lions@epfl

EE-731: ADVANCED TOPICS IN DATA SCIENCES
LABORATORY FOR INFORMATION AND INFERENCE SYSTEMS
SPRING 2016

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

INSTRUCTOR: VOLKAN CEVHER

SCRIBERS: ALP YURTSEVER AND BARAN GÖZCÜ

STOCHASTIC GRADIENT METHODS

Stochastic gradients (SG) methods have a long history in convex optimization theory, which dates back to 1951 [1]. Yet, there has been an outburst of interest in SG methods very recently, following the developments in machine learning and network applications. The appeal of these methods is mainly due to the efficiency of these methods in big data optimization, where the data is either too large to be processed in the full batch setting, or simply decentralized in space or time.

In the rest of this report, we assume that the reader is knowledgable on the basic probability theory, and we refer to [2] and [3] for the students who would like to refresh their knowledge on the topic.

1 Motivation

In this section, we present two motivating examples that show superiority of SG methods over the deterministic first order methods for some important problems.

1.1 Big data ($n \gg p$):

An important feature of the SG methods is their ability to solve problems by examining only a small fraction of the dataset. This feature is particularly important in the *big data* applications, characterized by $n \gg p$ condition, where n is the number of data points and p is the ambient dimension of the problem. We consider the following simple least squares problem:

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 \right\}, \tag{1}$$

where $\mathbf{A} \in \mathbb{R}^{n\times p}$ and $\mathbf{b} \in \mathbb{R}^n$ with $n = 10^4$ and $p = 10^2$.

We use a synthetic dataset, where the entries of the measurement matrix $\mathbf{A}$ are chosen according to the Gaussian distribution $\mathcal{N}(0, 1/\sqrt{p})$. We also generate the true solution $\mathbf{x}^\natural$ choosing its entries from the standard Gaussian distribution $\mathcal{N}(0, 1)$. Then the measurements are generated by $\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is white Gaussian noise, and the noise level is 30 dB.

Figure 1 shows the convergence of the iterates and the convergence in the objective value as a function of the data passes (epoch). Denote that the SG method achieves a moderately accurate solution in less than 1/5 epoch.

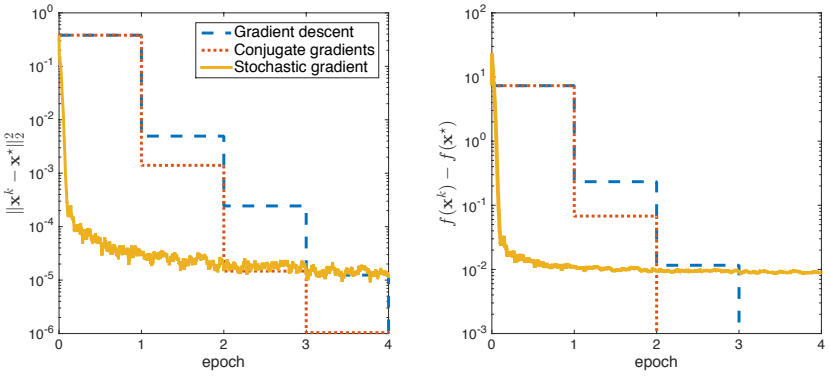


Figure 1: Comparison of SG methods with first order methods for solving (1).

1.2 Statistical Learning

The second motivating example for SG methods is the problems with statistical learning models. A statistical learning model consists of the following three elements:

1. A sample of i.i.d. random variables $(\mathbf{a}_i, b_i) \in \mathcal{A} \times \mathcal{B}$, $i = 1, \dots, n$, following an *unknown* probability distribution $\mathbb{P}$.
2. A class (set) $\mathcal{F}$ of functions $f : \mathcal{A} \to \mathcal{B}$.
3. A loss function $L : \mathcal{B} \times \mathcal{B} \to \mathbb{R}$.

Let $(\mathbf{a}, b)$ follow the probability distribution $\mathbb{P}$ and be independent of $(\mathbf{a}_1, b_1), \dots, (\mathbf{a}_n, b_n)$. Then, the *risk* corresponding to any $f \in \mathcal{F}$ is its expected loss:

$$R(f) := \mathbb{E}_{(\mathbf{a},b)} \left[L(f(\mathbf{a}), b) \right].$$

Statistical learning seeks to find a $f^\star \in \mathcal{F}$ that minimizes the risk, i.e., it solves

$$f^\star \in \arg\min_f \{ R(f) : f \in \mathcal{F} \}.$$

Note that, many machine learning problems indeed cast into the risk minimization formulation. Deterministic algorithms are not appropriate to tackle these problems simply because we do not have access to the objective function and the gradient. This formulation also models the problem with decentralized or streaming data.

By the law of large numbers, we can expect that for each $f \in \mathcal{F}$,

$$R(f) := \mathbb{E} \left[L(\mathbf{a}, b) \right] \approx \frac{1}{n} \sum_{i=1}^{n} L(f(\mathbf{a}_i), b_i),$$

when n is large enough, with high probability. Then, we approximate $f^\star$ by minimizing the *empirical average of the loss* instead of the risk, since we do not have access to the risk directly. That is, we consider the following optimization problem formulation:

$$\hat{f}_n \in \arg\min_f \left\{ \frac{1}{n} \sum_{i=1}^{n} L(f(\mathbf{a}_i), b_i) : f \in \mathcal{F} \right\}.$$

This is called the empirical risk minimization principle [4].

1.2.1 Markowitz portfolio optimization

Markowitz portfolio optimization problem considered in [6] is an example for this problem setup, where the main aim is to minimize the variance of return given a desired gain:

$$F^\star := \min_{\mathbf{x} \in \mathcal{X}} \left\{ \mathbb{E}[|\rho - \langle \mathbf{x}, \theta_t \rangle|^2] \right\}.$$

Here, $\rho \in \mathbb{R}$ is the desired return, and $\mathcal{X}$ is the intersection of the standard simplex and the constraint $\langle \mathbf{x}, \mathbb{E}[\theta_t] \rangle \geq \rho$. θ_t is the vector of returns at time t, which is modelled by a random variable. For the details of this model formulation, we refer to [7].

