High probability guarantees for stochastic convex optimization

Damek Davis DSD95@CORNELL.EDU
School of ORIE, Cornell University, Ithaca, NY 14850, USA

Dmitriy Drusvyatskiy DDRUSV@UW.EDU
Department of Mathematics, University of Washington

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

Standard results in stochastic convex optimization bound the number of samples that an algorithm needs to generate a point with small function value in expectation. More nuanced *high probability* guarantees are rare, and typically either rely on "light-tail" noise assumptions or exhibit worse sample complexity. In this work, we show that a wide class of stochastic optimization algorithms for strongly convex problems can be augmented with high confidence bounds at an overhead cost that is only logarithmic in the confidence level and polylogarithmic in the condition number. The procedure we propose, called `proxBoost`, is elementary and builds on two well-known ingredients: robust distance estimation and the proximal point method. We discuss consequences for both streaming (online) algorithms and offline algorithms based on empirical risk minimization.

Keywords: Proximal point, robust distance estimation, stochastic approximation, empirical risk

1. Introduction

Stochastic convex optimization lies at the core of modern statistical and machine learning. Standard results in the subject bound the number of samples that an algorithm needs to generate a point with small function value in *expectation*. Specifically, consider the problem

$$\min_x \; f(x) := \mathbb{E}_{z\sim\mathcal{P}}[f(x,z)], \tag{1}$$

where the random variable z follows a fixed unknown distribution $\mathcal{P}$ and $f(\cdot, z)$ is convex for a.e. $z \sim \mathcal{P}$. Given a tolerance $\epsilon > 0$, stochastic gradient methods produce a point x_ϵ satisfying $\mathbb{E}[f(x_\epsilon) - \min f] \leq \epsilon$. The cost of the algorithms, measured by the number of stochastic gradient evaluations, is $\mathcal{O}(1/\epsilon^2)$. The cost improves to $\mathcal{O}(1/\epsilon)$ if f is strongly convex (e.g. Nemirovsky and Yudin (1983); Polyak and Juditsky (1992); Ghadimi and Lan (2013); Hazan and Kale (2014)).

In this paper, we are interested in procedures that can produce an approximate solution with *high probability*, meaning a point $x_{\epsilon,p}$ satisfying

$$\mathbb{P}(f(x_{\epsilon,p}) - \min f \leq \epsilon) \geq 1 - p, \tag{2}$$

where $p > 0$ can be arbitrarily small. By Markov's inequality, one can guarantee (2) by generating a point $x_{\epsilon,p}$ with $\mathbb{E}[f(x_{\epsilon,p}) - \min f] \leq p\epsilon$. However, the resulting sample complexity can be very high for small p with the typical scaling of $\mathcal{O}(1/(p\epsilon))$ or $\mathcal{O}(1/(p\epsilon)^2)$. Existing literature does provide a path to reducing the dependence of the sample complexity on p to $\log(1/p)$, but this usually comes with cost of either worse dependence on ϵ (e.g., Bousquet and Elisseeff (2002); Nesterov and Vial

(2008); Shalev-Shwartz et al. (2009)) or more restrictive sub-Gaussian assumptions on the noise (e.g. Nemirovski et al. (2008); Juditsky and Nesterov (2014); Ghadimi and Lan (2012, 2013)).

The goal of this work. We aim to develop *generic* low-cost procedures that equip stochastic optimization algorithms with high confidence guarantees, without making restrictive noise assumptions. Consequently, it will be convenient to treat such algorithms as black boxes. More formally, suppose that the function f may only be accessed through a *minimization oracle* $\mathcal{M}(f, \epsilon)$, which on input $\epsilon > 0$, returns a point x_ϵ satisfying the low confidence bound

$$\mathbb{P}(f(x_\epsilon) - \min f \leq \epsilon) \geq \frac{2}{3}. \tag{3}$$

By Markov's inequality, minimization oracles arise from any algorithm that can generate x_ϵ satisfying $\mathbb{E}[f(x_\epsilon) - \min f] \leq \epsilon/3$. Let $\mathcal{C}_\mathcal{M}(f, \epsilon)$ denote the cost of the oracle call $\mathcal{M}(f, \epsilon)$. Given a minimization oracle and its cost, we investigate the following question:

> Is there a procedure within this oracle model of computation that returns a point $x_{\epsilon,p}$ satisfying the high confidence bound (2) at a cost on the order of $\mathcal{C}_\mathcal{M}(f, \epsilon) \cdot \log(\frac{1}{p})$?

We will see that when f is smooth and strongly convex, the answer is yes for a wide class of oracles $\mathcal{M}(f, \epsilon)$. Henceforth, suppose that f is μ-strongly convex with L-Lipschitz continuous gradient. Then the cost $\mathcal{C}_\mathcal{M}(f, \epsilon)$ typically depends on the condition number $\kappa := L/\mu \gg 1$, as well as scale sensitive quantities such as initialization quality and upper bound on the gradient variances, etc. The procedures introduced in this paper execute the minimization oracle multiple times in order to boost its confidence, with the total cost on the order of

$$\log\left(\frac{\log(\kappa)}{p}\right) \log(\kappa) \cdot \mathcal{C}_\mathcal{M}\left(f, \tfrac{\epsilon}{\log(\kappa)}\right).$$

Thus, high probability bounds are achieved with a small cost increase, which depends only logarithmically on $1/p$ and polylogarithmically on the condition number κ.

Known techniques and limitations. Before introducing our approach, we discuss two techniques for boosting the confidence of a minimization oracle, both of which have limitations. As a first approach, one may query the oracle $\mathcal{M}(f, \epsilon)$ multiple times and pick the "best" iterate from the batch. This strategy is flawed since testing which iterate is "best" is often costly. To illustrate, consider estimating the value $f(x) = \mathbb{E}_z[f(x, z)]$ to ϵ-accuracy for a fixed x—a mean estimation problem. Even under sub-Gaussian assumptions, this task may require on the order of $1/\epsilon^2$ samples Catoni (2012). In this paper, the cost $\mathcal{C}_\mathcal{M}(f, \epsilon)$ scales at worst as $1/\epsilon$, and therefore mean estimation would significantly degrade the sample complexity.

The second approach leverages that with strong convexity (3) implies

$$\mathbb{P}(\|x_\epsilon - \bar{x}\| \leq \sqrt{2\epsilon/\mu}) \geq \frac{2}{3},$$

where $\bar{x}$ is the minimizer of f. Given this bound, one may apply the *robust distance estimation* technique of (Nemirovsky and Yudin, 1983, p. 243) and Hsu and Sabato (2016) to choose a point near $\bar{x}$: Run m trials of $\mathcal{M}(f, \epsilon)$ and find one iterate x_{i^*} around which the other points "cluster". Then the point x_{i^*} will be within a distance of $\sqrt{18\epsilon/\mu}$ from $\bar{x}$ with probability $1 - \exp(-m/18)$. The downside of this strategy is that when converting naively back to function values, the suboptimality gap becomes $f(x_{i^*}) - \min f \leq \frac{L}{2}\|x_{i^*} - \bar{x}\|^2 \leq 9\kappa\epsilon$. Thus the function gap at x_{i^*} may be significantly larger than the expected gap at x_ϵ, by a factor of the condition number.

