Mathematics of Data: From Theory to Computation

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 6: Stochastic gradient methods

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-556 (Fall 2019)

License Information for Mathematics of Data Slides

Outline

- This class
    1. Stochastic programming
    2. Stochastic gradient descent
    3. Variance reduction technique
- Next class
    1. Non-convex optimization

Recommended reading materials

1. V. Cevher; S. Becker, and M. Schmidt. Convex optimization for big data. *IEEE Signal Process. Mag.*, vol. 31, pp. 32–43, 2014.
2. A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming.
3. L. Bottou., F. E. Curtis and J. Nocedal. Optimization methods for large-scale machine learning. *arXiv:1606.04838,* 2016 Jun 15.

Recall: Gradient descent

Problem (Unconstrained convex problem)

Consider the following convex minimization problem:

$$f^\star = \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x})$$

$f(\mathbf{x})$ *is proper, closed, and convex (perhaps strongly-convex and/or smooth).*

Gradient descent

Choose a starting point $\mathbf{x}^0$ and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla f(\mathbf{x}^k)$$

where γ_k is a step-size to be chosen so that $\mathbf{x}^k$ converges to $\mathbf{x}^\star$.

Recall: Gradient descent

Problem (Unconstrained convex problem)

Consider the following convex minimization problem:

$$f^\star = \min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x})$$

$f(\mathbf{x})$ *is proper, closed, and convex (perhaps strongly-convex and/or smooth).*

Gradient descent

Choose a starting point $\mathbf{x}^0$ and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla f(\mathbf{x}^k)$$

where γ_k is a step-size to be chosen so that $\mathbf{x}^k$ converges to $\mathbf{x}^\star$.

	f is L-smooth & **convex**	f is L-smooth & **non-convex**
GD	$O(1/T)$ (fast)	$O(1/T)$ (optimal)
AGD	$O(1/T^2)$ (optimal)	$O(1/T)$ (optimal) [6]

Why should we study anything else?

Statistical learning

A basic statistical learning model [16]

A statistical learning model consists of the following three elements.

1. A sample of i.i.d. random variables $(\mathbf{a}_j, b_j) \in \mathcal{A} \times \mathcal{B}$, $j = 1, \ldots, n$, 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 $L : \mathcal{B} \times \mathcal{B} \to \mathbb{R}$.

Statistical learning

A basic statistical learning model [16]

A statistical learning model consists of the following three elements.

1. A sample of i.i.d. random variables $(\mathbf{a}_j, b_j) \in \mathcal{A} \times \mathcal{B}$, $j = 1, \ldots, n$, 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 $L : \mathcal{B} \times \mathcal{B} \to \mathbb{R}$.

Definition (Risk)

Let $(\mathbf{a}, b)$ follow the probability distribution $\mathbb{P}$ and be independent of $\{(\mathbf{a}_i, b_i)\}_{i=1}^n$. 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

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

Many problems in machine learning cast into this formulation

Empirical risk minimization (ERM) I

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

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

when n is large enough, with high probability.

Statistical learning with Empirical risk minimization (ERM) [16]

We approximate $f^\star$ by minimizing the *empirical average of the loss* instead of the risk.

$$\arg\min_{f \in \mathcal{F}} \left\{ R_n(f) := \frac{1}{n}\sum_{j=1}^{n} L(f(\mathbf{a}_j), b_j) \right\}.$$

Example: Least squares

Recall that the LS estimator is given by

$$\arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ \frac{1}{2n} \|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 \right\} = \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ \frac{1}{2n} \sum_{j=1}^{n} (b_j - \langle \mathbf{a}_j, \mathbf{x}\rangle)^2 \right\},$$

where we define $\mathbf{b} := (b_1, \ldots, b_n)^T$ and $\mathbf{a}_j^T$ to be the j-th row of $\mathbf{A}$.

Empirical risk minimization (ERM) II

Example: Logistic regression

Recall the logistic regression formulation

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

where $\mathbf{b} := (b_1, \ldots, b_n)^T \in \{-1, 1\}^n$.

Gradient descent for ERM

$$f^{k+1} = f^k - \gamma_k \nabla R_n(f) = f^k - \gamma_k \frac{1}{n} \sum_{j=1}^{n} \nabla L(f(\mathbf{a}_j), b_j).$$

Computational cost per iteration is proportional to sample size n,which is expensive when n is large.

Statistical learning with streaming data

Recall that statistical learning seeks to find a $f^\star \in \mathcal{F}$ that minimizes the *expected* risk,

$$f^\star \in \arg\min_{f \in \mathcal{F}} \left\{ R(f) := \mathbb{E}_{(\mathbf{a},b)} \left[L(f(\mathbf{a}), b) \right] \right\}, \qquad .$$

In practice, data can arrive in a *streaming* way.

Example: Markowitz portfolio optimization

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

- $\rho \in \mathbb{R}$ is the desired return.
- $\mathcal{X}$ is intersection of the standard simplex and the constraint: $\langle \mathbf{x}, \mathbb{E}[\theta_t] \rangle \geq \rho$.

Gradient method

$$f^{k+1} = f^k - \gamma_k \nabla R(f) = f^k - \gamma_k \mathbb{E}_{(\mathbf{a},b)}[\nabla L(f^k(\mathbf{a}), b)].$$

This can not be implemented in practice as the distribution of (a, b) is unknown.