Figure 2 presents the convergence of the iterates and the variance of the return for three operator splitting method [5] and its stochastic variant [6], for four different datasets[1]. Note that the stochastic method solves the problem with a good accuracy before the end of first data pass (epoch).

Many important loss functions from convex optimization and machine learning problems can be considered from a statistical learning point of view. For instance, recall that the least squares estimator is given by

$$\hat{\mathbf{x}} \in \arg\min_{\mathbf{x}} \left\{ \frac{1}{2} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2n} \sum_{i=1}^{n} (b_i - \langle \mathbf{a}_i, \mathbf{x} \rangle)^2 : \mathbf{x} \in \mathbb{R}^p \right\}, \tag{2}$$

where we define $\mathbf{b} := (b_1, \dots, b_n)$ and $\mathbf{a}_i$ as i-th row of $\mathbf{A}$. Clearly this corresponds to a statistical learning model, for which

- the sample is given by $(\mathbf{a}_i, b_i) \in \mathbb{R}^p \times \mathbb{R}$, $i = 1, \dots, n$,
- the function class $\mathcal{F}$ is given by $\mathcal{F} := \{ f_{\mathbf{x}}(\cdot) := \langle \cdot, \mathbf{x} \rangle : \mathbf{x} \in \mathbb{R}^p \}$, and
- the loss function is given by $L(f_{\mathbf{x}}(\mathbf{a}), b) := (b - f_{\mathbf{x}}(\mathbf{a}))^2$.

[1] These datasets are available from `http://www.cs.technion.ac.il/~rani/portfolios`.

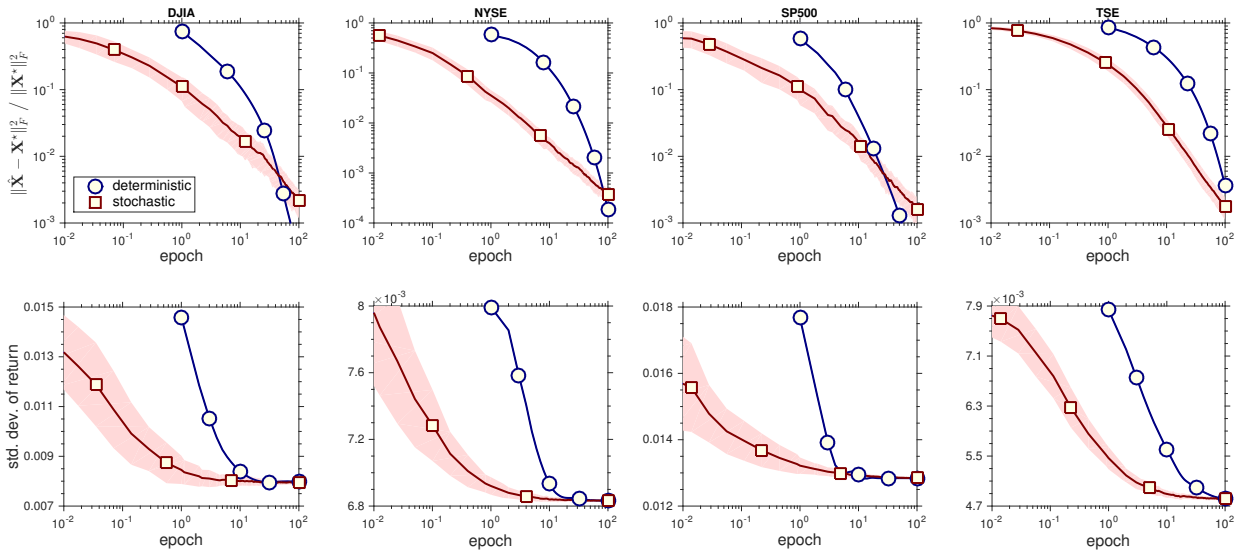


Figure 2: Comparison of stochastic and batch variants of three operator splitting method for Markowitz portfolio optimization [6].

Then, the corresponding ERM solution is

$$\hat{f}_n(\cdot) := \langle \cdot, \hat{\mathbf{x}} \rangle .$$

In other words, given $\mathbf{a}$, LS estimator seeks to minimize the error of predicting the corresponding b by a linear function, in terms of the squared error.

Similarly, recall the unconstrained SVM formulation

$$\hat{\mathbf{x}} \in \arg\min_{\mathbf{x}} \left\{ \frac{1}{n} \sum_{j=1}^{n} \max\{1 - b_j \langle \mathbf{a}_j, \mathbf{x} \rangle, 0\} + \lambda \|\mathbf{x}\|_2^2 : \mathbf{x} \in \mathbb{R}^p \right\}$$

where $\mathbf{b} := (b_1, \ldots, b_n) \in \{-1, 1\}^n$. This corresponds to a statistical learning model, for which

- the sample is given by $(\mathbf{a}_i, b_i) \in \mathbb{R}^p \times \mathbb{R}$, $i = 1, \ldots, n$,
- the function class $\mathcal{F}$ is given by $\mathcal{F} := \{f_{\mathbf{x}}(\cdot) := \langle \cdot, \mathbf{x} \rangle : \mathbf{x} \in \mathbb{R}^p\}$, and
- the loss function is given by hinge loss: $L(f_{\mathbf{x}}(\mathbf{a}), b) := \max\{0, 1 - b f_{\mathbf{x}}(\mathbf{a})\}$.

Then, the corresponding ERM solution is

$$\hat{f}_n(\cdot) := \langle \cdot, \hat{\mathbf{x}} \rangle .$$

So, given $\mathbf{a}$, SVM aims to minimize the error of predicting the corresponding b by a linear function, in terms of the hinge loss.