1.1. Contribution: The `proxBoost` algorithm

The procedure we introduce, called `proxBoost` , is based on the following simple observation: although robust distance estimation induces a trade-off between robustness and efficiency, this trade-off disappears for perfectly conditioned losses. Leveraging this fact, we design a continuation procedure that links together a short sequence of robust distance estimators for nearby problems with rapidly improving condition numbers. More specifically, the `proxBoost` algorithm generates a sequence of iterates $x_0, \ldots, x_T$. The first iterate, x_0, is simply the output of the robust distance estimator for minimizing f. Subsequently, given an iterate x_t, the procedure forms the better conditioned function $f^t(x) := f(x) + \frac{\mu 2^t}{2}\|x - x_t\|^2$ and declares the next iterate x_{t+1} to be the output of the robust distance estimator for minimizing f^t. We may in principle apply `proxBoost` with any minimization oracle $\mathcal{M}(f^t, \epsilon)$. The real benefit arises for concrete oracles, such as those based on streaming algorithms (e.g., stochastic gradient) or offline methods (e.g., empirical risk minimization), for which the cost of computing the robust distance estimator rapidly decreases as t increases and conditioning improves. **When used within the `proxBoost` method, these oracles benefit from new high confidence guarantees with only a modest logarithmic and polylogarithmic cost increase in $1/p$ and κ, respectively.** We now illustrate this claim.

1.1.1. STREAMING ALGORITHMS WITH HEAVY TAILS: HIGH CONFIDENCE ACCELERATION

Stochastic gradient methods may serve as minimization oracles $\mathcal{M}(f, \epsilon)$ in the `proxBoost` method. For these oracles, the cost $\mathcal{C}_{\mathcal{M}}(f, \epsilon)$ is measured by the number of stochastic gradient estimates that the algorithm must generate in order to reach functional accuracy ε in expectation. Although many such oracles exist and may be used within `proxBoost`, our goal is to use the *optimal* algorithm of Ghadimi and Lan (2013) as an oracle and and equip it with high confidence guarantees. This algorithm is optimal in the sense that it has minimal cost among stochastic gradient methods within a standard oracle model of computation. More specifically, the method generates a point x_ϵ satisfying $\mathbb{E}\left[f(x_\epsilon) - \min f\right] \leq \epsilon$ with $\mathcal{O}\left(\sqrt{\kappa}\ln\left(\Delta_{\text{in}}/\epsilon\right) + \sigma^2/\mu\epsilon\right)$ stochastic gradient estimates, where σ^2 and Δ_{in} are upper bounds on the variance of $\nabla f(x, z)$ and the initial function gap $f(x_0) - f^*$, respectively. In their original work—still the state-of-the-art—Ghadimi and Lan (2013, 2012) gave a variant of this algorithm with high confidence guarantees, but their result relies on a crucial assumption: the gradients must have light (subgaussian) tails.

Without an explicit light-tails assumption, it is unknown whether there is a high confidence analog of Ghadimi and Lan (2013) that preserves its optimal complexity. As a first attempt, one might try to adapt the robust distance estimation strategy, but this ultimately returns a point $x_{\epsilon,p}$ satisfying (2) with overall cost that is substantially higher than the optimal method, roughly by a factor of κ. In this work, we overcome this issue by embedding the optimal oracle of Ghadimi and Lan (2013) within `proxBoost` . Assuming only that σ^2 is finite and without any other light tail assumption, we show that this strategy returns a point $x_{\epsilon,p}$ satisfying (2) with overall cost

$$\widetilde{\mathcal{O}}\left(\log\left(\frac{1}{p}\right)\left(\sqrt{\kappa}\ln\left(\frac{\Delta_{\text{in}}}{\epsilon} \vee \kappa\right) + \frac{\sigma^2}{\mu\epsilon}\right)\right).$$

Here $\widetilde{\mathcal{O}}(\cdot)$ only hides logarithmic dependence on κ. **Thus, `proxBoost` endows the optimal low-confidence algorithm with high confidence guarantees at only a polylogarithmic rise in cost.**

It is worthwhile to note that `proxBoost` seeded with the stochastic gradient method resembles the weight decay schedule, which is commonly used in practice; see e.g. Ge et al. (2019); Yang

et al. (2018). The procedure is also related to, but distinct from, the SGD3 algorithm of Allen-Zhu (2018), which rapidly drives the gradient of the objective function to zero in expectation.

1.1.2. EMPIRICAL RISK MINIMIZATION WITH HEAVY TAILS: IMPROVED COMPLEXITY

An empirical risk minimization (ERM) procedure may also serve as a minimization oracle $\mathcal{M}(f, \epsilon)$ within `proxBoost`. Such procedures draw n i.i.d. samples $z_1, \ldots, z_n \sim \mathcal{P}$ and minimize the empirical average

$$\min_x \; f_S(x) := \frac{1}{n} \sum_{i=1}^{n} f(x, z_i). \tag{4}$$

For these oracles, the cost $\mathcal{C}_{\mathcal{M}}(f, \epsilon)$ is the the number of samples n needed to ensure that the minimizer y_S of the empirical risk f_S satisfies $\mathbb{E}\left[f(x_S) - \min f\right] \leq \epsilon$. While many works have analyzed the sample complexity of ERM under various settings (e.g., Hsu and Sabato (2016); Bartlett and Mendelson (2002); Shalev-Shwartz et al. (2009); Shalev-Shwartz and Ben-David (2014)) our goal is to adapt and improve the bounds of Hsu and Sabato (2016), who developed high confidence ERM guarantees for nonnegative losses $f(x, z)$. In their work, Hsu and Sabato (2016) couple ERM with the robust distance estimation strategy, producing a point $\hat{y}_S$ with relative error guarantee $\mathbb{P}\big[f(\hat{y}_S) \leq (1+\gamma)\min f\big] \geq 1 - p$ at a sample complexity cost on the order of $\mathcal{O}\left(\log(1/p) \cdot (\hat{\kappa}\,\kappa/\gamma)\right)$. Here, loosely speaking, κ and $\hat{\kappa}$ are the condition numbers of f and f_S, respectively. It is unknown whether this sample complexity can be improved, but the appearance of a squared "condition number" makes the sample complexity cost of ERM much larger than that of streaming algorithms from Section 1.1.1. **In this work, we provide such an improvement by embedding ERM within `proxBoost`, yielding an order of magnitude better complexity**

$$\widetilde{\mathcal{O}}\left(\log\left(\frac{1}{p}\right)\left(\frac{\hat{\kappa}}{\gamma} + \hat{\kappa}\right)\right).$$

1.2. Related literature

Robust distance estimation has a long history. The estimator we use was first introduced in (Nemirovsky and Yudin, 1983, p. 243), and is a multivariate generalization of the median of means Alon et al. (1999); Jerrum et al. (1986). Robust distance estimation was further investigated in Hsu and Sabato (2016) with a focus on high probability guarantees for ERM. Minsker (2015) analyzed a different generalization using the geometric median. Other articles related to the subject include median of means tournaments Lugosi and Mendelson (2016), robust multivariate mean estimators Joly et al. (2017); Lugosi and Mendelson (2019), and bandits with heavy tails Bubeck et al. (2013).

Most available high confidence guarantees for streaming algorithms make sub-Gaussian assumptions on the stochastic gradients Nemirovski et al. (2008); Juditsky and Nesterov (2014); Ghadimi and Lan (2012, 2013). Recently, there has been renewed interest in obtaining robust guarantees without the light-tails assumption. For example, the two works Chen et al. (2017); Yin et al. (2018) make use of the geometric median of means technique to robustly estimate the gradient in distributed optimization. A different technique was recently developed by Juditsky et al. (2019), establishing high confidence guarantees for mirror descent type algorithms by truncating the gradient.

Although this paper focuses on the problem (1), it is appealing to ask whether similar techniques are applicable in presence of convex constraints and/or regularizers. The answer is yes; the extension, however, is nontrivial and will appear in a forthcoming journal article. It is also worth pointing

out that if f is convex (but not strongly convex) and an upper bound R on $\|\bar{x}\|$ is known, then we may simply apply `proxBoost` to the the sum of f and a simple quadratic whose amplitude is on the order of R^2/ϵ. The outline of the paper is as follows. Sections 2 and 3 present the basic notation and robust distance estimation technique. Section 4 introduces the `proxBoost`framework, while Sections 5 and 6 discuss consequences for offline and streaming algorithms, respectively.

