Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problem

Beck Amir, Marcc Teboulle

UBC Okanagan

November 26, 2022

ToC

Sum of Two Functions

Consider a function f that can be written into the sum of two functions.

$$\min_{x} g(x) + h(x) \tag{1}$$

When $g(x)$ is Lipschitz smooth and $h(x)$, closed convex and proper, we can use Beck and Teboulle's FISTA algorithm.

1. The function h can be nonsmooth.
2. Taking the proximal operator of h has to be possible and easy to implement (More on that later).
3. Under the right conditions, the FISTA algorithm converges with $\mathcal{O}(1/k^2)$.

Contributions

1. In Beck's and Teboulle's paper[2], they popularized the use of Nesterov Momentum for nonsmooth functions.
2. Beck proved the convergence with the convexity assumption on g, h.
3. The FISTA algorithm is provably faster than the alternative algorithm: ISTA, TWIST.

A Major Assumption

Assumption (Convex Smooth Nonsmooth with Bounded Minimizers)

We will assume that $g : \mathbb{E} \mapsto \mathbb{R}$ *is* **strongly smooth** *with constant* L_g *and* $h : \mathbb{E} \mapsto \bar{\mathbb{R}}$ **is closed convex and proper**. *We define* $f := g + h$ *to be the summed function and* $ri \circ dom(g) \cap ri \circ dom(h) \neq \emptyset$. *We also assume that a set of minimizers exists for the function* f *and that the set is bounded. Denote the minimizer using* $\bar{x}$.

We refer to this as "**Assumption A.1**".

Lipschitz Smoothness

Definition (Strong Smoothness)

A differentiable function g is called strongly smooth with a constant α then it satisfies:

$$|g(y) - g(x) - \langle \nabla g(x), y - x \rangle| \leq \frac{\alpha}{2}\|x - y\|^2 \quad \forall x, y \in \mathbb{E}. \tag{2}$$

Remark

When g is convex, then the absolute value can be removed; the above condition is equivalent to:

$$\|\nabla g(x) - \nabla g(y)\| \leq \alpha \|y - x\| \quad \forall x, y \in \mathbb{E},$$

we assume $\|\cdot\|$ is the euclidean norm for simplicity. It has been shown in Beck's book, theorem 5.8[1].

Proximal Operator Definition

Definition (The Proximal Operator)

For a function f with $\alpha \geq 0$, the proximal operator is defined as:

$$\operatorname{prox}_{f,\alpha}(x) := \arg\min_{y} \left\{ f(y) + \frac{1}{2\alpha}\|y - x\|^2 \right\}.$$

Remark

The proximal operator is a singled-valued mapping when f is convex, closed, and proper.

Proixmal Operator and Set Projections

Observe that when f is an indicator function δ_Q defined as:

$$\delta_Q(x) := \begin{cases} 0 & x \in Q, \\ \infty & x \notin Q, \end{cases}$$

the proximal operator of δ_Q is

$$\operatorname{prox}_{\delta_Q,\alpha}(x) = \underset{y}{\operatorname{argmin}} \left\{ \delta_Q(y) + \frac{1}{\alpha}\|x - y\|^2 \right\} = \operatorname{argmin}_{y \in Q} \|x - y\|^2,$$

it searches for the closest point to the set Q for all $\alpha > 0$, and it is called a projection. When $Q \neq \emptyset$ is convex and closed, the point is unique.

Example of Prox

Definition (Soft Thresholding)

For some $x \in \mathbb{R}$, the proximal operator of the absolute value is:

$$\text{prox}_{\lambda\|\cdot\|_1,t}(x) = \text{sign}(x) \max(|x| - t\lambda, 0).$$

One could interpret the sign operator as projecting x onto the interval $[-1, 1]$ and the $\max(|x| - t\lambda, 0)$ as the distance of the point x to the interval $[-t\lambda, t\lambda]$.