Finally, recall the logistic regression formulation

$$\hat{\mathbf{x}} \in \arg\min_{\mathbf{x},\delta} \left\{ \frac{1}{n} \sum_{j=1}^{n} \log\left(1 + e^{-b_j(\langle \mathbf{x}, \mathbf{a}_j \rangle + \delta)}\right) : \mathbf{x} \in \mathbb{R}^p, \delta \in \mathbb{R} \right\}$$

where $\mathbf{b} := (b_1, \ldots, b_n) \in \{-1, 1\}^n$. This corresponds to a statistical learning model, for which

- the sample is given by $(\mathbf{a}_i, b_i) \in \mathbb{R}^p \times \mathbb{R}$, $i = 1, \ldots, n$,
- the function class $\mathcal{F}$ is given by $\mathcal{F} := \{f_{\mathbf{x}}(\cdot) := \langle \cdot, \mathbf{x} \rangle : \mathbf{x} \in \mathbb{R}^p\}$, and
- the loss function is given by the log-loss: $L(f_{\mathbf{x}}(\mathbf{a}), b) := \log\left(1 + e^{-b f_{\mathbf{x}}(\mathbf{a})}\right)$.

The corresponding ERM solution is

$$\hat{f}_n(\cdot) := \langle \cdot, \hat{\mathbf{x}} \rangle .$$

Hence, given $\mathbf{a}$, logistic regression aims to minimize the error of predicting the corresponding b by a linear function, in terms of the log-loss.

2 Stochastic projected gradient method (SG)

In this section we consider the following constrained convex minimization problem and present the stochastic projected gradient method (SG) to solve this problem:

$$f^\star = \min_{\mathbf{x}\in\mathcal{X}} \{f(\mathbf{x}) := \mathbb{E}[h(\mathbf{x},\theta)]\}$$

where $\mathcal{X} \subset \mathbb{R}^p$ is a non-empty bounded closed convex set, θ is a random vector whose probability distribution is supported on set Θ, f is continuous and convex on $\mathcal{X}$ and the solution set $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}(f) : f(\mathbf{x}^\star) = f^\star\}$ is nonempty. The pseudocode of SG is given below, where we denote by $P_\mathcal{X}$ the orthogonal projection operator onto the set $\mathcal{X}$.

Stochastic projected gradient method (SG)

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ and $(\gamma_k)_{k\in\mathbb{N}} \in]0, +\infty[^{\mathbb{N}}$.
2. For $k = 0, 1, \ldots$ perform:

$$\mathbf{x}^{k+1} = P_\mathcal{X}(\mathbf{x}^k - \gamma_k G(\mathbf{x}^k, \theta_k)).$$

Remark that, SG shares the same structure as the projected gradient descent method, but the gradient is replaced by an unbiased estimate in the 2nd step. The cost of computing this estimate is typically much cheaper than that of $\nabla f(\mathbf{x}^k)$. For instance, let us consider the simple least-squares example (2), where we can write the gradient $\nabla f(\mathbf{x}^k)$ and an unbiased estimate $G(\mathbf{x}^k, (\mathbf{a}_i, b_i))$ as follows:

$$\nabla f(\mathbf{x}^k) = \mathbf{A}^T(\mathbf{A}\mathbf{x}^k - \mathbf{b}), \quad G(\mathbf{x}^k, (\mathbf{a}_i, b_i)) = \mathbf{a}_i(\langle \mathbf{a}_i, \mathbf{x}^k\rangle - b_i),$$

where $\mathbf{A} \in \mathbb{R}^{n\times p}$ is a tall matrix, i.e., $n \gg p$. Note that the computational complexity of the gradient evaluation is $O(np)$, since it is dominated by the matrix vector product. On the other hand, evaluation of an unbiased estimator requires an inner product and its complexity is $O(p)$.

Theorem 1. [Mean convergence of SG [8]]
Suppose that:

1. f is μ-strongly convex,
2. $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2] \le M^2$,
3. $\gamma_k = \gamma_0/(k+1)$ with $\gamma_0 > \frac{\mu}{2}$.

Then,

$$\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|^2] \le \frac{1}{k}\max\left\{\frac{\gamma_0^2 M^2}{2\gamma_0\mu - 1}, \|\mathbf{x}^0 - \mathbf{x}^\star\|^2\right\}.$$

If, in addition,

1. $\mathbf{x}^\star \in \mathrm{int}(\mathcal{X})$,
2. ∇f is L-Lipschitz continuous.

Then,

$$\mathbb{E}[f(\mathbf{x}^k) - f(\mathbf{x}^\star)] \le \frac{L}{2k}\max\left\{\frac{\gamma_0^2 M^2}{2\gamma_0\mu - 1}, \|\mathbf{x}^0 - \mathbf{x}^\star\|^2\right\}.$$

Remark: $O(1/k)$ rate in the objective residual is optimal for stochastic gradient methods under strong convexity assumption.

Theorem 2. Almost sure convergence of SG [8]
Denote $\mathcal{F}_k = \sigma(\mathbf{x}^0, \theta_0, \ldots, \theta_{k-1})$. Suppose that:

1. ∇f is L-Lipschitz continuous,
2. $\sum_{k=0}^\infty \gamma_k = \infty$ and $\sum_{k=0}^\infty \gamma_k^2 < \infty$.
3. $\sum_{k=0}^\infty \gamma_k^2\, \mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|^2 | \mathcal{F}_k] < +\infty$ almost surely.

Then,

$$\mathbf{x}^k \to \mathbf{x}^\star \text{almost surely.}$$

Remarks:

- (2) is satisfied: $\gamma_k = \gamma_0/(k+1)$.
- (3) is satisfied: $\sup_{k\in\mathbb{N}} \mathbb{E}[\|G(\mathbf{x}^k,\theta_k) - \nabla f(\mathbf{x}^k)\|^2|\mathcal{F}_k] < +\infty$.

