w_{i-1}$,

$$\beta_b^2\ \triangleq\ \beta_e^2/B,\qquad \sigma_b^2\ \triangleq\ \sigma_e^2/B, \quad (37)$$

$$\beta_{b4}^4\ \triangleq\ \beta_{e4}^4,\qquad \sigma_{b4}^4\ \triangleq\ \sigma_{e4}^4. \quad (38)$$

and β_e^2, σ_e^2, β_{b4}^4, and σ_{b4}^4 are defined in Lemma 1. ■

Proof. The argument for (34) is similar to (19). To prove (35), we start by noting that

$$\begin{aligned}
&\mathbb{E}[\|\boldsymbol{s}_i^b(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}]\\
&=\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j=1}^{B}\Big(\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(j)})-\nabla J_e(\boldsymbol{w}_{i-1})\Big)\right\|^2\Bigg|\boldsymbol{\mathcal{F}}_{i-1}\right]\\
&=\frac{1}{B^2}\mathbb{E}\left[\sum_{j=1}^{B}\left\|\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(j)})-\nabla J_e(\boldsymbol{w}_{i-1})\right\|^2\Bigg|\boldsymbol{\mathcal{F}}_{i-1}\right]\\
&\quad+\frac{1}{B^2}\mathbb{E}\left[\sum_{j=1}^{B}\sum_{k\neq j}\Big(\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(j)})-\nabla J_e(\boldsymbol{w}_{i-1})\Big)^{\mathsf{T}}\right.\\
&\qquad\left.\Big(\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(k)})-\nabla J_e(\boldsymbol{w}_{i-1})\Big)\Big|\boldsymbol{\mathcal{F}}_{i-1}\right]\\
&\overset{(a)}{=}\frac{1}{B^2}\sum_{j=1}^{B}\mathbb{E}\left[\left\|\nabla_w Q(\boldsymbol{w}_{i-1};x_{\boldsymbol{n}_i(j)})-\nabla J_e(\boldsymbol{w}_{i-1})\right\|^2\Bigg|\boldsymbol{\mathcal{F}}_{i-1}\right]
\end{aligned} \quad (39)$$

where (a) holds because when $k\neq j$, $\boldsymbol{n}_i(j)$ is independent of $\boldsymbol{n}_i(k)$. Now using property (15) from Lemma 1 in (39) gives (35). The derivation for the fourth-order moment result follows from Jensen's inequality and property (16). We omit the proof for brevity. □

One important observation is that, with the mini-batch technique, the magnitude of the second-order moment of the gradient noise is reduced to $1/B$ of its original magnitude (see (37)), which suggests that we should expect both the MSD and ER of mini-batch implementations to improve by a factor of B. The analysis in the next section confirms this conclusion.

3.2. Performance of mini-batch gradient descent

First, the limiting covariance matrix of the mini batch gradient noise process is given by

$$R_s^b=\frac{1}{B^2}\sum_{j=1}^{B}R_s=\frac{1}{B}R_s. \quad (40)$$

Then, using Theorem 4.7 from [11] we deduce the following.

Theorem 3 (STEADY-STATE PERFORMANCE)**.** *Under Assumptions 1 and 2, for a sufficiently small step-size, the MSD and ER metrics for the mini-batch method* (32) *are given by:*

$$\mathrm{MSD}_b\ =\ \frac{\mu}{2}\mathrm{Tr}(H^{-1}R_s^b)=\frac{\mu}{2B}\mathrm{Tr}(H^{-1}R_s), \quad (41)$$

$$\mathrm{ER}_b\ =\ \frac{\mu}{4}\mathrm{Tr}(R_s^b)=\frac{\mu}{4B}\mathrm{Tr}(R_s), \quad (42)$$

where H is defined in (24) *and R_s is defined in* (40)*. Moreover, the algorithm converges at an exponential rate:*

