UNIVERSITY OF MICHIGAN

EECS 559
Optimization Methods for SIPML

Lecture 8 – Accelerated Proximal Gradient Methods

Instructor: Prof. Qing Qu (qingqu@umich.edu)

Lecture Agenda

- Accelerated Proximal Gradient Method
- Proximal Gradient Homotopy Method

Lecture Agenda

- **Accelerated Proximal Gradient Method**

- Proximal Gradient Homotopy Method

Unconstrained Minimization

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \; f(\boldsymbol{x})$$

- Suppose $f(\boldsymbol{x})$ belongs the class of convex and smooth with its gradient L-Lipschitz: $f \in \mathcal{F}_{L,R}$ with

$$\mathcal{F}_{L,R} := \{f : \mathcal{B}(\mathbf{0}, R) \mapsto \mathbb{R} \mid \|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{x}')\|_2 \leq L \|\boldsymbol{x} - \boldsymbol{x}'\|_2\}$$

- Solve the problem via *iterative optimization methods*, producing a sequence of points

$$\boldsymbol{x}_1, \; \boldsymbol{x}_2, \; \cdots, \boldsymbol{x}_k, \cdots$$

starting from an initialization $\boldsymbol{x}_0$.

Suboptimal Convergence Rate

- Now suppose the iterates $\{\boldsymbol{x}_i\}_{i\geq 0}$ are generated from a *black-box model*:

$$\boldsymbol{x}_{k+1} \;=\; \varphi_k\left(\{\boldsymbol{x}_i\}_{i=0}^{k},\; \{f(\boldsymbol{x}_i)\}_{i=0}^{k},\; \{\nabla f(\boldsymbol{x}_i)\}_{i=0}^{k}\right)$$

- We ask how well an algorithm does on the *worst-function* in $\mathcal{F}_{L,R}$?

$$\sup_{f\in\mathcal{F}_{L,R},\boldsymbol{x}_0}\left\{f(\boldsymbol{x}_k)-\inf_{\boldsymbol{x}} f(\boldsymbol{x})\right\}$$

Suboptimal Convergence Rate

- The gradient descent method achieves

$$\sup_{f\in\mathcal{F}_{L,R},\boldsymbol{x}_0}\left\{f(\boldsymbol{x}_k)-\inf_{\boldsymbol{x}} f(\boldsymbol{x})\right\} \leq \frac{CLR^2}{k}$$

- The best lower bound proved by Nesterov'83:

$$\sup_{f\in\mathcal{F}_{L,R},\boldsymbol{x}_0}\left\{f(\boldsymbol{x}_k)-\inf_{\boldsymbol{x}} f(\boldsymbol{x})\right\} \geq \frac{cLR^2}{k^2}$$

Suboptimal Convergence Rate

	Method	Iterate	Convergence
Smooth	Gradient Descent (GD)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \tau \nabla f(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated GD	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \tau \nabla f(\boldsymbol{p}_{k+1})$	$O(1/k^2)$
Nonsmooth	Proximal Gradient (PG)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu G_\mu(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated PG?	??	$O(1/k^2)$??

Suboptimal Convergence Rate

	Method	Iterate	Convergence
Smooth	Gradient Descent (GD)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \tau \nabla f(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated GD	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \tau \nabla f(\boldsymbol{p}_{k+1})$	$O(1/k^2)$
Nonsmooth	Proximal Gradient (PG)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu G_\mu(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated PG?	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1} \cdot (\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$, $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \mu G_\mu(\boldsymbol{p}_{k+1})$	$O(1/k^2)$??

Nonsmooth Composite Problems

$$\min_{\boldsymbol{x}} \; F(\boldsymbol{x}) \;=\; f(\boldsymbol{x}) \;+\; g(\boldsymbol{x})$$

- f: convex, continuously differentiable, and L-smooth,

$$\|\nabla f(\boldsymbol{x}') - \nabla f(\boldsymbol{x})\|_2 \;\leq\; L\,\|\boldsymbol{x}' - \boldsymbol{x}\|_2\,, \quad \forall \boldsymbol{x},\, \boldsymbol{x}' \in \mathbb{R}^n.$$

- g: convex but may *not* be differentiable.

Accelerated Proximal Gradient (APG)

- **Accelerated Gradient Descent**

$$\begin{aligned} \boldsymbol{p}_{k+1} &= \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}) \\ \boldsymbol{x}_{k+1} &= \boldsymbol{p}_{k+1} - \tau \nabla f(\boldsymbol{p}_{k+1}) \end{aligned}$$

- **Accelerated Proximal Gradient**

$$\begin{aligned} \boldsymbol{p}_{k+1} &= \boldsymbol{x}_k + \beta_{k+1} \cdot (\boldsymbol{x}_k - \boldsymbol{x}_{k-1}), \\ \boldsymbol{x}_{k+1} &= \operatorname{prox}_{\mu g}(\boldsymbol{p}_{k+1} - \mu \nabla f(\boldsymbol{p}_{k+1})) \\ &= \boldsymbol{p}_{k+1} - \mu G_\mu(\boldsymbol{p}_{k+1}) \end{aligned}$$

