Proximal Algorithms for Basis Pursuit Denoising

Francis Moran   Ali Zafari

Electrical & Computer Engineering Department
Rutgers University

Convex Optimization, Spring 2024

better late than never.

better late than never.

Proves Wrong!

1. N. Parikh, S. Boyd et al., "Proximal Algorithms," Foundations and Trends® in Optimization, 2014
2. (*FISTA*) A. Beck and M. Teboulle, "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems," SIAM Journal on Imaging Sciences, 2009
3. (*SpaRSA*) S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse Reconstruction by Separable Approximation," IEEE Transactions on Signal Processing, 2009

1. Introduction & Motivation
2. Proximal Algorithms
3. Basis Pursuit DeNoising (BPDN)
4. Accelerated Proximal Methods
5. Numerical Experiments
6. Conclusions
7. References

1. Introduction & Motivation
   - Unconstrained Optimization - Quick Recap
   - Non-smooth Minimization
   - Accelerating Algorithms
2. Proximal Algorithms
3. Basis Pursuit DeNoising (BPDN)
4. Accelerated Proximal Methods
5. Numerical Experiments
6. Conclusions
7. References

1 Introduction & Motivation
Unconstrained Optimization - Quick Recap
Non-smooth Minimization
Accelerating Algorithms

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Unconstrained Optimization - Quick Recap

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

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

- If f is convex and differentiable, $x^\star$ is a global minimizer if and only if $\nabla f(x^\star) = 0$.

Unconstrained Optimization - Quick Recap

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

- If f is convex and differentiable, $x^\star$ is a global minimizer if and only if $\nabla f(x^\star) = 0$.
- If f is convex, any local minimzer is also a global minimizer.

Unconstrained Optimization - Quick Recap

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

- If f is convex and differentiable, $x^\star$ is a global minimizer if and only if $\nabla f(x^\star) = 0$.
- If f is convex, any local minimzer is also a global minimizer.
- If f is strictly convex and has a global minimizer, then it is unique.

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

- If f is convex and differentiable, $x^\star$ is a global minimizer if and only if $\nabla f(x^\star) = 0$.
- If f is convex, any local minimzer is also a global minimizer.
- If f is strictly convex and has a global minimizer, then it is unique.
- If f is strongly convex then it has a global unique minimizer.

Gradient Descent Method

given a starting point $x_0 \in \mathrm{dom} f$;
repeat

$\Delta x_k = -\nabla f(x_k)$;
Line Search. Choose step size α_k;
Update. $x_{k+1} := x_k + \alpha_k \Delta x_k$;

until *stopping criterion is satisfied*;

given a starting point $x_0 \in \mathrm{dom} f$;
repeat
  $\Delta x_k = -\nabla f(x_k)$;
  Line Search. Choose step size α_k;
  Update. $x_{k+1} := x_k + \alpha_k \Delta x_k$;
until *stopping criterion is satisfied*;

- Differentiablity of f is assumed.

Gradient Descent Method

given a starting point $x_0 \in \mathrm{dom} f$;
repeat
  $\Delta x_k = -\nabla f(x_k)$;
  Line Search. Choose step size α_k;
  Update. $x_{k+1} := x_k + \alpha_k \Delta x_k$;
until *stopping criterion is satisfied*;

- Differentiablity of f is assumed.
- Despite slower convergence rate than Newton's method, its simplicity of implementation makes it more desirable.

Theorem (GD Convergence)

Let f be a convex L-smooth function. Suppose that step size $\alpha_k = \frac{1}{L}$ is fixed for all iterations k. Gradient descent optimization yields

$$f_k(x) - f(x^\star) \leq \frac{L}{2k}\|x_0 - x^\star\|_2^2$$

where $x^\star$ is any minizer of f.