2. Notation

Throughout, we follow standard notation of convex optimization, as set out for example in the monographs Nesterov (2018); Beck (2017). We let $\mathbf{R}^d$ denote an Euclidean space with inner product $\langle \cdot, \cdot \rangle$ and the induced norm $\|x\| = \sqrt{\langle x, x \rangle}$. The symbol $B_\varepsilon(x)$ will stand for the closed ball around x of radius $\varepsilon > 0$. We will use the shorthand interval notation $[1, m] := \{1, \ldots, m\}$ for any $m \in \mathbb{N}$.

Consider a function $f\colon \mathbf{R}^d \to \mathbf{R}$. The function f is called μ-strongly convex if the perturbed function $f - \frac{\mu}{2}\|\cdot\|^2$ is convex. We say that f is L-smooth if it differentiable with L-Lipschitz continuous gradient. If f is both μ-strongly convex and L-smooth, then standard results guarantee

$$\frac{\mu}{2}\|x - \bar{x}\|^2 \leq f(x) - f(\bar{x}) \leq \frac{L}{2}\|x - \bar{x}\|^2 \qquad \text{for all } x \in \mathbf{R}^d, \tag{5}$$

where $\bar{x}$ is the minimizer of f. The ratio $\kappa := L/\mu$ is called the condition number of f.

Assumption 1 Throughout this work, we consider the optimization problem

$$\min_{x \in \mathbf{R}^d} \; f(x) \tag{6}$$

where $f\colon \mathbf{R}^d \to \mathbf{R}$ is μ-strongly convex and L-smooth. We set $\{\bar{x}\} = \operatorname{argmin} f$ and $f^* = \min f$.

3. Background: a robust distance estimator

Let us suppose for the moment that the only access to f is by querying a black-box procedure that estimates $\bar{x}$. Namely, following Hsu and Sabato (2016) we will call a procedure $\mathcal{D}(\varepsilon)$ a *weak distance oracle* for the problem (6) if it returns a point x satisfying

$$\mathbb{P}[\|x - \bar{x}\| \leq \varepsilon] \geq \frac{2}{3}. \tag{7}$$

We will moreover assume that when querying $\mathcal{D}(\varepsilon)$ multiple times, the returned vectors are all statistically independent. Weak distance oracles arise naturally in stochastic optimization both in streaming and offline settings. We will discuss specific examples in Sections 5 and 6.

It is well known from (Nemirovsky and Yudin, 1983, p. 243) and Hsu and Sabato (2016) that the low-confidence estimate (7) can be improved to a high confidence guarantee by a clustering technique. We define the *robust distance estimator* $\mathcal{D}(\varepsilon, m)$ by the following procedure (Algorithm 1).

Thus the estimator $\mathcal{D}(\varepsilon, m)$ first generates m statistically independent points $y_1, \ldots, y_m$ by querying m times the weak distance oracle $\mathcal{D}(\varepsilon)$. Then the procedure computes the smallest radius ball around each point y_i that contains more than half of the generated points $\{y_1, \ldots, y_m\}$. Finally, the point y_{i^*} corresponding to the smallest such ball is returned. See Figure 1 for an illustration. The following lemma summarizes the guarantees of Algorithm 1.

Lemma 1 (Robust Distance Estimator) *The point x returned by $\mathcal{D}(\varepsilon, m)$ satisfies*

$$\mathbb{P}\big(\|x - \bar{x}\| \leq 3\varepsilon\big) \geq 1 - \exp\left(-\frac{m}{18}\right).$$

Algorithm 1 Robust Distance Estimation $\mathcal{D}(\varepsilon, m)$

Input: access to a weak distance oracle $\mathcal{D}(\varepsilon)$ and trial count m.
Query m times the oracle $\mathcal{D}(\varepsilon)$ and let $Y = \{y_1, \ldots, y_m\}$ consist of the responses.
for $i = 1, \ldots, m$ **do**
  Compute $r_i = \min\left\{r \geq 0 : |B_r(y_i) \cap Y| > \frac{m}{2}\right\}$.
end
Return y_{i^*} where $i^* = \operatorname{argmin}_{i \in [1,m]} r_i$.

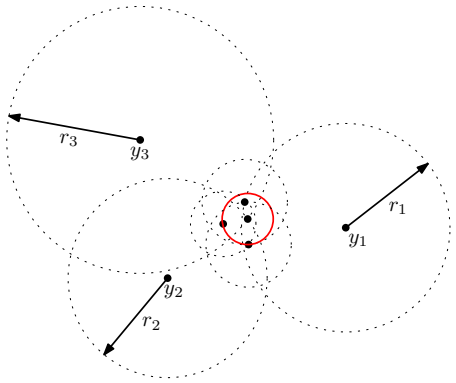

Figure 1: Illustration of the robust distance estimator $\mathcal{D}(\varepsilon, m)$.

4. The `proxBoost` Method

In this work, we explain how to efficiently use a robust distance estimator $\mathcal{D}(\varepsilon, m)$ to compute a point x satisfying $f(x) - \min f \leq \delta$ with high probability. One naive approach is to appeal to the upper bound in (5). Hence by Lemma 1, the point $x = \mathcal{D}(\varepsilon, m)$, with $\varepsilon = \sqrt{\frac{2\delta}{9L}}$, satisfies

$$\mathbb{P}\left(f(x) - f^* \leq \delta\right) \;\geq\; \mathbb{P}\left(\|x - \bar{x}\| \leq 3\varepsilon\right) \;\geq\; 1 - \exp\left(-\frac{m}{18}\right).$$

We will follow an alternative strategy, which can significantly decrease the overall cost when $\kappa \gg 1$. The optimistic goal is to replace $\varepsilon \approx \sqrt{\frac{\delta}{L}}$ used in the call $\mathcal{D}(\varepsilon, m)$ by the much larger quantity $\sqrt{\frac{\delta}{\mu}}$. The strategy we propose will apply a robust distance estimator $\mathcal{D}$ to a sequence of optimization problems that are better and better conditioned. We begin by applying $\mathcal{D}$ to f with the low accuracy $\sqrt{\frac{\delta}{\mu}}$. In step i, we will apply $\mathcal{D}$ to a new function f^i, which has condition number $\kappa_i \approx \frac{L+\mu 2^i}{\mu+\mu 2^i}$, with accuracy $\varepsilon_i \approx \sqrt{\frac{\delta}{\mu+\mu 2^i}}$. Continuing this process for $T \approx \log_2\left(\frac{L}{\mu}\right)$ rounds, we arrive at accuracy $\varepsilon_T \approx \sqrt{\frac{\delta}{\mu+L}}$ and a function f^T that is nearly perfectly conditioned with $\kappa_T \leq 2$. In this way, the total cost is amortized over the sequence of optimization problems. The key of course is to control the error incurred by varying the optimization problems along the iterations.

The outlined continuation procedure can be succinctly described using an *inexact proximal point method* in the sense of Martinet (1972, 1970); Rockafellar (1976). Henceforth, fix an increasing sequence of penalties $\lambda_0, \ldots, \lambda_T$ and a sequence of centers $x_0, \ldots, x_T$. For each index $i = 0, \ldots, T$,

define the perturbed functions and their minimizers:

$$f^i(x) := f(x) + \frac{\lambda_i}{2}\|x - x_i\|^2, \qquad \bar{x}_{i+1} := \operatorname*{argmin}_x f^i(x).$$

The exact proximal point method proceeds by inductively declaring $x_i = \bar{x}_i$ for $i \geq 1$. Since computing $\bar{x}_i$ exactly is in general impossible, we will instead monitor the error $\|\bar{x}_i - x_i\|$. The following elementary result will form the basis for the rest of the paper. To simplify notation, we will set $\bar{x}_0 := \operatorname{argmin} f$ and $\lambda_{-1} := 0$, throughout.

