Accelerated Stochastic Subgradient Methods under Local Error Bound Condition

Yi Xu

yi-xu@uiowa.edu
Computer Science Department
The University of Iowa
April 18, 2018

Co-authors: Tianbao Yang, Qihang Lin

Outline

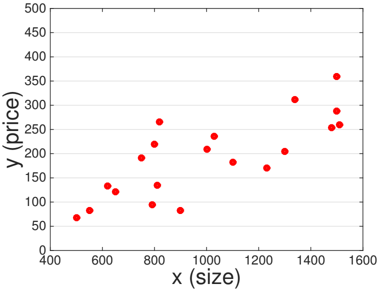
1. Introduction
2. Accelerated Stochastic Subgradient Methods
3. Applications and experiments
4. Conclusion

Outline

Example in machine learning

Table: house price

house	size (sqf)	price ($1k)
1	68	500
2	220	800
...	...	...
19	359	1500
20	266	820



Linear model:

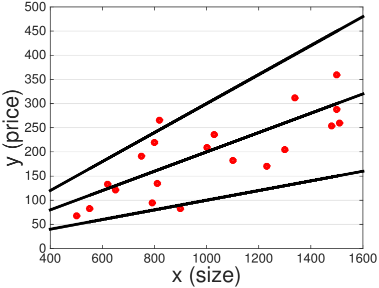
$$y = f(w) = xw,$$

where y = price, x = size.

Example in machine learning

Table: house price

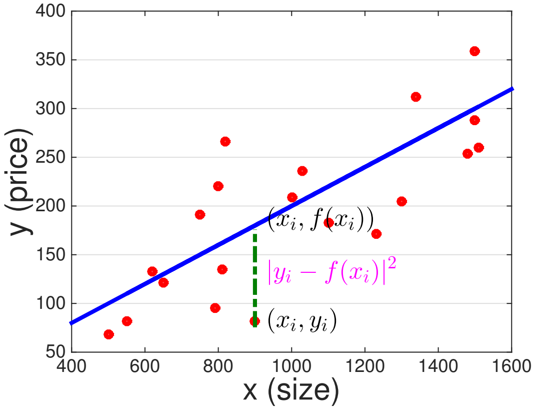
house	size (sqf)	price ($1k)
1	68	500
2	220	800
...	...	...
19	359	1500
20	266	820



Linear model:

$$y = f(w) = xw,$$

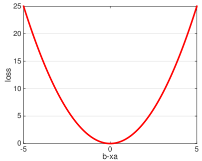
where y = price, x = size.



$$|y_1 - x_1 w|^2 + |y_2 - x_2 w|^2 + \ldots |y_{20} - x_{20} w|^2$$

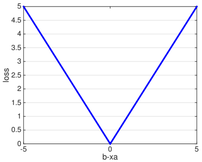
Least squares regression:

$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{(y_i - x_i w)^2}_{\text{square loss}}$$



Least absolute deviations:

$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{|y_i - x_i w|}_{\text{absolute loss}}$$



High dimensional model:

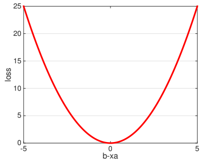
$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \mathbf{x}_i^\top \mathbf{w}| + \lambda\|\mathbf{w}\|_1 = \frac{1}{n}\|X\mathbf{w} - \mathbf{y}\|_1 + \underbrace{\lambda\|\mathbf{w}\|_1}_{\text{regularizer}}$$

- absolute loss is more robust to outliers problem
- ℓ_1 norm regularization is used for feature selection

Least squares regression:

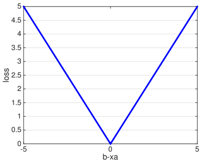
smooth

$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{(y_i - x_i w)^2}_{\text{square loss}}$$



Least absolute deviations:

$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{|y_i - x_i w|}_{\text{absolute loss}}$$



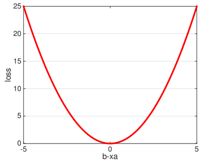
High dimensional model:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \mathbf{x}_i^\top \mathbf{w}| + \lambda\|\mathbf{w}\|_1 = \frac{1}{n}\|X\mathbf{w} - \mathbf{y}\|_1 + \underbrace{\lambda\|\mathbf{w}\|_1}_{\text{regularizer}}$$