The Proximal Gradient Algorithm

The Proximal Gradient Method

Algorithm Proximal Gradient With Fixed Step-sizes

```
Input: g, h, smooth and nonsmooth, L stepsize, x^(0) an initial guess of solution.
for k = 1, 2, ..., N do
    x^(k+1) = arg min_y {h(x^(k)) + <∇g(x^(k)), y - x^(k)> + (L/2)||y - x^(k)||^2}
    if x^(k+1), x^(k) close enough then
        Break
    end if
end for
```

1. **Input:** g, h, smooth and nonsmooth, L stepsize, $x^{(0)}$ an initial guess of solution.
2. **for** $k = 1, 2, \cdots, N$ **do**
3. $\quad x^{(k+1)} = \arg\min_y \{h(x^{(k)}) + \langle \nabla g(x^{(k)}), y - x^{(k)} \rangle + \frac{L}{2}\|y - x^{(k)}\|^2\}$
4. $\quad$ **if** $x^{(k+1)}, x^{(k)}$ close enough **then**
5. $\qquad$ **Break**
6. $\quad$ **end if**
7. **end for**

1. It takes the lowest point on the upper bounding function to go next.
2. It is a fixed-point iteration.

The Upper Bounding Function

Observe that when g is Lipschitz smooth with constant L_g then fix any point x and for all $y \in \mathbb{E}$:

$$g(x) + h(x) \le g(x) + \nabla g(x)^T(y - x) + \frac{\beta}{2}\|y - x\|^2 + h(y) =: m_x(y|\beta).$$

Moreover, it has shown that:

$$\underbrace{\text{prox}_{h,\beta^{-1}}(x - \beta^{-1}\nabla g(x))}_{=:\mathcal{P}^{g,h}_{\beta^{-1}}(x)} = \arg\min_{y}\{m_x(y)\}.$$

The proximal gradient algorithm performs fixed point iterations on the proximal gradient operator.

Facts About Proximal Gradient Descent

With our Assumption A.1:

1. It converges monotonically with a $\mathcal{O}(1/k)$ rate as shown in Beck's Boook[1] with a step size L^{-1} such that $L > L_g$.
2. When h is the indicator function, it is just the projected subgradient method. When $h = 0$, this is the smooth gradient descent method where the norm of the fixed point error converges with $\mathcal{O}(1/k)$[1].
3. The fixed point of the proximal gradient operator is the minimizer of f.

The Accelerated Proximal Gradient Method

Momentum Template Method

Algorithm Template Proximal Gradient Method With Momentum

1: **Input:** $x^{(0)}, x^{(-1)}, L, h, g$; 2 initial guesses and stepsize L
2: $y^{(0)} = x^{(0)} + \theta_k(x^{(0)} - x^{(-1)})$
3: **for** $k = 1, \cdots, N$ **do**
4: $\quad x^{(k)} = \text{prox}_{h,L^{-1}}(y^{(k)} + L^{-1}\nabla g(y^{(k)})) = \mathcal{P}^{g,h}_{L^{-1}}(y^{(k)})$
5: $\quad y^{(k+1)} = x^{(k)} + \theta_k(x^{(k)} - x^{(k-1)})$
6: **end for**

In the case of FISTA, we use:

$$t_k = \frac{1 + \sqrt{1 + 4t_k^2}}{2}, \theta_k = \frac{t_k - 1}{t_{k+1}}, t_0 = 1, x^{(-1)} = x^{(0)}$$

Facts about the Accelerated Proximal Gradient Method

1. When $h = 0$, this is Nesterov's famous accelerated gradient method proposed back in 1983.
2. It is no longer a descent method.
3. It has a convergence rate of $\mathcal{O}(1/k^2)$ under Assumption A.1, proved by Beck, Toboulle in the FISTA paper [2].

Some Important Questions to Address for the Second Half

1. Why does the sequence of t_k, θ_k makes sense?
2. What ideas are involved in proving that the convergence rate is $\mathcal{O}(1/k^2)$?
3. If the above is true, what secret sauce cooks up the sequence t_k, θ_k?
    - unfortunately, the secret cause is not the Nesterov momentum term t_k but rather an inequality involving two sequences that give us a bound.

Two Bounded Sequence

The proof for the convergence rate for deriving the momentum sequence hinges on the following lemma about two sequences of numbers:

Lemma (Two Bounded Sequences)

Consider the sequences $a_k, b_k \geq 0$ for $k \in \mathbb{N}$ with $a_1 + b_1 \leq c$. Inductively the two sequences satisfy $a_k - a_{k+1} \leq b_{k+1} - b_k$, which describes a sequence with oscillations bounded by the difference of another sequence. Consider the telescoping sum:

$$\begin{aligned} a_k - a_{k+1} &\geq b_{k+1} - b_k \quad \forall k \in \mathbb{N} \\ \implies -\sum_{k=1}^{N} a_{k+1} - a_k &\geq \sum_{k=1}^{N} b_{k+1} - b_k \\ -(a_{N+1} - a_1) &\geq b_{N+1} - b_1 \\ c \geq a_1 + b_1 &\geq b_{N+1} + a_{N+1} \\ \implies c &\geq a_{N+1}. \end{aligned}$$

The Nesterov Momentum Sequence

In the algorithm we listed, the Nesterov momentum sequence consists of θ_k, t_k with $t_0 = 1$ given by

$$t_k = \frac{1 + \sqrt{1 + 4t_k^2}}{2}, \theta_k = \frac{t_k - 1}{t_{k+1}}, t_0 = 1 \tag{3}$$

It is shown that the sequence of t_k grows linearly and $t_k > (1 + k)/2$.