Stochastic programming

Problem (Mathematical formulation)

Consider the following convex minimization problem:

$$f^\star = \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \mathbb{E}[f(\mathbf{x},\theta)] \right\}$$

- θ *is a random vector whose probability distribution is supported on set* Θ.
- $f(\mathbf{x}) := \mathbb{E}[f(\mathbf{x},\theta)]$ *is proper, closed, and convex.*
- *The solution set* $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}\,(f) : f(\mathbf{x}^\star) = f^\star\}$ *is nonempty.*

Stochastic gradient descent (SGD)

Stochastic gradient descent (SGD)

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:

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

- $G(\mathbf{x}^k, \theta_k)$ is an unbiased estimate of the full gradient:

$$\mathbb{E}[G(\mathbf{x}^k, \theta_k)] = \nabla f(\mathbf{x}^k).$$

Stochastic gradient descent (SGD)

Stochastic gradient descent (SGD)

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:

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

- $G(\mathbf{x}^k, \theta_k)$ is an unbiased estimate of the full gradient:

$$\mathbb{E}[G(\mathbf{x}^k, \theta_k)] = \nabla f(\mathbf{x}^k).$$

Remark

- The cost of computing $G(\mathbf{x}^k, \theta_k)$ is n times cheaper than that of $\nabla f(\mathbf{x}^k)$.
- As $G(\mathbf{x}^k, \theta_k)$ is an unbiased estimate of the full gradient, SG would perform well.
- We assume $\{\theta_k\}$ are jointly independent.
- SG is not a monotonic descent method.

Example: Convex optimization with finite sums

Convex optimization with finite sums

The problem

$$\arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x}) \right\},$$

can be rewritten as

$$\arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \mathbb{E}_i[f_i(\mathbf{x})] \right\}, \qquad i \text{ is uniformly distributed over } \{1, 2, \cdots, n\}.$$

Stochastic gradient descent (SGD)

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla f_i(\mathbf{x}^k) \qquad i \text{ is uniformly distributed over} \{1, ..., n\}$$

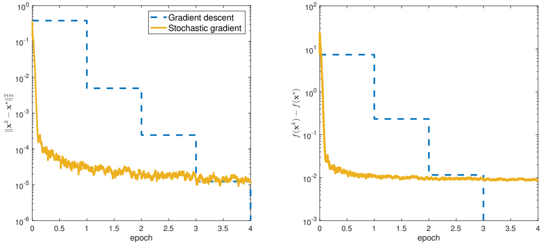
- Note: $\mathbb{E}_i[\nabla f_i(\mathbf{x}^k)] = \sum_{j=1}^{n} \nabla f_j(\mathbf{x}^k)/n = \nabla f(\mathbf{x}^k)$.
- The computational cost of SGD per iteration is p.

Synthetic least-squares problem

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

Setup

- $\mathbf{A} := \mathrm{randn}(n,p)$ - standard Gaussian $\mathcal{N}(0,\mathbb{I})$, with $n = 10^4$, $p = 10^2$.
- $\mathbf{x}^\natural$ is 50 sparse with zero mean Gaussian i.i.d. entries, normalized to $\|\mathbf{x}^\natural\|_2 = 1$.
- $\mathbf{b} := \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w}$ is Gaussian white noise with variance 1.



- 1 epoch $= 1$ pass over the full gradient

Convergence of SGD without strong convexity

Theorem (decaying step-size [14])

Assume

- $\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|^2] \leq D^2$ *for all* k,
- $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2] \leq M^2$, *(bounded gradient)*
- $\gamma_k = \gamma_0 / \sqrt{k}$

Then

$$\mathbb{E}[f(\mathbf{x}^k) - f(\mathbf{x}^\star)] \leq \left(\frac{D^2}{\gamma_0} + \gamma_0 M^2\right) \frac{2 + \log k}{\sqrt{k}}.$$

- $\mathcal{O}(1/\sqrt{k})$ rate is optimal for SG if we do not consider the strong convexity.

Convergence of SGD for strongly convex problems I

Theorem (strongly convex objective, fixed step-size [2])

Assume

- *f is μ-strongly convex and L-smooth,*
- $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2]_2 \leq \sigma^2 + M\|\nabla f(\mathbf{x}^k)\|_2^2$ *(Bounded variance),*
- $\gamma_k = \gamma \leq \frac{1}{LM}$.

Then

$$\mathbb{E}[f(\mathbf{x}^k) - f(\mathbf{x}^\star)] \leq \frac{\gamma L \sigma^2}{2\mu} + (1 - \mu\gamma)^{k-1} \left(f(\mathbf{x}^1) - f^\star\right).$$

- Converge fast (linearly) to a neighborhood around $\mathbf{x}^\star$
- Zero variance ($\sigma = 0$) $\Longrightarrow$ linear convergence
- Smaller step-sizes γ $\Longrightarrow$ converge to a better point, but with a slower rate

Convergence of SGD for strongly convex problems II

Theorem (strongly convex objective, decaying step-size [2])

Assume

- *f is μ-strongly convex and L-smooth,*
- $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2]_2 \leq \sigma^2 + M\|\nabla f(\mathbf{x}^k)\|_2^2$ *(bounded variance),*
- $\gamma_k = \frac{c}{k_0+k}$ *with some appropriate constants c and k_0.*

Then

$$\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|^2] \leq \frac{C}{k+1},$$

where C is a constant independent of k.

- Using the smooth property,

$$\mathbb{E}[f(\mathbf{x}^k) - f(\mathbf{x}^\star)] \leq L\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|^2] \leq \frac{C}{k+1}.$$

- The rate is optimal if $\sigma^2 > 0$ with the assumption of strongly-convexity.

*Randomized Kaczmarz algorithm

Problem

Given a full-column-rank matrix $\mathbf{A} \in \mathbb{R}^{n \times p}$ and $b \in \mathbb{R}^n$, solve the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}.$$

Notations: $\mathbf{b} := (b_1, \ldots, b_n)^T$ and $\mathbf{a}_j^T$ is the j-th row of $\mathbf{A}$.

