Stochastic optimization and sparse statistical recovery: Optimal algorithms for high dimensions

Alekh Agarwal
Microsoft Research
New York NY
alekha@microsoft.com

Sahand N. Negahban
Dept. of EECS
MIT
sahandn@mit.edu

Martin J. Wainwright
Dept. of EECS and Statistics
UC Berkeley
wainwrig@stat.berkeley.edu

Abstract

We develop and analyze stochastic optimization algorithms for problems in which the expected loss is strongly convex, and the optimum is (approximately) sparse. Previous approaches are able to exploit only one of these two structures, yielding a $\mathcal{O}(d/T)$ convergence rate for strongly convex objectives in d dimensions and $\mathcal{O}(\sqrt{s(\log d)/T})$ convergence rate when the optimum is s-sparse. Our algorithm is based on successively solving a series of ℓ_1-regularized optimization problems using Nesterov's dual averaging algorithm. We establish that the error of our solution after T iterations is at most $\mathcal{O}(s(\log d)/T)$, with natural extensions to approximate sparsity. Our results apply to locally Lipschitz losses including the logistic, exponential, hinge and least-squares losses. By recourse to statistical minimax results, we show that our convergence rates are optimal up to constants. The effectiveness of our approach is also confirmed in numerical simulations where we compare to several baselines on a least-squares regression problem.

1 Introduction

Stochastic optimization algorithms have many desirable features for large-scale machine learning, and have been studied intensively in the last few years (e.g., [18, 4, 8, 22]). The empirical efficiency of these methods is backed with strong theoretical guarantees on their convergence rates, which depend on various structural properties of the objective function. More precisely, for an objective function that is strongly convex, stochastic gradient descent enjoys a convergence rate ranging from $\mathcal{O}(1/T)$, when features vectors are extremely sparse, to $\mathcal{O}(d/T)$, when feature vectors are dense [9, 14, 10]. This strong convexity condition is satisfied for many common machine learning problems, including boosting, least squares regression, SVMs and generalized linear models among others.

A complementary condition is that of (approximate) sparsity in the optimal solution. Sparse models have proven useful in many applications (see e.g., [6, 5] and references therein), and many statistical procedures seek to exploit such sparsity. It has been shown [15, 19] that when the optimal solution θ^* is s-sparse, appropriate versions of the mirror descent algorithm converge at a rate $\mathcal{O}(s\sqrt{(\log d)/T})$. Srebro et al. [20] exploit the smoothness of common loss functions, and obtain improved rates of the form $\mathcal{O}(\eta\sqrt{(s \log d)/T})$, where η is the noise variance. While the $\sqrt{\log d}$ scaling makes these methods attractive in high dimensions, their scaling with respect to the iterations T is relatively slow—namely, $\mathcal{O}(1/\sqrt{T})$ as opposed to $\mathcal{O}(1/T)$ for strongly convex problems.

Many optimization problems encountered in practice exhibit both features: the objective function is strongly convex, and the optimum is (approximately) sparse. This fact leads to the natural question: is it possible to design algorithms for stochastic optimization that enjoy the best features of both types of structure? More specifically, an algorithm should have a $\mathcal{O}(1/T)$ convergence rate, as well as a logarithmic dependence on dimension. The main contribution of this paper is to answer this question in the affirmative, and to analyze a new algorithm that has convergence rate $\mathcal{O}((s \log d)/T)$

for a strongly convex problem with an s-sparse optimum in d dimensions. This rate is unimprovable (up to constants) in our setting, meaning that no algorithm can converge at a substantially faster rate. Our analysis also yields optimal rates when the optimum is only approximately sparse.

The algorithm proposed in this paper builds off recent work on multi-step methods for strongly convex problems [11, 10, 12], but involves some new ingredients so as to obtain optimal rates for statistical problems with sparse optima. In particular, we form a sequence of objective functions by decreasing the amount of regularization as the optimization algorithm proceeds which is quite natural from a statistical viewpoint. Each step of our algorithm can be computed efficiently, with a closed form update rule in many common examples. In summary, the outcome of our development is an *optimal one-pass* algorithm for many structured statistical problems in high dimensions, and with computational complexity linear in the sample size. Numerical simulations confirm our theoretical predictions regarding the convergence rate of the algorithm, and also establish its superiority compared to regularized dual averaging [22] and stochastic gradient descent algorithms. They also confirm that a direct application of the multi-step method of Juditsky and Nesterov [11] is inferior to our algorithm, meaning that our gradual decrease of regularization is quite critical. More details on our results and their proofs can be found in the full-length version of this paper [2].

2 Problem set-up and algorithm description

Given a subset $\Omega \subseteq \mathbb{R}^d$ and a random variable Z taking values in a space $\mathcal{Z}$, we consider an optimization problem of the form

$$\theta^* \in \arg\min_{\theta\in\Omega} \mathbb{E}[\mathcal{L}(\theta; Z)], \tag{1}$$

where $\mathcal{L} : \Omega \times \mathcal{Z} \to \mathbb{R}$ is a given loss function. As is standard in stochastic optimization, we do not have direct access to the *expected loss function* $\overline{\mathcal{L}}(\theta) := \mathbb{E}[\mathcal{L}(\theta; Z)]$, nor to its subgradients. Rather, for a given query point $\theta \in \Omega$, we observe a *stochastic subgradient*, meaning a random vector $g(\theta) \in \mathbb{R}^d$ such that $\mathbb{E}[g(\theta)] \in \partial\overline{\mathcal{L}}(\theta)$. The goal of this paper is to design algorithms that are suitable for solving the problem (1) when the optimum θ^* is (approximately) sparse.