Note that the conditions required for the convergence rate guarantee in the mean is stricter. We need f to be strongly convex and $\mathbb{E}[\|G(\mathbf{x}^k,\theta_k)\|^2] \leq M^2$. In fact, $\mathbb{E}[\|G(\mathbf{x}^k,\theta_k)\|^2] \leq M^2$ implies the condition $\mathbb{E}[f(\mathbf{x}^k) - f(\mathbf{x}^\star)] \leq \frac{L}{2k}\max\left\{\frac{\gamma_0^2 M^2}{2\gamma_0\mu - 1}, \|\mathbf{x}^0 - \mathbf{x}^\star\|^2\right\}$ that is assumed for almost sure convergence.

2.1 Numerical example: ℓ_1 constrained least squares problem

We consider the following simple problem formulation:

$$\min_{\mathbf{x}\in\mathbb{R}^p}\left\{f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 : \|\mathbf{x}\|_1 \leq 1\right\}. \tag{3}$$

We consider a synthetic problem setup, where each entry of the measurement matrix $\mathbf{A}$ follows the standard Gaussian distribution $\mathcal{N}(0,1)$. Problem dimensions n and p are 10^4 and 10^2 respectively. We generate a 10-sparse true signal $\mathbf{x}^\natural$, where the non-zero entries are randomly chosen with Gaussian distribution with 0 mean, and the signal is normalized to $\|\mathbf{x}^\natural\|_1 = 1$. We generate noisy measurements $\mathbf{b} := \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is Gaussian white noise and the SNR is $30dB$. We refer to [9] for efficient projections onto the ℓ_1 norm ball.

Figure 3 presents empirical convergence of SG and the theoretical bound provided by Theorem 1, with the optimum choice of the initial stepsize, i.e., $\gamma_0 = \mu/2$. Remark that, in many applications of big data optimization or machine learning problems, we do not have access to the strong convexity parameter μ a priori. Figure 4 shows the convergence of SG for different choices of the initial stepsize. Note that the empirical performance changes remarkable even with a small mismatch in the strong convexity constant estimation.

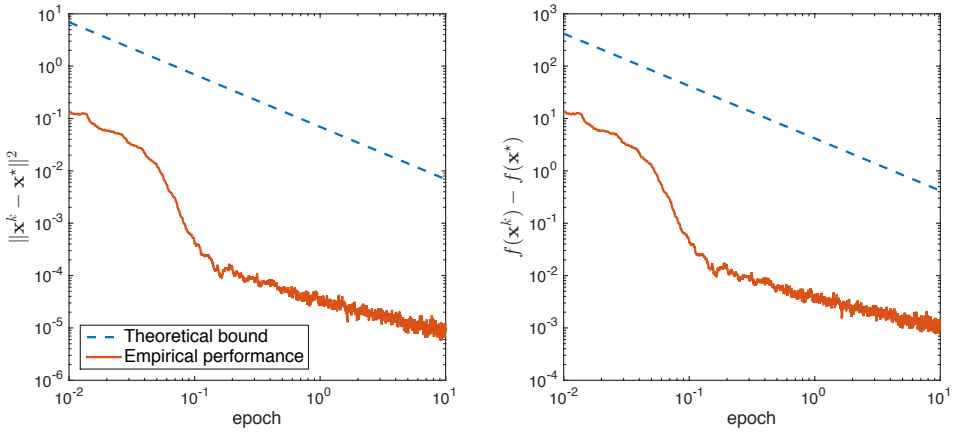


Figure 3: Empirical convergence of SG and the theoretical bound.

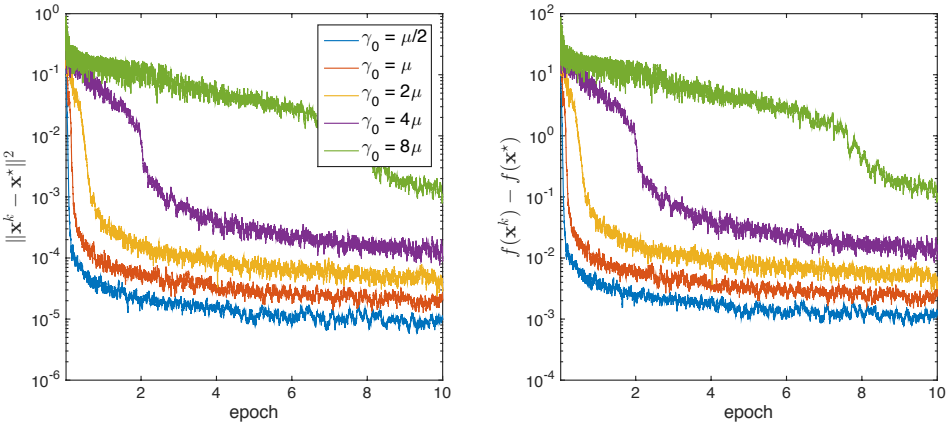


Figure 4: Empirical convergence of SG and with different learning rates.

3 Stochastic projected gradient method with averaging (ASG)

This section presents the first variant of SG methods which adopts a simple averaging step. This variant does not assume strong convexity, and it is more robust to the stepsize choice compared to SG. Convergence rate of this variant is $O(\frac{1}{\sqrt{k}})$, and this rate is optimal for stochastic methods we assume strong convexity. The pseudocode of ASG is given below, where we denote by $P_{\mathcal{X}}$ the orthogonal projection operator onto the set $\mathcal{X}$.

Stochastic gradient method with averaging (ASG)

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ and $(\gamma_k)_{k\in\mathbb{N}} \in]0, +\infty[^{\mathbb{N}}$.
2a. For $k = 0, 1, \ldots$ perform:

$$\mathbf{x}^{k+1} = P_{\mathcal{X}}(\mathbf{x}^k - \gamma_k G(\mathbf{x}^k, \theta_k)).$$

2b. $\bar{\mathbf{x}}^k = (\sum_{j=0}^k \gamma_j)^{-1} \sum_{j=0}^k \gamma_j \mathbf{x}^j$.