$$\alpha_b=1-2\nu\mu+(1+2/B)\delta^2\mu^2. \quad (43)$$

■

4. OPTIMAL IMPORTANCE SAMPLING

We have assumed so far that the data samples in an empirical SGD implementation are selected uniformly at random, according to (5). However, we can consider other selection policies in order to enhance performance. The works [19, 22] proposed to measure the importance of each sample according to its Lipschitz constant δ_n in (9). Specifically, they suggest selecting the sampling probability according to

$$p(n)=\frac{\delta_n}{\sum_{m=1}^{N}\delta_m}, \quad (44)$$

where $p(n)$ is the sampling probability of data x_n. This scheme assumes knowledge of the Lipschitz constants, which is usually not available in advance or even known. Moreover, this importance sampling method is not optimal, as the ensuing discussion will show where we derive the optimal sampling algorithm.

Let us denote the new pmf for the random variable $\boldsymbol{n}$ that we wish to determine optimally by

$$p(\boldsymbol{n})=\begin{cases}\alpha(1), & \text{if } \boldsymbol{n}=1,\\ \alpha(2), & \text{if } \boldsymbol{n}=2,\\ \vdots & \vdots\\ \alpha(N), & \text{if } \boldsymbol{n}=N,\end{cases} \quad (45)$$

where $\alpha(n)$ is the sampling probability for data x_n, and it holds that the $\{\alpha(n)\}$ add up to one. With this new pmf for $\boldsymbol{n}$, the empirical problem (1) can be interpreted as the following stochastic optimization problem

$$\min_{w\in\mathbb{R}^M}\ J_e(w)=\sum_{n=1}^{N}\frac{\alpha(n)}{\alpha(n)N}Q(w;x_n)=\mathbb{E}_{\boldsymbol{n}}[Q^l(w;x_{\boldsymbol{n}})], \quad (46)$$

where we defined

$$Q^l(w;x_n)\ \triangleq\ \frac{1}{\alpha(n)N}Q(w;x_n). \quad (47)$$

Now if we apply O-SGD to solve problem (46), we obtain the fol-

lowing importance sampling recursion:

$$\begin{aligned}\boldsymbol{w}_i &= \boldsymbol{w}_{i-1} - \mu\nabla_w Q^l(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) \\ &= \boldsymbol{w}_{i-1} - \frac{\mu}{p(\boldsymbol{n}_i)N}\nabla_w Q(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}).\end{aligned} \tag{48}$$

Next we will explain how to choose $p(\boldsymbol{n}_i)$ such that the above recursion can reach optimal steady-state performance. First, the gradient noise of the importance sampling approach is given by:

$$\boldsymbol{s}_i^l(\boldsymbol{w}_{i-1}) = \nabla_w Q^l(\boldsymbol{w}_{i-1}; x_{\boldsymbol{n}_i}) - \nabla_w J_e(\boldsymbol{w}_{i-1}) \tag{49}$$

Lemma 3 (GRADIENT NOISE PROPERTY)**.** *The gradient noise process in (49) satisfies the following conditions:*

$$\mathbb{E}[\boldsymbol{s}_i^l(\boldsymbol{w}_{i-1})|\boldsymbol{\mathcal{F}}_{i-1}] = 0, \tag{50}$$

$$\mathbb{E}[\|\boldsymbol{s}_i^l(\boldsymbol{w}_{i-1})\|^2|\boldsymbol{\mathcal{F}}_{i-1}] \le \beta_l^2\|\widetilde{w}_{i-1}\|^2 + \sigma_l^2, \tag{51}$$

where $\widetilde{w}_{i-1} = w^\star - w_{i-1}$ *and*

$$\beta_l^2 \triangleq 2\sum_{n=1}^N \frac{\delta^2}{\alpha(n)N^2}, \quad \sigma_l^2 \triangleq 2\sum_{n=1}^N \frac{1}{\alpha(n)N^2}\|\nabla_w Q(w^\star; x_n)\|^2$$

■

This Lemma can be established by following arguments similar to those used in Lemmas 1 and 2. Now, calling upon Theorem 4.7 from [11] we arrive at the following expression for the proposed importance sampling recursion (48):

$$\mathrm{ER}_l = \frac{\mu}{4}\mathrm{Tr}(R_s^l) = \frac{\mu}{4}\sum_{n=1}^N \frac{1}{\alpha(n)N^2}\|\nabla_w Q(w^\star; x_n)\|^2 \tag{52}$$