Accelerated Proximal Gradient (APG)

$$\boxed{\min_{\boldsymbol{x}} \; F(\boldsymbol{x}) \;=\; f(\boldsymbol{x}) + g(\boldsymbol{x}).}$$

$$\begin{aligned}
\boldsymbol{p}_{k+1} &= \boldsymbol{x}_k + \beta_{k+1} \cdot (\boldsymbol{x}_k - \boldsymbol{x}_{k-1}), \\
\boldsymbol{x}_{k+1} &= \operatorname{prox}_{\mu g}\left(\boldsymbol{p}_{k+1} - \mu \nabla f(\boldsymbol{p}_{k+1})\right) \\
&= \boldsymbol{p}_{k+1} - \mu G_\mu(\boldsymbol{p}_{k+1})
\end{aligned}$$

$$\tau_1 \;=\; 1, \quad \tau_{k+1} \frac{1 + \sqrt{1 + 4\tau_k^2}}{2}, \quad \beta_{k+1} \;=\; \frac{\tau_k - 1}{\tau_{k+1}}.$$

Convergence of APG

Theorem (Convergence of APG)
Let the sequence $\{\boldsymbol{x}_k\}_{k\geq 1}$ be generated by the APG for the convex composite function $F(\boldsymbol{x}) = f(\boldsymbol{x}) + g(\boldsymbol{x})$ with $\mu = 1/L$, where f is L-smooth. Let $\boldsymbol{x}_\star$ be a minimizer of $F(\boldsymbol{x})$, then for any $k \geq 1$,

$$F(\boldsymbol{x}_k) - F(\boldsymbol{x}_\star) \;\leq\; \frac{2L\,\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(k+1)^2}.$$

Suboptimal Convergence Rate

	Method	Iterate	Convergence
Smooth	Gradient Descent (GD)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \tau \nabla f(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated GD	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \tau \nabla f(\boldsymbol{p}_{k+1})$	$O(1/k^2)$
Nonsmooth	Proximal Gradient (PG)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu G_\mu(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated PG?	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1} \cdot (\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$, $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \mu G_\mu(\boldsymbol{p}_{k+1})$	$O(1/k^2)$

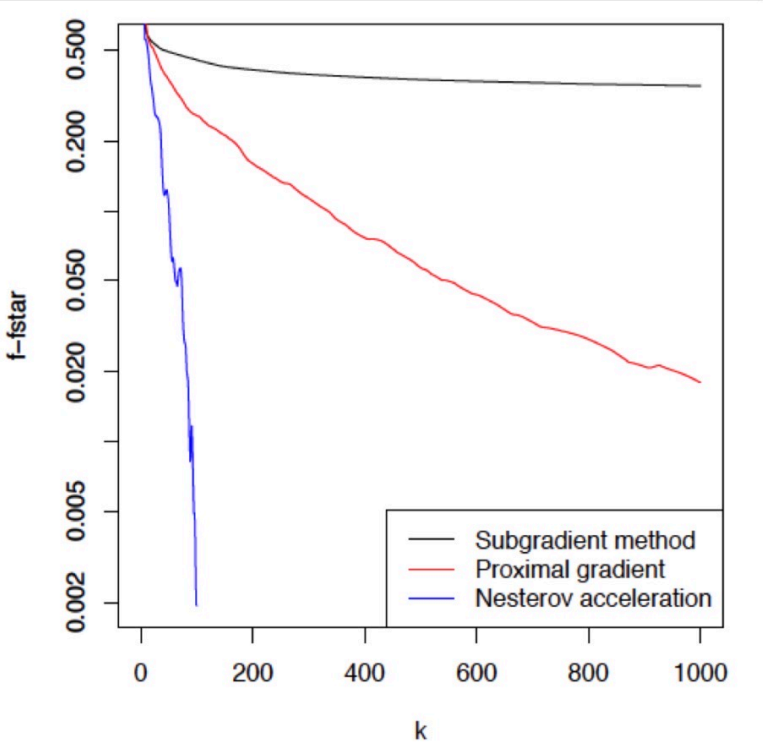
Example: Solving Lasso

Algorithm 8.3 Accelerated Proximal Gradient (APG) for BPDN

Problem: $\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda\|\boldsymbol{x}\|_1$, given $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{m \times n}$.
Input: $\boldsymbol{x}_0 \in \mathbb{R}^n$, $\boldsymbol{p}_1 = \boldsymbol{x}_1 \leftarrow \boldsymbol{x}_0$, and $t_1 \leftarrow 1$, and $L \geq \lambda_{\max}(\boldsymbol{A}^*\boldsymbol{A})$.
for $(k = 1, 2, \ldots, K-1)$ **do**
  $t_{k+1} \leftarrow \frac{1+\sqrt{1+4t_k^2}}{2}$; $\beta_{k+1} \leftarrow \frac{t_k - 1}{t_{k+1}}$.
  $\boldsymbol{p}_{k+1} \leftarrow \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$.
  $\boldsymbol{w}_{k+1} \leftarrow \boldsymbol{p}_{k+1} - \frac{1}{L}\boldsymbol{A}^*(\boldsymbol{A}\boldsymbol{p}_{k+1} - \boldsymbol{y})$.
  $\boldsymbol{x}_{k+1} \leftarrow \mathrm{soft}(\boldsymbol{w}_{k+1}, \lambda/L)$.
end for
Output: $\boldsymbol{x}_\star \leftarrow \boldsymbol{x}_K$.