Algorithm description: In order to solve a sparse version of the problem (1), our strategy is to consider a sequence of regularized problems of the form

$$\min_{\theta\in\Omega'} \big\{\overline{\mathcal{L}}(\theta) + \lambda\|\theta\|_1\big\}. \tag{2}$$

Our algorithm involves a sequence of K_T different epochs, where the regularization parameter $\lambda > 0$ and the constraint set $\Omega' \subset \Omega$ change from epoch to epoch. The epochs are specified by:

- a sequence of natural numbers $\{T_i\}_{i=1}^{K_T}$, where T_i specifies the length of the i^{th} epoch,
- a sequence of positive regularization weights $\{\lambda_i\}_{i=1}^{K_T}$, and
- a sequence of positive radii $\{R_i\}_{i=1}^{K_T}$ and d-dimensional vectors $\{y_i\}_{i=1}^{K_T}$, which specify the constraint set, $\Omega(R_i) := \big\{\theta \in \Omega \mid \|\theta - y_i\|_p \le R_i\big\}$, that is used throughout the i^{th} epoch.

We initialize the algorithm in the first epoch with $y_1 = 0$, and with any radius R_1 that is an upper bound on $\|\theta^*\|_1$. The norm $\|\cdot\|_p$ used in defining the constraint set $\Omega(R_i)$ is specified by $p = 2\log d/(2\log d - 1)$, a choice that will be clarified momentarily.

The goal of the i^{th} epoch is to update $y_i \mapsto y_{i+1}$, in such a way that we are guaranteed that $\|y_{i+1} - \theta^*\|_1^2 \le R_{i+1}^2$ for each $i = 1, 2, \ldots$. We choose the radii such that $R_{i+1}^2 = R_i^2/2$, so that upon termination, $\|y_{K_T} - \theta^*\|_1^2 \le R_1^2/2^{K_T-1}$. In order to update $y_i \mapsto y_{i+1}$, we run T_i rounds of the stochastic dual averaging algorithm [17] (henceforth DA) on the regularized objective

$$\min_{\theta\in\Omega(R_i)} \big\{\overline{\mathcal{L}}(\theta) + \lambda_i\|\theta\|_1\big\}. \tag{3}$$

The DA method generates two sequences of vectors $\{\mu^t\}_{t=0}^{T_i}$ and $\{\theta^t\}_{t=0}^{T_i}$ initialized as $\mu^0 = 0$ and $\theta^0 = y_i$, using a sequence of step sizes $\{\alpha^t\}_{t=0}^{T_i}$. At iteration $t = 0, 1, \ldots, T_i$, we let g^t be a stochastic subgradient of $\overline{\mathcal{L}}$ at θ^t, and we let ν^t be any element of the subdifferential of the ℓ_1-norm $\|\cdot\|_1$ at θ^t. The DA update at time t maps $(\mu^t, \theta^t) \mapsto (\mu^{t+1}, \theta^{t+1})$ via the recursions

$$\mu^{t+1} = \mu^t + g^t + \lambda_i\nu^t, \;\text{ and }\; \theta^{t+1} = \arg\min_{\theta\in\Omega(R_i)} \big\{\alpha^{t+1}\langle\mu^{t+1}, \theta\rangle + \psi_{y_i,R_i}(\theta)\big\}, \tag{4}$$

where the prox function ψ is specified below (5). The pseudocode describing the overall procedure is given in Algorithm 1. In the stochastic dual averaging updates (4), we use the prox function

$$\psi_{y_i,R_i}(\theta) = \frac{1}{2R_i^2\,(p-1)}\,\|\theta - y_i\|_p^2, \quad \text{where} \quad p = \frac{2\log d}{2\log d - 1}. \tag{5}$$

This particular choice of the prox-function and the specific value of p ensure that the function ψ is strongly convex with respect to the ℓ_1-norm, and has been previously used for sparse stochastic optimization (see e.g. [15, 19, 7]). In most of our examples, $\Omega = \mathbb{R}^d$ and owing to our choice of the prox-function and the feasible set in the update (4), we can compute θ^{t+1} from μ^{t+1} in closed form. Some algebra yields that the update (4) with $\Omega = \mathbb{R}^d$ is equivalent to

$$\theta^{t+1} = y_i + \frac{R_i^2\alpha^{t+1}}{(p-1)(1+\xi)}\,\frac{|\mu^{t+1}|^{(q-1)}\text{sign}(\mu^{t+1})}{\|\mu^{t+1}\|_q^{(q-2)}}, \;\text{ where }\; \xi = \max\left\{0, \frac{\alpha^{t+1}\|\mu^{t+1}\|_q R_i}{p-1} - 1\right\}.$$

Here $|\mu^{t+1}|^{(q-1)}$ refers to elementwise operations and $q = p/(p-1)$ is the conjugate exponent to p. We observe that our update (4) computes a subgradient of the ℓ_1-norm rather than computing an exact prox-mapping as in some previous methods [16, 7, 22]. Computing such a prox-mapping for $y_i \neq 0$ requires $\mathcal{O}(d^2)$ computation, which is why we adopt the update (4) with a complexity $\mathcal{O}(d)$.