Theorem 2 (Inexact proximal point method) *For all $j \geq 0$, the following estimate holds:*

$$f^j(\bar{x}_{j+1}) - f^* \leq \sum_{i=0}^{j} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2. \tag{8}$$

Consequently, for all $j \geq 0$ we have the error decompositions:

$$f(x_{j+1}) - f^* \leq (f^j(x_{j+1}) - f^j(\bar{x}_{j+1})) + \sum_{i=0}^{j} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2, \tag{9}$$

$$f(x_j) - f^* \leq \frac{L + \lambda_{j-1}}{2}\|\bar{x}_j - x_j\|^2 + \sum_{i=0}^{j-1} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2. \tag{10}$$

Proof We first establish (8) by induction. For the base case $j = 0$, observe $\lambda_{-1} = 0$ and $f^0(\bar{x}_1) = \min_x f^0(x) \leq f^0(\bar{x}_0) = f^* + \frac{\lambda_0}{2}\|\bar{x}_0 - x_0\|^2$. As the inductive assumption, suppose (8) holds up to iteration $j - 1$. We then conclude

$$f^j(\bar{x}_{j+1}) \leq f^j(\bar{x}_j) \overset{\text{(a)}}{\leq} f^{j-1}(\bar{x}_j) + \frac{\lambda_j}{2}\|\bar{x}_j - x_j\|^2 \overset{\text{(b)}}{\leq} f^* + \sum_{i=0}^{j} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2,$$

where (a) uses the estimate $f(\bar{x}_j) \leq f^{j-1}(\bar{x}_j)$ and (b) follows from the inductive assumption. This completes the proof of (8). Next, observe the estimate $f(x_{j+1}) - f^* \leq f^j(x_{j+1}) - f^* = (f^j(x_{j+1}) - f^j(\bar{x}_{j+1})) + f^j(\bar{x}_{j+1}) - f^*$. Upper-bounding the right-hand-side using (8) establishes (9). Inequality (10) for $j = 0$ follows directly from smoothness of f, while for $j \geq 1$, it follows by combining (9) with the fact that f^j is $(L + \lambda_j)$-smooth. ■

The main conclusion of Theorem 2 is the decomposition of the functional error described in (9). Namely, the estimate (9) upper bounds the error $f(x_{j+1}) - \min f$ as the sum of the suboptimality in the last step $f^T(x_{T+1}) - f^T(\bar{x}_{T+1})$ and the errors $\frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2$ incurred along the way. By choosing T and λ_i sufficiently large, we can be sure that the function f^T is well-conditioned. Moreover, in order to ensure that each term in the sum $\frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2$ is of order δ, it suffices to guarantee $\|\bar{x}_i - x_i\| \leq \sqrt{\frac{2\delta}{\lambda_i}}$ for each index i. Since λ_i is an increasing sequence, it follows that we may gradually decrease the tolerance on the errors $\|\bar{x}_i - x_i\|$, all the while improving the conditioning of the functions we encounter. With this intuition in mind, we introduce the `proxBoost` procedure (Algorithm 2). The algorithm depends on the amplitude sequence $\{\lambda_j\}_{j=1}^T$, which we will treat as a global parameter specified in theorem statements.

Algorithm 2 `proxBoost`(δ, p, T)

Input: $\delta \geq 0$, $p \in (0,1)$, $T \in \mathbb{N}$

Set $\lambda_{-1} = 0$, $\varepsilon_{-1} = \sqrt{\frac{2\delta}{\mu}}$.

Generate x_0 satisfying $\|x_0 - \bar{x}_0\| \leq \varepsilon_{-1}$ with probability $1-p$.

for $j = 0, \ldots, T$ **do**

Set $\varepsilon_j = \sqrt{\frac{2\delta}{\mu+\lambda_j}}$

Generate a point x_{j+1} satisfying

$$\mathbb{P}\left[\|x_{j+1} - \bar{x}_{j+1}\| \leq \varepsilon_j \mid E_j\right] \geq 1 - p, \tag{11}$$

where E_j denotes the event $E_j := \left\{x_i \in B_{\varepsilon_{i-1}}(\bar{x}_i) \text{ for all } i \in [0,j]\right\}$.

end

Return x_{T+1}

Thus `proxBoost` begins by generating a point x_0 that is a distance of $\sqrt{\frac{2\delta}{\mu}}$ away from the minimizer of f with probability $1-p$. This task can be achieved by applying a robust distance estimator on f, as discussed in Section 3. In each subsequent iteration, x_{j+1} is defined to be a point that is within a radius of $\varepsilon_j = \sqrt{\frac{2\delta}{\mu+\lambda_j}}$ from the minimizer of f^j with probability $1-p$ conditioned on the event E_j. The event E_j encodes that each previous iteration was successful in the sense that the point x_i indeed lies inside the ball $B_{\varepsilon_{i-1}}(\bar{x}_i)$ for all $i = 0, \ldots, j$. Thus x_{j+1} can be determined by a procedure that conditioned on the event E_j is a robust distance estimator on the function f^j.

The following theorem summarizes the guarantees of the `proxBoost` procedure.

Theorem 3 (proxBoost) *Fix a constant $\delta > 0$, failure probability $p \in (0,1)$ and a number $T \in \mathbb{N}$. Then with probability at least $1-(T+2)p$, the point $x_{T+1} =$ `proxBoost`(δ, p, T) satisfies*

$$f(x_{T+1}) - \min f \leq \delta\left(\frac{L+\lambda_T}{\mu+\lambda_T} + \sum_{i=0}^{T} \frac{\lambda_i}{\mu+\lambda_{i-1}}\right). \tag{12}$$

Proof We first prove by induction the estimate

$$\mathbb{P}[E_t] \geq 1 - (t+1)p \qquad \text{for all } t = 0, \ldots, T. \tag{13}$$

The base case $t = 0$ is immediate from the definition of x_0. Suppose now that (13) holds for some index $t-1$. Then the inductive assumption and the definition of x_t yield $\mathbb{P}[E_t] = \mathbb{P}[E_t | E_{t-1}]\mathbb{P}[E_{t-1}] \geq (1-p)(1-tp) \geq 1-(t+1)p$, thereby completing the induction. Using (10) and the definitions of x_{T+1} and ε_j within the event E_T immediately yields (12). ■

Looking at the estimate (12), we see that the final error $f(x_{T+1}) - \min f$ is controlled by the sum $\sum_{i=0}^{T} \frac{\lambda_i}{\mu+\lambda_{i-1}}$ and the condition number $\frac{L+\lambda_T}{\mu+\lambda_T}$ of f^T. A moment of thought yields an appealing choice $\lambda_i = \mu 2^i$ for the proximal parameters. Indeed, then every element in the sum $\frac{\lambda_i}{\mu+\lambda_{i-1}}$ is upper bounded by two and the quotient $\frac{L+\lambda_T}{\mu+\lambda_T}$ is upper bounded by two after only $T = \lceil \log(L/\mu) \rceil$ rounds.

Corollary 4 (Proximal boost with geometric decay) *Fix an iteration count T, a target accuracy $\epsilon > 0$, and a failure probability $p \in (0,1)$. Define the algorithm parameters:*

$$\delta = \frac{\epsilon}{3+2T} \qquad \text{and} \qquad \lambda_i = \mu 2^i \qquad \forall i \in [0,T].$$

Then the point $x_{T+1} = \texttt{proxBoost}(\delta, p, T)$ satisfies $\mathbb{P}(f(x_{T+1}) - \min f \leq \epsilon) \geq 1 - (T+2)p$.

