ECE 8201: Low-dimensional Signal Models for High-dimensional Data Analysis

Lecture 5: FISTA

Yuejie Chi
The Ohio State University
Reference


See also:


How to solve composite optimization problems?

General composite optimization problem:

\[(\text{COP}) : \hat{x} = \arg\min_x \{ F(x) = f(x) + g(x) \} \]

- \(f(x)\) is convex and differentiable,

- \(g(x)\) is convex, possibly non-differentiable

Examples:

- LASSO: \(f(x) = \frac{1}{2}\|y - Ax\|_2^2\), and \(g(x) = \lambda\|x\|_1\). (focus of this lecture)

- Nuclear norm minimization (later for matrix completion):

\[
f(X) = \|\mathcal{P}_\Omega(Y - X)\|_F^2, \quad g(X) = \lambda\|X\|_*
\]

where \(\|X\|_* = \sum_{i=1}^{\min(m,n)} \sigma_i(X)\), the sum of the singular values of \(X \in \mathbb{R}^{m \times n}\).
Motivation

Standard methods (e.g. subgradient methods) for solving COP has very slow convergence rate (need $O(1/\epsilon^2)$ iterations to reach $\epsilon$ accuracy).

We would discuss an algorithm called FISTA that

- is iterative, and has low computational cost (first-order algorithm, which requires computation of a single gradient per iteration);

- has quadratic convergence rate;

- performs well in practice and works for a large class of problems.

FISTA stands for Fast Iterative Shrinkage-Thresholding Algorithm.
Gradient descent

Consider the unconstrained minimization of a continuously differentiable function \( f(x) \) as

\[
\hat{x} = \arg\min_x f(x)
\]

using gradient descent: start with an initialization \( x_0 \in \mathbb{R}^n \), and iterate

\[
x_k = x_{k-1} - t_k \nabla f(x_{k-1})
\]

where \( t_k \) is a suitable step-size at step \( k \).

**Key observation:** we can view the gradient descent step as solving a *proximal regularization* of the linearized function \( f \) at \( x_{k-1} \),

\[
x_k = \arg\min_x \left\{ f(x_{k-1}) + \langle x - x_{k-1}, \nabla f(x_{k-1}) \rangle + \frac{1}{2t_k} \|x - x_{k-1}\|_2^2 \right\}.
\]
Generalized gradient descent

In the COP,

$$\hat{x} = \arg\min_x f(x) + g(x)$$

we would like to generalize the proximal regularization idea, by extending the update rule as

$$x_k = \arg\min_x \left\{ f(x_{k-1}) + \langle x - x_{k-1}, \nabla f(x_{k-1}) \rangle + \frac{1}{2t_k} \| x - x_{k-1} \|^2 + g(x) \right\}.$$  

This can be simplified (by ignoring constant terms) as

$$x_k = \arg\min_x \left\{ \frac{1}{2t_k} \| x - (x_{k-1} - t_k \nabla f(x_{k-1})) \|^2 + g(x) \right\} \quad (\ast)$$
**Proximal mapping**

**Definition 1.** The proximal mapping (operator) of a convex function $g(x)$ is written as

$$\text{prox}_g(x) = \arg\min_u \left\{ \frac{1}{2} \|u - x\|_2^2 + g(u) \right\}.$$ 

- $g(x) = 0$: $\text{prox}_g(x) = x$.

- $g(x) = I_C(x)$ is an indicator function of a convex set $C$, then

$$\text{prox}_g(x) = \arg\min_{u \in C} \|u - x\|_2^2$$

- $g(x) = \lambda \|x\|_1$: $\text{prox}_g(x)$ is the shrinkage (soft-thresholding) operator and can be decomposed entry-wise:

$$\text{prox}_g(x_i) := T_\lambda(x_i) = \begin{cases} 
    x_i - \lambda, & x_i \geq \lambda \\
    0, & |x_i| < \lambda \\
    x_i + \lambda, & x_i \leq -\lambda
\end{cases}$$
**Generalized gradient descent and ISTA**