Randomized Kaczmarz algorithm (RKA)

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$.
2. For $k = 0, 1, \ldots$ perform:
2a. Pick $j_k \in \{1, \cdots, n\}$ randomly with $\Pr(j_k = i) = \|\mathbf{a}_i\|_2^2 / \|\mathbf{A}\|_F^2$
2b. $\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\langle \mathbf{a}_{j_k}, \mathbf{x}^k \rangle - b_{j_k}\right) \mathbf{a}_{j_k} / \|\mathbf{a}_{j_k}\|_2^2$.

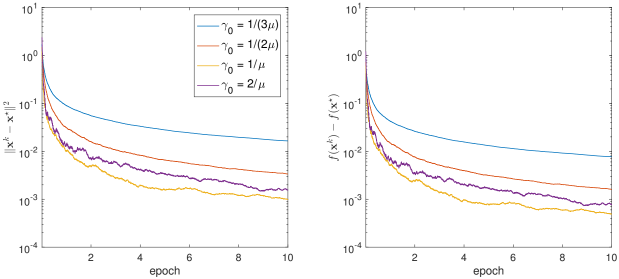
Linear convergence [15]

Let $\mathbf{x}^\star$ be the solution of $\mathbf{A}\mathbf{x} = \mathbf{b}$ and $\kappa = \|\mathbf{A}\|_F \|\mathbf{A}^{-1}\|$. Then

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|_2^2 \leq (1 - \kappa^{-2})^k \|\mathbf{x}^0 - \mathbf{x}^*\|_2^2$$

- RKA can be seen as a particular case of SGD [10].

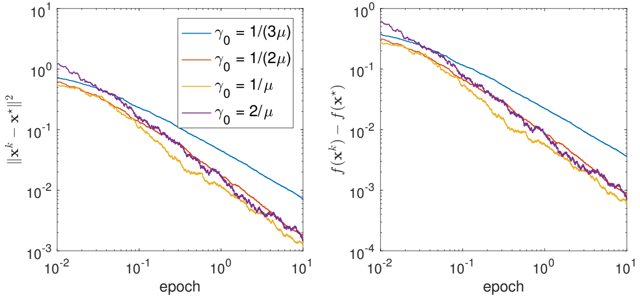
Example: SGD with different step sizes



Setup

- Synthetic least-squares problem as before
- $\gamma_k = \gamma_0/(k + k_0)$.

Example: SGD with different step sizes



Setup

- Synthetic least-squares problem as before
- $\gamma_k = \gamma_0/(k + k_0)$.

$\gamma_0 = 1/\mu$ *is the best choice.*

Comparison with GD

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x}) \right\}.$$

- f: μ-strongly convex with L-Lipschitz smooth.

	rate	iteration complexity	cost per iteration	total cost
GD	ρ^k	$\log(1/\epsilon)$	n	$n\log(1/\epsilon)$
SGD	$1/k$	$1/\epsilon$	1	$1/\epsilon$

- SGD is more favorable when n is large — large-scale optimization problems

Motivation for SGD with Averaging

- SGD iterates tend to oscillate around global minimizers
- Averaging iterates can reduce the oscillation effect
- Two types of averaging:

$$\bar{\mathbf{x}}^k = \frac{1}{k}\sum_{j=1}^{k} \gamma_j \mathbf{x}^j \quad \text{(vanilla averaging)}$$

$$\bar{\mathbf{x}}^k = \frac{\sum_{j=1}^{k} \gamma_j \mathbf{x}^j}{\sum_{j=1}^{k} \gamma_j} \quad \text{(weighted averaing)}$$

Convergence for SG-A I: strongly convex case

Stochastic gradient method with averaging (SG-A)

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:

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

2b. $\bar{\mathbf{x}}^k = \frac{1}{k}\sum_{j=1}^k \mathbf{x}^j$.

Theorem (Convergence of SG-A [13])

Assume

- *f is μ-strongly convex,*
- $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2] \leq M^2$,
- *$\gamma_k = \gamma_0/k$ for some $\gamma_0 \geq 1/\mu$.*

Then

$$\mathbb{E}[f(\bar{\mathbf{x}}^k) - f(\mathbf{x}^\star)] \leq \frac{\gamma_0 M^2(1 + \log k)}{2k}.$$

- Same convergence rate with vanilla SGD.

Convergence for SG-A II: non-strongly convex case

Stochastic gradient method with averaging (SG-A)

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:

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

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

Theorem (Convergence of SG-A [11])

Let $D = \|\mathbf{x}^0 - \mathbf{x}^\star\|$ *and* $\mathbb{E}[\|G(\mathbf{x}^k, \theta_k)\|^2] \leq M^2$.
Then,

$$\mathbb{E}[f(\bar{\mathbf{x}}^{k+1}) - f(\mathbf{x}^\star)] \leq \frac{D^2 + M^2 \sum_{j=0}^k \gamma_j^2}{2\sum_{j=0}^k \gamma_j}.$$

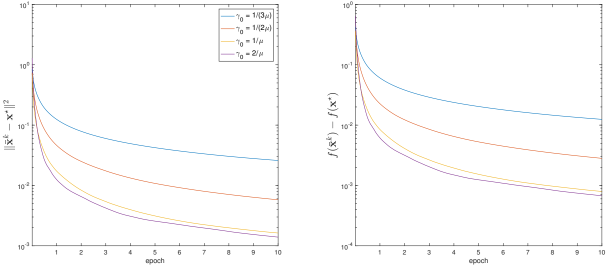
In addition, choosing $\gamma_k = D/(M\sqrt{k+1})$*, we get,*

$$\mathbb{E}[f(\bar{\mathbf{x}}^k) - f(\mathbf{x}^\star)] \leq \frac{MD(2 + \log k)}{\sqrt{k}}.$$

- Same convergence rate with vanilla SGD.