Algorithm 1 Regularization Annealed epoch Dual AveRaging (RADAR)

```
Require: Epoch length schedule $\{T_i\}_{i=1}^{K_T}$, initial radius $R_1$, step-size multiplier $\alpha$, prox-function
  $\psi$, initial prox-center $y_1$, regularization parameters $\lambda_i$.
  for Epoch $i = 1, 2, \ldots, K_T$ do
    Initialize $\mu^0 = 0$ and $\theta^0 = y_i$.
    for Iteration $t = 0, 1, \ldots, T_i - 1$ do
      Update $(\mu^t, \theta^t) \mapsto (\mu^{t+1}, \theta^{t+1})$ according to rule (4) with step size $\alpha^t = \alpha/\sqrt{t}$.
    end for
    Set $y_{i+1} = \frac{\sum_{t=1}^{T_i} \theta^t}{T_i}$.
    Update $R_{i+1}^2 = R_i^2/2$.
  end for
  Return $y_{K_T+1}$
```

Conditions: Having defined our algorithm, we now discuss the conditions on the objective function $\overline{\mathcal{L}}(\theta)$ and stochastic gradients that underlie our analysis.

Assumption 1 (Locally Lipschitz)**.** For each $R > 0$, there is a constant $G = G(R)$ such that

$$|\overline{\mathcal{L}}(\theta) - \overline{\mathcal{L}}(\tilde{\theta})| \leq G\,\|\theta - \tilde{\theta}\|_1 \tag{6}$$

for all pairs $\theta, \tilde{\theta} \in \Omega$ such that $\|\theta - \theta^*\|_1 \leq R$ and $\|\tilde{\theta} - \theta^*\|_1 \leq R$.

We note that it suffices to have $\|\nabla\overline{\mathcal{L}}(\theta)\|_\infty \leq G(R)$ for the above condition. As mentioned, our goal is to obtain fast rates for objectives satisfying a local strong convexity condition, defined below.

Assumption 2 (Local strong convexity (LSC))**.** The function $\overline{\mathcal{L}} : \Omega \to \mathbb{R}$ satisfies a R-local form of strong convexity (LSC) if there is a non-negative constant $\gamma = \gamma(R)$ such that

$$\overline{\mathcal{L}}(\tilde{\theta}) \geq \overline{\mathcal{L}}(\theta) + \langle\nabla\overline{\mathcal{L}}(\theta),\, \tilde{\theta} - \theta\rangle + \frac{\gamma}{2}\|\theta - \tilde{\theta}\|_2^2 \;\; \forall\theta, \tilde{\theta} \in \Omega \;\text{ with }\; \|\theta\|_1 \leq R \text{ and } \|\tilde{\theta}\|_1 \leq R. \tag{7}$$

Some of our results regarding stochastic optimization from a finite sample will use a weaker form of the assumption, called local RSC, exploited in our recent work on statistics and optimization [1, 13]. Our final assumption is a tail condition on the error in stochastic gradients: $e(\theta) := g(\theta) - \mathbb{E}[g(\theta)]$.

Assumption 3 (Sub-Gaussian stochastic gradients)**.** There is a constant $\sigma = \sigma(R)$ such that

$$\mathbb{E}\big[\exp(\|e(\theta)\|_\infty^2/\sigma^2)\big] \leq \exp(1) \;\text{ for all } \theta \text{ such that } \|\theta - \theta^*\|_1 \leq R. \tag{8}$$

Clearly, this condition holds whenever the error vector $e(\theta)$ has bounded components. More generally, the bound (8) holds whenever each component of the error vector has sub-Gaussian tails.

Some illustrative examples: We now describe some examples that satisfy the above conditions to illustrate how the various parameters of interest might be obtained in different scenarios.

Example 1 (Classification under Lipschitz losses)**.** In binary classification, the samples consist of pairs $z = (x, y) \in \mathbb{R}^d \times \{-1, 1\}$. Common choices for the loss function $\mathcal{L}(\theta; z)$ are the hinge loss $\max(0, 1 - y\langle\theta, x\rangle)$ or the logistic loss $\log(1 + \exp(-y\langle\theta, x\rangle)$. Given a distribution $\mathbb{P}$ over $\mathcal{Z}$ (either the population or the empirical distribution), a common strategy is to draw $(x_t, y_t) \sim \mathbb{P}$ at iteration t and use $g^t = \nabla\mathcal{L}(\theta; (x_t, y_t))$. We now illustrate how our conditions are satisfied in this setting.

- *Locally Lipschitz:* Both the above examples actually satisfy a stronger global Lipschitz condition since we have the bound $G \leq \|\nabla\bar{\mathcal{L}}(\theta)\|_\infty \leq \mathbb{E}\|x\|_\infty$. Often, the data satisfies the normalization $\|x\|_\infty \leq B$, in which case we get $G \leq B$. More generally, tail conditions on the marginal distribution of each coordinate of x ensure $G = \mathcal{O}(\sqrt{\log d})$ is valid with high probability.
- *LSC:* When the expectation in the objective (1) is under the population distribution, the above examples satisfy LSC. Here we focus on the example of the logistic loss, where we define the link function $\psi(\alpha) = \exp(\alpha)/(1 + \exp(\alpha))^2$. We also define $\Sigma = \mathbb{E}[xx^T]$ to be the covariance matrix and let $\sigma_{\min}(\Sigma)$ denote its minimum singular value. Then a second-order Taylor expansion yields