We can minimize this expression over the $\{\alpha(n)\}$ and solve:

$$\min_{\alpha} \quad \sum_{n=1}^N \frac{1}{\alpha(n)}\|\nabla_w Q(w^\star; x_n)\|^2 \tag{53}$$

$$\text{s.t.} \quad \sum_{n=1}^N \alpha(n) = 1, \;\; 0 \le \alpha(n) \le 1, \quad n = 1, 2, \cdots, N.$$

Fortunately, this problem has a closed-form solution, which can be derived by the Lagrangian multiplier method:

$$\alpha^\star(n) = \frac{\|\nabla_w Q(w^\star; x_n)\|}{\sum_{m=1}^N \|\nabla_w Q(w^\star; x_m)\|} \tag{54}$$

Substituting into (52) yields the optimal ER value:

$$\mathrm{ER}_l^\star = \frac{\mu}{4}\left(\sum_{n=1}^N \frac{1}{N}\|\nabla_w Q(w^\star; x_n)\|\right)^2 \tag{55}$$

From Jensen's inequality, we can verify that $\mathrm{ER}_l^\star$ is always smaller than or equal to the ER of E-SGD derived earlier in (26).

Although we determined the optimal pmf in (55), one practical problem is that the expression for $\alpha^\star(n)$ depends on the unknown $w^\star$. This problem can be overcome by replacing the minimizer by its estimate, which leads to an adaptive importance sampling method:

$$\alpha_i(n) = \frac{\|\nabla_w Q(w_{i-1}; x_n)\|}{\sum_{m=1}^N \|\nabla_w Q(w_{i-1}; x_m)\|}. \tag{56}$$

Expression (56) is still inefficient to update because at each iteration we have to compute $\|\nabla_w Q(w_{i-1}; x_n)\|$ for all data samples and then calculate the average. To reach an efficient update, we introduce an auxiliary variable $\boldsymbol{\psi} \in \mathbb{R}^N$, with its nth entry updated as follows:

$$\boldsymbol{\psi}_i(n) = \begin{cases}\gamma\boldsymbol{\psi}_{i-1}(n) + (1-\gamma)\|\nabla_w Q(\boldsymbol{w}_{i-1}; x_n)\|, & \text{if } n = \boldsymbol{n}_i \\ \boldsymbol{\psi}_{i-1}(n), & \text{if } n \ne \boldsymbol{n}_i\end{cases} \tag{57}$$

where $\gamma \in (0, 1)$; in the simulations we selected $\gamma \in (0.1, 0.5)$. Note that each entry $\boldsymbol{\psi}_i(n)$ is an estimate of $\|\nabla_w Q(w_i; x_n)\|$. Note further that at iteration i, only one entry of $\boldsymbol{\psi}_i$ is updated, and hence this update is cheap. We also update a scalar θ to maintain the sum of $\boldsymbol{\psi}$. Suppose $\boldsymbol{n}_i$ is picked up at iteration i, then

$$\begin{aligned}\theta_i &= \sum_{n=1}^N \boldsymbol{\psi}_i(n) = \sum_{n=1}^N \boldsymbol{\psi}_{i-1}(n) + \boldsymbol{\psi}_i(\boldsymbol{n}_i) - \boldsymbol{\psi}_{i-1}(\boldsymbol{n}_i) \\ &\overset{(57)}{=} \theta_{i-1} + (1-\gamma)\left(\|\nabla_w Q(\boldsymbol{w}_{i-1}; x_n)\| - \boldsymbol{\psi}_{i-1}(\boldsymbol{n}_i)\right).\end{aligned} \tag{58}$$

Note that each update of θ only requires $O(1)$ operations, which is also cheap. The algorithm is summarized in the following table, where $p_i \in \mathbb{R}^N$ is the sampling probability vector with each entry $p_i(n)$ indicating the probability that data x_n is selected.