Example: SG-A method with different step sizes

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

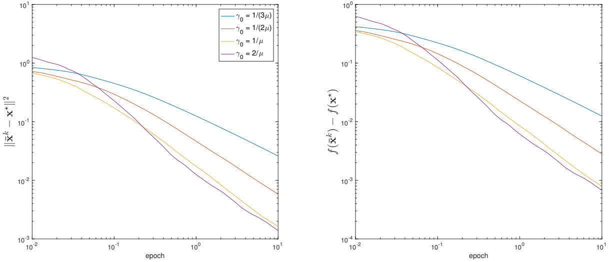


Setup

- Synthetic least-squares problem as before
- $\gamma_k = \gamma_0/(k + k_0)$.

Example: SG-A method with different step sizes

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



Setup

- Synthetic least-squares problem as before
- $\gamma_k = \gamma_0/(k + k_0)$.

SG-A is more stable than SG.
$\gamma_0 = 2/\mu$ is the best choice.

Least mean squares algorithm

Least-square regression problem

Solve

$$\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{2}\mathbb{E}_{(\mathbf{a},b)}(\langle \mathbf{a}, \mathbf{x}\rangle - b)^2 \right\},$$

given i.i.d. samples $\{(\mathbf{a}_j, b_j)\}_{j=1}^n$ (particularly in a streaming way).

Stochastic gradient method with averaging

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ and $\gamma > 0$.
2a. For $k = 1, \ldots, n$ perform:

$$\mathbf{x}^k = \mathbf{x}^{k-1} - \gamma\left(\langle \mathbf{a}_k, \mathbf{x}^{k-1}\rangle - b_k\right)\mathbf{a}_k.$$

2b. $\bar{\mathbf{x}}^k = \frac{1}{k+1}\sum_{j=0}^k \mathbf{x}^j$.

$O(1/n)$ convergence rate, without strongly convexity [1]

Let $\|\mathbf{a}_j\|_2 \le R$ and $|\langle \mathbf{a}_j, \mathbf{x}^\star\rangle - b_j| \le \sigma$ a.s.. Pick $\gamma = 1/(4R^2)$. Then

$$\mathbb{E}f(\bar{\mathbf{x}}^{n-1}) - f^* \le \frac{2}{n}\left(\sigma\sqrt{p} + R\|\mathbf{x}^0 - \mathbf{x}^\star\|_2\right)^2.$$

Popular SGD Variants

- Mini-batch SGD: For each iteration,

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \frac{1}{b} \sum_{\theta \in \Gamma} G(\mathbf{x}^k, \theta).$$

  - γ_k: step-size
  - b : mini-batch size
  - Γ : a set of random variables θ of size b

- Accelerated SGD (Nesterov accelerated technique)

- SGD with Momentum

- Adaptive stochastic methods: AdaGrad...

Adaptive methods for stochastic optimization

Remark

- Adaptive methods have extensive applications in stochastic optimization.
- We will see **another nature** of adaptive methods in this lecture.
- Mild additional assumption: **bounded variance** of gradient estimates.

AdaGrad for stochastic optimization

- Only modification: $\nabla f(\mathbf{x}) \Rightarrow G(\mathbf{x}, \theta)$

AdaGrad with $H_k = \lambda_k I$ [8]
1. Set Q_0 =0.   **2.** For $k = 0, 1, \ldots, T$, iterate $\begin{cases} Q^k & = Q^{k-1} + \|G(x^k, \theta)\|^2 \\ H_k & = \sqrt{Q_t} I \\ x_{k+1} & = x_t - \alpha_k H_k^{-1} G(x^k, \theta) \end{cases}$

Theorem (Convergence rate: stochastic, convex optimization [8])

Assume f is convex and L-smooth, such that minimizer of f lies in a convex, compact set $\mathcal{K}$ with diameter D. Also consider bounded variance for unbiased gradient estimates, i.e., $\mathbb{E}\left[\|G(x,\theta) - \nabla f(x)\|^2 | x\right] \leq \sigma^2$. Then,

$$\mathbb{E}[f(x)] - \min_{x \in \mathbb{R}^d} f(x) = O\left(\frac{\sigma D}{\sqrt{T}}\right)$$

- AdaGrad is **adaptive** also in the sense that it adapt to nature of the oracle.

AcceleGrad for stochastic optimization

- Similar to AdaGrad, replace $\nabla f(\mathbf{x}) \Rightarrow G(\mathbf{x}, \theta)$

AcceleGrad (Accelerated Adaptive Gradient Method)
Input : Number of iterations T, $x_0 \in \mathcal{K}$, diameter D, weights $\{\alpha_t\}_{t\in[T]}$, learning rate $\{\eta_t\}_{t\in[T]}$
1. Set $y_0 = z_0 = x_0$ **2.** For $k = 0, 1, \dots, T$, iterate $\begin{cases} \tau_t & := 1/\alpha_t \\ x_{t+1} & = \tau_t z_t + (1-\tau_t) y_t, \text{define } g_t := \nabla f(x_{t+1}) \\ z_{t+1} & = \Pi_{\mathcal{K}}(z_t - \alpha_t \eta_t g_t) \\ y_{t+1} & = x_{t+1} - \eta_t g_t \end{cases}$
Output : $\overline{y}_T \propto \sum_{t=0}^{T-1} \alpha_t y_{t+1}$

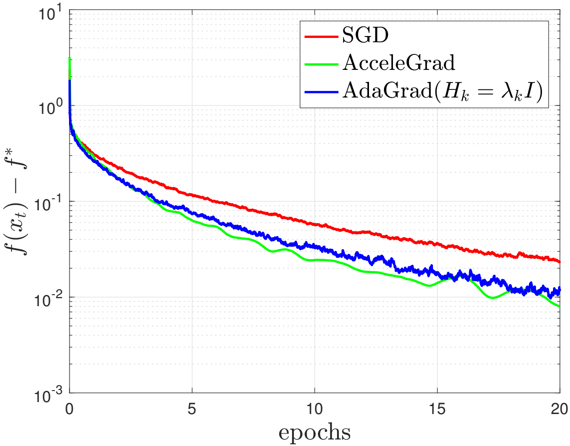
Theorem (Convergence rate [9])