- The generalized gradient descent (*) can be regarded as a proximal mapping:

\[
x_k = \underset{x}{\text{argmin}} \left\{ \frac{1}{2t_k} \| x - (x_{k-1} - t_k \nabla f(x_{k-1})) \|_2^2 + g(x) \right\}
\]

\[
= \text{prox}_{t_k g}(x_{k-1} - t_k \nabla f(x_{k-1}))
\]

- When \( f(x) = \frac{1}{2} \| y - Ax \|_2^2 \), and \( g(x) = \lambda \| x \|_1 \), this gives the update rule for ISTA (Iterative Shrinkage-Thresholding Algorithm), or proximal gradient descent:

\[
x_k = \text{prox}_{\lambda t_k \| x \|_1}(x_{k-1} - t_k \nabla f(x_{k-1}))
\]

\[
= \text{prox}_{\lambda t_k \| x \|_1}(x_{k-1} - t_k \nabla f(x_{k-1}))
\]

\[
= T_{\lambda t_k}(x_{k-1} - t_k \nabla f(x_{k-1}))
\]

where \( \nabla f(x_{k-1}) = A^T(Ax - y) \). This can be efficiently computed.
Choice of step size

- Constant step-size: $t_k = t$
- Backtracking line search: start with $t_0$ and do $t = \beta t$ until

$$f(x - t\nabla f(x)) \leq f(x) - \alpha t\|\nabla f(x)\|_2^2$$

with $0 < \alpha, \beta < 1$, e.g. $\alpha = 1/2$. 
Assumptions

- $g : \mathbb{R}^n \rightarrow \mathbb{R}$ is a continuous convex function, possibly nonsmooth;
- $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a smooth convex function that is continuously differentiable with Lipschitz constant:
  \[ \|\nabla f(x) - \nabla f(y)\| \leq L_f \|x - y\|, \quad \forall x, y \in \mathbb{R}^n. \]

Example: For LASSO problems, we have $L_f = \sigma_{\text{max}}(A^T A)$.

- The optimal value of $F = f + g$ is $F^\star$ with optimal solution $x^\star$. 
Theorem 1. [Convergence for generalized gradient descent] Fix step size $t_k = t \leq 1/L$,
\[
F(x_k) - F^* \leq \frac{\|x_0 - x^*\|_2^2}{2tk}
\]

Similar results hold with backtracking for step size.

- Similar to the convergence of gradient descent
- The best possible is $O(1/k^2)$ for first-order methods – can we achieve it?

The answer is yes, with minimal additional computational cost.
Accelerated Gradient Descent

ISTA reaches an accuracy within $O(1/k)$ after $k$ steps; this is not optimal (which is $O(1/k^2)$). The methods of Nesterov meet the optimal bound with the same computational cost (one gradient computation per iteration).

- We will first examine Nesterov’s acceleration method (1983) for smooth convex functions;

- We then extend it to optimizing composite functions, using FISTA (Beck and Teboulle, 2009), which extends Nesterov’s method.
Nesterov’s ACG for convex smooth function

Consider minimizing a convex smooth function $f(x)$ with Lipschitz constant $L$:

$$\hat{x} = \arg\min_x f(x)$$

Nesterov’s Accelerated Gradient Descent performs attains a rate of $O(1/k^2)$. It proceeds as below:

- Start with an initialization $x_0 = x_{-1}$, $\theta_0 = 0$;

- for $k = 1, 2, \ldots$,

$$\theta_k = \frac{1 + \sqrt{1 + 4\theta_{k-1}^2}}{2},$$

$$y_k = x_{k-1} + \left(\frac{\theta_{k-1} - 1}{\theta_k}\right)(x_{k-1} - x_{k-2})$$

$$x_k = y_k - t_k \nabla f(y_k)$$
Remark: other choice of the momentum term with $\theta_k = \frac{k+1}{2}$:

$$y_k = x_{k-1} + \frac{k-2}{k+1}(x_{k-1} - x_{k-2})$$

**Theorem 2. [Nesterov 1983]** *The Nesterov’s AGD satisfies*

$$f(y_k) - f(x^*) \leq \frac{2\|x_0 - x^*\|^2}{Lk^2}$$

Achieves the optimal rate!
The FISTA algorithm with step size $t_k$ (e.g. $t_k = \frac{1}{L}$, where $L_f$ is the Lipschitz constant of $f$):