In the next two sections, we seed `proxBoost` with streaming and ERM algorithms. The reader, however, should keep in mind that `proxBoost` is agnostic to the inner workings of the robust distance estimators it uses. The only caveat is that some distance estimators require auxiliary quantities as input (e.g. initial function gap). Therefore, we may have to iteratively update such estimates.

5. Empirical risk minimization with heavy tails: improved complexity

In this section, we explore the consequences of `proxBoost` for empirical risk minimization. Setting the stage, fix a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ and equip $\mathbf{R}^d$ with the Borel σ-algebra. Consider the stochastic optimization problem (1), where $f\colon \mathbf{R}^d \times \Omega \to \mathbf{R}_+$ is a measurable nonnegative function. A common approach to problems of the form (1) is based on empirical risk minimization (ERM): collect *i.i.d.* samples $z_1, \ldots, z_n \sim \mathcal{P}$ and compute a minimizers x_S of the empirical average $f_S(x) := \frac{1}{n}\sum_{i=1}^n f(x, z_i)$. A central question is to determine the number n of samples that would ensure low generalization error $f(x_S) - \min f$, with high probability. There is a vast literature on this subject; representative works include Hsu and Sabato (2016); Bartlett and Mendelson (2002); Shalev-Shwartz et al. (2009); Shalev-Shwartz and Ben-David (2014). We build on Hsu and Sabato (2016), who focused on high confidence bounds for smooth strongly convex minimization.

Assumption 2 Following Hsu and Sabato (2016), we make the following assumptions on the loss.

1. **(Strong convexity)** There exist a real $\mu > 0$ and a natural number $N \in \mathbb{N}$ such that:
   (a) the population loss f is μ-strongly convex,
   (b) the loss $x \mapsto f_S(x)$ is μ-strongly convex with probability at least $5/6$, whenever $|S| \geq N$.
2. **(Smoothness)** There exist constants $L, \hat{L} > 0$ such that:
   (a) for a.e. $z \sim \mathcal{P}$, the loss $x \mapsto f(x,z)$ is nonnegative and $\hat{L}$-smooth,
   (b) the population objective $x \mapsto f(x)$ is L-smooth. (Note $L \leq \hat{L}$.)

The following result proved in (Hsu and Sabato, 2016, Theorem 15) shows that the empirical risk minimizer is a weak distance oracle for the problem (1).

Lemma 5 *Fix an i.i.d. sample $z_1, \ldots, z_n \sim \mathcal{P}$ of size $n \geq N$. Suppose Assumption 2 holds. Then the minimizer x_S of the empirical risk (4) satisfies the bound $\mathbb{P}\left[\|x_S - \bar{x}\| \leq \sqrt{\frac{96\hat{L}f^*}{n\mu^2}}\right] \geq 2/3$.*

In particular, using Algorithm 1 one may turn ERM into a robust distance estimator for (5). It is an easy computation, using Lemma 1 and the bound (5), to estimate the functional sub-optimality of the returned point x. Namely, the main result of (Hsu and Sabato, 2016, Corollary 16) shows that by setting $m = \lceil 18\ln(1/p)\rceil$ and $n = \max\{\lceil \frac{432\hat{\kappa}\kappa}{\gamma}\rceil, N\}$, the returned point x satisfies

$$\mathbb{P}[f(x) \leq (1+\gamma)f^*] \geq 1 - p, \tag{14}$$

while the overall sample complexity is

$$\left\lceil 18 \ln\left(\frac{1}{p}\right)\right\rceil \cdot \max\left\{\left\lceil \frac{432\hat{\kappa}\kappa}{\gamma}\right\rceil, N\right\}, \tag{15}$$

where $\hat{\kappa} = \hat{L}/\mu$ and $\kappa = L/\mu$. Notice that the guarantee (14) measures relative error.

We will now see how to find a point x satisfying (14) with significantly fewer samples than (15) by embedding ERM within `proxBoost`. Algorithm 3 is the robust distance estimator induced by ERM, while Algorithm 4 is the `proxBoost` algorithm specialized to ERM.

Algorithm 3 `ERM-R`(n, m, λ, x)

Input: sample count $n \in \mathbb{N}$, trial count $m \in \mathbb{N}$, center $x \in \mathbf{R}^d$, amplitude $\lambda > 0$.
▷ *Form the points* $Y = \{y_1, \ldots, y_m\}$ *by running ERM* m *times*:
for $i = 1, \ldots, m$ **do**
  Draw i.i.d. samples $z_1, \ldots, z_n \sim \mathcal{P}$ and compute $y_i = \operatorname{argmin}_y \frac{1}{n}\sum_{i=1}^n f(y, z_i) + \frac{\lambda}{2}\|y - x\|^2$.
end
▷ *Form the robust distance estimator using* $Y = \{y_1, \ldots, y_m\}$:
for $i = 1, \ldots, m$ **do**
  Compute $r_i = \min\{r \geq 0 : |B_r(y_i) \cap Y| > \frac{m}{2}\}$.
end
Return y_{i^*} where $i^* = \operatorname{argmin}_{i \in [1,m]} r_i$

Algorithm 4 `BoostERM`(γ, T, m)

Input: $T, m \in \mathbb{N}$, $\gamma > 0$
Set $\lambda_{-1} = 0$, $x_{-1} = 0$, $n_{-1} = \frac{432\hat{L}}{\gamma\mu}$
for $j = 0, \ldots, T+1$ **do**
  $x_j = \texttt{ERM-R}(n_{j-1}, m, \lambda_{j-1}, x_{j-1})$
  $n_j = 432 \left\lceil \frac{\hat{L}+\lambda_j}{\mu+\lambda_j}\left(\frac{1}{\gamma} + \sum_{i=0}^{j} \frac{\lambda_i}{\mu+\lambda_{i-1}}\right)\right\rceil \vee N$
end
Return x_{T+1}

The following result follows quickly from Theorem 3; see Appendix A for details.

Theorem 6 (Efficiency of `BoostERM` with geometric decay) *Fix a target relative accuracy* $\gamma > 0$ *and a probability of failure* $p \in (0, 1)$. *Define the algorithm parameters:*

$$T = \lceil \log_2(\kappa) \rceil, \qquad m = \left\lceil 18 \ln\left(\tfrac{T+2}{p}\right)\right\rceil, \qquad \tilde{\gamma} = \frac{\gamma}{3+2T}, \qquad \lambda_i = \mu 2^i.$$

Then with probability of at least $1 - p$, *the point* $x_{T+1} = \texttt{BoostERM}(\tilde{\gamma}, T, m)$ *satisfies* $f(x^{T+1}) \leq (1+\gamma)f^*$. *Moreover, the total number of samples used by the algorithm is*

$$\mathcal{O}\left(\ln(\kappa)\ln\left(\frac{\ln(\kappa)}{p}\right) \cdot \max\left\{\left(1 + \tfrac{1}{\gamma}\right)\hat{\kappa}\ln(\kappa), N\right\}\right).$$

Notice that the sample complexity provided by Theorem 6 is an order of magnitude better than (15) in terms of the dependence on the condition numbers $\hat{\kappa}$ and κ.