- absolute loss is more robust to outliers problem
- ℓ_1 norm regularization is used for feature selection

Least squares regression:

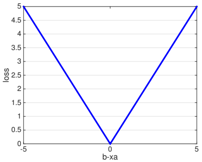
$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{(y_i - x_i w)^2}_{\text{square loss}}$$



Least absolute deviations:

non-smooth

$$\min_{w\in\mathbb{R}} F(w) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{|y_i - x_i w|}_{\text{absolute loss}}$$



High dimensional model:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} |y_i - \mathbf{x}_i^\top \mathbf{w}| + \lambda\|\mathbf{w}\|_1 = \frac{1}{n}\|X\mathbf{w} - \mathbf{y}\|_1 + \underbrace{\lambda\|\mathbf{w}\|_1}_{\text{regularizer}}$$

- absolute loss is more robust to outliers problem
- ℓ_1 norm regularization is used for feature selection

Machine learning problems:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\ell(\mathbf{w};\mathbf{x}_i,y_i)}_{\text{loss function}} + \underbrace{r(\mathbf{w})}_{\text{regularizer}}$$

- Classification:
  - hinge loss: $\ell(\mathbf{w};\mathbf{x},y) = \max(0, 1 - y\mathbf{x}^\top\mathbf{w})$
- Regression:
  - absolute loss: $\ell(\mathbf{w};\mathbf{x},y) = |\mathbf{x}^\top\mathbf{w} - y|$
  - square loss: $\ell(\mathbf{w};\mathbf{x},y) = (\mathbf{x}^\top\mathbf{w} - y)^2$
- Regularizer:
  - ℓ_1 norm: $r(\mathbf{w}) = \lambda\|\mathbf{w}\|_1$
  - ℓ_2^2 norm: $r(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2$

Convex optimization problem

- Problem:

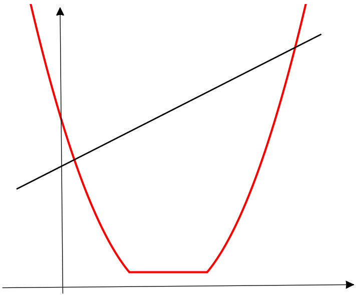
$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w})$$

  - $F(\mathbf{w}) : \mathbb{R}^d \rightarrow \mathbb{R}$ is convex
  - optimal value: $F(\mathbf{w}_*) = \min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w})$
  - optimal solution: $\mathbf{w}_*$

- Goal: to find a solution $\widehat{\mathbf{w}}$

$$F(\widehat{\mathbf{w}}) - F(\mathbf{w}_*) \leq \epsilon$$

  - $0 < \epsilon \ll 1$, (e.g. 10^{-7})
  - ϵ-optimal solution: $\widehat{\mathbf{w}}$



Complexity measure

- Most optimization algorithms are iterative

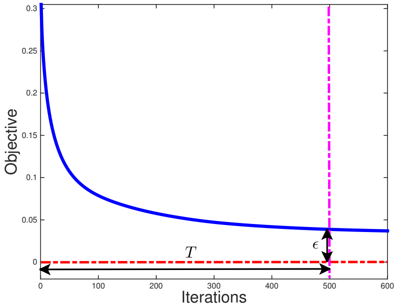
$$\mathbf{w}_{t+1} = \mathbf{w}_t + \nabla \mathbf{w}_t$$

- **Iteration complexity**: number of iterations $T(\epsilon)$ that

$$F(\mathbf{w}_T) - F(\mathbf{w}_*) \leq \epsilon$$

  where $0 < \epsilon \ll 1$.

- **Time complexity**: $T(\epsilon) \times C(n, d)$
  - $C(n, d)$: Per-iteration cost

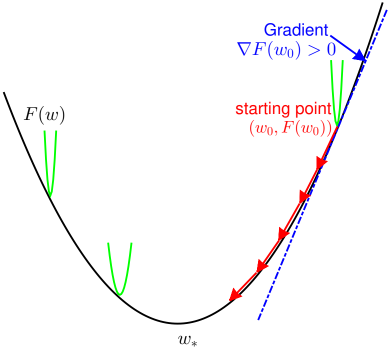


Gradient Descent (GD)