Convergence Rate of Gradient Descent

Theorem (GD Convergence)

Let f be a convex L-smooth function. Suppose that step size $\alpha_k = \frac{1}{L}$ is fixed for all iterations k. Gradient descent optimization yields

$$f_k(x) - f(x^\star) \leq \frac{L}{2k} \|x_0 - x^\star\|_2^2$$

where $x^\star$ is any minizer of f.

- In class we saw better rate with additional strong convexity assumption of f.

1 Introduction & Motivation
Unconstrained Optimization - Quick Recap
Non-smooth Minimization
Accelerating Algorithms

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Subgradient Method

- The subgradient method is the non-smooth version of gradient descent. The basic algorithm is straightforward, consisting of the iterations:

$$x_{k+1} = x_k - \alpha_k \Delta x_k$$

where the Δx_k is any member of $\partial f(x_k)$.

- The subgradient method is the non-smooth version of gradient descent. The basic algorithm is straightforward, consisting of the iterations:

$$x_{k+1} = x_k - \alpha_k \Delta x_k$$

where the Δx_k is any member of $\partial f(x_k)$.

Definition (Subgradient)

Subgradient of function f at x is a vector $g \in \mathbb{R}^n$ such that

$$f(y) \geq f(x) + g^T(x - y) \;\; \forall y \in \mathrm{dom} f$$

Collection of subradients at x is called the subdifferential at x denoting as $\partial f(x)$.

Theorem (Subgradient Non-convergence!)

Let f be a convex Lipschitz continuous function with $M > 0$. Suppose that step size $\alpha_k = \alpha > 0$ is fixed for all k. Then

$$f_k^{best} - f(x^\star) \leq \frac{1}{2\alpha k}\|x_0 - x^\star\|_2^2 + \frac{\alpha M^2}{2}$$

Theorem (Subgradient Non-convergence!)

Let f be a convex Lipschitz continuous function with $M > 0$. Suppose that step size $\alpha_k = \alpha > 0$ is fixed for all k. Then

$$f_k^{best} - f(x^\star) \leq \frac{1}{2\alpha k}\|x_0 - x^\star\|_2^2 + \frac{\alpha M^2}{2}$$

- Subgradient method does not guarantee a decrease at each iteration, so we keep the best after k^{th} iteration, f_k^{best}.

Theorem (Subgradient Non-convergence!)

Let f be a convex Lipschitz continuous function with $M > 0$. Suppose that step size $\alpha_k = \alpha > 0$ is fixed for all k. Then

$$f_k^{best} - f(x^\star) \leq \frac{1}{2\alpha k}\|x_0 - x^\star\|_2^2 + \frac{\alpha M^2}{2}$$

- Subgradient method does not guarantee a decrease at each iteration, so we keep the best after k^{th} iteration, f_k^{best}.
- For fixed step size, convergence is not guaranteed.

Theorem (Subgradient Convergence)

Let f be a convex Lipschitz continuous function with $M > 0$. Suppose that step sizes satisfy $\alpha_k \to 0$ as $k \to \infty$ and $\sum_{k=1}^{\infty} \alpha_k = \infty$. Then the achievable rate for a general f is

$$f_k^{best} - f(x^\star) \leq \frac{1}{\sqrt{k}} \|x_0 - x^\star\|_2^2 + Const. \frac{M^2 \log k}{\sqrt{k}}$$

- This convergence rate is slow compared to gradient descent.
- The choice of step size heavily affects the convergence rate of subgradient method.

1 Introduction & Motivation
Unconstrained Optimization - Quick Recap
Non-smooth Minimization
Accelerating Algorithms

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Nestrov's Momentum Based Acceleration

Using the same gradient descent method but with new updates:

$$y_k = x_k + \beta_k(x_k - x_{k-1})$$
$$x_{k+1} = y_k - \alpha_k \nabla f(y_k)$$

where $\beta_k = \frac{k-1}{k+2}$.

Using the same gradient descent method but with new updates:

$$y_k = x_k + \beta_k(x_k - x_{k-1})$$
$$x_{k+1} = y_k - \alpha_k \nabla f(y_k)$$

where $\beta_k = \frac{k-1}{k+2}$.