Theorem 3. [Mean convergence of ASG [8]]
Denote $D_{\mathcal{X}} = \max_{\mathbf{x}\in\mathcal{X}} \|\mathbf{x}^0 - \mathbf{x}\|^2$ and assume that $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2] \leq M^2$ for some $M \in]0, +\infty[$. Then,

$$\mathbb{E}[f(\bar{\mathbf{x}}^k) - f(\mathbf{x}^\star)] \leq \frac{D_{\mathcal{X}}^2 + M^2 \sum_{j=0}^k \gamma_j^2}{2 \sum_{j=0}^k \gamma_j},$$

In addition, choosing $\gamma_k = D_{\mathcal{X}}/(M\sqrt{k})$, we get,

$$\mathbb{E}[f(\bar{\mathbf{x}}^k) - f(\mathbf{x}^\star)] \leq \frac{M D_{\mathcal{X}}}{\sqrt{k}}.$$

Figure 5 presents the convergence of this variant with different stepsize choices for the same problem setup considered in section 2.1. In comparison to the Figure 4, it provides an empirical evidence to the robustness of this variant for the stepsize choice.

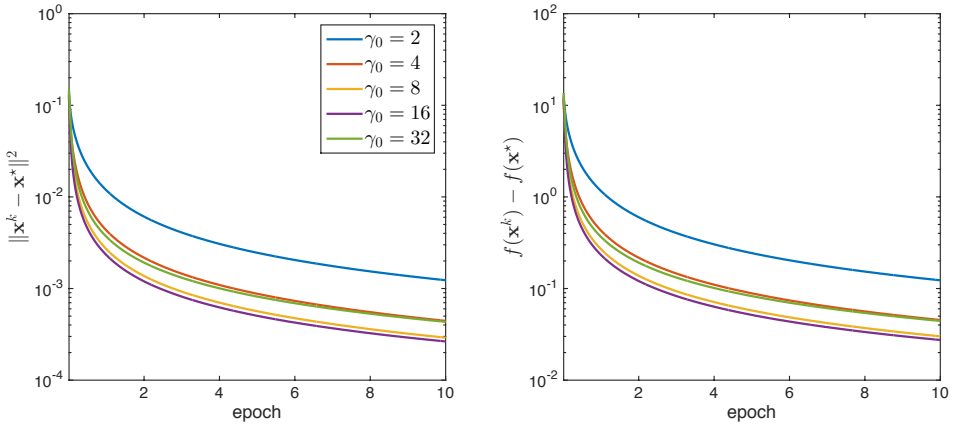


Figure 5: Empirical convergence of ASG with different learning rates.

Note that the results above that we have discussed in sections 2-3 can be genralized for the problems with nonsmooth objectives, simply replacing the gradient estimates by the subgradient estimates in the algorithmic flow.

4 Stochastic proximal gradient method (SPGM)

In many real world applications in convex optimization and machine learning, underlying objective function can be written as the summation of a smooth and nonsmooth terms. This generic problem formulation is called as the convex composite minimization problem. Let us denote by f and g (both are proper, closed and convex functions) to be the smooth and the nonsmooth parts respectively, then we can write this problem formulation as

$$F^\star = \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ F(\mathbf{x}) := f(\mathbf{x}) + g(\mathbf{x}) \Big\}. \tag{4}$$

Here, we assume that the solution set $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}(F) : F(\mathbf{x}^\star) = F^\star\}$ is nonempty. Typically in the applications, smooth term f represents the data fidelity and the nonsmooth term g is a regularizer which encourages some desired structures and/or constraints in the solution.

While these problems can be solved using the subgradient methods, considering the subgradient of overall nonsmooth objective F, these methods are inefficient in terms of iteration complexity. On the other hand, if the proximal operator of the nonsmooth part,

$$\mathrm{prox}_g(\mathbf{x}) \equiv \arg\min_{\mathbf{y}\in\mathbb{R}^p} \Big\{ g(\mathbf{y}) + \frac{1}{2}\|\mathbf{y}-\mathbf{x}\|_2^2 \Big\}, \tag{5}$$

is cheap to compute, then solving (4) is as efficient as solving an unconstraint smooth convex minimization problem in terms of complexity. Fortunately, proximal operator of many well-known convex regularization functions can be computed either analytically or very efficiently. Table 1 provides a non-exhaustive list of proximal operators of common regularization functions and their computational complexities.

In this section, we will specifically consider the composite convex optimization problems of the following formulation:

$$F^\star = \min_{\mathbf{x}\in\mathbb{R}^p} \Big\{ F(\mathbf{x}) := \mathbb{E}[h(\mathbf{x},\theta)] + g(\mathbf{x}) \Big\}, \tag{6}$$

where $f := \mathbb{E}[h(\mathbf{x},\theta)]$ and g are both proper, closed and convex functions. f is assumed to be a smooth function, in the sense that ∇f is Lipschitz continuous, and g is possibly nonsmooth. θ is a random vector whose distribution is supported on Θ, and we assume that it is possible to generate an i.i.d. sample $(\theta_k)_{k\in\mathbb{N}}$ of realizations of θ.

Assuming that we can find a vector $G(\mathbf{x},\theta)$ such that $\mathbb{E}[G(\mathbf{x},\theta)] = \nabla f(\mathbf{x})$ given $(\mathbf{x},\theta) \in \mathbb{R}^p \times \Omega$, we can solve (6) using the stochastic proximal gradient method (SPGM).