Optimal adaptive importance sampling for SGD

Initialization:
  $\boldsymbol{\psi}_0$ is initialized to be some large positive vector;
  θ_0 is initialized as the sum of the entires in $\boldsymbol{\psi}_0$;
  p_0 is initialized as uniform distribution;
for $i = 1, 2, 3, \ldots$
  Pick $\boldsymbol{n}_i$ according to sampling probability p_{i-1};
  Update $\boldsymbol{\psi}_i$ and θ_i according to (57) and (58) respectively;
  Update sampling probability $p_i = \boldsymbol{\psi}_i/\theta_i$;
  Update $\boldsymbol{w}_i$ according to (48)
end

In the above algorithm, we initialize $\boldsymbol{\psi}_0$ to large entries so that p_i is not too small for some indices. In this way, we can guarantee that all data samples are accessed with large enough probability during the initial stages. The key feature of this algorithm is that it does not depend on any pre-knowledge of each data sample (such as the Lipschitz constants needed in [19, 22]), and can automatically learn the optimal sampling probability distribution. Moreover, the algorithm is very efficient in computational cost.

5. NUMERICAL EXPERIMENTS

We illustrate the results by considering the regularized logistic regression problem:

$$J_e(w) = \frac{\rho}{2}\|w\|^2 + \frac{1}{N}\sum_{n=1}^N \ln\left(1 + \exp(-\gamma(n)h_n^{\mathsf{T}}w)\right), \tag{59}$$

where $h_n \in \mathbb{R}^{10}$ is the feature and $\gamma(n) \in \{\pm 1\}$ is the label scalar. In the simulation, we generate a random data set $\{h_n, \gamma(n)\}$ with $N = 500$. We set $\rho = 0.01$ and $\mu = 0.01$. We run the empirical SGD and mini-batch algorithms over 25 epochs. All simulation results shown below are averaged over 100 trials. From Fig 1, it is clear that our bound is significantly tighter than the bound from [19], which is also shown in (27). We also observe that the MSD performance is inversely proportionally to the size of the mini-batch, as predicted by Theorem 3. Moreover, the figure shows that our theoretical performance expressions match well with the simulated results.

Next, in Fig. 2 we illustrate the behavior of our optimal importance sampling algorithm with the same problem setting. All algorithms use a 10 mini-batch size. The red curve is the standard SGD learning curve, which is used as reference; The blue curve is

using the fixed optimal importance sampling probability, which is precalculated with the $w^\star$ information (54). The green curve is our proposed adaptive importance sampling method, which is seen to be as good as the optimal solution. We also compare against the resampling technique from [19,22], which use the Lipschitz constants. The result is the black curve, which is only matching the performance of the standard SGD implementation and is away from the optimal performance.

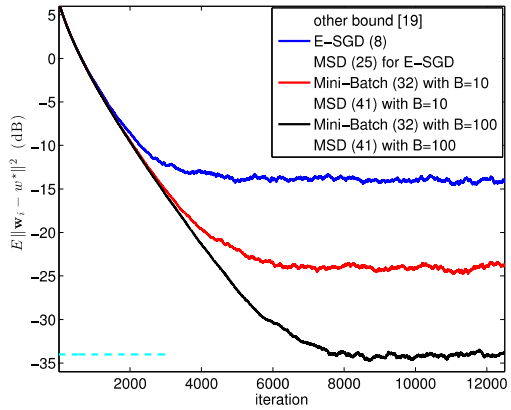

Fig. 1. Convergence behavior of mini-batch SGD for regularized logistic regression problem. B indicates the size of the mini-batch.

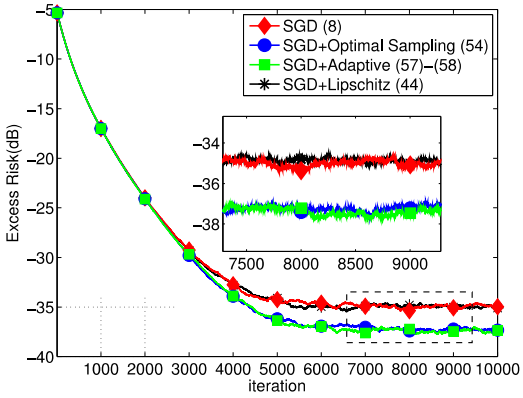

Fig. 2. Optimal adaptive importance sampling algorithm for regularized logistic regression problem.