Theorem (Nestrov's Optimal Method)

Let f be a convex L-smooth function ($L > 0$). Nestrov's updates for suitable choices of step size α_k using gradient-based descent optimization yields:

$$f(x_k) - f(x^\star) \leq \frac{2L}{(k+1)^2} \|x_0 - x^\star\|_2^2$$

1 Introduction & Motivation

2 Proximal Algorithms
- Proximal Map Definition
- Proximal Gradient Descent
- Convergence Analysis

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

1 Introduction & Motivation

2 Proximal Algorithms
Proximal Map Definition
Proximal Gradient Descent
Convergence Analysis

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Proximal Map Definition

Definition (Proximal Map)

The proximal operator $\text{prox}_{\alpha f} : \mathbb{R}^n \to \mathbb{R}^n$ of function $\lambda f : \mathbb{R}^n \to \mathbb{R}$ where $\alpha > 0$:

$$\text{prox}_{\alpha f}(x) = \underset{y}{\text{argmin}} \left(f(y) + \frac{1}{2\alpha} \|y - x\|_2^2 \right)$$

Definition (Proximal Map)

The proximal operator $\text{prox}_{\alpha f} : \mathbb{R}^n \to \mathbb{R}^n$ of function $\lambda f : \mathbb{R}^n \to \mathbb{R}$ where $\alpha > 0$:

$$\text{prox}_{\alpha f}(x) = \underset{y}{\text{argmin}} \left(f(y) + \frac{1}{2\alpha} \|y - x\|_2^2 \right)$$

- The mapping itself includes an optimization problem.

Definition (Proximal Map)

The proximal operator $\text{prox}_{\alpha f} : \mathbb{R}^n \to \mathbb{R}^n$ of function $\lambda f : \mathbb{R}^n \to \mathbb{R}$ where $\alpha > 0$:

$$\text{prox}_{\alpha f}(x) = \underset{y}{\text{argmin}} \left(f(y) + \frac{1}{2\alpha} \|y - x\|_2^2 \right)$$

- The mapping itself includes an optimization problem.
- Based on choice of f, the proximal mapping might have a closed form solution.

Definition (Proximal Map)

The proximal operator $\mathrm{prox}_{\alpha f} : \mathbb{R}^n \to \mathbb{R}^n$ of function $\lambda f : \mathbb{R}^n \to \mathbb{R}$ where $\alpha > 0$:

$$\mathrm{prox}_{\alpha f}(x) = \underset{y}{\mathrm{argmin}} \left(f(y) + \frac{1}{2\alpha} \|y - x\|_2^2 \right)$$

- The mapping itself includes an optimization problem.
- Based on choice of f, the proximal mapping might have a closed form solution.
- f need not be differentiable.

Gradient Descent Steps - Quadratic Upperbounds

Recall: For L-smooth convex function f we can see that gradient descent step minimizes its quadratic upperbound at each iteration:

$$f(x^\star) \leq f(y) \leq f_{qup,x_k}(y) \qquad \forall y \in \mathrm{dom} f$$

Recall: For L-smooth convex function f we can see that gradient descent step minimizes its quadratic upperbound at each iteration:

$$\begin{aligned} f(x^\star) \leq f(y) \leq f_{qup,x_k}(y) \qquad & \forall y \in \mathrm{dom} f \\ = f(x_k) + \nabla f(x_k)^T (y - x_k) + \frac{L}{2} \|y - x_k\|_2^2 & \end{aligned}$$

Recall: For L-smooth convex function f we can see that gradient descent step minimizes its quadratic upperbound at each iteration:

$$\begin{aligned} f(x^\star) \leq f(y) \leq f_{qup,x_k}(y) & \qquad \forall y \in \mathrm{dom} f \\ &= f(x_k) + \nabla f(x_k)^T (y - x_k) + \frac{L}{2}\|y - x_k\|_2^2 \\ &= f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|_2^2 + \frac{L}{2}\|y - x_k + \frac{1}{L}\nabla f(x_k)\|_2^2 \end{aligned}$$

$y = x_k - \frac{1}{L}\nabla f(x_k)$ minimizes the last term.