Stochastic proximal gradient method (SPGM)

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ and $(\gamma_k)_{k\in\mathbf{N}} \in]0, +\infty[^{\mathbf{N}}$.
2. For $k = 0, 1, \ldots$ perform:

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\gamma_k g}(\mathbf{x}^k - \gamma_k G(\mathbf{x}^k, \theta_k)).$$

Name	Function	Proximal operator	Complexity
ℓ_1-norm	$f(\mathbf{x}) := \|\mathbf{x}\|_1$	$\text{prox}_{\lambda f}(\mathbf{x}) = \text{sign}(\mathbf{x}) \otimes [\|\mathbf{x}\| - \lambda]_+$	$O(p)$
ℓ_2-norm	$f(\mathbf{x}) := \|\mathbf{x}\|_2$	$\text{prox}_{\lambda f}(\mathbf{x}) = [1 - \lambda/\|\mathbf{x}\|_2]_+\mathbf{x}$	$O(p)$
Support function	$f(\mathbf{x}) := \max_{\mathbf{y}\in\mathcal{C}} \mathbf{x}^T\mathbf{y}$	$\text{prox}_{\lambda f}(\mathbf{x}) = \mathbf{x} - \lambda P_{\mathcal{C}}(\mathbf{x})$	
Box indicator	$f(\mathbf{x}) := \delta_{[\mathbf{a},\mathbf{b}]}(\mathbf{x})$	$\text{prox}_{\lambda f}(\mathbf{x}) = P_{[\mathbf{a},\mathbf{b}]}(\mathbf{x})$	$O(p)$
Positive semidefinite cone indicator	$f(\mathbf{X}) := \delta_{\mathbb{S}^p_+}(\mathbf{X})$	$\text{prox}_{\lambda f}(\mathbf{X}) = \mathbf{U}[\Sigma]_+\mathbf{U}^T$, where $\mathbf{X} = \mathbf{U}\Sigma\mathbf{U}^T$	$O(p^3)$
Hyperplane indicator	$f(\mathbf{x}) := \delta_{\mathcal{X}}(\mathbf{x}), \mathcal{X} := \{\mathbf{x} \,:\, \mathbf{a}^T\mathbf{x} = b\}$	$\text{prox}_{\lambda f}(\mathbf{x}) = P_{\mathcal{X}}(\mathbf{x}) = \mathbf{x} + (\frac{b-\mathbf{a}^T\mathbf{x}}{\|\mathbf{a}\|_2})\mathbf{a}$	$O(p)$
Simplex indicator	$f(\mathbf{x}) = \delta_{\mathcal{X}}(\mathbf{x}), \mathcal{X} := \{\mathbf{x} \,:\, \mathbf{x} \geq 0,\ \mathbf{1}^T\mathbf{x} = 1\}$	$\text{prox}_{\lambda f}(\mathbf{x}) = (\mathbf{x} - \nu\mathbf{1})$ for some $\nu \in \mathbb{R}$, which can be efficiently calculated	$\tilde{O}(p)$
Convex quadratic	$f(\mathbf{x}) := (1/2)\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{q}^T\mathbf{x}$	$\text{prox}_{\lambda f}(\mathbf{x}) = (\lambda\mathbb{I} + \mathbf{Q})^{-1}\mathbf{x}$	$O(p \log p) \to O(p^3)$
Square ℓ_2-norm	$f(\mathbf{x}) := (1/2)\|\mathbf{x}\|_2^2$	$\text{prox}_{\lambda f}(\mathbf{x}) = (1/(1+\lambda))\mathbf{x}$	$O(p)$
log-function	$f(\mathbf{x}) := -\log(x)$	$\text{prox}_{\lambda f}(x) = ((x^2 + 4\lambda)^{1/2} + x)/2$	$O(1)$
log det-function	$f(\mathbf{x}) := -\log\det(\mathbf{X})$	$\text{prox}_{\lambda f}(\mathbf{X})$ is the log-function prox applied to the individual eigenvalues of $\mathbf{X}$	$O(p^3)$

Table 1: Non-exhaustive list of proximal operators of common regularization functions with computational complexities. Here: $[\mathbf{x}]_+ := \max\{0, \mathbf{x}\}$ and $\delta_{\mathcal{X}}$ is the indicator function of the convex set $\mathcal{X}$, sign is the sign function, $\mathbb{S}^p_+$ is the cone of symmetric positive semidefinite matrices, $P_{\mathcal{X}}$ is the orthogonal projection on $\mathcal{X}$, $\otimes$ is the pointwise multiplication operator and $[\mathbf{x}]_{+i} = \max\{x_i, 0\}$. For more functions, see [10, 11].

Remark that, SPGM shares the same structure as the classical proximal gradient method, but the gradient is replaced by an unbiased estimate in the 2nd step. Moreover, SPGM reduces to SG as a special case, by setting g as the indicator function of the constraint set $\mathcal{X}$.

Theorem 4. [Mean convergence of the iterates of SPGM [12]]
Suppose that:

1. f and g are (strongly) convex with $\mu_f \geq 0$ and $\mu_g \geq 0$ such that $\mu := \mu_f + \mu_g > 0$.
2. $\sup_{k\in\mathbb{N}} \mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|^2 | \mathcal{F}_k] \leq M^2$.
3. $\gamma_k = \gamma_0/k^\alpha$ with $0 < \alpha \leq 1$.

Then, for k large enough,

$$\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|_2^2] = \begin{cases} O(k^\alpha) & \text{if} \quad 0 < \alpha < 1, \\ O(1/k^\beta) + O(1/k) & \text{if} \quad \alpha = 1, \end{cases}$$

where $\beta = 2\gamma_0(\mu_g + \mu_f\varepsilon)/(1+\mu_g)^2$ for some fixed $0 < \varepsilon < 1$. Note that, if γ_0 is large enough, then $\beta > 1$.

Theorem 5. [Almost sure convergence of the iterates of SPGM [15]]
Suppose that:

1. $0 < \gamma_k \leq 1/L$ and $\sum_{k=0}^{\infty} \gamma_k = \infty$.
2. $\limsup_k \|\mathbf{x}^k\| < +\infty$ almost surely.
3. $\sum_{k\geq 0} \gamma_k^2 \, \mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|^2 | \mathcal{F}_k] < +\infty$.