Assume f is convex and G-Lipschitz and that minimizer of f lies in a convex, compact set $\mathcal{K}$ with diameter D. Also consider bounded variance for unbiased gradient estimates, i.e., $\mathbb{E}\left[\|G(x,\theta) - \nabla f(x)\|^2 | x\right] \leq \sigma^2$. Then,

$$\mathbb{E}[f(\overline{y}_T)] - \min_x f(x) = O\left(\frac{GD\sqrt{\log T}}{\sqrt{T}}\right).$$

Example: Synthetic least squares

- $A \in \mathbb{R}^{n \times d}$, where $n = 200$ and $d = 50$.
- Number of epochs: 20.
- Algorithms: SGD, AdaGrad & AcceleGrad.



Convex optimization with finite sums

Problem (Convex optimization with finite sums)

We consider the following simple example in the next few slides:

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x}) \right\}$$

- f_j *is proper, closed, and convex.*
- ∇f_j *is L_j-Lipschitz continuous for $j = 1, \ldots, n$.*
- *The solution set $\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathrm{dom}(f) : f(\mathbf{x}^\star) = f^\star\}$ is nonempty.*

- One prevalent choice is given by

$$G(\mathbf{x}^k, i_k) = \nabla f_{i_k}(\mathbf{x}^k), \qquad i_k \text{ is uniformly distributed over } \{1, 2, \cdots, n\}$$

An observation of SGD step

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k \nabla f(\mathbf{x}^k) \quad \text{(GD)}$$

Lemma

Assume f is Lipschitz smooth with constant L. Then,

$$f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k) \leq (\gamma_k^2 L - \gamma_k)\|\nabla f(\mathbf{x}^k)\|^2.$$

An observation of SGD step

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k G(\mathbf{x}^k, i_k) \quad \text{(SGD)}$$

Lemma

Assume f is Lipschitz smooth with constant L. Then,

$$\mathbb{E}[f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k)] \leq (\gamma_k^2 L - \gamma_k)\mathbb{E}[\|\nabla f(\mathbf{x}^k)\|^2] + L\gamma_k^2 \mathbb{E}[\|G(\mathbf{x}^k, i_k) - \nabla f(\mathbf{x}^k)\|^2]$$

An observation of SGD step

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma_k G(\mathbf{x}^k, i_k) \quad \text{(SGD)}$$

Lemma

Assume f is Lipschitz smooth with constant L. Then,

$$\mathbb{E}[f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k)] \leq (\gamma_k^2 L - \gamma_k)\mathbb{E}[\|\nabla f(\mathbf{x}^k)\|^2] + L\gamma_k^2\mathbb{E}[\|G(\mathbf{x}^k, i_k) - \nabla f(\mathbf{x}^k)\|^2]$$

- The variance in gradient dominates later (as if $\nabla f(\mathbf{x}^k) \to 0$).
- To ensure convergence, $\gamma_k \to 0$. $\Longrightarrow$ Slow convergence!

Can we decrease the variance while using a constant step-size?

- Choose a stochastic gradient, s.t. $\mathbb{E}\left[\|G(\mathbf{x}^k; i_k)\|^2\right] \to 0$.

Variance reduction techniques: SVRG

- Select the stochastic gradient ∇f_{i_k}, and compute a gradient estimate

$$\mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}) + \nabla f(\tilde{\mathbf{x}}),$$

where $\tilde{\mathbf{x}}$ is a good approximation of $\mathbf{x}^\star$.

- As $\tilde{\mathbf{x}} \to \mathbf{x}^\star$ and $\mathbf{x}^k \to \mathbf{x}^\star$,

$$\nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}) + \nabla f(\tilde{\mathbf{x}}) \to 0.$$

- Therefore,

$$\mathbb{E}\left[\|\nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}) + \nabla f(\tilde{\mathbf{x}})\|^2\right] \to 0.$$

Stochastic gradient algorithm with variance reduction

Stochastic gradient with variance reduction (SVRG) [7, 18]

1. Choose $\widetilde{\mathbf{x}}^0 \in \mathbb{R}^p$ as a starting point and $\gamma > 0$ and $q \in \mathbb{N}_+$.
2. For $s = 0, 1, 2 \cdots$, perform:
   - **2a**. $\widetilde{\mathbf{x}} = \widetilde{\mathbf{x}}^s, \quad \widetilde{\mathbf{v}} = \nabla f(\widetilde{\mathbf{x}}), \quad \mathbf{x}^0 = \widetilde{\mathbf{x}}$.
   - **2b**. For $k = 0, 1, \cdots q-1$, perform:

$$\begin{cases} \text{Pick } i_k \in \{1, \ldots, n\} \text{ uniformly at random} \\ \mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\widetilde{\mathbf{x}}) + \widetilde{\mathbf{v}} \\ \mathbf{x}^{k+1} := \mathbf{x}^k - \gamma \mathbf{r}_k, \end{cases} \tag{1}$$

   - **2c**. Update $\widetilde{\mathbf{x}}^{s+1} = \frac{1}{m} \sum_{j=0}^{q-1} \mathbf{x}^j$.

Common features

- The SVRG method uses a multistage scheme to reduce the variance of the stochastic gradient $\mathbf{r}_k$ where $\mathbf{x}^k$ and $\widetilde{\mathbf{x}}^s$ tend to $\mathbf{x}_\star$.
- Learning rate γ does not necessarily tend to 0.
- Each stage, SVRG uses $n + 2q$ component gradient evaluations: n for the full gradient at the beginning of each stage, and $2q$ for each of the q stochastic gradient steps.