We Define the Following Quantities

1. $v^{(k)} = x^{(k)} - x^{(k-1)}$ is the velocity term.
2. $\bar{v}^{(k)} = \theta_k v^{(k)}$ is the weighed velocity term.
3. $e^{(k)} := x^{(k)} - \bar{x}$, where $\bar{x} \in \arg\min_x(f(x))$, where $\bar{x}$ is fixed.
4. $\Delta_k := f(x^{(k)}) - f(\bar{x})$ which represent the optimality gap at step k.

We Define the Following Quantities

1. $v^{(k)} = x^{(k)} - x^{(k-1)}$ is the velocity term.
2. $\bar{v}^{(k)} = \theta_k v^{(k)}$ is the weighed velocity term.
3. $e^{(k)} := x^{(k)} - \bar{x}$, where $\bar{x} \in \arg\min_x(f(x))$, where $\bar{x}$ is fixed.
4. $\Delta_k := f(x^{(k)}) - f(\bar{x})$ which represent the optimality gap at step k.

We Define the Following Quantities

1. $v^{(k)} = x^{(k)} - x^{(k-1)}$ is the velocity term.
2. $\bar{v}^{(k)} = \theta_k v^{(k)}$ is the weighed velocity term.
3. $e^{(k)} := x^{(k)} - \bar{x}$, where $\bar{x} \in \arg\min_x(f(x))$, where $\bar{x}$ is fixed.
4. $\Delta_k := f(x^{(k)}) - f(\bar{x})$ which represent the optimality gap at step k.

We Define the Following Quantities

1. $v^{(k)} = x^{(k)} - x^{(k-1)}$ is the velocity term.
2. $\bar{v}^{(k)} = \theta_k v^{(k)}$ is the weighed velocity term.
3. $e^{(k)} := x^{(k)} - \bar{x}$, where $\bar{x} \in \arg\min_x(f(x))$, where $\bar{x}$ is fixed.
4. $\Delta_k := f(x^{(k)}) - f(\bar{x})$ which represent the optimality gap at step k.

Form Matching to the Sequences

From lemma 2.3 in Beck's FISTA paper, one can obtain the following two expressions using the quantities introduced:

Substitute $x = x^{(k)}, y = y^{(k+1)}$ lemma 2.3

$$2L^{-1}(\Delta_k - \Delta_{k+1}) \geq \|v^{(k+1)} - \bar{v}^{(k)}\|^2 + 2\langle v^{(k+1)} - \bar{v}^{(k)}, \bar{v}^{(k)}\rangle \tag{*}$$

Substitute $x = \bar{x}, y = y^{(k+1)}$ lemma 2.3

$$-2L^{-1}\Delta_{k+1} \geq \|v^{(k+1)} - \bar{v}^{(k)}\|^2 + 2\langle v^{(k+1)} - \bar{v}^{(k)}, e^{(k)} + \bar{v}^{(k)}\rangle. \tag{$\star$}$$

The Sequences t_k

For now, we don't know what the sequence t_k is, but we can assume that $t_k > 1$ for all k and using $(*), (\star)$ and consider $(t_{k+1}^2 - t_{k+1})(*) + t_{k+1}(\star)$ which gives us:

$$
\begin{aligned}
&2L^{-1}(\underbrace{(t_{k+1}^2 - t_{k+1})\Delta_k}_{a_k} - \underbrace{t_{k+1}^2\Delta_{k+1}}_{a_{k+1}}) \\
&\geq \text{...Non-Trivial Amount of Math is Skipped...} \\
&\geq \underbrace{\|t_{k+1}v^{(k+1)} + e^{(k)}\|^2}_{b_{k+1}} - \underbrace{\|e^{(k-1)} + (t_{k+1}\theta_k + 1)v^{(k)}\|^2}_{b_k}, \qquad (\star\star)
\end{aligned}
$$

If the form were to match the Two Bounded Sequences, it has to be the case that $t_{k+1}\theta_k + 1 = t_k$ and $t_{k+1}^2 - t_{k+1}^2 = t_k^2$. In this case, the Nesterov Momentum terms satisfy the conditions perfectly.