- Problem: $\min_{w \in \mathbb{R}} F(w)$
- $w_{t+1} = \arg\min_{w \in \mathbb{R}} F(w_t) + \langle \nabla F(w_t), w - w_t \rangle + \frac{L}{2}\|w - w_t\|_2^2$
- **GD**: initial $w_0 \in \mathbb{R}$, for $t = 0, 1, \ldots$

$$w_{t+1} = w_t - \eta \nabla F(w_t)$$

- $\eta = \frac{1}{L} > 0$: step size.
- simple & easy to implement



Theorem ([Nesterov, 2004])

$$\text{After } T = O\left(\frac{1}{\epsilon}\right), F(\mathbf{w}_T) - F(\mathbf{w}_*) \le \epsilon$$

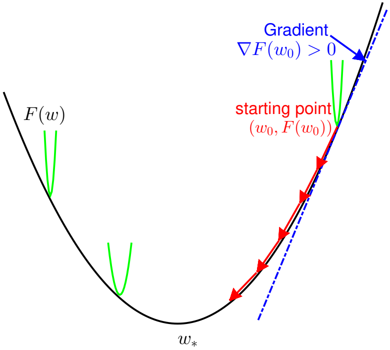
Gradient Descent (GD)

smooth

- Problem: $\min_{w\in\mathbb{R}} F(w)$
- $w_{t+1} = \arg\min_{w\in\mathbb{R}} F(w_t) + \langle \nabla F(w_t), w - w_t\rangle + \frac{L}{2}\|w - w_t\|_2^2$
- **GD**: initial $w_0 \in \mathbb{R}$, for $t = 0, 1, \ldots$

$$w_{t+1} = w_t - \eta \nabla F(w_t)$$

- $\eta = \frac{1}{L} > 0$: step size.
- simple & easy to implement



Theorem ([Nesterov, 2004])

$$\text{After } T = O\left(\frac{1}{\epsilon}\right), F(\mathbf{w}_T) - F(\mathbf{w}_*) \le \epsilon$$

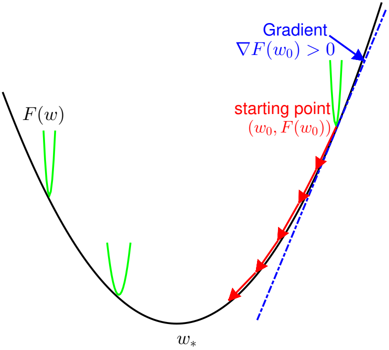
Gradient Descent (GD)

$F(w) \leq F(w_t) + \langle \nabla F(w_t), w - w_t \rangle + \frac{L}{2}\|w - w_t\|_2^2$

- Problem: $\min_{w \in \mathbb{R}} F(w)$
- $w_{t+1} = \arg\min_{w \in \mathbb{R}} F(w_t) + \langle \nabla F(w_t), w - w_t \rangle + \frac{L}{2}\|w - w_t\|_2^2$
- **GD**: initial $w_0 \in \mathbb{R}$, for $t = 0, 1, \ldots$

$$w_{t+1} = w_t - \eta \nabla F(w_t)$$

- $\eta = \frac{1}{L} > 0$: step size.
- simple & easy to implement



Theorem ([Nesterov, 2004])

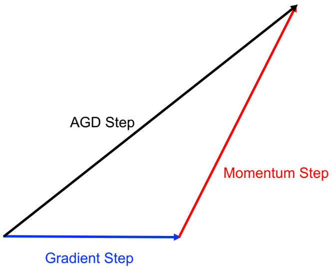
After $T = O\left(\frac{1}{\epsilon}\right)$, $F(\mathbf{w}_T) - F(\mathbf{w}_*) \leq \epsilon$

Accelerated Gradient Descent (AGD)

- Nesterov's momentum trick
- **AGD**: initial $\mathbf{w}_0$, $\mathbf{v}_1 = \mathbf{w}_0$, for $t = 1, 2, \ldots$:

$$\mathbf{w}_t = \mathbf{v}_t - \eta \nabla F(\mathbf{v}_t)$$
$$\mathbf{v}_{t+1} = \mathbf{w}_t + \beta_t(\mathbf{w}_t - \mathbf{w}_{t-1})$$

