Optimal first-order convex minimization methods

with applications to image reconstruction and ML

Donghwan Kim & Jeffrey A. Fessler

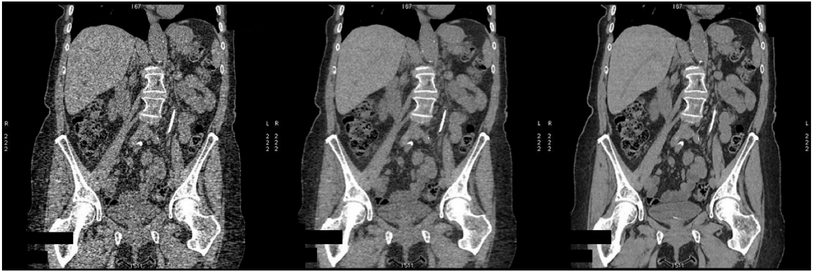
EECS Dept., BME Dept., Dept. of Radiology
University of Michigan

http://web.eecs.umich.edu/~fessler

Cornell ECE

2018-10-15

Lower-dose X-ray CT image reconstruction



Thin-slice FBP	ASIR	Statistical
Seconds	A bit longer	Much longer

Image reconstruction as an optimization problem:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x} \succeq \boldsymbol{0}} \frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_{\boldsymbol{W}}^2 + \mathsf{R}(\boldsymbol{x}),$$

$\boldsymbol{y}$ data, $\boldsymbol{A}$ system model, $\boldsymbol{W}$ statistics, $\mathsf{R}(\boldsymbol{x})$ regularizer.

(Same sinogram, so all at same dose.)

Outline

Optimization problem setting

Standard first-order algorithms

- Gradient descent
- Nesterov's "optimal" first-order method

Optimizing first-order minimization methods

Numerical examples

- Logistic regression for machine learning
- Adaptive restart of OGM
- CT image reconstruction

Generalizing OGM

- Sparsity and constraints
- Dynamic MRI / robust PCA: low-rank + sparse
- Matrix completion

Summary / future work

Outline

Optimization problem setting

$$\hat{\boldsymbol{x}} \in \arg\min_{\boldsymbol{x}} f(\boldsymbol{x})$$

- Unconstrained
- Large-scale (Hessian $\nabla^2 f$ too big to store and/or undefined)
  - image reconstruction / inverse problems
  - big-data / machine learning
  - ...
- Cost function assumptions
  - $f : \mathbb{R}^M \mapsto \mathbb{R}$
  - convex (need not be strictly convex)
  - non-empty set of global minimizers:

$$\hat{\boldsymbol{x}} \in \mathcal{X}^* = \left\{ \boldsymbol{x}_\star \in \mathbb{R}^M : f(\boldsymbol{x}_\star) \leq f(\boldsymbol{x}),\ \forall \boldsymbol{x} \in \mathbb{R}^M \right\}$$

  - smooth (differentiable with L-Lipschitz gradient)

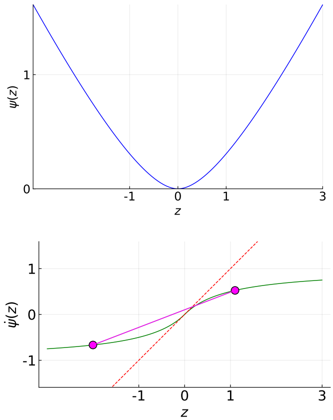
$$\left\| \nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{z}) \right\|_2 \leq L \left\| \boldsymbol{x} - \boldsymbol{z} \right\|_2, \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^M$$

Example: Fair potential function

Fair's potential function [1] (similar to Huber function and hyperbola):

$$\psi(z) = \delta^2 \left[|z/\delta| - \log(1 + |z/\delta|) \right]$$
$$\dot{\psi}(z) = \frac{z}{1 + |z/\delta|}$$
$$\ddot{\psi}(z) = \frac{1}{(1 + |z/\delta|)^2} \leq 1.$$

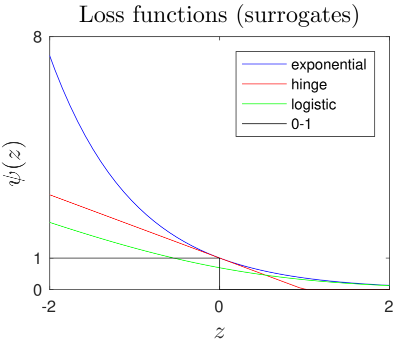
Thus $L = 1$.



Example: Machine learning for classification

To learn weights $\boldsymbol{x}$ of binary classifier given feature vectors $\{\boldsymbol{v}_i\}$ and labels $\{y_i = \pm 1\}$:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}), \qquad f(\boldsymbol{x}) = \sum_i \psi(y_i \langle \boldsymbol{x},\, \boldsymbol{v}_i \rangle)\,.$$



loss functions $\psi(z)$

- 0-1: $\mathbb{I}_{\{z \leq 0\}}$
- exponential: $\exp(-z)$
- logistic: $\log(1 + \exp(-z))$
- hinge: $\max\{0, 1 - z\}$

Which of these ψ fit our conditions?

Outline

Gradient descent

- Problem:
$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}) \,.$$
- Initial guess $\boldsymbol{x}_0$.
- Simple *recursive* iteration:
$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \nabla f(\boldsymbol{x}_n) \,.$$
- Step size $1/L$ ensures monotonic descent of f.
- Telescoping sum (for intuition, not implementation):
$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_0 - \frac{1}{L} \sum_{k=0}^{n} \nabla f(\boldsymbol{x}_k) \,.$$

Gradient descent convergence rate

- Classic $O(1/n)$ convergence rate of cost function descent:

$$\underbrace{f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star)}_{\text{inaccuracy}} \leq \frac{L \left\| \boldsymbol{x}_0 - \boldsymbol{x}_\star \right\|_2^2}{2n}.$$

- Drori & Teboulle (2014) derive tight inaccuracy bound:

$$f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star) \leq \frac{L \left\| \boldsymbol{x}_0 - \boldsymbol{x}_\star \right\|_2^2}{4n + 2}.$$

- They specify a Huber function f for which GD achieves that bound $\Longrightarrow$ case closed for GD with step size $1/L$.
- $O(1/n)$ rate is undesirably slow.

Generalizing GD slightly

- GD with general step size h:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{h}{L} \nabla f(\boldsymbol{x}_n) \,.$$

- Classical monotone descent result:
  $h \in (0,2) \Longrightarrow f(\boldsymbol{x}_{n+1}) < f(\boldsymbol{x}_n)$ when $\boldsymbol{x}_n$ is not a minimizer.
- What is best h?
- If f is quadratic, then *asymptotic* best choice is:

$$h_* = \frac{2L}{\lambda_{\max}(\nabla^2 f) + \lambda_{\min}(\nabla^2 f)}.$$

(Impractical for large-scale problems.)

Generalizing GD slightly

- GD with general step size h:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{h}{L} \nabla f(\boldsymbol{x}_n) .$$

- More generally, Taylor et al. [3] recently (2017) conjectured:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \le \frac{L \left\| \boldsymbol{x}_0 - \boldsymbol{x}_\star \right\|_2^2}{2} \max \left\{ \frac{1}{2Nh+1}, \; (1-h)^{2N} \right\} .$$

  - Proof for $0 < h \le 1$ by Drori and Teboulle, 2014 [2]
  - Upper bounds achieved by a Huber function and by a quadratic function $f(x) = (L/2)x^2$ respectively.
  - Best h depends on N !
    (For $N = 1$, $h_* = 1.5$; for $N = 100$, $h_* = 1.9705$.)
  - Must select N in advance?
  - Still $O(1/N)$...