$$\bar{\mathcal{L}}(\tilde{\theta}) - \bar{\mathcal{L}}(\theta) - \langle\nabla\bar{\mathcal{L}}(\theta), \tilde{\theta} - \theta\rangle = \frac{\psi(\langle\widetilde{\theta}, x\rangle)}{2}\|\Sigma^{1/2}(\theta - \tilde{\theta})\|_2^2 \geq \frac{\psi(BR)\sigma_{\min}(\Sigma)}{2}\|\theta - \tilde{\theta}\|_2^2,$$

where $\widetilde{\theta} = a\theta + (1 - a)\tilde{\theta}$ for some $a \in (0, 1)$. Hence $\gamma \geq \psi(BR)\sigma_{\min}(\Sigma)$ in this example.
- *Sub-Gaussian gradients:* Assuming the bound $\mathbb{E}\|x\|_\infty \leq B$, this condition is easily verified. A simple calculation yields $\sigma = 2B$, since

$$\|e(\theta)\|_\infty = \|\nabla\mathcal{L}(\theta; (x, y)) - \nabla\bar{\mathcal{L}}(\theta)\|_\infty \leq \|\nabla\mathcal{L}(\theta; (x, y))\|_\infty + \|\nabla\bar{\mathcal{L}}(\theta)\|_\infty \leq 2B.$$

Example 2 (Least-squares regression)**.** In the regression setup, we are given samples of the form $z = (x, y) \in \mathbb{R}^d \times \mathbb{R}$. The loss function of interest is $\mathcal{L}(\theta; (x, y)) = (y - \langle\theta, x\rangle)^2/2$. To illustrate the conditions more clearly, we assume that our samples are generated as $y = \langle x, \theta^*\rangle + w$, where $w \sim \mathcal{N}(0, \eta^2)$ and $\mathbb{E}xx^T = \Sigma$ so that $\mathbb{E}\mathcal{L}(\theta; (x, y)) = \|\Sigma^{1/2}(\theta - \theta^*)\|_2^2/2$.

- *Locally Lipschitz:* For this example, the Lipschitz parameter $G(R)$ depends on the bound R. If we define $\rho(\Sigma) = \max_i \Sigma_{ii}$ to be the largest variance of a coordinate of x, then a direct calculation yields the bound $G(R) \leq \rho(\Sigma)R$.
- *LSC:* Again we focus on the case where the expectation is taken under the population distribution, where we have $\gamma = \sigma_{\min}(\Sigma)$.
- *Sub-Gaussian gradients:* Once again we assume that $\|x\|_\infty \leq B$. It can be shown with some work that Assumption 3 is satisfied with $\sigma^2(R) = 8\rho(\Sigma)^2R^2 + 4B^4R^2 + 10B^2\eta^2$.

3 Main results and their consequences

In this section we state our main results, regarding the convergence of Algorithm 1. We focus on the cases where Assumptions 1 and 3 hold over the entire set Ω, and RSC holds uniformly for all $\|\theta\|_1 \leq R_1$; key examples being the hinge and logistic losses from Example 1. Extensions to examples such as least-squares loss, which are not Lipschitz on all of Ω require a more delicate treatment and these results as well the proofs of our results can be found in the long version [2].

Formally, we assume that $G(R) \equiv G$ and $\sigma(R) \equiv \sigma$ in Assumptions 1 and 3. We also use γ to denote $\gamma(R_1)$ in Assumption 2. For a constant $\omega > 0$ governing the error probability in our results, we also define $\omega_i^2 = \omega^2 + 24\log i$ at epoch i. Our results assume that we run Algorithm 1 with

$$T_i \geq c_1 \left[\frac{s^2}{\gamma^2 R_i^2}\left((G^2 + \sigma^2)\log d + \omega_i^2\sigma^2\right) + \log d\right], \tag{9}$$

where c_1 is a universal constant. For a total of T iterations in Algorithm 1, we state our results for the parameter $\widehat{\theta}_T = y_{(K_T+1)}$ where K_T is the last epoch completed in T iterations.

3.1 Main theorem and some remarks

We start with our main result which shows an overall convergence rate of $O(1/T)$ after T iterations. This $\mathcal{O}(1/T)$ convergence is analogous to earlier work on multi-step methods for strongly convex objectives [11, 12, 10]. For each subset $S \subseteq \{1, 2, \ldots, d\}$ of cardinality s, we define

$$\varepsilon^2(\theta^*; S) := \|\theta^*_{S^c}\|_1^2 / s. \tag{10}$$

This quantity captures the degree of sparsity in the optimum θ^*; for instance, $\varepsilon^2(\theta^*; S) = 0$ if and only if θ^* is supported on S. Given the probability parameter $\omega > 0$, we also define the shorthand

$$\kappa_T = \log_2 \left[\frac{\gamma^2 R_1^2 T}{s^2((G^2 + \sigma^2) \log d + \omega^2 \sigma^2)} \right] \log d. \tag{11}$$

Theorem 1. Suppose the expected loss $\overline{\mathcal{L}}$ satisfies Assumptions 1— 3 with parameters $G(R) \equiv G$, γ and $\sigma(R) \equiv \sigma$, and we perform updates (4) with epoch lengths (9) and parameters

$$\lambda_i^2 = \frac{R_i \gamma}{s\sqrt{T_i}} \sqrt{(G^2 + \sigma^2) \log d + \omega_i^2 \sigma^2} \quad \text{and} \quad \alpha(t) = 5 R_i \sqrt{\frac{\log d}{(G^2 + \lambda_i^2 + \sigma^2) t}}. \tag{12}$$