Comparison on Lasso

$$\min_{\boldsymbol{x}} \; F(\boldsymbol{x}) = \underbrace{\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2}_{f(\boldsymbol{x})} + \underbrace{\lambda\|\boldsymbol{x}\|_1}_{g(\boldsymbol{x})}$$



Example: Solving Stable PCP

Algorithm 8.4 Accelerated Proximal Gradient (APG) for Stable PCP

Problem: $\min_{\boldsymbol{L},\boldsymbol{S}} \|\boldsymbol{L}\|_* + \lambda\|\boldsymbol{S}\|_1 + \frac{\mu}{2}\|\boldsymbol{Y} - \boldsymbol{L} - \boldsymbol{S}\|_F^2$, given $\boldsymbol{Y}$, $\lambda, \mu > 0$.
Input: $\boldsymbol{L}_0 \in \mathbb{R}^{m\times n}$, $\boldsymbol{S}_0 \in \mathbb{R}^{m\times n}$, $\boldsymbol{P}_1^S = \boldsymbol{S}_1 \leftarrow \boldsymbol{S}_0$, $\boldsymbol{P}_1^L = \boldsymbol{L}_1 \leftarrow \boldsymbol{L}_0$, $t_1 \leftarrow 1$.
for $(k = 1, 2, \ldots, K-1)$ **do**
  $t_{k+1} \leftarrow \frac{1+\sqrt{1+4t_k^2}}{2}$, $\beta_{k+1} \leftarrow \frac{t_k - 1}{t_{k+1}}$.
  $\boldsymbol{P}_{k+1}^L \leftarrow \boldsymbol{L}_k + \beta_{k+1}\big(\boldsymbol{L}_k - \boldsymbol{L}_{k-1}\big)$.
  $\boldsymbol{P}_{k+1}^S \leftarrow \boldsymbol{S}_k + \beta_{k+1}\big(\boldsymbol{S}_k - \boldsymbol{S}_{k-1}\big)$.
  $\boldsymbol{W}_{k+1} \leftarrow \boldsymbol{Y} - \boldsymbol{P}_{k+1}^S$ and compute the SVD: $\boldsymbol{W}_{k+1} = \boldsymbol{U}_{k+1}\boldsymbol{\Sigma}_{k+1}\boldsymbol{V}_{k+1}^*$.
  $\boldsymbol{L}_{k+1} \leftarrow \boldsymbol{U}_{k+1}\mathrm{soft}(\boldsymbol{\Sigma}_{k+1}, 1/\mu)\boldsymbol{V}_{k+1}^*$.
  $\boldsymbol{S}_{k+1} \leftarrow \mathrm{soft}((\boldsymbol{Y} - \boldsymbol{P}_{k+1}^L), \lambda/\mu)$.
end for
Output: $\boldsymbol{L}_\star \leftarrow \boldsymbol{L}_K$; $\boldsymbol{S}_\star \leftarrow \boldsymbol{S}_K$.

Convergence for Strongly Convex Function

Theorem. (Convergence for γ-Strongly Convex Function) Suppose f is L-smooth and γ-strongly convex. Let $\mu = 1/L$ and $\kappa = L/\gamma$, then for any $k \geq 1$,

$$F(\boldsymbol{x}_k) - F(\boldsymbol{x}_\star) \leq \left(1 - \frac{1}{\sqrt{\kappa}}\right)^k \left(F(\boldsymbol{x}_0) - F(\boldsymbol{x}_\star) + \frac{\mu \left\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\right\|_2^2}{2}\right)$$

$$\begin{aligned} \boldsymbol{p}_{k+1} &= \boldsymbol{x}_k + \beta_k \left(\boldsymbol{x}_k - \boldsymbol{x}_{k-1}\right), \quad \beta_k = \frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1}, \\ \boldsymbol{x}_{k+1} &= \operatorname{prox}_{\mu g}\left(\boldsymbol{p}_{k+1} - \mu \nabla f(\boldsymbol{p}_{k+1})\right). \end{aligned}$$

A Practical Issue

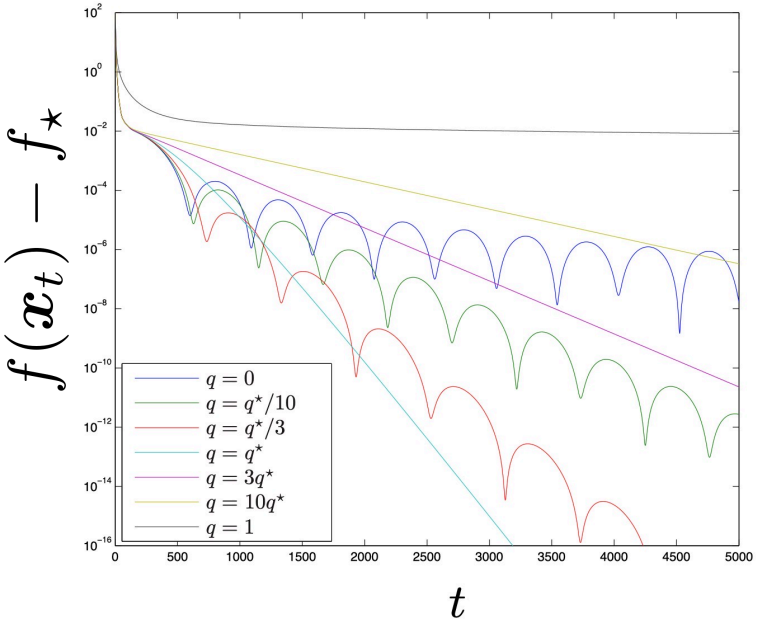
- Fast convergence requires knowledge of estimating $\kappa = L/\gamma$, strong convexity parameter γ could be challenging in practice.

- For sparse recovery, the APG method is also called the Fast Iterative Shrinkage-Thresholding Algorithm (FISTA).