- $\beta_t \in (0, 1)$ is momentum parameter.
- Nesterov's Accelerated Gradient



Theorem ([Beck and Teboulle, 2009])

Let $\eta = \frac{1}{L}$, $\beta_t = \frac{\theta_t - 1}{\theta_{t+1}} \in (0, 1)$ *with* $\theta_{t+1} = \frac{1 + \sqrt{1 + 4\theta_t^2}}{2}$ *and* $\theta_1 = 1$, *then after*

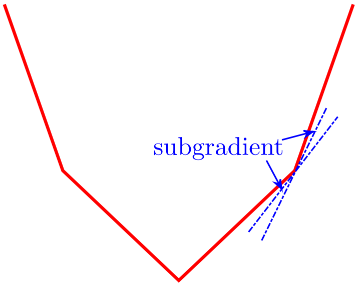
$$T = O\left(\frac{1}{\sqrt{\epsilon}}\right), F(\mathbf{w}_T) - F(\mathbf{w}_*) \le \epsilon$$

SubGradient (SG) descent

- Problem: $\min_{w \in \mathbb{R}} F(w)$
- **SG**: initial w_0, for $t = 0, 1, \ldots$

$$w_{t+1} = w_t - \eta \partial F(w_t)$$

- decrease η every iteration.



Theorem ([Nesterov, 2004])

$$\textit{After } T = O\left(\frac{1}{\epsilon^2}\right), F(\mathbf{w}_T) - F(\mathbf{w}_*) \leq \epsilon$$

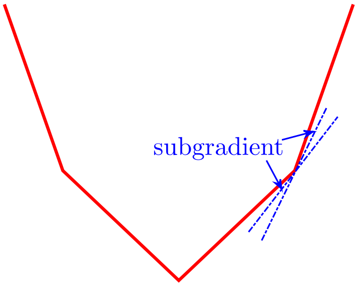
SubGradient (SG) descent

non-smooth

- Problem: $\min_{w \in \mathbb{R}} F(w)$
- **SG**: initial w_0, for $t = 0, 1, \ldots$

$$w_{t+1} = w_t - \eta \partial F(w_t)$$

- decrease η every iteration.



Theorem ([Nesterov, 2004])

After $T = O\left(\frac{1}{\epsilon^2}\right)$, $F(\mathbf{w}_T) - F(\mathbf{w}_*) \leq \epsilon$

Summary of time complexity

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w}; \mathbf{x}_i, y_i)$$

Method	Time complexity	Smooth
GD	$O\left(\frac{nd}{\epsilon}\right)$	YES
AGD	$O\left(\frac{nd}{\sqrt{\epsilon}}\right)$	YES
SG	$O\left(\frac{nd}{\epsilon^2}\right)$	NO

GD: Gradient Descent
AGD: Accelerated Gradient Descent
SG: SubGradient descent

Challenge of deterministic methods

Computing gradient is expensive

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w};\mathbf{x}_i, y_i)$$

$$\nabla F(\mathbf{w}) := \frac{1}{n}\sum_{i=1}^{n} \nabla f_i(\mathbf{w};\mathbf{x}_i, y_i)$$

- When n/d is large: Big Data
- To compute the gradient, need to pass through all data points.
- At each updating step, need this expensive computation.

Stochastic Gradient Descent (SGD)

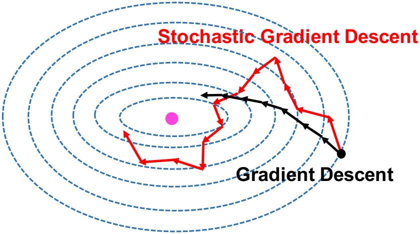
- **SGD**: initial w_0, for $t = 0, 1, \ldots$

  sample one data $\xi_t = (\mathbf{x}_t, y_t)$

  $\mathbf{w}_{t+1} = \mathbf{w}_t - \eta \nabla f(\mathbf{w}_t; \xi_t)$

$$\min_{\mathbf{w} \in \mathbb{R}^d} F(\mathbf{w}) := \mathrm{E}_{\xi \sim \mathcal{P}}[f(\mathbf{w}; \xi)]$$

- decrease η every iteration
- simple & memory efficient
- problem: variance of stochastic gradient, slow convergence



Theorem ([Nemirovski et al., 2009])

After $T = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$, $F(w_T) - F(w_*) \le \epsilon$ *with a probability* $1 - \delta$.