Heavy ball method and momentum

- Quest for accelerated convergence.
- Heavy ball iteration (Polyak, 1987):

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{\alpha}{L}\nabla f(\boldsymbol{x}_n) + \underbrace{\beta\,(\boldsymbol{x}_n - \boldsymbol{x}_{n-1})}_{\text{momentum!}}$$

(recursive form to implement)

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L}\sum_{k=0}^{n}\underbrace{\alpha\beta^{n-k}}_{\text{coefficients}}\nabla f(\boldsymbol{x}_k)$$

(summation form to analyze)

- How to choose α and β?
- How to optimize coefficients more generally?

General first-order method classes

- General "first-order" (GFO) method:

$$\boldsymbol{x}_{n+1} = \mathrm{function}(\boldsymbol{x}_0, f(\boldsymbol{x}_0), \nabla f(\boldsymbol{x}_0), \ldots, f(\boldsymbol{x}_n), \nabla f(\boldsymbol{x}_n)).$$

- First-order (FO) methods with fixed step-size coefficients:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \sum_{k=0}^{n} h_{n+1,k} \nabla f(\boldsymbol{x}_k).$$

Primary goals:

- Analyze convergence rate of FO for any given $\{h_{n,k}\}$
- Optimize step-size coefficients $\{h_{n,k}\}$
  - fast convergence
  - efficient recursive implementation
  - universal (design *prior* to iterating, independent of L)

GFO vs fixed-step FO

General FO

- Steepest descent (with line search)
- Conjugate gradients
- Quasi-Newton methods
- Barzilai & Borwein method
- any with "backtracking"
- ...

Fixed-step FO

- GD
- Heavy ball method
- Nesterov's fast GM
- OGM
- Proximal methods like ISTA, FISTA, POGM (without back-tracking)
- ...

Nesterov's fast gradient method (FGM1)

Nesterov (1983) iteration: Initialize: $t_0 = 1$, $\boldsymbol{z}_0 = \boldsymbol{x}_0$

$$\boldsymbol{z}_{n+1} = \boldsymbol{x}_n - \frac{1}{L}\nabla f(\boldsymbol{x}_n) \qquad \text{(usual GD update)}$$

$$t_{n+1} = \frac{1}{2}\left(1+\sqrt{1+4t_n^2}\right) \qquad \text{(magic momentum factors)}$$

$$\boldsymbol{x}_{n+1} = \boldsymbol{z}_{n+1} + \frac{t_n - 1}{t_{n+1}}\left(\boldsymbol{z}_{n+1} - \boldsymbol{z}_n\right) \qquad \text{(update with momentum)}\ .$$

Reverts to GD if $t_n = 1, \forall n$.

FGM1 is in class FO: $$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L}\sum_{k=0}^{n} h_{n+1,k}\nabla f(\boldsymbol{x}_k)$$

$$h_{n+1,k} = \begin{cases} \dfrac{t_n - 1}{t_{n+1}} h_{n,k}, & k = 0,\dots,n-2 \\ \dfrac{t_n - 1}{t_{n+1}}\left(h_{n,n-1} - 1\right), & k = n-1 \\ 1 + \dfrac{t_n - 1}{t_{n+1}}, & k = n. \end{cases}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1.25 & 0 & 0 & 0 \\ 0 & 0.10 & 1.40 & 0 & 0 \\ 0 & 0.05 & 0.20 & 1.50 & 0 \\ 0 & 0.03 & 0.11 & 0.29 & 1.57 \end{bmatrix}$$

Nesterov's FGM1 optimal convergence rate

Shown by Nesterov to be $O(1/n^2)$ for "primary" sequence $\{\boldsymbol{z}_n\}$:

$$f(\boldsymbol{z}_n) - f(\boldsymbol{x}_\star) \leq \frac{2L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(n+1)^2}.$$

Nesterov constructed a simple quadratic function f such that, for any general FO method:

$$\frac{\frac{3}{32} L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(n+1)^2} \leq f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star).$$

Thus $O(1/n^2)$ rate of FGM1 is optimal.

New results (Donghwan Kim & JF, 2016):

- Bound on convergence rate of "secondary" sequence $\{\boldsymbol{x}_n\}$:

$$f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star) \leq \frac{2L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(n+2)^2}.$$

- Verifies (numerically inspired) conjecture of Drori & Teboulle (2014).

Outline

Overview

First-order (FO) method with fixed step-size coefficients:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \sum_{k=0}^{n} h_{n+1,k} \nabla f(\boldsymbol{x}_k)$$

- Analyze (*i.e.*, bound) convergence rate as a function of
  - number of iterations N
  - Lipschitz constant L
  - step-size coefficients $H = \{h_{n+1,k}\}$
  - initial distance to a solution: $R = \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$.
- Optimize H by minimizing the bound.
  "Optimizing the optimizer" (meta-optimization?)
- Seek an equivalent recursive form for efficient implementation.

Ideal "universal" bound for first-order methods

For given

- number of iterations N
- Lipschitz constant L
- step-size coefficients $H = \{h_{n+1,k}\}$
- initial distance to a solution: $R = \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|$,

try to bound the worst-case convergence rate of a FO method:

$$B_1(H, R, L, N, M) \triangleq \max_{f \in \mathcal{F}_L} \max_{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \in \mathbb{R}^M} \max_{\substack{\boldsymbol{x}_\star \in \mathcal{X}^*(f) \\ \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\| \leq R}} f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star)$$

$$\text{such that} \quad \boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \sum_{k=0}^{n} h_{n+1,k} \nabla f(\boldsymbol{x}_k), \quad n = 0, \ldots, N-1.$$

Clearly for any FO method, this cost-function bound would hold:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \leq B_1(H, R, L, N, M).$$

Towards practical bounds for first-order methods

For convex functions with L-Lipschitz gradients:

$$\frac{1}{2L}\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{z})\|^2 \leq f(\boldsymbol{x}) - f(\boldsymbol{z}) - \langle \nabla f(\boldsymbol{z}), \boldsymbol{x} - \boldsymbol{z}\rangle, \quad \forall \boldsymbol{x}, \boldsymbol{z} \in \mathbb{R}^M.$$

Drori & Teboulle (2014) use this inequality to propose a "more tractable" (finite-dimensional) relaxed bound:

$$B_2(H, R, L, N, M) \triangleq \max_{\boldsymbol{g}_0,\ldots,\boldsymbol{g}_N \in \mathbb{R}^M} \max_{\delta_0,\ldots,\delta_N \in \mathbb{R}} \max_{\boldsymbol{x}_0,\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N \in \mathbb{R}^M} \max_{\boldsymbol{x}_\star : \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\| \leq R} LR\delta_N^2$$

$$\text{such that} \quad \boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L}\sum_{k=0}^{n} h_{n+1,k} R\, \boldsymbol{g}_k, \quad n = 0, \ldots, N-1,$$

$$\frac{1}{2}\left\|\boldsymbol{g}_i - \boldsymbol{g}_j\right\|^2 \leq \delta_i - \delta_j - \frac{1}{R}\langle \boldsymbol{g}_j, \boldsymbol{x}_i - \boldsymbol{x}_j\rangle, \quad i, j = 0, \ldots, N, *,$$

where $\boldsymbol{g}_n = \frac{1}{LR}\nabla f(\boldsymbol{x}_n)$ and $\delta_n = \frac{1}{LR}\left(f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star)\right)$.
For any FO method:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \leq B_1(H, R, L, N, M) \leq B_2(H, R, L, N, M)$$

However, even B_2 is as of yet unsolved (for general H).