- A common observation: *ripples/bumps* in the traces of cost values.

Rippling Behavior

Numerical example: $\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \frac{1-\sqrt{q}}{1+\sqrt{q}}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$, $q^\star = 1/\kappa$



- Period of ripples is often proportional to $\kappa^{1/2} = \sqrt{L/\gamma}$
- when $q < q^\star$: we overestimate momentum — rippling behavior
- when $q > q^\star$: we underestimate momentum — slow convergence

Adaptive Restart (O'Donoghue, Candes'12)

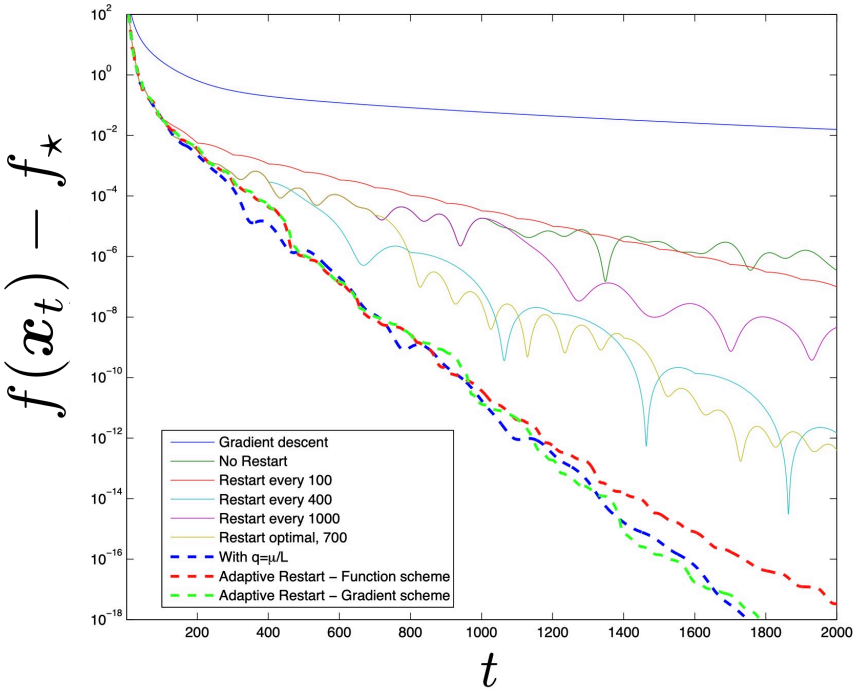
When a certain criterion is met, *restart* running APG with

$$\begin{aligned} \boldsymbol{p}_0 &\leftarrow \boldsymbol{x}_k \\ \boldsymbol{x}_0 &\leftarrow \boldsymbol{x}_k \\ t_0 &\leftarrow 1 \end{aligned}$$

- Take the current iterate as a new start pointing;
- Erase all memory of previous iterates and reset the momentum back to zero.

"Adaptive restart for accelerated gradient schemes," B. O'donoghue, and E. Candes, Foundations of computational mathematics, 2012

Adaptive Restart (O'Donoghue, Candes'12)



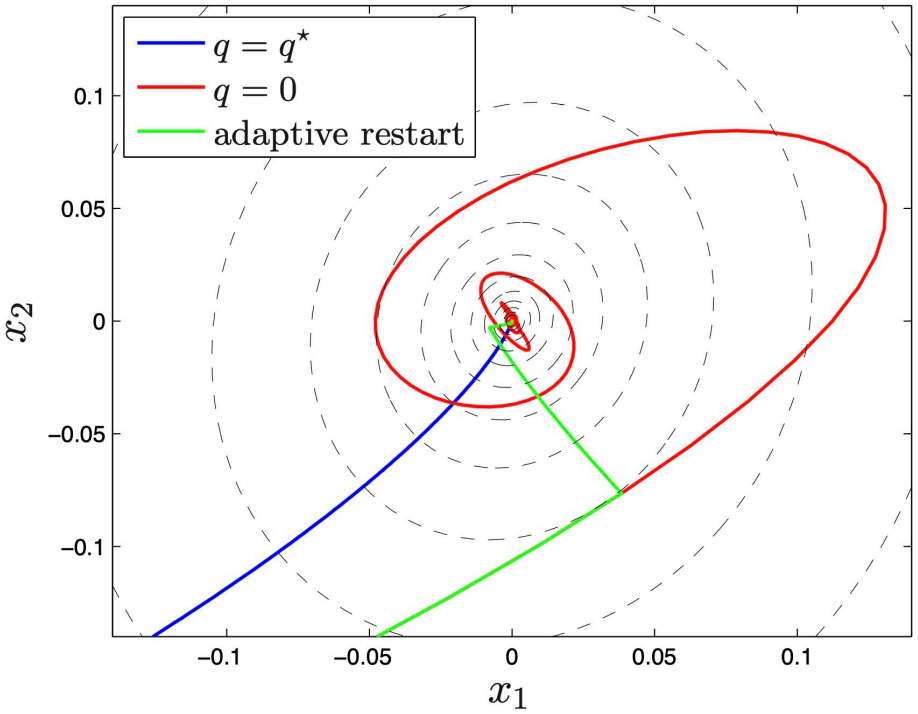
- **function scheme:**

restart if $f(\boldsymbol{x}_k) > f(\boldsymbol{x}_{k-1})$

- **gradient scheme:**

restart if $\langle \nabla f(\boldsymbol{p}_k), \boldsymbol{x}_k - \boldsymbol{x}_{k-1} \rangle > 0$

Adaptive Restart (O'Donoghue, Candes'12)



- With overestimated momentum (e.g., $q = 0$), one sees spiraling trajectory.
- Adaptive restart helps mitigate this issue.