Then for any subset $S \subseteq \{1, \ldots, d\}$ of cardinality s and any $T \geq 2\kappa_T$, there is a universal constant c_0 such that with probability at least $1 - 6\exp(-\omega^2/12)$ we have

$$\|\widehat{\theta}_T - \theta^*\|_2^2 \leq c_3 \left[\frac{s}{\gamma^2 T} ((G^2 + \sigma^2) \log d + \sigma^2(\omega^2 + \log \frac{\kappa_T}{\log d})) + \varepsilon^2(\theta^*; S) \right]. \tag{13}$$

Consequently, the theorem predicts a convergence rate of $O(1/\gamma^2 T)$ which is the best possible under our assumptions. Under the setup of Example 1, the error bound of Theorem 1 further simplifies to

$$\|\widehat{\theta}_T - \theta^*\|_2^2 = \mathcal{O}\left(\frac{sB^2}{\gamma^2 T} (\log d + \omega^2) + \varepsilon^2(\theta^*; S) \right). \tag{14}$$

We note that for an approximately sparse θ^*, Theorem 1 guarantees convergence only to a tolerance $\varepsilon^2(\theta^*; S)$ due to the error terms arising out of the approximate sparsity. Overall, the theorem provides a family of upper bounds, one for each choice of S. The best bound can be obtained by optimizing this choice, trading off the competing contributions of s and $\|\theta^*_{S^c}\|_1$.

At this point, we can compare the result of Theorem 1 to some of the previous work. One approach to minimize the objective (1) is to perform stochastic gradient descent on the objective, which has a convergence rate of $\mathcal{O}((\widetilde{G}^2 + \widetilde{\sigma}^2)/(\gamma^2 T))$ [10, 14], where $\|\nabla \overline{\mathcal{L}}(\theta)\|_2 \leq \widetilde{G}$ and $\mathbb{E} \exp\left(\frac{\|e(\theta)\|_2^2}{\widetilde{\sigma}^2} \right) \leq \exp(1)$. In the setup of Example 1, $\widetilde{G}^2 = Bd$ and similarly for $\widetilde{\sigma}$; giving an exponentially worse scaling in the dimension d. An alternative is to perform mirror descent [15, 19] or regularized dual averaging [22] using the same prox-function as Algorithm 1 but without breaking it up into epochs. As mentioned in the introduction, this single-step method fails to exploit the strong convexity of our problem and obtains inferior convergence rates of $\mathcal{O}(s\sqrt{\log d/T})$ [19, 22, 7].

A proposal closer to our approach is to minimize the regularized objective (3), but with a fixed value of λ instead of the decreasing schedule of λ_i used in Theorem 1. This amounts to using the method of Juditsky and Nesterov [11] on the regularized problem, and by using the proof techniques developed in this paper, it can be shown that setting $\lambda = \sigma\sqrt{\log d/T}$ leads to an overall convergence rate of $\widetilde{\mathcal{O}}\left(\frac{sB^2}{\gamma^2 T} (\log d + \omega^2) \right)$, which exhibits the same scaling as Theorem 1. However, with this fixed setting of λ, the initial epochs tend to be much longer than needed for halving the error. Indeed, our setting of λ_i is based on minimizing the upper bound at each epoch, and leads to an improved performance in our numerical simulations. The benefits of slowly decreasing the regularization in the context of deterministic optimization were also noted in the recent work of Xiao and Zhang [23].

3.2 Some illustrative corollaries

We now present some consequences of Theorem 1 by making specific assumptions regarding the sparsity of θ^*. The simplest situation is when θ^* is supported on some subset S of size s. More generally, Theorem 1 also applies to the case when the optimum θ^* is only approximately sparse. One natural form of approximate sparsity is to assume that $\theta^* \in \mathbb{B}_q(R_q)$ for $0 < q \leq 1$, where

$$\mathbb{B}_q(R_q) := \left\{ \theta \in \mathbb{R}^d \;\middle|\; \sum_{i=1}^d |\theta_i|^q \leq R_q \right\}.$$

For $0 < q \leq 1$, membership in the set $\mathbb{B}_q(R_q)$ enforces a decay rate on the components of the vector θ. We now present a corollary of Theorem 1 under such an approximate sparsity condition. To facilitate comparison with minimax lower bounds, we set $\omega^2 = \delta \log d$ in the corollaries.

Corollary 1. *Under the conditions of Theorem 1, for all $T > 2\kappa_T$ with probability at least $1 - 6\exp(-\delta \log d/12)$, there is a universal constant c_0 such that*

$$\|\widehat{\theta}_T - \theta^*\|_2^2 \leq \begin{cases} c_0 \left[\frac{G^2+\sigma^2(1+\delta)}{\gamma^2} \, \frac{s \log d}{T} + \frac{s\sigma^2}{\gamma^2 T} \log \frac{\kappa_T}{\log d}\right] & \theta^* \text{ is } s\text{-sparse}, \\ c_0 R_q \left[\left\{\frac{(G^2+\sigma^2(1+\delta)) \log d}{\gamma^2 T}\right\}^{\frac{2-q}{2}} + \left(\frac{\sigma^2}{\gamma^2 T}\right)^{\frac{2-q}{2}} \frac{\log \frac{\kappa_T}{\log d}}{((1+\delta) \log d)^{\frac{q}{2}}}\right] & \theta^* \in \mathbb{B}_q(R_q). \end{cases}$$

The first part of the corollary follows directly from Theorem 1 by noting that $\varepsilon^2(\theta^*; S) = 0$ under our assumptions. Note that as q ranges over the interval $[0, 1]$, reflecting the degree of sparsity, the convergence rate ranges from from $\widetilde{\mathcal{O}}(1/T)$ (for $q = 0$ corresponding to exact sparsity) to $\widetilde{\mathcal{O}}(1/\sqrt{T})$ (for $q = 1$). This is a rather interesting trade-off, showing in a precise sense how convergence rates vary quantitatively as a function of the underlying sparsity.