Numerical bounds for first-order methods

- Drori & Teboulle (2014) further relax the bound:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \leq B_1(H, \ldots) \leq B_2(H, \ldots) \leq B_3(H, R, L, N).$$

- For given step-size coefficients H, and given number of iterations N, they use a semi-definite program (SDP) to compute B_3 numerically.
- They find numerically that for the FGM1 choice of H, the convergence bound B_3 is slightly below $\frac{2L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(N+1)^2}$.
- This suggested that improvements on FGM1 could exist.

Optimizing step-size coefficients numerically

Drori & Teboulle (2014) also computed numerically the minimizer over H of their relaxed bound for given N using a SDP:

$$H^* = \arg\min_H B_3(H, R, L, N).$$

Numerical solution for H^* for $N = 5$ iterations: [2, Ex. 3]

0. Input: $f \in C_L^{1,1}(\mathbb{R}^d)$, $x_0 \in \mathbb{R}^d$,
1. $x_1 = x_0 - \frac{1.6180}{L} f'(x_0)$,
2. $x_2 = x_1 - \frac{0.1741}{L} f'(x_0) - \frac{2.0194}{L} f'(x_1)$,
3. $x_3 = x_2 - \frac{0.0756}{L} f'(x_0) - \frac{0.4425}{L} f'(x_1) - \frac{2.2317}{L} f'(x_2)$,
4. $x_4 = x_3 - \frac{0.0401}{L} f'(x_0) - \frac{0.2350}{L} f'(x_1) - \frac{0.6541}{L} f'(x_2) - \frac{2.3656}{L} f'(x_3)$,
5. $x_5 = x_4 - \frac{0.0178}{L} f'(x_0) - \frac{0.1040}{L} f'(x_1) - \frac{0.2894}{L} f'(x_2) - \frac{0.6043}{L} f'(x_3) - \frac{2.0778}{L} f'(x_4)$.

Drawbacks:

- Must choose N in advance
- Requires $O(N)$ memory for all gradient vectors $\{\nabla f(\boldsymbol{x}_n)\}_{n=1}^N$
- $O(N^2)$ computation for N iterations

Benefit: convergence bound (for specific N) $\approx 2\times$ lower than for Nesterov's FGM1.

Analytical solution (D. Kim, JF, 2016)

- Analytical solution for optimized step-size coefficients [8, 9]:

$$H^* \; : \quad h_{n+1,k} = \begin{cases} \frac{\theta_n - 1}{\theta_{n+1}} h_{n,k}, & k = 0, \dots, n-2 \\ \frac{\theta_n - 1}{\theta_{n+1}} \left(h_{n,n-1} - 1 \right), & k = n-1 \\ 1 + \frac{2\theta_n - 1}{\theta_{n+1}}, & k = n. \end{cases}$$

$$\theta_n = \begin{cases} 1, & n = 0 \\ \frac{1}{2}\left(1 + \sqrt{1 + 4\theta_{n-1}^2}\right), & n = 1, \dots, N-1 \\ \frac{1}{2}\left(1 + \sqrt{1 + 8\theta_{n-1}^2}\right), & n = N. \end{cases}$$

- Analytical convergence bound for this optimized H^*:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \le B_3(H^*, R, L, N) = \frac{1 L \left\| \boldsymbol{x}_0 - \boldsymbol{x}_\star \right\|_2^2}{(N+1)(N+1+\sqrt{2})}.$$

  - Of course bound is $O(1/N^2)$, but constant is twice better.
  - No numerical SDP needed $\Longrightarrow$ feasible for large N.
  - (History: sought banded / structured lower-triangular form)

Optimized gradient method (OGM1)

Donghwan Kim & JF (2016) also found efficient recursive iteration:
Initialize: $\theta_0 = 1$, $\boldsymbol{z}_0 = \boldsymbol{x}_0$

$$\boldsymbol{z}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \nabla f(\boldsymbol{x}_n)$$

$$\theta_n = \begin{cases} \frac{1}{2}\left(1 + \sqrt{1 + 4\theta_{n-1}^2}\right), & n = 1, \ldots, N-1 \\ \frac{1}{2}\left(1 + \sqrt{1 + 8\theta_{n-1}^2}\right), & n = N \end{cases}$$

$$\boldsymbol{x}_{n+1} = \boldsymbol{z}_{n+1} + \frac{\theta_n - 1}{\theta_{n+1}} (\boldsymbol{z}_{n+1} - \boldsymbol{z}_n) + \underbrace{\frac{\theta_n}{\theta_{n+1}} (\boldsymbol{z}_{n+1} - \boldsymbol{x}_n)}_{\text{new momentum}}.$$

Reverts to Nesterov's FGM1 by removing the new term.

- Very simple modification of existing Nesterov code.
- No need to solve SDP.
- Factor of 2 better bound than Nesterov's "optimal" FGM1.
- Similar momentum to Güler's 1992 proximal point algorithm [10].

(Proofs omitted.)

Recent refinement of OGM1

New version OGM1' (D. Kim and JF, 2017) [11, 12]:

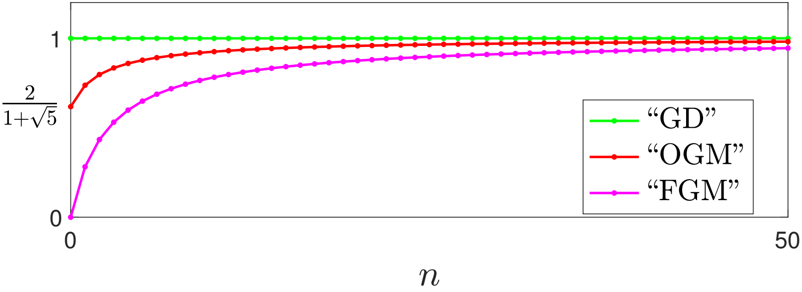
$$\begin{aligned}
\boldsymbol{z}_{n+1} &= \boldsymbol{x}_n - \frac{1}{L} \nabla f(\boldsymbol{x}_n) \\
t_{n+1} &= \frac{1}{2}\left(1 + \sqrt{1 + 4t_n^2}\right) \qquad \text{(momentum factors)} \\
\boldsymbol{x}_{n+1} &= \underbrace{\boldsymbol{x}_n - \frac{1 + t_n/t_{n+1}}{L} \nabla f(\boldsymbol{x}_n)}_{\text{over-relaxed GD}} + \underbrace{\frac{t_n - 1}{t_{n+1}} (\boldsymbol{z}_{n+1} - \boldsymbol{z}_n)}_{\text{FGM momentum}}.
\end{aligned}$$

- New convergence bound for *every iteration*:

$$f(\boldsymbol{z}_n) - f(\boldsymbol{x}_\star) \le \frac{1 L \left\| \boldsymbol{x}_0 - \boldsymbol{x}_\star \right\|_2^2}{(n+1)^2}.$$

- Simpler and more practical implementation.
- Need not pick N in advance.

OGM1’ momentum factors



$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1 + t_n / t_{n+1}}{L} \nabla f(\boldsymbol{x}_n) + \frac{t_n - 1}{t_{n+1}} (\boldsymbol{z}_{n+1} - \boldsymbol{z}_n)$$

Intuition: $1 + t_n / t_{n+1} \to 2$ as $n \to \infty$

Optimized gradient method (OGM) is optimal!