6. Streaming algorithms with heavy tails: high confidence acceleration

In this section, we will seed `proxBoost` with the robust distance estimator, induced by (accelerated) stochastic gradient methods. An important point is that the complexity of such methods depends on the initial gap $f(x_0) - f^*$. Consequently, in order to know how many iterations are needed to reach an accuracy $\mathbb{E}[f(x_i)] - f^* \leq \delta$, we must have available an upper bound $\Delta \geq f(x_0) - f^*$. Thus, we will have to dynamically update an estimate of the initialization quality for each proximal subproblem along the iterations of `proxBoost`. The following assumption formalizes this idea.

Assumption 3 Consider the proximal minimization problem

$$\min_y \ \varphi_x(y) := f(y) + \frac{\lambda}{2}\|y - x\|^2.$$

Let $\Delta > 0$ be a real number satisfying $\varphi_x(x) - \min \varphi_x \leq \Delta$. We will let $\texttt{Alg}(\delta, \lambda, \Delta, x)$ be a procedure that returns a point y satisfying $\mathbb{P}[\varphi_x(y) - \min \varphi_x \leq \delta] \geq \frac{2}{3}$.

Clearly, since φ_x is $(\mu + \lambda)$-strongly convex, $\texttt{Alg}(\delta, \lambda, \Delta, x)$ is a weak distance oracle for φ_x. Indeed, denoting by $\bar{y}_x$ the minimizer of φ_x, the procedure returns a point y satisfying $\mathbb{P}(\|y - \bar{y}_x\| \leq \varepsilon) \geq \frac{2}{3}$ with $\varepsilon = \sqrt{\frac{2\delta}{\mu+\lambda}}$. Following the recipe in Section 2, we may turn it into a robust distance estimator for φ_x, as long as Δ upper bounds the initialization error. We record the robust distance estimator induced by $\texttt{Alg}(\cdot)$ as Algorithm 5 and the resulting `proxBoost` as Algorithm 6.

Algorithm 5 $\texttt{Alg-R}(\delta, \lambda, \Delta, x, m)$

Input: accuracy $\delta > 0$, amplitude $\lambda > 0$, upper bound $\Delta > 0$, center $x \in \mathbf{R}^d$, trial count $m \in \mathbb{N}$.
Query m times $\texttt{Alg}(\delta, \lambda, \Delta, x)$ and let $Y = \{y_1, \ldots, y_m\}$ consist of the responses.
for $i = 1, \ldots, m$ **do**
  Compute $r_i = \min\{r \geq 0 : |B_r(y_i) \cap Y| > \frac{m}{2}\}$.
end
Return y_{i^*} where $i^* = \operatorname{argmin}_{i \in [1,m]} r_i$.

Algorithm 6 $\texttt{BoostAlg}(\delta, \Delta_{\text{in}}, x_{\text{in}}, T, m)$

Input: accuracy $\delta > 0$, upper bound $\Delta_{\text{in}} > 0$, center $x_{\text{in}} \in \mathbf{R}^d$, and $m, T \in \mathbb{N}$
Set $\lambda_{-1} = 0$, $\Delta_{-1} = \Delta_{\text{in}}$, $x_{-1} = x_{\text{in}}$
for $j = 0, \ldots, T+1$ **do**
  $x_j = \texttt{Alg-R}(\delta/9, \lambda_{j-1}, \Delta_{j-1}, x_{j-1}, m)$
  $\Delta_j = \delta\left(\frac{L+\lambda_{j-1}}{\mu+\lambda_{j-1}} + \sum_{i=0}^{j-1} \frac{\lambda_i}{\mu+\lambda_{i-1}}\right)$
end
Return x_{T+1}.

The following result follows quickly from Theorem 3; see Appendix B for details.

Theorem 7 (Efficiency of `BoostAlg` with geometric decay) *Fix an arbitrary point $x_{\text{in}} \in \mathbf{R}^d$ and let Δ_{in} be any upper bound $\Delta_{\text{in}} \geq f(x_{\text{in}}) - \min f$. Fix a target accuracy $\epsilon > 0$ and probability of failure $p \in (0,1)$, and set the algorithm parameters*

$$T = \lceil \log_2(\kappa) \rceil, \qquad m = \left\lceil 18 \ln\left(\tfrac{2+T}{p}\right)\right\rceil, \qquad \delta = \frac{\epsilon}{3+2T}, \qquad \lambda_i = \mu 2^i.$$

Then the point $x_{T+1} = \texttt{BoostAlg}(\delta, \Delta_{\text{in}}, x_{\text{in}}, T, m)$ *satisfies* $\mathbb{P}(f(x_{T+1}) - \min f \leq \epsilon) \geq 1 - p$. *Moreover, the total number of calls to* $\texttt{Alg}(\cdot)$ *is* $\left\lceil 18 \ln\left(\frac{\lceil 2+\log_2(\kappa)\rceil}{p}\right)\right\rceil \lceil 2 + \log_2(\kappa)\rceil$, *while the initialization errors satisfy,* $\Delta_i \leq \frac{\kappa+1+2\lceil\log_2(\kappa)\rceil}{3+2\lceil\log_2(\kappa)\rceil}\epsilon$, *for all* $i \in [0, T+1]$.

Illustration: robust (accelerated) stochastic gradient methods

We now concretely describe how to use (accelerated) stochastic gradient methods as $\texttt{Alg}(\cdot)$ within the `proxBoost` procedure. Following the standard literature on streaming algorithms, we suppose that the only access to f is through a stochastic gradient oracle. Namely, fix a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ and let $G\colon \mathbf{R}^d \times \Omega \to \mathbf{R}$ be a measurable map satisfying

$$\mathbb{E}_z G(x, z) = \nabla f(x) \quad \text{and} \quad \mathbb{E}_z \|G(x, z) - \nabla f(x)\|^2 \leq \sigma^2.$$

The performance of numerical methods is judged by the number of stochastic gradient evaluations $G(x, z)$ with $z \sim \mathcal{P}$ required by the algorithm to produce an approximate minimizer of the problem.

Fix an initial point x_{in} and let $\Delta_{\text{in}} > 0$ satisfy $\Delta_{\text{in}} \geq f(x_0) - f^*$. It is well known that an appropriately modified stochastic gradient method can generate a point x satisfying $\mathbb{E} f(x) - f^* \leq \epsilon$ with sample complexity $\mathcal{O}\left(\kappa \log\left(\frac{\Delta_{\text{in}}}{\epsilon}\right) + \frac{\sigma^2}{\mu\epsilon}\right)$. The accelerated stochastic gradient methods Ghadimi and Lan (2013); Kulunchakov and Mairal (2019) have the much better sample complexity $\mathcal{O}\left(\sqrt{\kappa} \log\left(\frac{\Delta_{\text{in}}}{\epsilon}\right) + \frac{\sigma^2}{\mu\epsilon}\right)$. We may use either procedure as $\texttt{Alg}(\cdot)$ within `proxBoost`. Theorem 7 then guarantees that we will find x satisfying $\mathbb{P}[f(x) - f^* \leq \epsilon] \geq 1 - p$ with sample complexities

$$\mathcal{O}\left(\ln(\kappa) \ln\left(\frac{\ln \kappa}{p}\right) \cdot \left(\kappa \ln\left(\frac{\Delta_{\text{in}} \ln(\kappa)}{\epsilon} \vee \kappa\right) + \frac{\sigma^2 \ln(\kappa)}{\mu\epsilon}\right)\right),$$
$$\mathcal{O}\left(\ln(\kappa) \ln\left(\frac{\ln \kappa}{p}\right) \cdot \left(\sqrt{\kappa} \ln\left(\frac{\Delta_{\text{in}} \ln(\kappa)}{\epsilon} \vee \kappa\right) + \frac{\sigma^2 \ln(\kappa)}{\mu\epsilon}\right)\right),$$

for the unaccelerated and accelerated methods, respectively. Notice that the overhead cost for obtaining the high confidence guarantee is only polylogarithmic in κ and logarithmic in $1/p$.