Stochastic SubGradient (SSG) descent

- Problem:

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \mathrm{E}_{\xi\sim\mathcal{P}}[f(\mathbf{w};\xi)]$$

- **SSG**: initial w_0, for $t = 0, 1, \ldots$

sample one data ξ_t

$$\mathbf{w}_{t+1} = \mathbf{w}_t - \eta\partial f(\mathbf{w}_t;\xi_t)$$

- decrease η every iteration

Theorem ([Hazan and Kale, 2011])

After $T = O\left(\frac{\log(1/\delta)}{\epsilon^2}\right)$, $F(w_T) - F(w_*) \leq \epsilon$ with a probability $1-\delta$.

Summary of time complexity

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w};\mathbf{x}_i, y_i)$$

Method	Time complexity	Smooth
SGD	$\widetilde{O}\left(\frac{d}{\epsilon^2}\right)$	YES
SSG	$\widetilde{O}\left(\frac{d}{\epsilon^2}\right)$	NO

SGD: Stochastic Gradient Descent
SSG: Stochastic SubGradient descent

- SGD can not enjoy the smoothness property to obtain faster rate.

How can we do better?

- Assume Strong Global Assumptions (e.g., strong convexity, smoothness): smaller family of problems
- Strongly convex problems

$$F(\mathbf{x}) \geq F(\mathbf{y}) + \partial F(\mathbf{y})^{\top}(\mathbf{x}-\mathbf{y}) + \frac{\lambda}{2}\|\mathbf{x}-\mathbf{y}\|_2^2$$

- $\lambda > 0$: strong convexity parameter.
- SSG with $\eta_t = 1/(\lambda t)$ enjoys $O\left(\frac{1}{\lambda\epsilon}\right)$ iteration complexity.

Strong convexity is sometimes too good to be true

Non-smooth and non-strongly problems in ML

Robust Regression:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n}|\mathbf{w}^\top\mathbf{x}_i - y_i|^p, \quad p \in [1,2)$$

Sparse Classification:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i\mathbf{w}^\top\mathbf{x}_i) + \lambda\|\mathbf{w}\|_1$$

Outline

The contributions of our paper

Y. Xu, Q. Lin, and T. Yang. ***Stochastic convex optimization: Faster local growth implies faster global convergence.*** *In ICML, pages 3821-3830, 2017.*

- A New Theory of Stochastic Convex Optimization
  - A Broader Family of Conditions: Local Error Bound Condition
  - Faster Global Convergence under Local Error Bound Condition
  - Applications in Machine Learning

Local error bound (LEB) condition

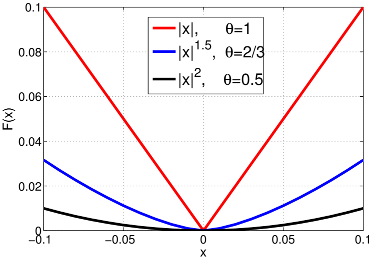
Definition

If there exists a constant $c > 0$ and a **local growth rate** $\theta \in (0, 1]$ such that:

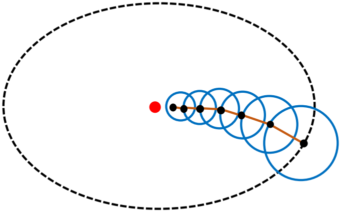
$$\|\mathbf{w} - \mathbf{w}_*\|_2 \leq c(F(\mathbf{w}) - F(\mathbf{w}_*))^{\theta}, \quad \forall \mathbf{w} \in \mathcal{S}_\epsilon, \tag{1}$$

then we say $F(\mathbf{w})$ satisfies a **local error bound condition** (also know as local growth condition).