For the class of first-order (FO) methods with fixed step sizes:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \sum_{k=0}^{n} h_{n+1,k} \nabla f(\boldsymbol{x}_k),$$

we optimized OGM and proved the convergence rate upper bound:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) \leq \frac{L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{N^2}.$$

Recently Y. Drori [13] considered the class of general FO methods:

$$\boldsymbol{x}_{n+1} = F(\boldsymbol{x}_0, f(\boldsymbol{x}_0), \nabla f(\boldsymbol{x}_0), \ldots, f(\boldsymbol{x}_n), \nabla f(\boldsymbol{x}_n)),$$

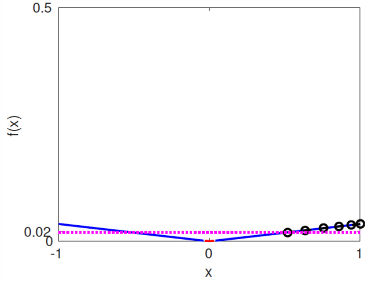
and showed *any* algorithm in this class has a function f such that

$$\frac{L \|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{N^2} \leq f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star),$$

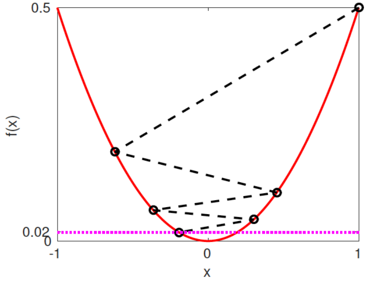
for $d > N$ (large-scale). Thus OGM has optimal (worst-case) complexity among all FO methods, not just fixed-step FO methods!

Worst-case functions for OGM

From (D. Kim and JF, 2017) [11, 12], worst-case behavior is:



(c) $N = 5$: $f_{1,\mathrm{OGM}}\ (\boldsymbol{x};5)$



(d) $N = 5$: $f_2(\boldsymbol{x})$

OGM has two worst-case functions (like GD):
a Huber function and a quadratic function.
Worst-case means:

$$f(\boldsymbol{x}_N) - f(\boldsymbol{x}_\star) = \frac{LR^2}{\theta_N^2} \leq \frac{LR^2}{(N+1)(N+1+\sqrt{2})} \leq \frac{LR^2}{(N+1)^2}.$$

Outline

Machine learning (logistic regression)

To learn weights $\boldsymbol{x} \in \mathbb{R}^N$ of binary classifier $\operatorname{sign}(\langle \boldsymbol{x},\, \boldsymbol{v} \rangle)$ given M feature vectors $\{\boldsymbol{v}_i\} \in \mathbb{R}^N$ and corresponding labels $\{y_i = \pm 1\}$:

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} f(\boldsymbol{x}), \qquad f(\boldsymbol{x}) = \sum_i \psi(y_i \langle \boldsymbol{x},\, \boldsymbol{v}_i \rangle) + \beta \frac{1}{2} \|\boldsymbol{x}\|_2^2 .$$

logistic loss:

$$\psi(z) = \log(1 + \mathrm{e}^{-z}), \quad \dot{\psi}(z) = \frac{-1}{\mathrm{e}^z + 1}, \quad \ddot{\psi}(z) = \frac{\mathrm{e}^z}{(\mathrm{e}^z + 1)^2} \in \left(0, \frac{1}{4}\right].$$

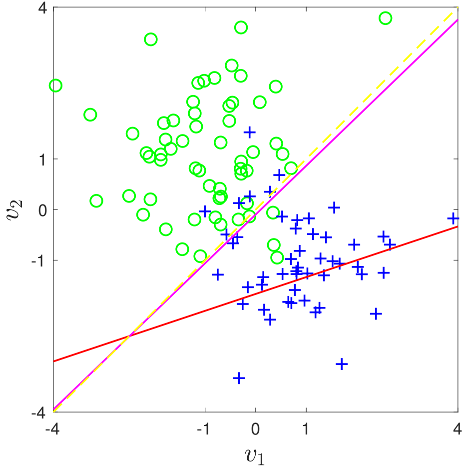
Gradient $\nabla f(\boldsymbol{x}) = \sum_i y_i \, \boldsymbol{v}_i \, \dot{\psi}(y_i \langle \boldsymbol{x},\, \boldsymbol{v}_i \rangle) + \beta \boldsymbol{x}$

Hessian is positive definite so strictly convex:

$$\nabla^2 f(\boldsymbol{x}) = \sum_i \boldsymbol{v}_i \, \ddot{\psi}(y_i \langle \boldsymbol{x},\, \boldsymbol{v}_i \rangle) \, \boldsymbol{v}_i' + \beta \boldsymbol{I} \preceq \frac{1}{4} \sum_i \boldsymbol{v}_i \, \boldsymbol{v}_i' + \beta \boldsymbol{I}$$

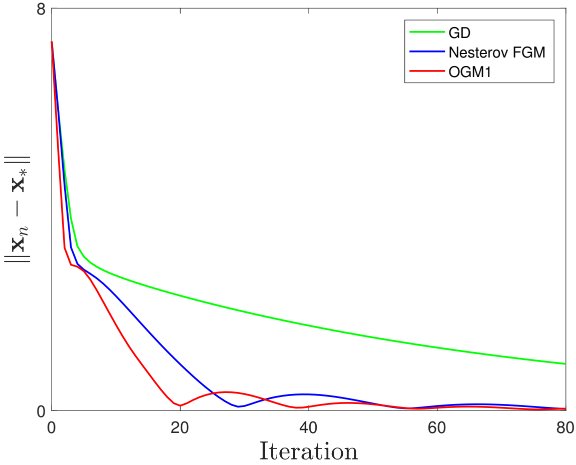
$$\Longrightarrow L \triangleq \frac{1}{4} \rho\left(\sum_i \boldsymbol{v}_i \, \boldsymbol{v}_i' \right) + \beta \geq \max_{\boldsymbol{x}} \rho\left(\nabla^2 f(\boldsymbol{x}) \right)$$

Numerical Results: logistic regression



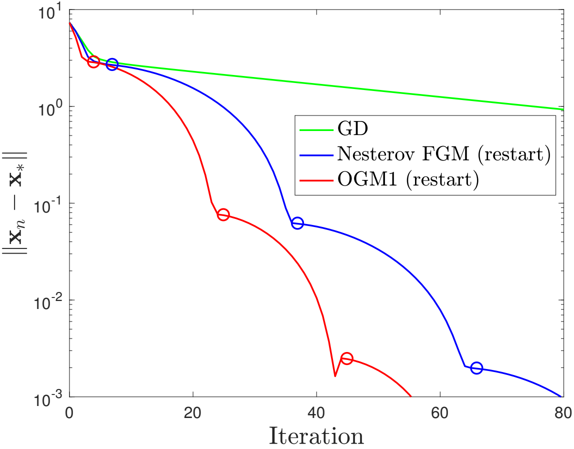
Training data (points); initial decision boundary (red);
final decision boundary (magenta); ideal boundary (yellow).
$M = 100$, $N = 7$ (cf "large scale" ?)