It is useful to note that the results on recovery for generalized linear models presented here exactly match those that have been developed in the statistics literature [13, 21], which are optimal under our assumptions on the design vectors. Concretely, ignoring factors of $\mathcal{O}(\log T)$, we get a parameter $\widehat{\theta}_T$ having error at most $\mathcal{O}(s \log d/(\gamma^2 T)$ with an error probability decyaing to zero with d. Moreover, in doing so our algorithm only goes over at most T data samples, as each stochastic gradient can be evaluated with one fresh data sample drawn from the underlying distribution. Since the statistical minimax lower bounds [13, 21] demonstrate that this is the smallest possible error that any method can attain from T samples, our method is statistically optimal in the scaling of the estimation error with the number of samples. We also observe that it is easy to instead set the error probability to $\delta = \omega^2 \log T$, if an error probability decaying with T is desired, incurring at most additional $\log T$ factors in the error bound. Finally, we also remark that our techniques extend to handle examples such as the least-squares loss that are not uniformly Lipschitz. The details of this extension are deferred to the long version of this paper [2].

Stochastic optimization over finite pools: A common setting for the application of stochastic optimization methods in machine learning is when one has a finite pool of examples, say $\{Z_1, \ldots, Z_n\}$, and the objective (1) takes the form

$$\theta^* = \arg\min_{\theta \in \Omega} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\theta; Z_i) \tag{15}$$

In this setting, a stochastic gradient $g(\theta)$ can be obtained by drawing a sample Z_j at random *with replacement* from the pool $\{Z_1, \ldots, Z_n\}$, and returning the gradient $\nabla \mathcal{L}(\theta; Z_j)$.

In high-dimensional problems where $d \gg n$, the sample loss is not strongly convex. However, it has been shown by many researchers [3, 13, 1] that under suitable conditions, this objective does satisfy restricted forms of the LSC assumption, allowing us to appeal to a generalized form of Theorem 1. We will present this corollary only for settings where θ^* is exactly sparse and also specialize to the logistic loss, $\mathcal{L}(\theta; (x, y)) = \log(1 + \exp(-y\langle\theta, x\rangle))$ to illustrate the key aspects of the result. We recall the definition of the link function $\psi(\alpha) = \exp(\alpha)/(1 + \exp(\alpha))^2$. We will state the result for sub-Gaussian data design with parameters (Σ, η_x^2), meaning that the $\mathbb{E}[x_i x_i^T] = \Sigma$ and $\langle u, x_i\rangle$ is η_x-sub-Gaussian for any unit norm vector $u \in \mathbb{R}^d$.

Corollary 2. *Consider the finite-pool loss* (15)*, based on n i.i.d. samples from a sub-Gaussian design with parameters (Σ, η_x^2). Suppose that Assumptions 1-3 are satisfied and the optimum θ^* of* (15) *is s-sparse. Then there are universal constants (c_0, c_1, c_2, c_3) such that for all $T \geq 2\kappa_T$ and $n \geq c_3 \frac{\log d}{\sigma_{\min}^2(\Sigma)} \max(\sigma_{\min}^2(\Sigma), \eta_x^4)$, we have*

$$\|\widehat{\theta}_T - \theta^*\|_2^2 \leq \frac{c_0}{\sigma_{\min}^2(\Sigma)} \, \frac{s \log d}{T} \left\{\frac{1}{\psi^2(2BR_1)} \left\{B^2(1+\delta)\right\}\right\} + c_0 \frac{s\sigma^2}{\sigma_{\min}^2(\Sigma)\psi^2(2BR_1)T} \log \frac{\kappa_T}{\log d}.$$

with probability at least $1 - 2\exp(-c_1 n \min(\sigma_{\min}^2(\Sigma)/\eta_x^4, 1)) - 6\exp(-\delta \log d/12)$.

We observe that the bound only holds when the number of samples n in the objective (15) is large enough, which is necessary for the restricted form of the LSC condition to hold with non-trivial parameters in the finite sample setting.

A modified method with constant epoch lengths: Algorithm 1 as described is efficient and simple to implement. However, the convergence results critically rely on the epoch length T_i to be set appropriately in a doubling manner. This could be problematic in practice, where it might be tricky to know when an epoch should be terminated. Following Juditsky and Nesterov [11], we next demonstrate how a variant of our algorithm with constant epoch lengths enjoys similar rates of convergence. The key challenge here is that unlike the previous set-up [11], our objective function changes at each epoch which leads to significant technical difficulties. At a very coarse level, if we have a total budget of T iterations, then this version of our algorithm allows us to set the epoch lengths to $\mathcal{O}(\log T)$, and guarantees convergence rates that are $\mathcal{O}((\log T)/T)$.