Convergence analysis

Assumption A5.

(i) f is μ-strongly convex

(ii) The learning rate $0 < \gamma < 1/(4L_{\max})$, where $L_{\max} = \max_{1 \leq j \leq n} L_j$.

(iii) q is large enough such that

$$\kappa = \frac{1}{\mu\gamma(1 - 4\gamma L_{\max})q} + \frac{4\gamma L_{\max}(q+1)}{(1 - 4\gamma L_{\max})q} < 1.$$

Theorem

Assumptions:

- *The sequence $\{\widetilde{\mathbf{x}^s}\}_{k \geq 0}$ is generated by SVRG.*
- *Assumption A5 is satisfied.*

Conclusion: *Linear convergence is obtained:*

$$\mathbb{E}f(\widetilde{\mathbf{x}^s}) - f(\mathbf{x}^\star) \leq \kappa^s (f(\widetilde{\mathbf{x}^0}) - f(\mathbf{x}^\star)).$$

Choice of γ and q, and complexity

Chose γ and q such that $\kappa \in (0, 1)$:

For example

$$\gamma = 0.1/L_{\max}, q = 100(L_{\max}/\mu) \Longrightarrow \kappa \approx 5/6.$$

Complexity

$$\mathbb{E}f(\widetilde{\mathbf{x}^s}) - f(\mathbf{x}^\star) \leq \varepsilon, \quad \text{when } s \geq \log((f(\widetilde{\mathbf{x}^0}) - f(\mathbf{x}^\star))/\epsilon)/\log(\kappa^{-1})$$

Since at each stage needs $n + 2q$ component gradient evaluations, with $q = \mathcal{O}(L_{\max}/\mu)$, we get the overall complexity is

$$\mathcal{O}\Big((n + L_{\max}/\mu)\log(1/\epsilon)\Big).$$

*Variance reduction techniques: SAGA

Stochastic Average Gradient (SAGA) [4]
1a. Choose $\tilde{\mathbf{x}}_i^0 = \mathbf{x}^0 \in \mathbb{R}^p, \forall i,\ q \in \mathbb{N}_+$ and stepsize $\gamma > 0$.
1b. Store $\nabla f_i(\tilde{\mathbf{x}}_i^0)$ in a table data-structure with length n.
2. For $k = 0, 1 \ldots$ perform:
2a. pick $i_k \in \{1, \ldots, n\}$ uniformly at random
2b. Take $\tilde{\mathbf{x}}_{i_k}^{k+1} = \mathbf{x}^k$, store $\nabla f_{i_k}(\tilde{\mathbf{x}}_{i_k}^{k+1})$ in the table and leave other entries the same.
2c. $\mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}_{i_k}^k) + \frac{1}{n}\sum_{j=1}^n \nabla f_j(\tilde{\mathbf{x}}_j^k)$
3. $\mathbf{x}^{k+1} = \mathbf{x}^k - \gamma \mathbf{r}_k$

Recipe:

In each iteration:

- Store last gradient evaluated at each datapoint.
- Previous gradient for datapoint j is $\nabla f_j(\tilde{\mathbf{x}}_j^k)$.
- Perform SG-iterations with the following stochastic gradient

$$\mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}_{i_k}^k) + \frac{1}{n}\sum_{j=1}^n \nabla f_j(\tilde{\mathbf{x}}_j^k).$$

*Variance reduction techniques: SAGA

- Select the stochastic gradient $\mathbf{r}_k$ as

$$\mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}^k_{i_k}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\mathbf{x}}^k_j),$$

where, at each iteration, $\tilde{\mathbf{x}}$ is updated as $\tilde{\mathbf{x}}^k_{i_k} = \mathbf{x}^k$ and $\tilde{\mathbf{x}}^k_j$ stays the same for $j \neq i_k$.

- As $\tilde{\mathbf{x}}^k_j \to \mathbf{x}^\star$ and $\mathbf{x}^k \to \mathbf{x}^\star$,

$$\nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}^k_{i_k}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\mathbf{x}}^k_j) \to 0.$$

- Therefore,

$$\mathbb{E}\left[\|\nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}^k_{i_k}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\mathbf{x}}^k_j)\|^2\right] \to 0.$$

*Convergence of SAGA

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x}) \right\}.$$

Theorem (Convergence of SAGA [4])

Suppose that f *is* μ*-strongly convex and that the stepsize is* $\gamma = \frac{1}{2(\mu n + L)}$ *with*

$$\rho = 1 - \frac{\mu}{2(\mu n + L)} < 1,$$

$$C = \|\mathbf{x}^0 - \mathbf{x}^\star\|^2 + \frac{n}{\mu n + L}[f(\mathbf{x}^0) - \langle \nabla f(\mathbf{x}^\star), \mathbf{x}^0 - \mathbf{x}^\star\rangle - f(\mathbf{x}^\star)]$$

Then

$$\mathbb{E}[\|\mathbf{x}^k - \mathbf{x}^\star\|^2] \leq \rho^k C.$$

- Allows the constant step-size.
- Obtains linear rate convergence.

SVRG vs SAGA

- SVRG update:

$$\begin{cases} \mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\widetilde{\mathbf{x}}) + \nabla f(\tilde{\mathbf{x}}) \\ \mathbf{x}^{k+1} := \mathbf{x}^k - \gamma \mathbf{r}_k, \end{cases}$$

- SAGA update:

$$\begin{cases} \mathbf{r}_k = \nabla f_{i_k}(\mathbf{x}^k) - \nabla f_{i_k}(\tilde{\mathbf{x}}^k_{i_k}) + \frac{1}{n}\sum_{j=1}^{n} \nabla f_j(\tilde{\mathbf{x}}^k_j) \\ \mathbf{x}^{k+1} := \mathbf{x}^k - \gamma \mathbf{r}_k, \end{cases}$$

	SVRG	SAGA
Storage of gradients	no	yes
Epoch-base	yes	no
Parameters	stepsize & epoch lengths	stepsize
Gradient evaluations per step	at least 2	1

Table: Comparisons of SVRG and SAGA [4]

Taxonomy of algorithms

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) := \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x}) \right\}.$$

- $f(\mathbf{x}) = \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{x})$: μ-strongly convex with L-Lipschitz continuous gradient.