Numerical Results: convergence rates



OGM faster than FGM in early iterations...
by roughly the predicted $\sqrt{2}$ factor

Numerical Results: adaptive restart



FGM restart, O'Donoghue & Candès, 2015.
OGM restart (D. Kim & JF, 2018) [16]

Adaptive restart of OGM

Recall:

$$\boldsymbol{x}_{n+1} = \boldsymbol{x}_n - \frac{1 + t_n/t_{n+1}}{L} \nabla f(\boldsymbol{x}_n) + \frac{t_n - 1}{t_{n+1}} (\boldsymbol{z}_{n+1} - \boldsymbol{z}_n)$$

Heuristic: restart momentum (set $t_n = 1$) if

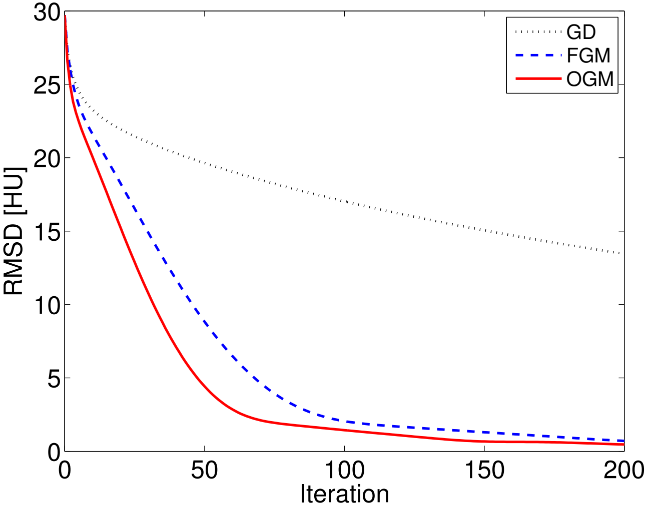
$$\langle -\nabla f(\boldsymbol{x}_n), \boldsymbol{z}_{n+1} - \boldsymbol{z}_n \rangle < 0.$$

But wait, what about optimality?

- worst-case...
- cost functions are often locally *strongly* convex

Formal analysis for strongly convex quadratic functions:
(D. Kim & JF, 2017) [17]

Low-dose 2D X-ray CT image reconstruction simulation



Outline

Combining ordered subsets (OS) with momentum

- Optimization problems in image reconstruction (and machine learning) involve sums of many similar terms:

$$f(\boldsymbol{x}) = \sum_{m=1}^{M} f_m(\boldsymbol{x}) \,.$$

- Approximate gradients using just one term at a time:

$$\nabla f(\boldsymbol{x}) \approx M \nabla f_m(\boldsymbol{x})$$

  - Ordered subsets (OS) in tomography [18]
  - Incremental gradients in optimization / machine learning
- Combining OS with momentum dramatically accelerates!

OS + OGM1 method

Initialize: $\theta_0 = 1$, $\boldsymbol{z}_0 = \boldsymbol{x}_0$ (D. Kim, S. Ramani, JF, 2015) [19]

For each iteration n

For each subset $m = 1, \ldots, M$

$$k = nM + m - 1$$

$$\boldsymbol{z}_{k+1} = \boldsymbol{x}_k - \frac{M}{L} \nabla f_m(\boldsymbol{x}_k) \qquad \text{(usual OS update)}$$

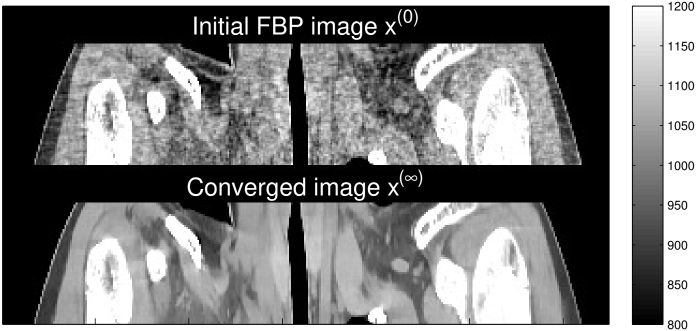
$$\theta_k = \frac{1}{2}\left(1 + \sqrt{1 + 4\theta_{k-1}^2}\right) \qquad \text{(momentum factors)}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{z}_{k+1} + \frac{\theta_k - 1}{\theta_{k+1}} (\boldsymbol{z}_{k+1} - \boldsymbol{z}_k) + \underbrace{\frac{\theta_k}{\theta_{k+1}} (\boldsymbol{z}_{k+1} - \boldsymbol{x}_k)}_{\text{new momentum}}.$$

- Simple modification of existing OS code
- $\approx O(1/(Mn)^2)$ decrease of cost function f in early iterations

Results: 3D X-ray CT patient scan

- 3D cone-beam helical CT scan with pitch 0.5

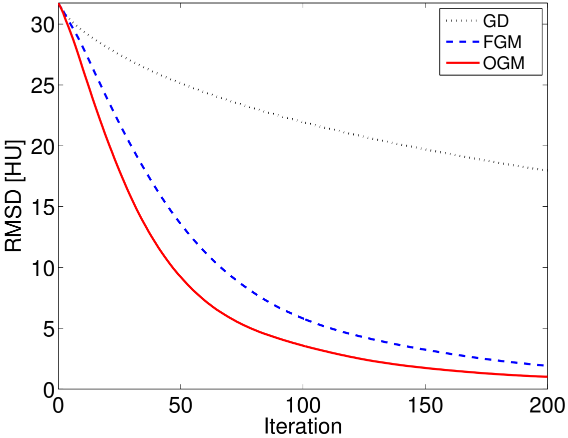


- Convergence rate in RMSD [HU], within ROI, versus iteration:

$$\mathrm{RMSD}_{\mathrm{ROI}}(\boldsymbol{x}_n) \triangleq \frac{||x^{(n)}_{\mathrm{ROI}} - \hat{x}_{\mathrm{ROI}}||_2}{\sqrt{N_{\mathrm{ROI}}}}.$$

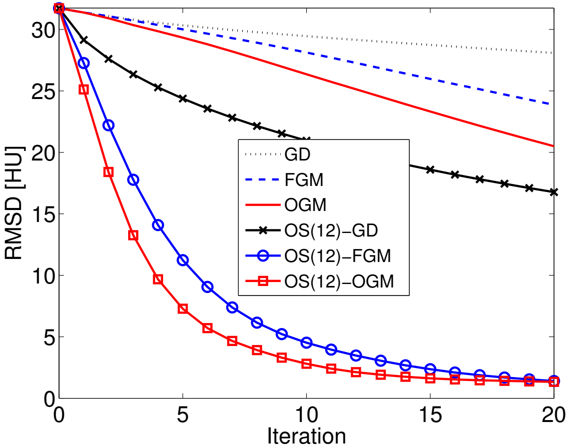
(Disclaimer: RMSD may not relate to task performance...)

Results: RMSD [HU] vs. iteration: without OS



- Computation time: OGM $<$ FGM $\ll$ GD
- OGM requires about $\frac{1}{\sqrt{2}}$-times fewer iterations than FGM to reach the same RMSD.

Results: RMSD [HU] vs. iteration: with OS

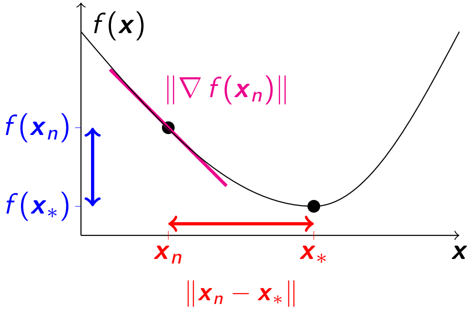


- $M = 12$ subsets in OS algorithm.
- Proposed OS-OGM converges faster than OS-FGM.
- Computation time per iteration of all algorithms are similar.