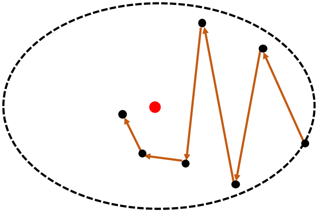
- $\mathcal{S}_\epsilon = \{\mathbf{w} \in \mathbb{R}^d : F(\mathbf{w}) - F_* \leq \epsilon\}$: ϵ-sublevel set.
- A local sharpness measure of the function



Sketch of accelerated algorithm



ASSG



SSG

Accelerated Stochastic SubGradient (ASSG) method

1: Set η_1, K and t
2: **for** $k = 1, \ldots, K$ **do**
3: $\quad \mathbf{w}_k = \text{SSG}(\mathbf{w}_{k-1}, \eta_k, D_k, t)$
4: $\quad \eta_{k+1} = \eta_k/2$, $D_{k+1} = D_k/2$
5: **end for**

SSG$(\mathbf{w}_1, \eta, D, t)$: for $\tau = 1, \ldots, t$

$$\mathbf{w}_{\tau+1} = \text{Proj}_{\|\mathbf{w}-\mathbf{w}_1\|_2 \leq D}[\mathbf{w}_\tau - \eta\partial f_\tau(\mathbf{w}_\tau, \mathbf{z}_\tau)]$$

Output: $\widehat{\mathbf{w}} = \sum_{\tau=1}^{t} \mathbf{w}_\tau/t$

Theorem [Xu et al., 2017]

After $T = O\left(t\log\left(\frac{1}{\epsilon}\right)\right)$ iterations with $t \geq \frac{\log(1/\delta)G^2c^2}{\epsilon^{2(1-\theta)}}$, $F(\mathbf{w}_K) - F_* \leq 2\epsilon$ with a probability $1-\delta$.

Practical Variant: ASSG with Restarting (RASSG)

Setting $t \geq \frac{\log(1/\delta)G^2c^2}{\epsilon^{2(1-\theta)}}$ requires c, which is usually unknown

A Practical Variant:

1: **Input**: $D_1^{(1)}$, t_1, $\mathbf{w}^{(0)}$ and $\eta_1 = \epsilon_0/(3G^2)$
2: **for** $s = 1, 2, \ldots, S$ **do**
3: Let $\mathbf{w}^{(s)}$ =ASSG$(\mathbf{w}^{(s-1)}, K, t_s, D_1^{(s)})$
4: Let $t_{s+1} = t_s 2^{2(1-\theta)}$, $D_1^{(s+1)} = D_1^{(s)} 2^{1-\theta}$
5: **end for**

- another level of restarting
- increasing t by a factor of $2^{2(1-\theta)}$
- iteration complexity remains the same

Summary of time complexity

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \mathrm{E}_{\xi\sim\mathcal{P}}[f(\mathbf{w};\xi)]$$

Table: Time complexities for non-smooth stochastic optimization methods[1]

Method	Time complexity	Condition
SSG	$O\left(\frac{d}{\epsilon^2}\right)$	Stochastic structure
ASSG	$\widetilde{O}\left(\frac{d}{\epsilon^{2(1-\theta)}}\right)$	Stochastic structure and LEB

SSG: Stochastic SubGradient descent
ASSG: Accelerated Stochastic SubGradient descent

[1] $\theta \in (0, 1]$

Outline

Piecewise linear convex optimization

$\theta = 1 \Longrightarrow$ ASSG achieves $O(\log(1/\epsilon))$ iteration complexity

Examples:

- Robust Regression

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n} |\mathbf{w}^\top \mathbf{x}_i - y_i|$$

- Sparse Classification:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n} \max(0, 1 - y_i \mathbf{w}^\top \mathbf{x}_i) + \lambda \|\mathbf{w}\|_1$$

Piecewise quadratic convex optimization

$\theta = 1/2 \Longrightarrow$ ASSG achieves $\widetilde{O}(1/\epsilon)$ iteration complexity

Examples:

- Least-squares regression + ℓ_1 regularizer

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n}(\mathbf{w}^\top \mathbf{x}_i - y_i)^2 + \lambda\|\mathbf{w}\|_1$$

- Squared hinge loss + ℓ_1 regularizer:

$$\min_{\mathbf{w}} \frac{1}{n}\sum_{i=1}^{n}\max(0, 1 - y_i\mathbf{w}^\top \mathbf{x}_i)^2 + \lambda\|\mathbf{w}\|_1$$

- Hurbe loss: $\ell(\mathbf{w}^\top \mathbf{x}_i, y_i) = \begin{cases} \frac{1}{2}(\mathbf{w}^\top \mathbf{x}_i - y_i)^2 & \text{for} \quad |\mathbf{w}^\top \mathbf{x} - y_i| \le \gamma \\ \gamma(|\mathbf{w}^\top \mathbf{x}_i - y_i| - \frac{1}{2}\gamma) & \text{for} \quad |\mathbf{w}^\top \mathbf{x}_i - y_i| > \gamma \end{cases}$