Theorem 2. Suppose the expected loss satisfies Assumptions 1- 3 with parameters G, γ, and σ resp. Let S be any subset of $\{1, \ldots, d\}$ of cardinality s. Suppose we run Algorithm 1 for a total of T iterations with epoch length $T_i \equiv T \log d / \kappa_T$ and with parameters as in Equation 12. Assuming that this setting ensures $T_i = \mathcal{O}(\log d)$, for any set S, with probability at least $1 - 3\exp(\omega^2/12)$

$$\|\widehat{\theta}_T - \theta^*\|_2^2 = \mathcal{O}\left(s\, \frac{(G^2 + \sigma^2)\log d + (\omega^2 + \log(\kappa/\log d))\sigma^2}{T}\, \frac{\log d}{\kappa}\right).$$

The theorem shows that up to logarithmic factors in T, setting the epoch lengths optimally is not critical. A similar result can also be proved for the case of least-squares regression.

4 Simulations

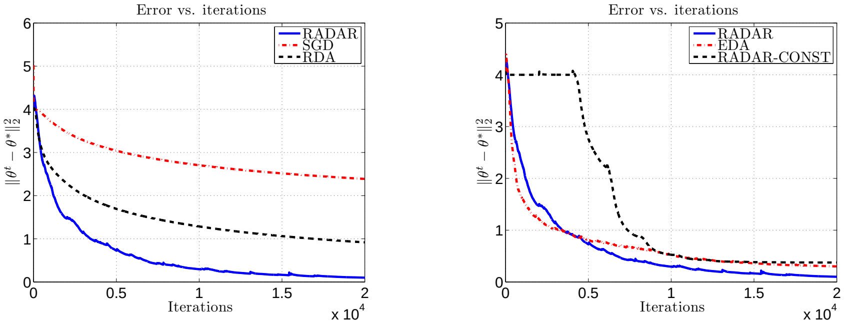
In this section we will present numerical simulations that back our theoretical convergence results. We focus on least-squares regression, discussed in Example 2. Specifically, we generate samples (x_t, y_t) with each coordinate of x_t distributed as Unif$[-B, B]$ and $y_t = \langle \theta^*, x_t \rangle + w_t$. We pick θ^* to be s-sparse vector with $s = \lceil \log d \rceil$, and $w_t \sim \mathcal{N}(0, \eta^2)$ with $\eta^2 = 0.5$. Given an iterate θ^t, we generate a stochastic gradient of the expected loss (1) at (x_t, y_t). For the ℓ_1-norm, we pick the sign vector of θ^t, with 0 for any component that is zero, a member of the ℓ_1-sub-differential.

Our first set of results evaluate Algorithm 1 against other stochastic optimization baselines assuming a complete knowledge of problem parameters. Specifically, we epoch i is terminated once $\|y_{i+1} - \theta^*\|_p^2 \leq \|y_i - \theta^*\|_p^2/2$. This ensures that θ^* remains feasible throughout, and tests the performance of Algorithm 1 in the most favorable scenario. We compare the algorithm against two baselines. The first baseline is the regularized dual averaging (RDA) algorithm [22], applied to the regularized objective (3) with $\lambda = 4\eta\sqrt{\log d/T}$, which is the statistically optimal regularization parameter with T samples. We use the same prox-function $\psi(\theta) = \frac{\|\theta\|_p^2}{2(p-1)}$, so that the theory for RDA predicts a convergence rate of $\mathcal{O}(s\sqrt{\log d/T})$ [22]. Our second baseline is the stochastic gradient (SGD) algorithm which exploits the strong convexity but not the sparsity of the problem (1). Since the squared loss is not uniformly Lipschitz, we impose an additional constraint $\|\theta\|_1 \leq R_1$, without which the algorithm does not converge. The results of this comparison are shown in Figure 1(a), where we present the error $\|\theta^t - \theta^*\|_2^2$ averaged over 5 random trials. We observe that RADAR comprehensively outperforms both the baselines, confirming the predictions of our theory.

The second set of results focuses on evaluating algorithms better tailored for our assumptions. Our first baseline here is the approach that we described in our remarks following Theorem 1. In this approach we use the same multi-step strategy as Algorithm 1 but keep λ fixed. We refer to this as Epoch Dual Averaging (henceforth EDA), and again employ $\lambda = 4\eta\sqrt{\log d/T}$ with this strategy. Our epochs are again determined by halving of the squared ℓ_p-error measured relative to θ^*.

Finally, we also evaluate the version of our algorithm with constant epoch lengths that we analyzed in Theorem 2 (henceforth RADAR-CONST), using epochs of length $\log(T)$. As shown in Figure 1(b), the RADAR-CONST has relatively large error during the initial epochs, before converging quite rapidly, a phenomenon consistent with our theory.[1] Even though the RADAR-CONST method does not use the knowledge of θ^* to set epochs, all three methods exhibit the same eventual convergence rates, with RADAR (set with optimal epoch lengths) performing the best, as expected. Although RADAR-CONST is very slow in initial iterations, its convergence rate remains competitive with EDA (even though EDA *does* exploit knowledge of θ^*), but is worse than RADAR as expected.

Overall, our experiments demonstrate that RADAR and RADAR-CONST have practical performance consistent with our theoretical predictions. Although optimal epoch length setting is not too critical for our approach, better data-dependent empirical rules for determining epoch lengths remains an interesting question for future research. The relatively poorer performance of EDA demonstrates the importance of our decreasing regularization schedule.