Outline

OGM bound relates cost function decrease to initial distance:

$$f(\boldsymbol{z}_n) - f(\boldsymbol{x}_\star) \le \frac{L\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{(n+1)^2}.$$



Desiderata:

- $f(\boldsymbol{x}_n) \to f(\boldsymbol{x}_*)$
- $\|\boldsymbol{x}_n - \boldsymbol{x}_*\| \to 0$
- $\|\nabla f(\boldsymbol{x}_n)\| \to 0$

- OGM is one of 3^2 possible "optimal" FO optimizer formulations.
- Taylor et al. explore all 9 for strongly convex functions [20].
- For non-strongly convex cases, 3 of 9 have non-trivial bounds [20, 21].

Generalizing OGM - faster gradient norm decrease

- Cost function decrease: $f(\boldsymbol{x}_n) - f(\boldsymbol{x}_\star) \sim O(1/n^2)$
- Gradient norm decrease? $\|\nabla f(\boldsymbol{x}_n)\| \to 0$ at what rate?

Important especially for problems involving duality.

Bounds on gradient norm decrease

- Known bounds for gradient norm [22] [24]:

$$\text{GD:} \quad \min_{0 \le n \le N} \|\nabla f(\boldsymbol{x}_n)\| = \quad \|\nabla f(\boldsymbol{x}_N)\| \le \frac{\sqrt{2}}{N} LR$$

$$\text{FGM:} \qquad \|\nabla f(\boldsymbol{x}_N)\| \le \frac{2}{N} LR.$$

- New recent bounds (DK & JF, 2016) [25]:

$$\text{FGM:} \quad \min_{0 \le n \le N} \|\nabla f(\boldsymbol{x}_n)\| \le \frac{2\sqrt{3}}{N^{3/2}} LR$$

$$\text{OGM:} \quad \min_{0 \le n \le N} \|\nabla f(\boldsymbol{x}_n)\| \le \|\nabla f(\boldsymbol{x}_N)\| \le \frac{\sqrt{2}}{N} LR.$$

- Can one do better than FGM?

Generalized OGM (GOGM) recursive iteration

Recent generalization (DK & JF, 2016) [25]

Input: $f \in \mathcal{F}_L$, $\boldsymbol{x}_0 \in \mathbb{R}^N$, $\boldsymbol{z}_0 = \boldsymbol{x}_0$, $t_0 \in (0, 1]$.
for $n = 0, 1, \ldots$

$$\boldsymbol{z}_{n+1} = \boldsymbol{x}_n - \frac{1}{L} \nabla f(\boldsymbol{x}_n)$$

$$t_{n+1} > 0 \text{ s.t. } t_{n+1}^2 \leq T_{n+1} \triangleq \sum_{k=0}^{n+1} t_k \qquad \text{(momentum factors)}$$

$$\boldsymbol{x}_{n+1} = \boldsymbol{z}_{n+1} + \frac{(T_n - t_n)t_{n+1}}{T_{n+1}t_n} (\boldsymbol{z}_{n+1} - \boldsymbol{z}_n) + \frac{(2t_n^2 - T_n)t_{n+1}}{T_{n+1}t_n} (\boldsymbol{z}_{n+1} - \boldsymbol{x}_n) \,.$$

- Simple implementation
- Best choice of factors t_n (in terms of gradient norm decrease)?

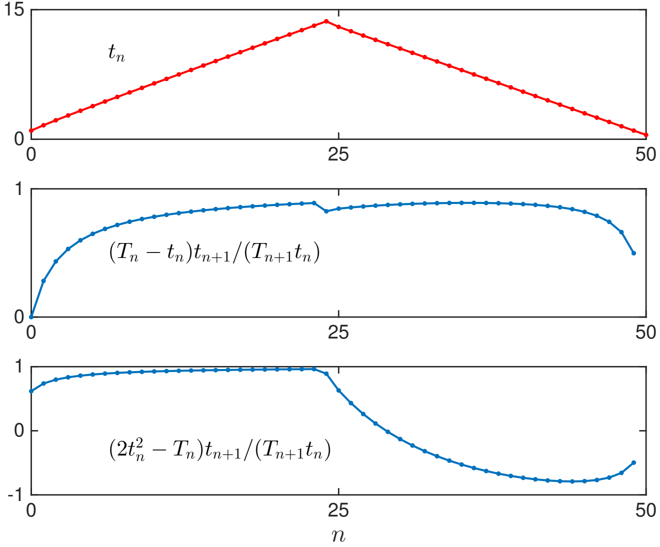
Generalized OGM (GOGM)

Optimized choice of momentum factors (for decreasing gradient norm) (DK & JF, 2016) [25, 26] :

$$t_n \triangleq \begin{cases} 1, & n = 0, \\ \frac{1}{2}\left(1 + \sqrt{1 + 4t_{n-1}^2}\right), & n = 0, \dots, \lfloor N/2 \rfloor - 1, \\ (N - n + 1)/2, & n = \lfloor N/2 \rfloor, \dots N. \end{cases}$$

Dubbed "OGM-OG" for OGM with optimized gradients.

Optimized parameters for OGM-OG



OGM-OG convergence rate bounds

- Convergence bound for cost function for OGM-OG:

$$f(\boldsymbol{z}_N) - f(\boldsymbol{x}_\star) \le \frac{2L\,\|\boldsymbol{x}_0 - \boldsymbol{x}_\star\|_2^2}{N^2}.$$

- Same as Nesterov's FGM.
- Convergence bound for gradient norm is best known among fixed-step FO methods:

$$\min_{0 \le n \le N} \|\nabla f(\boldsymbol{z}_n)\| \le \min_{0 \le n \le N} \|\nabla f(\boldsymbol{x}_n)\| \le \frac{\sqrt{6}}{N^{3/2}} LR.$$

- $\sqrt{2}$ better than FGM's *smallest* gradient norm bound.
- Variations that do not require choosing N in advance, but that have slightly larger constants in bounds.
- Derivation uses relaxations that are not tight.
- Is $N^{3/2}$ best possible? What is best possible constant?

Summary of (fast?) gradient decreasing FO methods

From [25, 26]:

Algorithm	Asymptotic convergence rate bound		Require selecting N in advance
	Cost function	Gradient norm	
GM	$\frac{1}{4}N^{-1}$	$\sqrt{2}N^{-1}$	No
FGM	$2N^{-2}$	$2\sqrt{3}N^{-\frac{3}{2}}$	No
OGM	N^{-2}	$\sqrt{2}N^{-1}$	No
OGM-H	$4N^{-2}$	$4N^{-\frac{3}{2}}$	Yes
OGM-OG	$2N^{-2}$	$\sqrt{6}N^{-\frac{3}{2}}$	Yes
OGM-a $(a > 2)$	$\frac{a}{2}N^{-2}$	$\frac{a\sqrt{6}}{2\sqrt{a-2}}N^{-\frac{3}{2}}$	No
OGM-$a=4$	$2N^{-2}$	$2\sqrt{3}N^{-\frac{3}{2}}$	No

Numerical examples are work-in-progress.
Trade-off between cost function rate and gradient norm rate?