Summary of Proximal Methods

	Method	Iterate	Convergence
Smooth	Gradient Descent (GD)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \tau \nabla f(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated GD	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \tau \nabla f(\boldsymbol{p}_{k+1})$	$O(1/k^2)$
Nonsmooth	Proximal Gradient (PG)	$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \mu G_\mu(\boldsymbol{x}_k)$	$O(1/k)$
	Accelerated PG?	$\boldsymbol{p}_{k+1} = \boldsymbol{x}_k + \beta_{k+1}(\boldsymbol{x}_k - \boldsymbol{x}_{k-1})$ $\boldsymbol{x}_{k+1} = \boldsymbol{p}_{k+1} - \tau G_\mu(\boldsymbol{p}_{k+1})$	$O(1/k^2)$

Further Readings

- *High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications.* John Wright, Yi Ma. (**Chapter 8.3**)

- *Proximal Algorithms, Foundations & Trends in Optimization.* Neal Parikh, Stephen Boyd, 2014. https://web.stanford.edu/~boyd/papers/prox_algs.html

- *A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems.* Amir Beck and Marc Teboulle, 2009. *https://epubs.siam.org/doi/abs/10.1137/080716542?mobileUi=0*

Lecture Agenda

- Accelerated Proximal Gradient Method
- **Proximal Gradient Homotopy Method**

Nonsmooth Composite Problems

$$\min_{\boldsymbol{x}} \; F(\boldsymbol{x}) \;=\; f(\boldsymbol{x}) \;+\; g(\boldsymbol{x})$$

- f: convex, continuously differentiable, and L-smooth,
- g: convex but may *not* be differentiable.

We know that

- APG achieves $O(1/k^2)$ convergence

Can we have faster convergence of proximal method by exploiting extra data structures?

Sparse Recovery

$\boldsymbol{y} \in \mathbb{R}^m \quad \boldsymbol{A} \in \mathbb{R}^{m \times n} \quad \boldsymbol{x}_\star \in \mathbb{R}^n$

Problem: given the measurement $\boldsymbol{y} \in \mathbb{R}^m$ and sensing matrix $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ $(m \ll n)$ recover the sparsest $\boldsymbol{x} \in \mathbb{R}^n$ from

$$\boldsymbol{y} \quad = \quad \boldsymbol{A} \cdot \boldsymbol{x}_\star \; + \; \boldsymbol{n}.$$

Stable Sparse Recovery via Lasso

Basis pursuit denoising (BPDN)

$$\min_{\boldsymbol{x}} \|\boldsymbol{x}\|_1, \quad \text{s.t.} \quad \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\| \leq \delta,$$

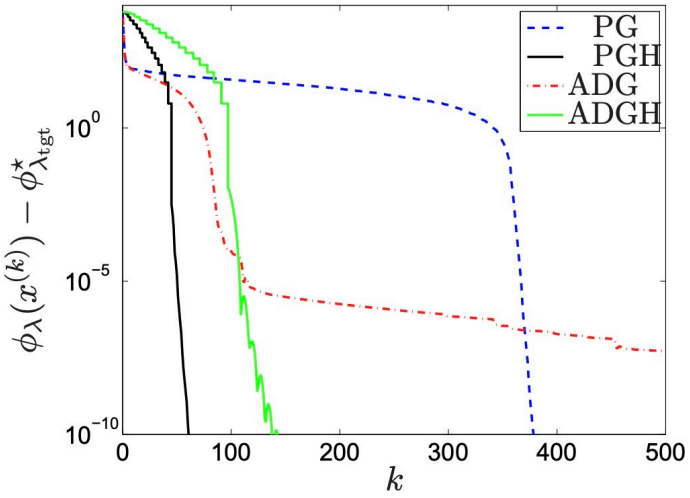
equivalent to

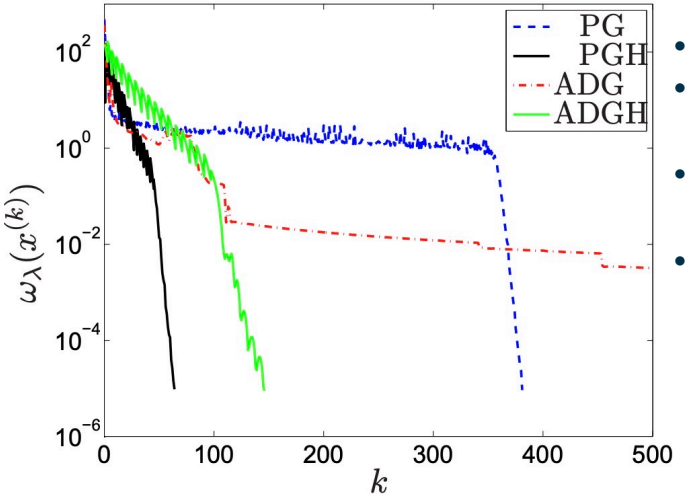
$$\boxed{\min_{\boldsymbol{x}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2 + \lambda \cdot \|\boldsymbol{x}\|_1,}$$

for some properly chosen $\lambda > 0$. The problem is termed as Lasso (least absolute shrinkage and selection operator).