(a)

(b)

Figure 1. A comparison of RADAR with other stochastic optimization algorithms for $d = 40000$ and $s = \lceil \log d \rceil$. The left plot compares RADAR with the RDA and SGD algorithms, neither of which exploits both the sparsity and the strong convexity structures simultaneously. The right one compares RADAR with the EDA and RADAR-CONST algorithms, all of which exploit the problem structure but with varying degrees of effectiveness. We plot $\|\theta^t - \theta^*\|_2^2$ averaged over 5 random trials versus the number of iterations.

5 Discussion

In this paper we present an algorithm that is able to take advantage of the strong convexity and sparsity conditions that are satsified by many common problems in machine learning. Our algorithm is simple and efficient to implement, and for a d-dimensional objective with an s-sparse optima, it achieves the minimax-optimal convergence rate $\mathcal{O}(s \log d/T)$. We also demonstrate optimal convergence rates for problems that have weakly sparse optima, with implications for problems such as sparse linear regression and sparse logistic regression. While we focus our attention exclusively on sparse vector recovery due to space constraints, the ideas naturally extend to other structures such as group sparse vectors and low-rank matrices. It would be interesting to study similar developments for other algorithms such as mirror descent or Nesterov's accelerated gradient methods, leading to multi-step variants of those methods with optimal convergence rates in our setting.

Acknowledgements

The work of all three authors was partially supported by ONR MURI grant N00014-11-1-0688 to MJW. In addition, AA was partially supported by a Google Fellowship, and SNN was partially supported by the Yahoo KSC award.

[1] To clarify, the epoch lengths in RADAR-CONST are set large enough to guarantee that we can attain an overall error bound of $\mathcal{O}(1/T)$, meaning that the initial epochs for RADAR-CONST are much longer than for RADAR. Thus, after roughly 500 iterations, RADAR-CONST has done only 2 epochs and operates with a crude constraint set $\Omega(R_1/4)$. During epoch i, the step size scales proportionally to $R_i/\sqrt{t}$, where t is the iteration number within the epoch; hence the relatively large initial steps in an epoch can take us to a bad solution even when we start with a good solution y_i when R_i is large. As R_i decreases further with more epochs, this effect is mitigated and the error of RADAR-CONST does rapidly decrease like our theory predicts.

References

[1] A. Agarwal, S. N. Negahban, and M. J. Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *To appear in The Annals of Statistics*, 2012. Full-length version http://arxiv.org/pdf/1104.4824v2.

[2] A. Agarwal, S. N. Negahban, and M. J. Wainwright. Stochastic optimization and sparse statistical recovery: An optimal algorithm for high dimensions. 2012. URL http://arxiv.org/abs/1207.4421.

[3] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of Lasso and Dantzig selector. *Ann. Stat.*, 37(4):1705–1732, 2009.

[4] L. Bottou and O. Bousquet. The tradeoffs of large scale learning. In *NIPS*, 2007.

[5] P. Bühlmann and S. Van De Geer. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer Series in Statistics. Springer, 2011.

[6] D. L. Donoho. High-dimensional data analysis: The curses and blessings of dimensionality, 2000.

[7] J. Duchi, S. Shalev-Shwartz, Y. Singer, and A. Tewari. Composite objective mirror descent. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pages 14–26. Omnipress, 2010.

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

[9] E. Hazan, A. Kalai, S. Kale, and A. Agarwal. Logarithmic regret algorithms for online convex optimization. In *Proceedings of the Nineteenth Annual Conference on Computational Learning Theory*, 2006.

[10] E. Hazan and S. Kale. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. *Journal of Machine Learning Research - Proceedings Track*, 19:421–436, 2011.

[11] A. Juditsky and Y. Nesterov. Primal-dual subgradient methods for minimizing uniformly convex functions. Available online http://hal.archives-ouvertes.fr/docs/00/50/89/33/PDF/Strong-hal.pdf, 2010.

[12] G. Lan and S. Ghadimi. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, part ii: shrinking procedures and optimal algorithms. 2010.

[13] S. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. In *NIPS Conference*, Vancouver, Canada, December 2009. Full length version arxiv:1010.2731v1.

[14] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609, 2009.

[15] A. Nemirovski and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley, New York, 1983.

[16] Y. Nesterov. Gradient methods for minimizing composite objective function. Technical Report 76, Center for Operations Research and Econometrics (CORE), Catholic University of Louvain (UCL), 2007.

[17] Y. Nesterov. Primal-dual subgradient methods for convex problems. *Mathematical Programming A*, 120(1):261–283, 2009.

[18] S. Shalev-Shwartz, Y. Singer, and N. Srebro. Pegasos: Primal estimated sub-gradient solver for SVM. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[19] S. Shalev-Shwartz and A. Tewari. Stochastic methods for l_1 regularized loss minimization. *Journal of Machine Learning Research*, 12:1865–1892, June 2011.

[20] N. Srebro, K. Sridharan, and A. Tewari. Smoothness, low noise, and fast rates. In *Advances in Neural Information Processing Systems 23*, pages 2199–2207, 2010.

[21] S. A. van de Geer. High-dimensional generalized linear models and the lasso. *The Annals of Statistics*, 36:614–645, 2008.

[22] L. Xiao. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research*, 11:2543–2596, 2010.

[23] L. Xiao and T. Zhang. A proximal-gradient homotopy method for the sparse least-squares problem. *ICML*, 2012. URL http://arxiv.org/abs/1203.3002.