Conclusion. This work developed a generic continuation procedure for minimizing stochastic smooth and strongly convex functions. The procedure, `proxBoost`, when paired with typical algorithms, boosts low confidence guarantees to high confidence outcomes. We presented two applications to streaming and offline algorithms. First, we showed that `proxBoost` equips the (accelerated) stochastic gradient methods of Ghadimi and Lan (2013) with high confidence guarantees at an overhead cost that is only polylogarithmic in κ and logarithmic in $1/p$. Second, we improved by a factor of the condition number the sample efficiency of ERM in Hsu and Sabato (2016).

Acknowledgments

Research of Drusvyatskiy was in part supported by by the NSF DMS 1651851 and CCF 1740551 awards and a research visiting position at Microsoft Research, Redmond, WA 98052. The auhors would like to thank Sebastian Bubeck, Lin Xiao, and Junyu Zhang for insightful discussions.

References

Z. Allen-Zhu. How to make the gradients small stochastically: Even faster convex and nonconvex sgd. In *Advances in Neural Information Processing Systems*, pages 1157–1167, 2018.

N. Alon, Y. Matias, and M. Szegedy. The space complexity of approximating the frequency moments. *J. Comput. System Sci.*, 58(1, part 2):137–147, 1999. ISSN 0022-0000. doi: 10.1006/jcss.1997.1545. URL https://doi-org.offcampus.lib.washington.edu/10.1006/jcss.1997.1545. Twenty-eighth Annual ACM Symposium on the Theory of Computing (Philadelphia, PA, 1996).

P.L. Bartlett and S. Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(Nov):463–482, 2002.

A. Beck. *First-order methods in optimization*, volume 25. SIAM, 2017.

O. Bousquet and A. Elisseeff. Stability and generalization. *Journal of Machine Learning Research*, 2:499–526, 2002.

S. Bubeck, N. Cesa-Bianchi, and G. Lugosi. Bandits with heavy tail. *IEEE Transactions on Information Theory*, 59(11):7711–7717, 2013.

O. Catoni. Challenging the empirical mean and empirical variance: a deviation study. *Ann. Inst. Henri Poincaré Probab. Stat.*, 48(4):1148–1185, 2012. ISSN 0246-0203. doi: 10.1214/11-AIHP454. URL https://doi.org/10.1214/11-AIHP454.

Y. Chen, L. Su, and J. Xu. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 1(2):44, 2017.

R. Ge, S.M. Kakade, R. Kidambi, and P. Netrapalli. The step decay schedule: A near optimal, geometrically decaying learning rate procedure. *arXiv preprint arXiv:1904.12838*, 2019.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization I: A generic algorithmic framework. *SIAM J. Optim.*, 22(4):1469–1492, 2012. ISSN 1052-6234. doi: 10.1137/110848864. URL https://doi-org.offcampus.lib.washington.edu/10.1137/110848864.

S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, II: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.

E. Hazan and S. Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1):2489–2512, 2014.

D. Hsu and S. Sabato. Loss minimization and parameter estimation with heavy tails. *The Journal of Machine Learning Research*, 17(1):543–582, 2016.

M.R. Jerrum, L.G. Valiant, and V.V. Vazirani. Random generation of combinatorial structures from a uniform distribution. *Theoret. Comput. Sci.*, 43(2-3):169–188, 1986. ISSN 0304-3975. doi: 10.1016/0304-3975(86)90174-X. URL https://doi-org.offcampus.lib.washington.edu/10.1016/0304-3975(86)90174-X.

E. Joly, G. Lugosi, and R.I. Oliveira. On the estimation of the mean of a random vector. *Electronic Journal of Statistics*, 11(1):440–451, 2017.

A. Juditsky and Y. Nesterov. Deterministic and stochastic primal-dual subgradient algorithms for uniformly convex minimization. *Stoch. Syst.*, 4(1):44–80, 2014. ISSN 1946-5238. doi: 10.1214/10-SSY010. URL https://doi-org.offcampus.lib.washington.edu/10.1214/10-SSY010.

A. Juditsky, A. Nazin, A. Nemirovsky, and A. Tsybakov. Algorithms of robust stochastic optimization based on mirror descent method. *arXiv:1907.02707*, 2019.

A. Kulunchakov and J. Mairal. Estimate sequences for stochastic composite optimization: Variance reduction, acceleration, and robustness to noise. *arXiv:1901.08788*, 2019.

G. Lugosi and S. Mendelson. Risk minimization by median-of-means tournaments. *arXiv:1608.00757*, 2016.

G. Lugosi and S. Mendelson. Sub-gaussian estimators of the mean of a random vector. *The Annals of Statistics*, 47(2):783–794, 2019.

B. Martinet. Régularisation d'inéquations variationnelles par approximations successives. *Rev. Française Informat. Rech. Opérationnelle*, 4(Sér. R-3):154–158, 1970.

B. Martinet. Détermination approchée d'un point fixe d'une application pseudo-contractante. Cas de l'application prox. *C. R. Acad. Sci. Paris Sér. A-B*, 274:A163–A165, 1972.

S. Minsker. Geometric median and robust estimation in Banach spaces. *Bernoulli*, 21(4):2308–2335, 2015. ISSN 1350-7265. doi: 10.3150/14-BEJ645. URL https://doi-org.offcampus.lib.washington.edu/10.3150/14-BEJ645.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM J. Optim.*, 19(4):1574–1609, 2008. ISSN 1052-6234. doi: 10.1137/070704277. URL https://doi-org.offcampus.lib.washington.edu/10.1137/070704277.

A. S. Nemirovsky and D. B. Yudin. *Problem complexity and method efficiency in optimization*. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York, 1983. ISBN 0-471-10345-4. Translated from the Russian and with a preface by E. R. Dawson, Wiley-Interscience Series in Discrete Mathematics.

Yu. Nesterov. *Lectures on convex optimization*, volume 137 of *Springer Optimization and Its Applications*. Springer, 2018. ISBN 978-3-319-91577-7; 978-3-319-91578-4. doi: 10.1007/978-3-319-91578-4. URL https://doi.org/10.1007/978-3-319-91578-4.

Yu. Nesterov and J.-P. Vial. Confidence level solutions for stochastic programming. *Automatica*, 44:1559–1568, 2008.

B.T. Polyak and A.B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM J. Control Optim.*, 30(4):838–855, 1992. ISSN 0363-0129. doi: 10.1137/0330046. URL https://doi.org/10.1137/0330046.

R.T. Rockafellar. Monotone operators and the proximal point algorithm. *SIAM J. Control Optimization*, 14(5):877–898, 1976. ISSN 0363-0129. doi: 10.1137/0314056. URL https://doi-org.offcampus.lib.washington.edu/10.1137/0314056.

S. Shalev-Shwartz and S. Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

S. Shalev-Shwartz, O. Shamir, N. Srebro, and K. Sridharan. Stochastic convex optimization. In *COLT*, 2009.

T. Yang, Y. Yan, Z. Yuan, and R. Jin. Why does stagewise training accelerate convergence of testing error over sgd. *arXiv preprint arXiv:1812.03934*, 2018.

D. Yin, Y. Chen, R. Kannan, and P. Bartlett. Byzantine-robust distributed learning: Towards optimal statistical rates. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5650–5659, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018. PMLR. URL http://proceedings.mlr.press/v80/yin18a.html.

Appendix A. Proof of Theorem 6

In this section we prove Theorem 6. Indeed, we will prove a more general result, Theorem A.1. Theorem 6 will then follow immediately under the parameter setting specified in its statement.