Solving Lasso

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \ \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2 \ + \ \lambda\cdot\left\|\boldsymbol{x}\right\|_1,$$





- PG: proximal gradient
- PGH: proximal gradient-homotopy
- ADG: accelerated proximal gradient
- ADGH: accelerated proximal gradient - homotopy

Solving Lasso: Linear Convergence?

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \; F(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\|_2^2 \; + \; \lambda\cdot\|\boldsymbol{x}\|_1 \,,$$

- In theory, for convex (but *not* strongly convex) problems, the convergence rate for APG is $O(1/k^2)$.
- For solving Lasso, why is the final stage convergence *linear*?

Intuition: in the final stage, the solution is *sparse*, the problem satisfies certain *restricted strong convexity property*.

Restricted Strong Convexity and Smoothness

Theorem. Let $f(\boldsymbol{x}) = \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$ and suppose $\boldsymbol{A}$ satisfies REC. Suppose $\boldsymbol{x}$ and $\boldsymbol{x}'$ are two *sparse vectors* such that $|\text{supp}(\boldsymbol{x}) \cup \text{supp}(\boldsymbol{x}')| \leq s$ for some integer $s < m$. Then the following hold:

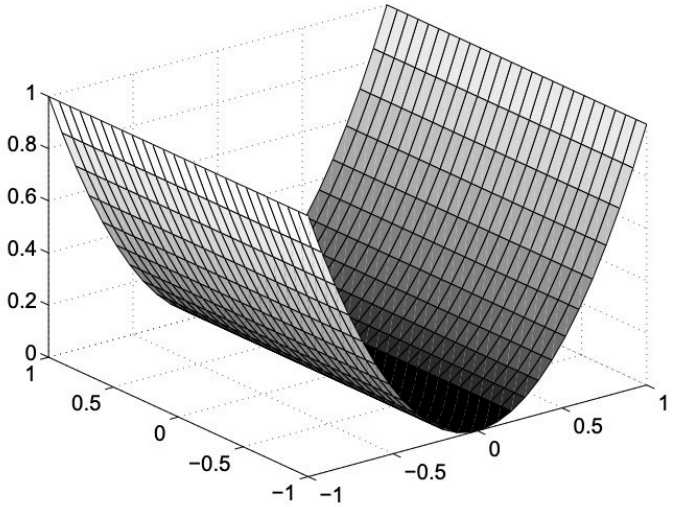
- **Restricted smoothness:**

$$f(\boldsymbol{x}') \leq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}' - \boldsymbol{x} \rangle + \frac{\rho_+(\boldsymbol{A}, s)}{2}\|\boldsymbol{x}' - \boldsymbol{x}\|_2^2,$$

- **Restricted strong convexity:**

$$f(\boldsymbol{x}') \geq f(\boldsymbol{x}) + \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}' - \boldsymbol{x} \rangle + \frac{\rho_-(\boldsymbol{A}, s)}{2}\|\boldsymbol{x}' - \boldsymbol{x}\|_2^2.$$

Restricted Strong Convexity and Smoothness



- When $\boldsymbol{A}$ satisfies REC, the function $F(\boldsymbol{x})$ is restricted strongly convex along *sparse* directions.
- Restricted strong convexity implies *linear* convergence.

Restricted Eigenvalue Condition

Definition. (Restricted eigenvalue condition, REC) Given an integer $s > 0$, we say that $\boldsymbol{A}$ satisfies the REC at the sparsity level s if there exist positive constants $\rho_+(\boldsymbol{A}, s)$ and $\rho_-(\boldsymbol{A}, s)$ such that

$$\rho_+(\boldsymbol{A}, s) = \sup_{\|\boldsymbol{x}\|_2=1,\ \|\boldsymbol{x}\|_0 \le s} \|\boldsymbol{A}\boldsymbol{x}\|_2^2,$$

$$\rho_-(\boldsymbol{A}, s) = \inf_{\|\boldsymbol{x}\|_2=1,\ \|\boldsymbol{x}\|_0 \le s} \|\boldsymbol{A}\boldsymbol{x}\|_2^2.$$

- If $\boldsymbol{A}$ satisfies RIP, then $\rho_+ = 1 + \delta$ and $\rho_- = 1 - \delta$.

Restricted Isometry Property (RIP)

Definition. (Restricted isometry property, RIP) Given an integer $s > 0$, we say that $\boldsymbol{A}$ satisfies the RIP at sparsity level s if there exist a positive constant $\delta > 0$ such that

$$(1-\delta) \cdot \|\boldsymbol{x}\|_2^2 \;\leq\; \|\boldsymbol{A}\boldsymbol{x}\|_2^2 \;\leq\; (1+\delta) \cdot \|\boldsymbol{x}\|_2^2$$

for all $\boldsymbol{x}$ with $\|\boldsymbol{x}\|_0 \leq s$.

Example: when $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ is i.i.d. Gaussian $A_{ij} \sim_{i.i.d.} \mathcal{N}(0,1)$ and $m \geq \Omega(\frac{n}{\delta} \log n)$, it satisfies δ-RIP with high probability.

Idea: Homotopy Continuation of λ

$$\min_{\boldsymbol{x}\in\mathbb{R}^n} \ \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{A}\boldsymbol{x}\right\|_2^2 \ + \ \lambda\cdot\left\|\boldsymbol{x}\right\|_1,$$

Adaptively adjust the λ to make $\{\boldsymbol{x}_k\}_{k\geq 1}$ *all sparse* along the solution path, so that the restricted strong convexity is satisfied.