Outline

Non-smooth (composite) convex problems

Composite cost function:

$$\arg\min_{\boldsymbol{x}} F(\boldsymbol{x}), \quad F(\boldsymbol{x}) \triangleq f(\boldsymbol{x}) + g(\boldsymbol{x})$$

$f(\boldsymbol{x})$: convex, smooth with Lipshitz gradient
$g(\boldsymbol{x})$: convex but possibly (usually) non-smooth
Examples:

- $g(\boldsymbol{x}) = \|\boldsymbol{x}\|_1$
- $g(\boldsymbol{x})$ characteristic function of a convex constraint

Fast iterative soft thresholding algorithm (FISTA) (Beck & Teboulle, 2009) [27]
AKA "fast proximal gradient method" (FPGM)
Simple recursive iteration with $O(1/n^2)$ cost function convergence rate

Improving on FISTA

DK & JF, 2016 [28, 29]

Algorithm	Asymptotic convergence rate bound		Require selecting N in advance
	Cost function ($\times LR^2$)	Proximal gradient ($\times LR$)	
PGM	$\frac{1}{2}N^{-1}$	$2N^{-1}$	No
FPGM [5]	$\mathbf{2N^{-2}}$	$2N^{-1}$	No
FPGM-σ $(0 < \sigma < 1)$ [22]	$\frac{2}{\sigma^2}N^{-2}$	$\frac{2\sqrt{3}}{\sigma^2}\sqrt{\frac{1+\sigma}{1-\sigma}}N^{-\frac{3}{2}}$	No
FPGM-$\sigma = 0.78$	$3.3N^{-2}$	$16.2N^{-\frac{3}{2}}$	
FPGM-H	$8N^{-2}$	$5.7N^{-\frac{3}{2}}$	Yes
FPGM-OPG	$4N^{-2}$	$\mathbf{4.9N^{-\frac{3}{2}}}$	Yes
FPGM-a $(a > 2)$	aN^{-2}	$\frac{a\sqrt{6}}{\sqrt{a-2}}N^{-\frac{3}{2}}$	No
FPGM-$a = 4$	$4N^{-2}$	$6.9N^{-\frac{3}{2}}$	

FPGM with "optimized proximal gradient" (FPGM-OPG).
Best known bound on proximal gradient convergence rate, among fixed-step FO methods.

Proximal OGM (POGM)

OGM extension for composite problems by Taylor et al. [20]:

1306 A. B. TAYLOR, J. M. HENDRICKX, F. GLINEUR

Proximal optimized gradient method (POGM)

Input: $F^{(1)} \in \mathcal{F}_{0,L}(\mathbb{E})$, $F^{(2)} \in \mathcal{F}_{0,\infty}(\mathbb{E})$, $x_0 \in \mathbb{E}$, $y_0 = x_0$, $\theta_0 = 1$.

For $k = 1 : N$

$$y_k = x_{k-1} - \frac{1}{L} B^{-1} \nabla F^{(1)}(x_{k-1})$$

$$z_k = y_k + \frac{\theta_{k-1} - 1}{\theta_k}(y_k - y_{k-1}) + \frac{\theta_{k-1}}{\theta_k}(y_k - x_{k-1}) + \frac{\theta_{k-1} - 1}{L\gamma_{k-1}\theta_k}(z_{k-1} - x_{k-1})$$

$$x_k = \operatorname{prox}_{\gamma_k F^{(2)}}(z_k)$$

In this algorithm, we use the sequence $\gamma_k = \frac{1}{L}\frac{2\theta_{k-1}+\theta_k-1}{\theta_k}$ and the inertial coefficients proposed in [23]:

$$\theta_k = \begin{cases} \frac{1+\sqrt{4\theta_{k-1}^2+1}}{2}, & i \le N-1, \\ \frac{1+\sqrt{8\theta_{k-1}^2+1}}{2}, & i = N. \end{cases}$$

Simply trying to generalize OGM using the standard proximal step on the primary sequence $\{y_i\}$ (as for FPGM1) does not lead to a converging algorithm. We obtained

Application: Dynamic MRI image reconstruction

Object model: dynamic image sequence $\boldsymbol{X} = \boldsymbol{L} + \boldsymbol{S}$

- $\boldsymbol{L}$ is low rank
- $\boldsymbol{S}$ is (transform) sparse

Composite cost function for DMRI image reconstruction:

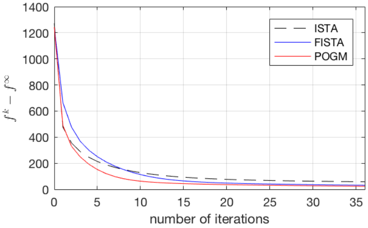
$$\hat{\boldsymbol{X}} = \operatorname*{arg\,min}_{\boldsymbol{L},\boldsymbol{S}} \underbrace{\frac{1}{2} \|\boldsymbol{y} - \boldsymbol{A}\,\mathrm{vec}(\boldsymbol{L}+\boldsymbol{S})\|_2^2}_{f\left(\begin{bmatrix}\boldsymbol{L}\\ \boldsymbol{S}\end{bmatrix}\right)} + \underbrace{\beta_1 |||\boldsymbol{L}|||_* + \beta_2 |||\boldsymbol{T}\boldsymbol{S}|||_1}_{g\left(\begin{bmatrix}\boldsymbol{L}\\ \boldsymbol{S}\end{bmatrix}\right)}$$

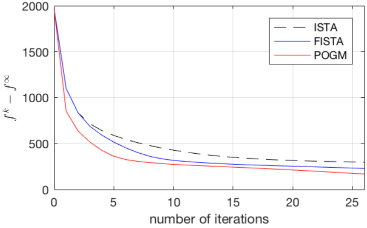
(Akin to robust PCA but with a MRI physics sensing matrix $\boldsymbol{A}$)

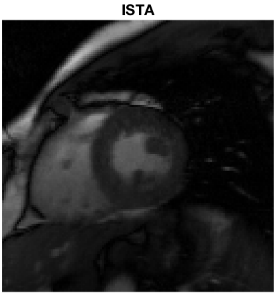
- $f(\boldsymbol{x})$ is smooth with tractable Lipschitz constant
- $g(\boldsymbol{x})$ is convex and non-smooth with simple proximal operations

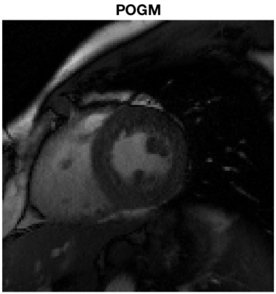
POGM results for dynamic MRI

Claire Lin, ISBI 2018 submission [30]; data from [31]





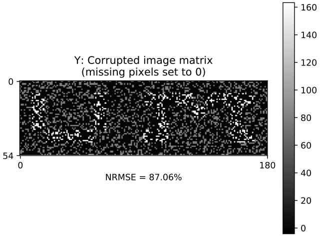


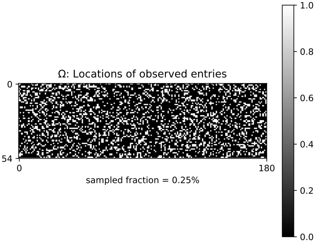


Model: $\boldsymbol{Y} = \boldsymbol{M} \odot (\boldsymbol{X} + \boldsymbol{\varepsilon})$,
$\boldsymbol{M}$: sampling mask
$\boldsymbol{X}$: assumed low-rank latent matrix
$\boldsymbol{\varepsilon}$: noise in measured samples
Matrix completion using Schatten p-norm regularizer with $p = 1/2$:

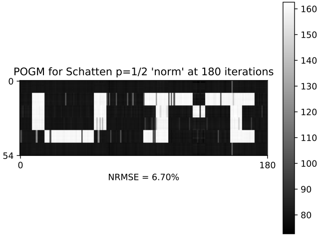
$$\hat{\boldsymbol{X}} = \arg\min_{\boldsymbol{X}} \frac{1}{2} \|\boldsymbol{M} \cdot (\boldsymbol{Y} - \boldsymbol{X})\|_{\mathrm{Frob}}^2 + \beta R(\boldsymbol{X}), \quad R(\boldsymbol{X}) = \sum_k \sigma_k^{1/2}(\boldsymbol{X})$$