- **Initialization:** $x_0 = x_{-1} \in \mathbb{R}^n$, $\theta_0 = 1$,

- **For** $k = 1, 2, \ldots$,

\[
\theta_k = \frac{1 + \sqrt{1 + 4\theta_{k-1}^2}}{2}
\]

\[
y_k = x_{k-1} + \left(\frac{\theta_{k-1} - 1}{\theta_k}\right)(x_{k-1} - x_{k-2})
\]

\[
x_k = \text{prox}_{t_k g}(y_k - t_k \nabla f(y_k))
\]

FISTA is computationally efficient when the proximal operator can be computed efficiently (e.g. LASSO).
Interpretation

- first iteration is a proximal gradient step at $y_1 = x_0$

- next iterations are proximal gradient steps at extrapolated points $y_k, k \geq 2$, with the linear combinations carefully chosen.
For LASSO: set \( y_1 = x_0 \in \mathbb{R}^n \), \( \theta_1 = 0 \), and \( t_k = 1/\sigma_{\text{max}}(A^TA) \) (constant step-size), iterate

\[
\theta_k = \frac{1 + \sqrt{1 + 4\theta_{k-1}^2}}{2}
\]

\[
y_k = x_{k-1} + \left( \frac{\theta_{k-1} - 1}{\theta_k} \right) (x_{k-1} - x_{k-2})
\]

\[
x_k = T_{\lambda t_k} \left( y_k - t_k A^T(Ay_k - y) \right)
\]

The main computation cost to apply \( A \) and \( A^T \); no matrix inversion is needed.
Convergence of FISTA

Theorem 3.

\[ F(x_k) - F(x^*) \leq \frac{2L\|x_0 - x^*\|_2^2}{(k+1)^2} \sim O\left(\frac{1}{k^2}\right) \]
Proof of Theorem 3

• Introduce another sequence \( v_k \), which satisfies

\[
v_k := x_{k-1} + \theta_k (x_k - x_{k-1})
\]

\[
y_k = \frac{1}{\theta_k} v_{k-1} + \left(1 - \frac{1}{\theta_k}\right) x_{k-1}
\]

• Two useful facts:

1. \( v_k = v_{k-1} + \theta_k (x_k - y_k) \)

2. \( \left(1 - \frac{1}{\theta_k}\right) \theta_k^2 = \theta_{k-1}^2 \)
Important inequalities

Upper bound of $f$ from Lipschitz property:

$$
\| \nabla f(x) - \nabla f(y) \| \leq L_f \| x - y \|, \quad \forall x, y \in \mathbb{R}^n.
$$

we have

$$
f(y) \leq f(x) + \nabla f(x)^T (y - x) + \frac{L_f}{2} \| y - x \|^2, \quad \forall x, y
$$
Important inequalities

Upper bound of $g$ from definition of proximal operator:

$$g(y) \leq g(z) + \frac{1}{t}(w - y)^T(y - z), \quad \forall w, z, y = \text{prox}_{tg}(w)$$

Proof: since $y = \text{prox}_{tg}(w)$ minimizes $tg(u) + \frac{1}{2}\|w - u\|^2_2$ by definition, we have

$$0 \in t\partial g(y) + (y - w)$$

i.e.

$$\frac{1}{t}(w - y) \in \partial g(y),$$

By the definition of subgradient we have $\forall z,

$$g(z) \geq g(y) + \frac{1}{t}(w - y)^T(z - y)$$
Progress in one iteration

Define \( x^+ = x_k, \ x = x_{k-1}, \ y = y_k, \ \theta = \theta_k, \ v = v_{k-1}, \ v^+ = v_k, \)