Then,

$$\mathbf{x}^k \to \mathbf{x}^\star \text{ almost surely.}$$

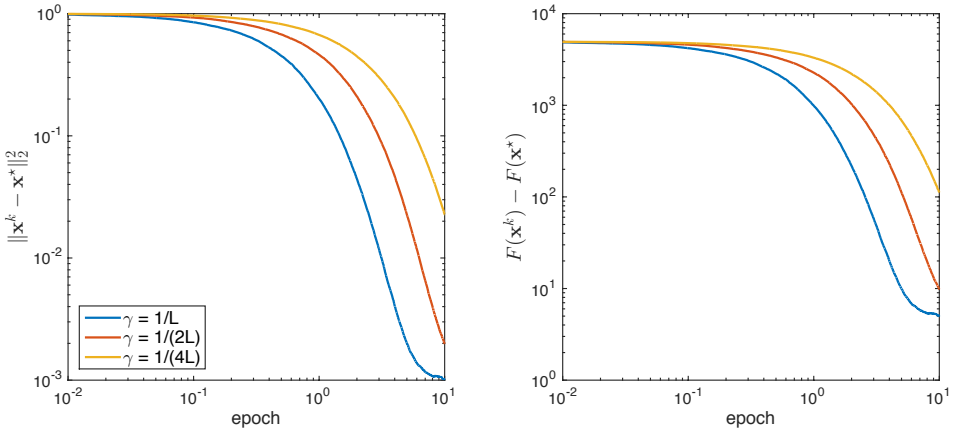
4.1 Numerical example: ℓ_1 regularized least squares problem (LASSO)

We consider the following simple problem formulation:

$$\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \rho\|\mathbf{x}\|_1 \right\}. \tag{7}$$

We consider a synthetic problem setup, where each entry of the measurement matrix $\mathbf{A}$ follows the standard Gaussian distribution $\mathcal{N}(0,1)$. Problem dimensions n and p are 10^4 and 10^2 respectively. We generate a 10-sparse true signal $\mathbf{x}^\natural$, where the non-zero entries are randomly chosen with Gaussian distribution with 0 mean, and the signal is normalized to $\|\mathbf{x}^\natural\|_1 = 1$. We generate noisy measurements $\mathbf{b} := \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is Gaussian white noise and the SNR is $30dB$. Regularization constant *rho* is chosen as 10^{-2}.

Figure 4.1 presents empirical performance of SPGM in solving problem (7), with different initial stepsize choices. As in the case for SG, robustness of SPGM against the change in stepsize parameter can be improved by adding a simple averaging step. See [15] for the details and convergence properties of this variant of SPGM.



5 Accelerated stochastic proximal gradient methods

In the same way as the proximal gradient methods in the batch setting, stochastic proximal gradient methods can be accelerated adding a so-called momentum term. In this lecture, we consider two different accelerated variants of SPGM and provide the corresponding convergence guarantees. Per-iteration computational complexity of both of these two variants are essentially same as per iteration complexity of SPGM.

We first consider the accelerated stochastic proximal gradient method I (ASPGM I), pseudocode of which is given below.

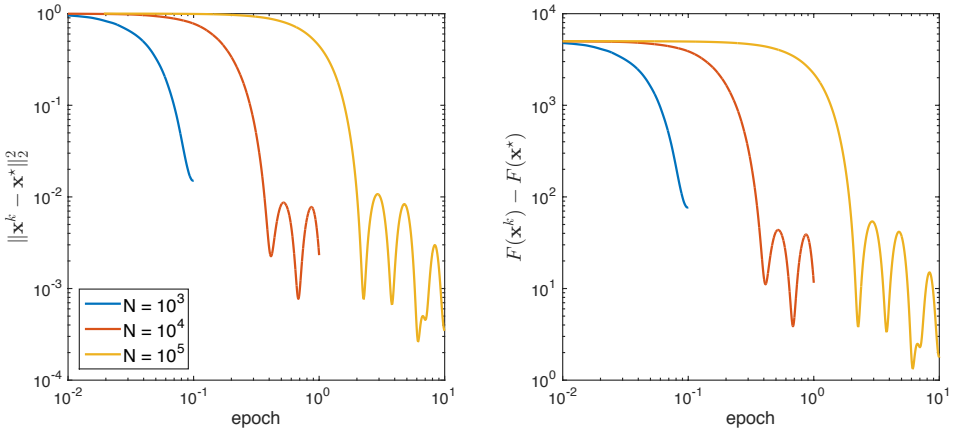
Accelerated stochastic proximal gradient method I (ASPGM I)

1. Choose $\mathbf{x}_0 = \mathbf{z}_0 = \mathbf{0}$. Define $\alpha_k := 2/(k+2)$ and $\gamma_k := \alpha_k(2 + N^{3/2}/L)$
2. For $k = 0, 1, \dots N$ perform:
2a. $\mathbf{y}^k = (1-\alpha_k)\mathbf{x}^k + \alpha_k \mathbf{z}^k$
2b. $\mathbf{z}^{k+1} = \text{prox}_{\frac{1}{\gamma_k L} g}(\mathbf{z}^k - \frac{1}{\gamma_k L} G(\mathbf{y}^k, \theta_k))$
2c. $\mathbf{x}^{k+1} = (1-\alpha_k)\mathbf{x}^k + \alpha_k \mathbf{z}^{k+1}$.

Theorem 6. [Mean convergence of ASPGM I [13]]
Suppose that $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|] \leq M^2$ for some $M \in]0, \infty[$. Then,

$$\mathbb{E}[[F(\mathbf{x}^{N+1}) - F(\mathbf{x}^\star)] \leq \frac{2\|\mathbf{x}^\star - \mathbf{z}_0\|^2 + M^2}{\sqrt{N+2}} + L\frac{4\|\mathbf{x}^\star - \mathbf{z}_0\|^2 + 2M^2}{N+2}.$$

While ASPGM I has optimal convergence rate in the absence of strong convexity assumption, $O(\frac{1}{\sqrt{k}})$, it requires the number of iterations N to be fixed in advance. This highly limits the practical applicability of this variant as there is no way to deduce the number of iterations required to solve the problem in advance in most of the practical applications. Figure 5 represents the empirical performance of this variant in solving the LASSO formulation (7) with the same experimental setup considered in section 4.1. Three different plots correspond to three different choices of N. While underestimation of N may cause your algorithm to halt before convergence, overestimation of N results in a much slower convergence.