Structured composite non-smooth problems

$$F(\mathbf{w}) = h(A\mathbf{w}) + R(\mathbf{w})$$

- $h(\cdot)$ is strongly convex (no smoothness assumption is required)
- $R(\mathbf{w})$ is polyhedral
- $\theta = 1/2 \Longrightarrow$ ASSG achieves $\widetilde{O}(1/\epsilon)$ iteration complexity

Examples:

- Robust Regression + ℓ_1 regularizer

$$\min_{\mathbf{w}} \frac{1}{n} \sum_{i=1}^{n} |\mathbf{w}^\top \mathbf{x}_i - y_i|^p + \lambda \|\mathbf{w}\|_1, p \in (1, 2)$$

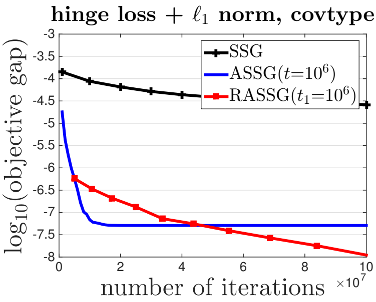
Problems with intermediate θ

ℓ_p norm regression with ℓ_1 constraint

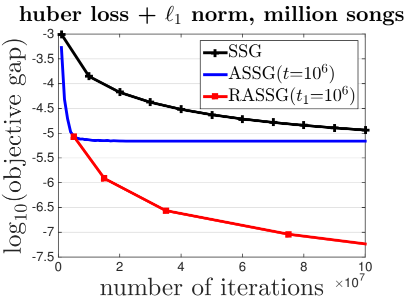
$$\min_{\|\mathbf{w}\|_1 \leq B} \frac{1}{n} \sum_{i=1}^{n} (\mathbf{w}^\top \mathbf{x}_i - y_i)^{2p}, p \in \mathbb{N}^+$$

where $\theta = 1/(2p)$

Experiments: SSG vs. ASSG

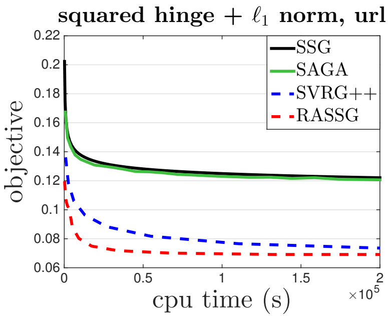


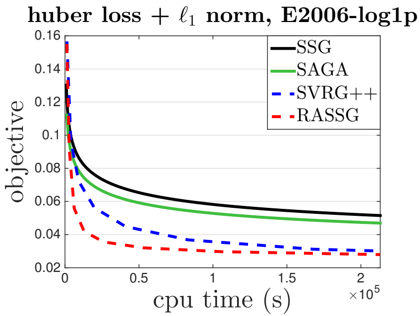
Classification



Regression

Experiments: ASSG vs Other Baselines





Outline

Conclusion

- Present our recent improved work ASSG with a lower iteration complexity for solving **non-smooth** optimization problems.

Method	Time complexity	Problem
SSG	$O\left(\frac{d}{\epsilon^2}\right)$	Stochastic structure
ASSG	$\widetilde{O}\left(\frac{d}{\epsilon^{2(1-\theta)}}\right)$	Stochastic structure + LEB

- Study examples satisfying LEB in machine learning.
- RASSG for $\theta = 1$?
- Nonconvex problems?

Thank You! Questions?

Reference

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Img. Sci.*, 2:183–202, 2009.

Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: an optimal algorithm for stochastic strongly-convex optimization. In *Proceedings of the 24th Annual Conference on Learning Theory (COLT)*, pages 421–436, 2011.

Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19:1574–1609, 2009.

Yurii Nesterov. *Introductory lectures on convex optimization : a basic course*. Applied optimization. Kluwer Academic Publ., 2004. ISBN 1-4020-7553-7.

Yi Xu, Qihang Lin, and Tianbao Yang. Stochastic convex optimization: Faster local growth implies faster global convergence. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 3821–3830, 2017.