6. CONCLUSION

This paper establishes a useful connection between empirical stochastic gradient methods for learning from finite data samples, and online stochastic gradient methods for learning from streaming data. Using performance expressions for the excess risk (ER), an optimal sampling strategy is devised to attain the best ER performance. Simulation runs illustrate the results.

7. REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice Hall, NJ, 1989.

[2] B. T. Polyak and A. B. Juditsky, "Acceleration of stochastic approximation by averaging," *SIAM Journal on Control and Optimization*, vol. 30, no. 4, pp. 838–855, 1992.

[3] B. T. Polyak, *Introduction to Optimization*, Optimization Software, New York, 1987.

[4] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. International Conference on Computational Statistics* (COMPSTAT), Paris, France, 2010, pp. 177–186.

[5] O. Bousquet and L. Bottou, "The tradeoffs of large scale learning," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Vancouver, Canada, 2008, pp. 161–168.

[6] E. Moulines and F. R. Bach, "Non-asymptotic analysis of stochastic approximation algorithms for machine learning," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Granada, Spain, 2011, pp. 451–459.

[7] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Proc. International Conference on Machine Learning* (ICML), Alberta, Canada, 2004, pp. 116–124.

[8] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM Journal on Optimization*, vol. 19, no. 4, pp. 1574–1609, 2009.

[9] S. S. Haykin, *Adaptive Filter Theory*, Fourth Edition, Prentice-Hall, NJ, 2008.

[10] A. H. Sayed, *Adaptive Filters*, Wiley, NY, 2008.

[11] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4–5, pp. 311–801, 2014.

[12] M. V. Solodov, "Incremental gradient algorithms with stepsizes bounded away from zero," *Computational Optimization and Applications*, vol. 11, no. 1, pp. 23–35, 1998.

[13] A. Nedić and D. P. Bertsekas, "Convergence rate of incremental subgradient algorithms," in *Stochastic Optimization: Algorithms and Applications*, pp. 223–264. Springer, 2001.

[14] Y. Koren, "Factorization meets the neighborhood: a multifaceted collaborative filtering model," in *Proc. International Conference on Knowledge Discovery and Data Mining* (SIGKDD), Las Vegas, USA, 2008, pp. 426–434.

[15] I. Sutskever, J. Martens, G. Dahl, and G. Hinton, "On the importance of initialization and momentum in deep learning," in *Proc. International Conference on Machine Learning* (ICML), Atlanta, USA, 2013, pp. 1139–1147.

[16] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, April 2014.

[17] M. Rupp and A. H. Sayed, "A time-domain feedback analysis of filtered-error adaptive gradient algorithms," *IEEE Transactions on Signal Processing*, vol. 44, no. 6, pp. 1428–1439, 1996.

[18] A. H. Sayed and M. Rupp, "An ℓ_2 stable feedback structure for nonlinear adaptive filtering and identification," *Automatica*, vol. 33, no. 1, pp. 13–30, Jan. 1997.

[19] D. Needell, R. Ward, and N. Srebro, "Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Montreal, Canada, 2014, pp. 1017–1025.

[20] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: Primal estimated sub-gradient solver for SVM," *Mathematical Programming*, vol. 127, no. 1, pp. 3–30, 2011.

[21] A. Cotter, O. Shamir, N. Srebro, and K. Sridharan, "Better mini-batch algorithms via accelerated gradient methods," in *Proc. Advances in Neural Information Processing Systems* (NIPS), Granada, Spain, 2011, pp. 1647–1655.

[22] P. Zhao and T. Zhang, "Stochastic optimization with importance sampling for regularized loss minimization," in *Proc. International Conference on Machine Learning* (ICML), Lille, France, 2015, pp. 1355–1363.