Gradient descent	SVRG/SAGA	SGM
Linear	Linear	Sublinear

Table: Rate of convergence.

- $\kappa = L/\mu$ and $s_0 = 8\sqrt{\kappa}n(\sqrt{2}\alpha(n-1) + 8\sqrt{\kappa})^{-1}$ for $0 < \alpha \leq 1/8$.

SVRG/SAGA	AccGrad	SGM
$\mathcal{O}((n+\kappa)\log(1/\varepsilon))$	$\mathcal{O}((n\kappa)\log(1/\varepsilon))$	$1/\epsilon$

Table: Complexity to obtain ε-solution.

Stochastic methods for non-convex problems

Remark (Convex optimization)

- Large scale convex optimization $\Rightarrow$ demands stochastic methods.
- SGD, AdaGrad & AcceleGrad are optimal for general convex functions.
- Adaptive methods can also adapt to **the stochasticity of the gradient oracle**.

Remark (Non-convex optimization)

- Large scale non-convex optimization $\Rightarrow$ demands stochastic methods.
- AdaGrad, ADAM, RMSProp are frequently used in neural network optimization (more on next lecture!)

SGD - Non-convex stochastic optimization

- SGD is not as well-studied for non-convex problems as for convex problems.
- There is a gap between SGD's practical performance and theoretical understanding.
- Recall SGD update rule:

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k G(\mathbf{x}^k, \theta)$$

Theorem (A well-known result for SGD & Non-convex problems [5])

Let f be a non-convex and L-smooth function. Set $\alpha_k = \min\left\{\frac{1}{L}, \frac{C}{\sigma\sqrt{T}}\right\}$, $\forall k = 1, ..., T$, where σ^2 is the variance of the gradients and $C > 0$ is constant. Then,

$$\mathbb{E}[\|\nabla f(\mathbf{x}^R)\|^2] = O\left(\frac{\sigma}{\sqrt{T}}\right),$$

where $\mathbb{P}(R = k) = \frac{2\alpha_k - L\alpha_k^2}{\sum_{k=1}^{T}(2\alpha_k - L\alpha_k^2)}$.

Non-convergence of ADAM and a new method: AmsGrad

- It has been shown that ADAM may not converge for *some* objective functions [12].
- An ADAM alternative is proposed that is proved to be convergent [12].

AmsGrad

Input. Step size $\{\alpha_k\}_{k=1}^T$, exponential decay rates $\{\beta_{1k}\}_{k=1}^T$, β_2

1. Set $m_0 = 0, v_0 = 0$ and $\hat{v}_0 = 0$
2. For $k = 1, 2, \ldots, T$, iterate

$$\left\{\begin{array}{ll} g_k & = G(x^k, \theta) \\ m_k & = \beta_{1k} m_{k-1} + (1 - \beta_{1k}) g_k \leftarrow \text{1st order estimate} \\ v_k & = \beta_2 v_{k-1} + (1 - \beta_2) g_k^2 \leftarrow \text{2nd order estimate} \\ \hat{v}_k & = \max\{\hat{v}_{k-1}, v_k\} \text{ and } \hat{V}_k = \mathrm{diag}(\hat{v}_k) \\ H_k & = \sqrt{\hat{v}_k} \\ x^{k+1} & = \Pi_{\mathcal{X}}^{\sqrt{\hat{V}_k}}(x^k - \alpha_k \hat{m}_k / H_k) \end{array}\right.$$

where $\Pi_{\mathcal{X}}^{A}(y) = \arg\min_{x \in \mathcal{X}} \langle (x - y), A(x - y) \rangle$ (weighted projection onto $\mathcal{X}$).

AdaGrad & AmsGrad for non-convex optimization

Theorem (AdaGrad convergence rate: stochastic, non-convex [17])

Assume f is non-convex and L-smooth, such that $\|\nabla f(x)\|^2 \leq G^2$ and $f^\star = \inf_x f(x) > \infty$. Also consider bounded variance for unbiased gradient estimates, i.e., $\mathbb{E}\left[\|G(x,\theta) - \nabla f(x)\|^2 | x\right] \leq \sigma^2$. Then with probability $1-\delta$,

$$\min_{k \in \{1,..,T-1\}} \|\nabla f(x^k)\|^2 = \tilde{O}\left(\frac{\sigma}{\delta^{3/2}\sqrt{T}}\right)$$

- **Note:** As $1-\delta \to 1$, the rate deteriorates by a factor of $\delta^{-3/2}$.

Theorem (AmsGrad convergence rate 1: stochastic, non-convex [3])

Let $g_k = G(x^k, \theta)$. Assume $|g_{1,i}| > c > 0$, $\forall i \in [d]$ and $\|g_k\| \leq G$. Consider a non-increasing sequence β_{1k} and $\beta_{1k} \leq \beta_1 \in [0,1)$. Set $\alpha_k = 1/\sqrt{t}$. Then,

$$\min_{t \in [T]} \mathbb{E}\left[\|\nabla f(x^k)\|^2\right] = O\left(\frac{\log T}{\sqrt{T}}\right).$$

AdaGrad & AmsGrad for non-convex optimization

Theorem (AdaGrad convergence rate: stochastic, non-convex [17])

Assume f is non-convex and L-smooth, such that $\|\nabla f(x)\|^2 \leq G^2$ and $f^\star = \inf_x f(x) > \infty$. Also consider bounded variance for unbiased gradient estimates, i.e., $\mathbb{E}\left[\|G(x,\theta) - \nabla f(x)\|^2 | x\right] \leq \sigma^2$. Then with probability $1-\delta$,

$$\min_{k\in\{1,..,T-1\}} \|\nabla f(x^k)\|^2 = \tilde{O}\left(\frac{\sigma}{\delta^{3/2}\sqrt{T}}\right)$$

- **Note:** As $1-\delta \to 1$, the rate deteriorates by a factor of $\delta^{-3/2}$.