Using the Two Bounded Sequences

It is not hard to show that the base case $a_1 + b_1 < c$ is bouned using Assumption A1, The Two Bounded Sequences give

$$a_N \leq c \tag{4}$$
$$t_N^2 \Delta_N \leq c \tag{5}$$
$$\Delta_N \leq \frac{c}{t_N^2}, \tag{6}$$

And recall that $t_k \geq (k+1)/2$, we can conclude that Δ_N convergences with rante $\mathcal{O}(1/N^2)$.

Simple LASSO

The Lasso Problem

Lasso minimizes the 2-norm objective with one norm penalty.

$$\min_x \left\{ \frac{1}{2}\|Ax - b\|_2^2 + \lambda\|x\|_1 \right\}$$

And the prox for $\|\cdot\|_1$ is given by:

$$(\text{prox}_{\lambda\|\cdot\|,t}(x))_i = \text{sign}(x_i)\max(|x_i| - t\lambda, 0),$$

For our experiment:

1. A has diagonal elements that are numbers equally spaced on the interval $[0, 2]$.
2. Vector b is the diagonal of A and every odd index is changed into $\epsilon \sim N(0, 10^{-3})$.

Results

The plot of Δ_k:

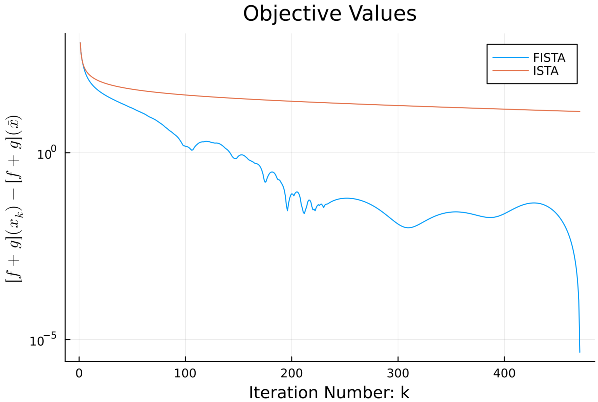
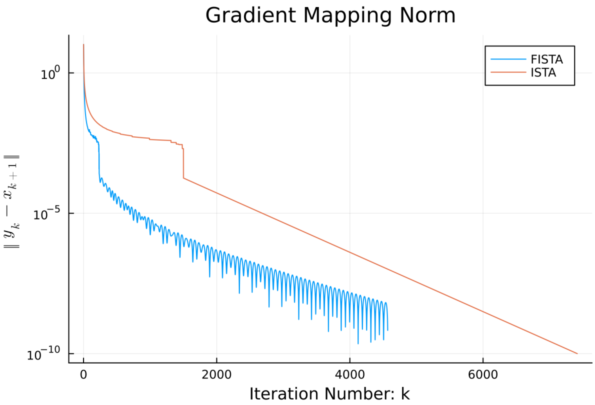


Figure: The left is the objective value of the function during all iterations.

Results

The plot of $\|y^{(k)} - x^{(k+1)}\|_\infty$:



Experiment Setup

Given an image that is convoluted by a Guassian kernel with some guassian noise, we want to recover the image, given the parameters for convolutions.

1. Guassian blur with a discrete 15 by 15 kernel is a linear transform represented by a sparse matrix A in the computer.
2. When an image is 500 by 500 with three color channels, A is 750000×750000.
3. Let the noise be on all normalized colors values with $N(0, 10^{-2})$
4. We let $\lambda = \alpha \times (3 \times 500^2)^{-1}$.
5. Implemented in Julia, the code is too long to be shown here.

The Blurred Image

We consider blurring the image of a pink unicorn that I own.

Figure: The image is blurred by the Gaussian Blurred matrix A with a tiny amount of noise on the level of 2×10^{-2} that is barely observable. Zoom in to observe the tiny amount of Gaussian noise on top of the blur.

Results

(a) (b) (c)

Figure: (a) $\alpha = 0$, without any one norm penalty, is not robust to the additional noise. (b) $\alpha = 0.01$, there is a tiny amount of λ. (c) $\alpha = 0.1$, it is more penalty compared to (a).

References

Amir Beck.
First-Order Methods in Optimization.
Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017.

Amir Beck and Marc Teboulle.
A fast iterative shrinkage-thresholding algorithm for linear inverse problems, 2009.