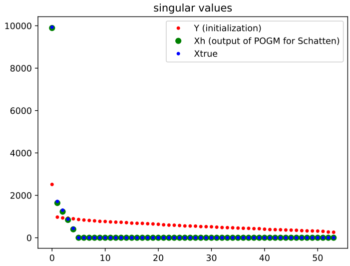
Compromise between $\operatorname{rank}\{\boldsymbol{X}\}$ and nuclear norm $\|\boldsymbol{X}\|_*$
Nonconvex because $p < 1$



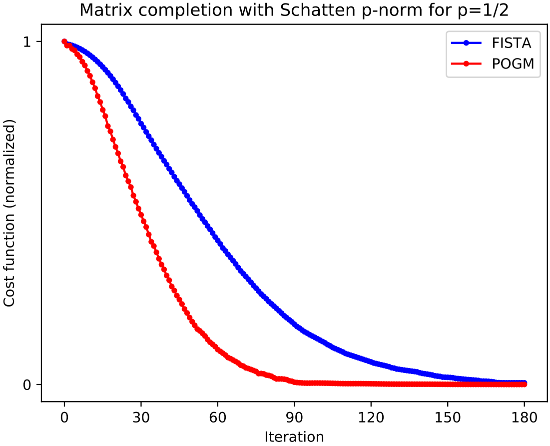


POGM results for matrix completion





POGM converges faster than FISTA



Useful acceleration despite nonconvexity of this matrix completion problem

Convergence bounds are an open problem

Summary

- Optimized first-order minimization algorithm (optimal!)
- Simple implementation akin to Nesterov's FGM
- Analytical converge rate bound
- Bound on cost function decrease is $2\times$ better than Nesterov
- Recent extensions:
  - Adaptive restart
  - Decrease gradient norm
  - Constraints and non-smooth cost functions, *e.g.*, ℓ_1
- Take-away:
  use OGM / POGM instead of Nesterov's FGM / FISTA

Future work

- Tighter bounds
- Strongly convex case
- Nonconvex problems
- Asymptotic / local convergence rates
- Incremental gradients
- Stochastic gradient descent
- Distributed computation
- Cost-function specific algorithms?
  cf. "Learning to optimize" *e.g.*, [32]
- Low-dose 3D X-ray CT image reconstruction

Bibliography I

[1] R. C. Fair. "On the robust estimation of econometric models." In: *Ann. Econ. Social Measurement* 2 (Oct. 1974), 667–77.

[2] Y. Drori and M. Teboulle. "Performance of first-order methods for smooth convex minimization: A novel approach." In: *Mathematical Programming* 145.1-2 (June 2014), 451–82.

[3] A. B. Taylor, J. M. Hendrickx, and Francois Glineur. "Smooth strongly convex interpolation and exact worst-case performance of first- order methods." In: *Mathematical Programming* 161.1 (Jan. 2017), 307–45.

[4] B. T. Polyak. *Introduction to optimization*. New York: Optimization Software Inc, 1987.

[5] J. Barzilai and J. Borwein. "Two-point step size gradient methods." In: *IMA J. Numerical Analysis* 8.1 (1988), 141–8.

[6] Y. Nesterov. "A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$." In: *Dokl. Akad. Nauk. USSR* 269.3 (1983), 543–7.

[7] Y. Nesterov. "Smooth minimization of non-smooth functions." In: *Mathematical Programming* 103.1 (May 2005), 127–52.

[8] D. Kim and J. A. Fessler. *Optimized first-order methods for smooth convex minimization*. 2014.

[9] D. Kim and J. A. Fessler. "Optimized first-order methods for smooth convex minimization." In: *Mathematical Programming* 159.1 (Sept. 2016), 81–107.

[10] O. Güler. "New proximal point algorithms for convex minimization." In: *SIAM J. Optim.* 2.4 (1992), 649–64.

[11] D. Kim and J. A. Fessler. *On the convergence analysis of the optimized gradient methods*. 2015.

[12] D. Kim and J. A. Fessler. "On the convergence analysis of the optimized gradient methods." In: *J. Optim. Theory Appl.* 172.1 (Jan. 2017), 187–205.

Bibliography II

[13] Y. Drori. "The exact information-based complexity of smooth convex minimization." In: *J. Complexity* 39 (Apr. 2017), 1–16.

[14] D. Böhning and B. G. Lindsay. "Monotonicity of quadratic approximation algorithms." In: *Ann. Inst. Stat. Math.* 40.4 (Dec. 1988), 641–63.

[15] B. O'Donoghue and E. Candes. "Adaptive restart for accelerated gradient schemes." In: *Found. Comp. Math.* 15.3 (June 2015), 715–32.

[16] D. Kim and J. A. Fessler. "Adaptive restart of the optimized gradient method for convex optimization." In: *J. Optim. Theory Appl.* 178.1 (July 2018), 240–63.

[17] D. Kim and J. A. Fessler. *Adaptive restart of the optimized gradient method for convex optimization*. 2017.

[18] H. Erdogan and J. A. Fessler. "Ordered subsets algorithms for transmission tomography." In: *Phys. Med. Biol.* 44.11 (Nov. 1999), 2835–51.

[19] D. Kim, S. Ramani, and J. A. Fessler. "Combining ordered subsets and momentum for accelerated X-ray CT image reconstruction." In: *IEEE Trans. Med. Imag.* 34.1 (Jan. 2015), 167–78.

[20] A. B. Taylor, J. M. Hendrickx, and Francois Glineur. "Exact worst-case performance of first-order methods for composite convex optimization." In: *SIAM J. Optim.* 27.3 (Jan. 2017), 1283–313.

[21] A. S. Nemirovsky. "Information-based complexity of linear operator equations." In: *J. of Complexity* 8.2 (1992), 153–75.

[22] Y. Nesterov. *How to make the gradients small*. Optima 88. 2012.

[23] A. Beck and M. Teboulle. "A fast dual proximal gradient algorithm for convex minimization and applications." In: *Operations Research Letters* 42.1 (Jan. 2014), 1–6.

[24] I. Necoara and A. Patrascu. "Iteration complexity analysis of dual first order methods for conic convex programming." In: *Optimization Methods and Software* 31.3 (2016), 645–78.

Bibliography III

[25] D. Kim and J. A. Fessler. "Generalizing the optimized gradient method for smooth convex minimization." In: *SIAM J. Optim.* 28.2 (2018), 1920–50.

[26] D. Kim and J. A. Fessler. *Generalizing the optimized gradient method for smooth convex minimization*. 2016.

[27] A. Beck and M. Teboulle. "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems." In: *IEEE Trans. Im. Proc.* 18.11 (Nov. 2009), 2419–34.

[28] D. Kim and J. A. Fessler. *Another look at the fast iterative shrinkage/thresholding algorithm (FISTA)*. 2016.

[29] D. Kim and J. A. Fessler. "Another look at the Fast Iterative Shrinkage/Thresholding Algorithm (FISTA)." In: *SIAM J. Optim.* 28.1 (2018), 223–50.

[30] C. Y. Lin and J. A. Fessler. "Accelerated methods for low-rank plus sparse image reconstruction." In: *Proc. IEEE Intl. Symp. Biomed. Imag.* 2018, 48–51.

[31] R. Otazo, E. Candes, and D. K. Sodickson. "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components." In: *Mag. Res. Med.* 73.3 (Mar. 2015), 1125–36.

[32] Y. Chen, W. Yu, and T. Pock. "On learning optimized reaction diffusion processes for effective image restoration." In: *Proc. IEEE Conf. on Comp. Vision and Pattern Recognition*. 2015, 5261–9.