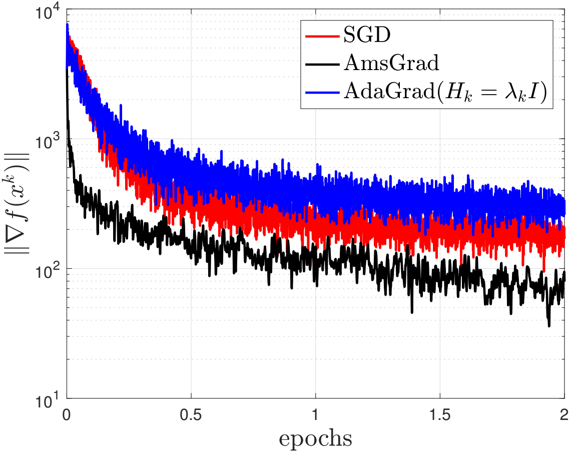
Theorem (AmsGrad convergence rate 2: stochastic, non-convex [19])

Consider $f : \mathbb{R}^d \to \mathbb{R}$ to be non-convex ans L-smooth. Assume $\|G(x,\theta)\|_\infty \leq G_\infty$ and set $\alpha_k = 1/\sqrt{dT}$. Also define $x_{out} = x^k$, for $k = 1, \ldots, T$ with probability $\alpha^k / \sum_{i=1}^T \alpha_i$. Then,

$$\mathbb{E}\left[\|\nabla f(x_{out})\|^2\right] = O\left(\sqrt{\frac{d}{T}}\right).$$

Example: Logistic regression with non-convex regularizer

- Synthetic data: $A \in \mathbb{R}^{n \times d}$, $n = 2000$, $d = 200$.
- Batch size: 20 samples.
- Algorithms: SGD, AdaGrad, AmsGrad.



References I

[1] Francis Bach and Eric Moulines.
Non-strongly-convex smooth stochastic approximation with convergence rate o(1/n).
In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'13, pages 773–781, USA, 2013. Curran Associates Inc.

[2] Léon Bottou, Frank E. Curtis, and Jorge Nocedal.
Optimization methods for large-scale machine learning, 2016.
quantization overview.

[3] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong.
On the convergence of a class of adam-type algorithms for non-convex optimization.
In *International Conference on Learning Representations*, 2019.

[4] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien.
Saga: A fast incremental gradient method with support for non-strongly convex composite objectives.
In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1646–1654. Curran Associates, Inc., 2014.

References II

[5] Saeed Ghadimi and Guanghui Lan.
Stochastic first-and zeroth-order methods for nonconvex stochastic programming.
SIAM Journal on Optimization, 23(4):2341–2368, 2013.

[6] Saeed Ghadimi and Guanghui Lan.
Accelerated gradient methods for nonconvex nonlinear and stochastic programming.
Math. Program., 156(1-2):59–99, 2016.

[7] Rie Johnson and Tong Zhang.
Accelerating stochastic gradient descent using predictive variance reduction.
In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 315–323. Curran Associates, Inc., 2013.

[8] Kfir Levy.
Online to offline conversions, universality and adaptive minibatch sizes.
In *Advances in Neural Information Processing Systems*, pages 1613–1622, 2017.

[9] Kfir Levy, Alp Yurtsever, and Volkan Cevher.
Online adaptive methods, universality and acceleration.
In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018.

References III

[10] Deanna Needell, Rachel Ward, and Nati Srebro.
Stochastic gradient descent, weighted sampling, and the randomized kaczmarz algorithm.
In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1017–1025. Curran Associates, Inc., 2014.

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

[12] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar.
On the convergence of adam and beyond.
In *International Conference on Learning Representations*, 2018.

[13] Shai Shalev-Shwartz, Yoram Singer, Nathan Srebro, and Andrew Cotter.
Pegasos: primal estimated sub-gradient solver for svm.
Mathematical Programming, 127(1):3–30, Mar 2011.

[14] Ohad Shamir and Tong Zhang.
Stochastic gradient descent for non-smooth optimization: Convergence results and optimal averaging schemes.
In *International Conference on Machine Learning*, pages 71–79, 2013.

References IV

[15] Thomas Strohmer and Roman Vershynin.
A randomized kaczmarz algorithm with exponential convergence.
Journal of Fourier Analysis and Applications, 15(2):262, Apr 2008.

[16] Vladimir N. Vapnik.
An overview of statistical learning theory.
IEEE Trans. Inf. Theory, 10(5):988–999, September 1999.

[17] Rachel Ward, Xiaoxia Wu, and Leon Bottou.
AdaGrad stepsizes: Sharp convergence over nonconvex landscapes.
In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 6677–6686, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

[18] Lin Xiao and Tong Zhang.
A proximal stochastic gradient method with progressive variance reduction.
SIAM Journal on Optimization, 24, 03 2014.

[19] Dongruo Zhou, Yiqi Tang, Ziyan Yang, Yuan Cao, and Quanquan Gu.
On the convergence of adaptive gradient methods for nonconvex optimization.
ArXiv, abs/1808.05671, 2018.