- upper bound from Lipschitz property: if \( 0 < t \leq 1/L, \)

\[
f(x^+) \leq f(y) + \nabla f(y)^T(x^+ - y) + \frac{1}{2t}||y - x^+||^2_2
\]

- upper bound from the definition of prox-operator \((x^+ = \text{prox}_{t g}(y - t \nabla f(y))):\)

\[
g(x^+) \leq g(z) + \nabla f(y)^T(z - x^+) + \frac{1}{t}(x^+ - y)^T(z - x^+), \quad \forall z
\]

- add the upper bounds and use convexity of \( f: \)

\[
F(x^+) \leq F(z) + \frac{1}{t}(x^+ - y)^T(z - x^+) + \frac{1}{2t}||y - x^+||^2_2, \quad \forall z
\]
• make convex combination of upper bounds for \( z = x \) and \( z = x^* \):

\[
F(x^+) - F^* - \left( 1 - \frac{1}{\theta} \right) (F(x) - F^*) = F(x^+) - \frac{1}{\theta} F^* - \left( 1 - \frac{1}{\theta} \right) F(x)
\]

\[
\leq \frac{1}{t} (x^+ - y)^T \left( \frac{1}{\theta} x^* + (1 - \frac{1}{\theta}) x - x^+ \right) + \frac{1}{2t} \| y - x^+ \|^2
\]

\[
= \frac{1}{2t} \left( \| y - \frac{1}{\theta} x^* + (1 - \frac{1}{\theta}) x \|^2 - \| x^+ - \frac{1}{\theta} x^* + (1 - \frac{1}{\theta}) x \|^2 \right)
\]

\[
= \frac{1}{2\theta^2 t} \left( \| v - x^* \|^2 - \| v^+ - x^* \|^2 \right)
\]

We now have, at the \( k \)th iteration:

\[
\theta_k^2 \cdot t (F(x_k) - F^*) + \frac{1}{2} \| v_k - x^* \|^2 \leq (\theta_k^2 - \theta_k) \cdot t (F(x_{k-1}) - F^*) + \frac{1}{2} \| v_{k-1} - x^* \|^2
\]

\[
= \theta_{k-1}^2 \cdot t (F(x_{k-1}) - F^*) + \frac{1}{2} \| v_{k-1} - x^* \|^2
\]
Applying the above relationship recursively, we obtain

\[
\theta_k^2 t (F(x_k) - F^*) + \frac{1}{2} \|v_k - x^*\|^2_2 \leq \theta_0^2 t (F(x_0) - F^*) + \frac{1}{2} \|v_0 - x^*\|^2_2 \\
= \frac{1}{2} \|x_0 - x^*\|^2_2
\]

therefore, plug in \(t = \frac{1}{L}\),

\[
F(x_k) - F^* \leq \frac{1}{2\theta_k^2 t} \|x_0 - x^*\|^2_2 \leq \frac{2L}{(k + 1)^2} \|x_0 - x^*\|^2_2.
\]
Alternative formulation:

- Initialization: $y_1 = x_0 \in \mathbb{R}^n$, and $L_f$ is the Lipschitz constant;

- Fix step size $t_k = \frac{1}{L}$.

- For $k = 1, 2, \ldots$,

  \[ x_k = \text{prox}_{t_k g}(y_k - t_k \nabla f(y_k)) \]
  \[ y_{k+1} = x_k + \left( \frac{k - 2}{k + 1} \right) (x_k - x_{k-1}) \]

Convergence speed $O(1/k^2)$ in $k$ steps.
Computational-Statistical Trade-off

If there is indeed a ground truth $x^*$ and we wish $\hat{x}$ is close to $x^*$; we have a sequence of $\{x_k\}$ and hope $x_k$ converges to $\hat{x}$. At a fixed $k$, we may bound

$$\|x_k - x^*\|_2 \leq \underbrace{\|x_k - \hat{x}\|_2}_{\text{computational error}} + \underbrace{\|\hat{x} - x^*\|_2}_{\text{statistical error}}$$

Active research in studying the computational-statistical trade-offs in statistical estimation.