Theorem A.1 (Efficiency of `BoostERM`) *Fix a target accuracy $\gamma > 0$ and numbers $T, m \in \mathbb{N}$. Then with probability at least $1 - (T+2)\exp\left(-\frac{m}{18}\right)$, the point $x_{T+1} = \texttt{BoostERM}(\gamma, T, m)$ satisfies*

$$f(x_{T+1}) - f^* \leq \left(\frac{L + \lambda_T}{\mu + \lambda_T} + \sum_{i=0}^{T} \frac{\lambda_i}{\mu + \lambda_{i-1}}\right) \gamma f^*.$$

Proof We will verify that Algorithm 4 is an instantiation of Algorithm 2 with $\delta = \gamma f^*$ and $p = \exp(-\frac{m}{18})$. More precisely, we will prove by induction that with this choice of p and δ, the iterates x_j satisfy (11) for each index $j = 0, \ldots, T$. As the base case, consider the evaluation $x_0 = \texttt{ERM-R}(n_{-1}, m, \lambda_{-1}, x_{-1})$, where x_{-1} can be arbitrary since $\lambda_{-1} = 0$. Then Lemma 1 and Theorem 5 guarantee

$$\mathbb{P}\left[\|x_0 - \bar{x}_0\| \leq 3\sqrt{\frac{96\hat{L}f^*}{n_{-1}\mu^2}}\right] \geq 1 - \exp\left(-\frac{m}{18}\right).$$

Taking into account the definitions of n_{-1} in Algorithm 4 and ε_{-1} in Algorithm 2, we deduce

$$\mathbb{P}\left[\|x_0 - \bar{x}_0\| \leq \epsilon_{-1}\right] \geq 1 - p,$$

as claimed. As an inductive hypothesis, suppose that (11) holds for $x_0, x_1, \ldots, x_{j-1}$. We will prove it holds for $x_j = \texttt{ERM-R}(n_{j-1}, m, \lambda_{j-1}, x_{j-1})$. To this end, suppose that the event E_{j-1} occurs. Then by the same reasoning as in the base case, the point x_j satisfies

$$\mathbb{P}\left[\|x_j - \bar{x}_j\| \leq 3\sqrt{\frac{96(\hat{L} + \lambda_{j-1})f^{j-1}(\bar{x}_j)}{n_{j-1}(\mu + \lambda_{j-1})^2}}\right] \geq 1 - \exp\left(-\frac{m}{18}\right). \tag{16}$$

Now, using (8) and the inductive assumption that $\|x_i - \bar{x}_i\| \leq \varepsilon_{i-1} = \sqrt{\frac{2\delta}{\mu+\lambda_{i-1}}}$ for all $i \in [0, j-1]$ (conditioned on E_{j-1}), we have

$$f^{j-1}(\bar{x}_j) - f^* \leq \sum_{i=0}^{j-1} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2 \leq \delta \sum_{i=0}^{j-1} \frac{\lambda_i}{\mu + \lambda_{i-1}},$$

which, together with $\delta = \gamma f^*$, implies

$$f^{j-1}(\bar{x}_j) \leq f^* + \delta \sum_{i=0}^{j-1} \frac{\lambda_i}{\mu + \lambda_{i-1}} = \left(1 + \gamma \sum_{i=0}^{j-1} \frac{\lambda_i}{\mu + \lambda_{i-1}}\right) f^*.$$

Combining this inequality with (16), we conclude that conditioned on the event E_{j-1}, we have with probability $1 - p$ the guarantee

$$\frac{\mu + \lambda_{j-1}}{2}\|x_j - \bar{x}_j\|^2 \leq \frac{432(\hat{L} + \lambda_{j-1})(1 + \gamma \sum_{i=0}^{j-1} \frac{\lambda_i}{\mu+\lambda_{i-1}})}{n_{j-1}(\mu + \lambda_{j-1})} \cdot f^* \leq \gamma f^* = \delta, \tag{17}$$

where the last inequality follows from the definition of n_{j-1}. This implies that the estimate (11) holds for x_j with $\epsilon_{j-1} = \sqrt{\frac{2\delta}{\mu+\lambda_{j-1}}}$. An application of Theorem 3 completes the proof. ■

Appendix B. Proof of Theorem 7

In this section, we prove Theorem 7. Indeed, we will prove a more general result, Theorem B.1. Theorem 7 will then follow immediately under the parameter setting specified in its statement.

Theorem B.1 (Efficiency of `BoostAlg`) *Fix an arbitrary point $x_{\text{in}} \in \mathbf{R}^d$ and let Δ_{in} be any constant satisfying $\Delta_{\text{in}} \geq f(x_{\text{in}}) - \min f$. Fix natural numbers $T, m \in \mathbb{N}$. Then with probability at least $1 - (T+2)\exp\left(-\frac{m}{18}\right)$, the point $x_{T+1} = \texttt{BoostAlg}(\delta, \Delta_{\text{in}}, x_{\text{in}}, T, m)$ satisfies*

$$f(x_{T+1}) - \min f \leq \delta\left(\frac{L + \lambda_T}{\mu + \lambda_T} + \sum_{i=0}^{T} \frac{\lambda_i}{\mu + \lambda_{i-1}}\right).$$

Proof We will verify that Algorithm 6 is an instantiation of Algorithm 2 with $p = \exp(-\frac{m}{18})$. More precisely, we will prove by induction that with this choice of p, the iterates x_j satisfy (11) for each index $j = 0, \ldots, T$. For the base case $j = 0$, Lemma 1 guarantees that with probability $1 - p$, the point x_0 produced by the robust distance estimator `Alg-R` satisfies

$$\|x_0 - \bar{x}_0\| \leq 3\sqrt{\frac{2 \cdot \delta/9}{\mu}} = \varepsilon_{-1}.$$

As an inductive hypothesis, suppose that (11) holds for the iterates $x_0, \ldots, x_{j-1}$ for some $j \geq 1$. We will prove it holds for x_j. To this end, suppose that the event E_{j-1} occurs. Using (10) we deduce

$$\begin{aligned} f(x_{j-1}) - f^* &\leq \frac{L + \lambda_{j-2}}{2}\|\bar{x}_{j-1} - x_{j-1}\|^2 + \sum_{i=0}^{j-2} \frac{\lambda_i}{2}\|\bar{x}_i - x_i\|^2 \\ &\leq \frac{\delta(L + \lambda_{j-2})}{\mu + \lambda_{j-2}} + \sum_{i=0}^{j-2} \frac{\delta\lambda_i}{\mu + \lambda_{i-1}} = \Delta_{j-1}, \end{aligned}$$

where the second inequality follows from the inclusion $x_i \in B_{\varepsilon_{i-1}}(\bar{x}_i)$ with $\varepsilon_{i-1} = \sqrt{\frac{2\delta}{\mu+\lambda_{i-1}}}$ for all $i = 0, \ldots, j-1$. By examining the definition of f^{j-1}, we deduce $f^{j-1}(x_{j-1}) = f(x_{j-1})$ and $\min f^{j-1} \geq \min f = f^*$, which imply

$$f^{j-1}(x_{j-1}) - \min f^{j-1} \leq f(x_{j-1}) - f^* \leq \Delta_{j-1}. \tag{18}$$

That is, Δ_{j-1} is an upper bound on the initial gap $f^{j-1}(x_{j-1}) - \min f^{j-1}$ for all j, whenever the event E_{j-1} occurs. Moreover, Lemma 1 guarantees that conditioned on E_{j-1} with probability $1-p$, the following estimate holds:

$$\|x_j - \bar{x}_j\| \leq 3\sqrt{\frac{2 \cdot \delta/9}{\mu + \lambda_{j-1}}} = \varepsilon_{j-1}.$$

Thus (11) holds for the iterate x_j, as desired. An application of Theorem 3 completes the proof. ■