- Pick a large initial $\lambda_0 = \left\|\boldsymbol{A}^\top\boldsymbol{y}\right\|_\infty$ so that $\boldsymbol{x}_0$ is sparse.
- *Shrink* the penalty λ to the desired λ_{tgt} *gradually*, making sure all $\{\boldsymbol{x}_k\}_{k\geq 1}$ are always *sparse*.

Proximal Gradient Homotopy Method

Algorithm 1 Proximal Gradient Homotopy Method

Input: $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{y} \in \mathbb{R}^{m}$, $\lambda_{tgt} > 0$, $\varepsilon > 0$, $L_{\min} > 0$
Parameters: $\eta \in (0,1)$, $\delta \in (0,1)$
Initialize: $\lambda_0 \leftarrow \left\| \boldsymbol{A}^\top \boldsymbol{y} \right\|_\infty$, $\varepsilon_0 \leftarrow \delta \lambda_0$, $\widehat{\boldsymbol{x}}_0 \leftarrow \boldsymbol{0}$, $M_0 \leftarrow L_{\min}$
$N \leftarrow \lfloor \log(\lambda_0 / \lambda_{tgt}) \,/\, \log(1/\eta) \rfloor$
for $K = 0, 1, \cdots, N-1$ **do**
    $\{\widehat{\boldsymbol{x}}_{K+1}, M_{K+1}\} \leftarrow \text{ProxGrad}(\lambda_K, \varepsilon_K, \widehat{\boldsymbol{x}}_K, M_K)$
    $\lambda_{K+1} \leftarrow \eta \lambda_K$, $\varepsilon_{K+1} \leftarrow \delta \lambda_{K+1}$
end for
$\{\widehat{\boldsymbol{x}}_{tgt}, M_{tgt}\} \leftarrow \text{ProxGrad}(\lambda_{tgt}, \varepsilon, \widehat{\boldsymbol{x}}_N, M_N)$
Return: $\widehat{\boldsymbol{x}}_{tgt}$

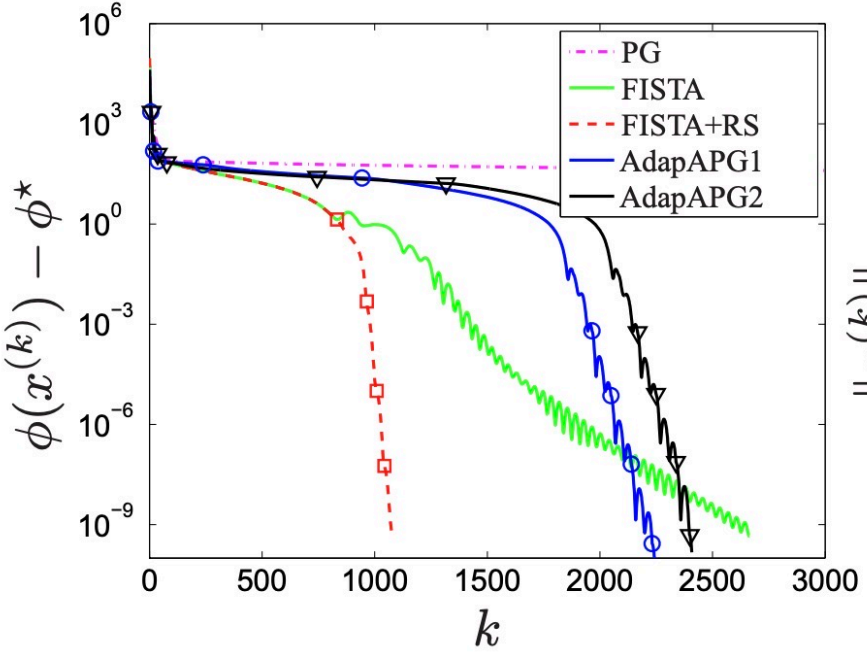
Linear Convergence

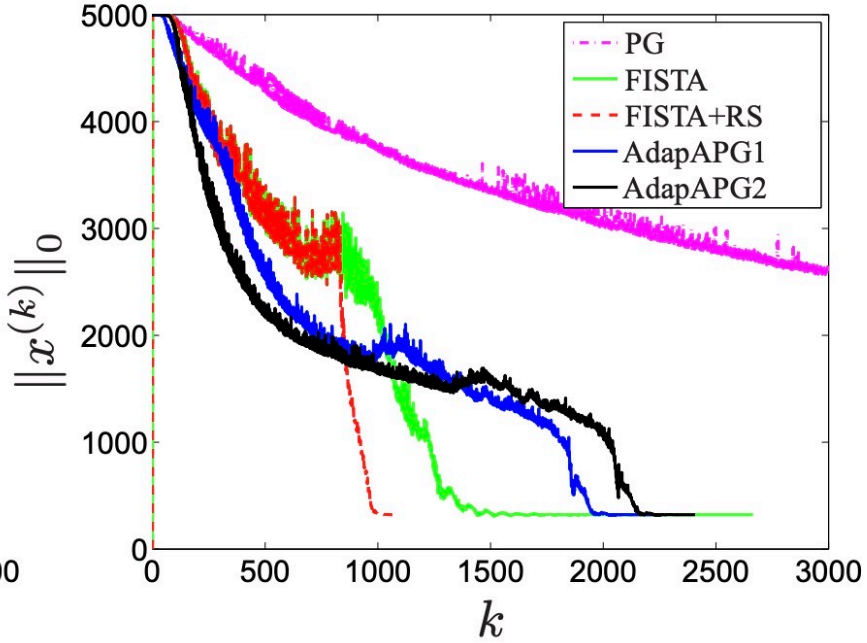
Theorem. (Informal)
Suppose $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ satisfies REC. Under mild assumptions and with properly chosen δ and η, the function value $F(\boldsymbol{x}_k)$ of the proximal gradient homotopy method converges with

$$\#\text{Iter} \leq O\left(\kappa \log\left(\lambda_0/\varepsilon\right)\right),$$