Finally, we present the second accelerated variant, which we call as accelerated stochastic proximal gradient method II (ASPGM II). ASPGM II tackles to the problem of choosing N in advance of ASPGM I.

Accelerated stochastic proximal gradient method II (ASPGM II)

1. Choose $\mathbf{y}^0 = \mathbf{z}^0 = \mathbf{0}$, $(\gamma_k)_{k\in\mathbb{N}}$, $(\alpha_k)_{k\in\mathbb{N}} \in]0, +\infty[^{\mathbb{N}}$, $\alpha_0 = 1, \gamma_0 = L + \mu$.
2. For $k = 0, 1, \ldots$ perform:
2a. $\mathbf{x}^{k+1} = (1 - \alpha_k)\mathbf{y}^k + \alpha_k \mathbf{z}^k$.
2b. $\mathbf{y}^{k+1} = \text{prox}_{\frac{1}{\gamma_k} g}(\mathbf{x}^{k+1} - \frac{1}{\gamma_k} G(\mathbf{x}^{k+1}, \theta_k))$.
2c. $\mathbf{z}^{k+1} = \mathbf{z}^k - \frac{1}{\gamma_k \alpha_k + \mu}\left(\gamma_k(\mathbf{x}^{k+1} - \mathbf{y}^{k+1}) + \mu(\mathbf{z}^k - \mathbf{x}^{k+1})\right)$.

Theorem 7. [Mean convergence of ASPGM II [14]]
Suppose that:

1. $\mathbb{E}[\|\mathbf{z}^k - \mathbf{x}^\star\|^2] \leq D^2$ for some $D \in]0, \infty[$.
2. $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|^2] \leq M^2$ for some $M \in]0, \infty[$.
3. $\gamma_k = c(k+1)^{3/2} + L$ for a fixed $c > 0$, and $\alpha_k = 2/(k+2)$.

Then,

$$\mathbb{E}[F(\mathbf{y}^{k+1}) - F(\mathbf{x}^\star)] \leq \frac{3D^2 L}{k^2} + \left(3D^2 c + \frac{5M^2}{3c}\right)\frac{1}{\sqrt{k}}.$$

Note that this rate is optimal without strong convexity assumption.

Theorem 8. [Mean convergence of ASPGM II with strong convexity [14]]
Define $\lambda_k = \prod_{j=1}^{k}(1 - \alpha_j)$ and $\lambda_0 = 1$. Suppose that:

1. f is μ-strongly convex.
2. $\mathbb{E}[\|\mathbf{z}^k - \mathbf{x}^\star\|^2] \leq D^2$ for some $D \in]0, \infty[$.
3. $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k) - \nabla f(\mathbf{x}^k)\|^2] \leq M^2$ for some $M \in]0, \infty[$.
4. $\gamma_k = L + \frac{\mu}{\lambda_{k-1}}$ and $\alpha_k = \sqrt{\lambda_{k-1} + \frac{\lambda_{k-1}^2}{4}} - \frac{\lambda_{k-1}}{2}$.

Then,

$$\mathbb{E}[F(\mathbf{y}^{k+1}) - F(\mathbf{x}^\star)] \leq \frac{2(L+\mu)D^2}{k^2} + \frac{6M^2}{\mu k}.$$

Note that this rate is optimal under strong convexity assumption.

References

[1] H. Robbins and S. Monro. A stochastic approximation method.. *Ann. Math. Statist.*, 22(3):400–407, 1951.

[2] R. Durrett. Probability: Theory and examples.. *Cambridge university press*, 2010.

[3] M. Ledoux and M. Talagrand. Probability in Banach spaces: Isoperimetry and processes. *Springer, New York.*, 1991.

[4] V. Vapnik. An overview of statistical learning theory. *IEEE Trans. Inf. Theory*, 10(5):988–999, 1999.

[5] D. Davis and W. Yin. A three-Operator splitting scheme and its optimization applications. *arXiv:1504.01032v1, 2015*.

[6] V. Cevher, B. C. Vũ, and A. Yurtsever. Stochastic forward Douglas-Rachford splitting for monotone inclusions. *infoscience.epfl.ch/record/215759*.

[7] J. Brodie, I. Daubechies, C. de Mol, D. Giannone, and I. Loris. Sparse and stable Markowitz portfolios. *Proc. Natl. Acad. Sci.*, 106:12267-12272, 2009.

[8] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM J. Optim.*, vol. 19, pp. 1574–1609, 2008.

Machine Learning (ICML), 2008

[9] J. Duchi, S. S-Schwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. In *Proc. of the Int. Conf. on Machine Learning (ICML)*, 2008.

[10] P. Combettes and Pesquet J.-C. Signal recovery by proximal forward-backward splitting. In *Fixed-Point Algorithms for Inverse Problems in Science and Engineering*, pages 185–212. Springer-Verlag, 2011.

[11] N. Parikh and S. Boyd. Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):123–231, 2013.

[12] L. Rosasco, S. Villa, and B. C. Vũ. Convergence of stochastic proximal gradient algorithm. *http://arxiv.org/abs/1403.5074*.

[13] Q. Lin, X. Chen and J. Peña. A smoothing stochastic gradient method for composite optimization. *Optimization Methods and Software*, vol. 29, pp. 1281–1301, 2014.

[14] J. T. Kwok, C. Hu and W. Pan. Accelerated gradient methods for stochastic optimization and online learning. *Advances in Neural Information Processing Systems*, vol. 22, pp. 781–789, 2009.

[15] Y. F. Atchade, G. Fort and E. Moulines. On stochastic proximal gradient algorithms. *arXiv:1402.2365, 2014*.