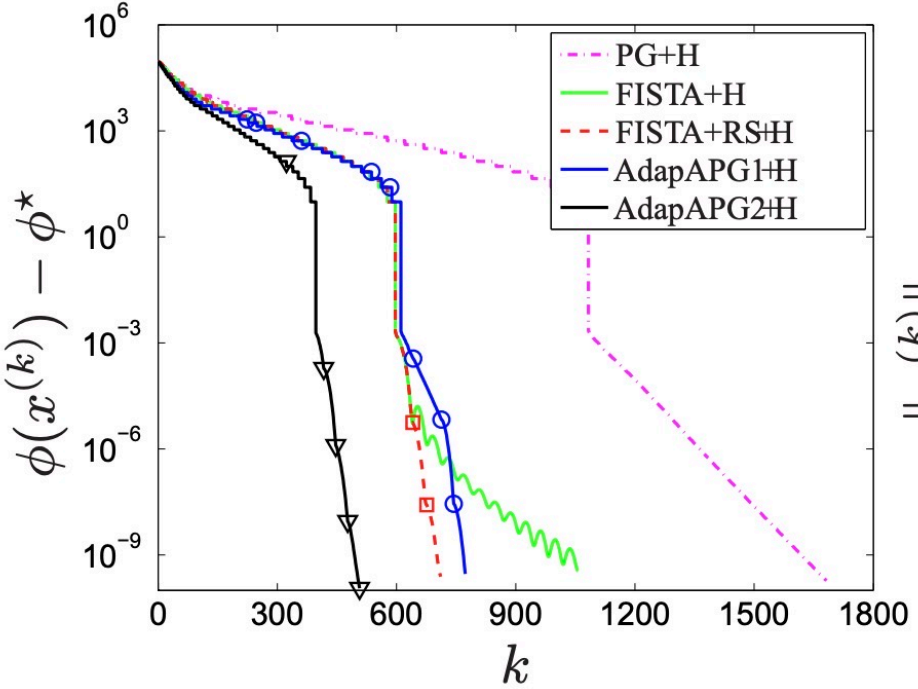
where $\varepsilon > 0$ is the precision and $\kappa := \rho_+(\boldsymbol{A}, s)/\rho_-(\boldsymbol{A}, s)$.

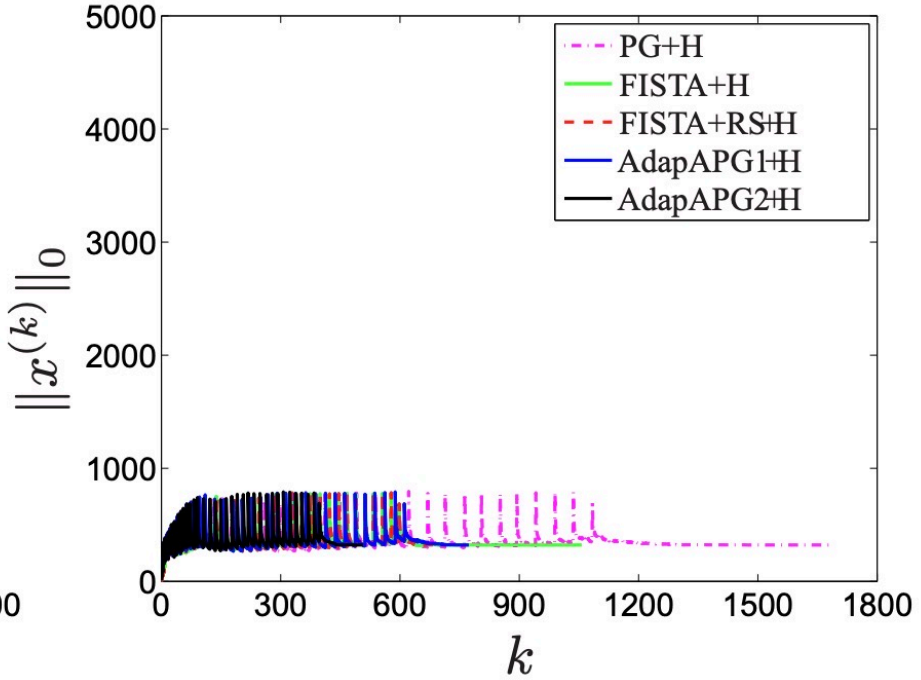
Sublinear Convergence





Linear Convergence with Homotopy





Further Readings

- *Fixed-Point Continuation for ℓ_1-Minimization: Methodology and Convergence.* SIAM Journal on Optimization, Elaine T. Hale, Wotao Yin, and Yin Zhang, 2008. https://locus.siam.org/doi/abs/10.1137/070698920
- *A Proximal-Gradient Homotopy Method for the Sparse Least-Squares Problem.* Lin Xiao, Tong Zhang, 2012. https://arxiv.org/abs/1203.3002
- *An Adaptive Accelerated Proximal Gradient Method and its Homotopy Continuation for Sparse Optimization.* Qihang Lin, Xiao Lin, 2014. https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/apg_homotopy.pdf