Recall: For L-smooth convex function f we can see that gradient descent step minimizes its quadratic upperbound at each iteration:

$$
\begin{aligned}
f(x^\star) \le f(y) \le f_{qup,x_k}(y) & \qquad \forall y \in \mathrm{dom} f \\
&= f(x_k) + \nabla f(x_k)^T (y - x_k) + \frac{L}{2}\|y - x_k\|_2^2 \\
&= f(x_k) - \frac{1}{2L}\|\nabla f(x_k)\|_2^2 + \frac{L}{2}\|y - x_k + \frac{1}{L}\nabla f(x_k)\|_2^2
\end{aligned}
$$

$y = x_k - \frac{1}{L}\nabla f(x_k)$ minimizes the last term.

Proximal map showed itself:

$$x_{k+1} = \mathrm{prox}_{\frac{1}{L} f_{lin,x_k}}(x_k) = x_k - \frac{1}{L}\nabla f(x_k)$$

1 Introduction & Motivation

2 Proximal Algorithms
- Proximal Map Definition
- Proximal Gradient Descent
- Convergence Analysis

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Proximal Gradient Descent

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Let's only consider proximal map for linear approximate of g, as we did for gradient descent update:

$$x_{k+1} = \text{prox}_{\alpha_k f_{g_{lin,x_k}}}(x_k)$$

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Let's only consider proximal map for linear approximate of g, as we did for gradient descent update:

$$
\begin{aligned}
x_{k+1} &= \text{prox}_{\alpha_k f_{g_{lin,x_k}}}(x_k) \\
&= \underset{x}{\text{argmin}} \left(g(x_k) + \nabla g(x_k)^T (x - x_k) + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 + h(x) \right)
\end{aligned}
$$

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Let's only consider proximal map for linear approximate of g, as we did for gradient descent update:

$$
\begin{aligned}
x_{k+1} &= \operatorname{prox}_{\alpha_k f_{g_{lin,x_k}}}(x_k) \\
&= \underset{x}{\operatorname{argmin}} \left(g(x_k) + \nabla g(x_k)^T (x - x_k) + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 + h(x) \right) \\
&= \underset{x}{\operatorname{argmin}} \left(-\frac{\alpha_k}{2} \|\nabla g(x_k)\|_2^2 + \frac{1}{2\alpha_k} \|x - x_k + \alpha_k \nabla g(x_k)\|_2^2 + h(x) \right)
\end{aligned}
$$

Proximal Gradient Descent

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Let's only consider proximal map for linear approximate of g, as we did for gradient descent update:

$$
\begin{aligned}
x_{k+1} &= \mathrm{prox}_{\alpha_k f_{g_{lin,x_k}}}(x_k) \\
&= \underset{x}{\mathrm{argmin}} \left(g(x_k) + \nabla g(x_k)^T (x - x_k) + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 + h(x) \right) \\
&= \underset{x}{\mathrm{argmin}} \left(-\frac{\alpha_k}{2} \|\nabla g(x_k)\|_2^2 + \frac{1}{2\alpha_k} \|x - x_k + \alpha_k \nabla g(x_k)\|_2^2 + h(x) \right) \\
&= \underset{x}{\mathrm{argmin}} \left(\frac{1}{2\alpha_k} \|x - x_k + \alpha_k \nabla g(x_k)\|_2^2 + h(x) \right)
\end{aligned}
$$

Proximal Gradient Descent

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Let's only consider proximal map for linear approximate of g, as we did for gradient descent update:

$$
\begin{aligned}
x_{k+1} &= \text{prox}_{\alpha_k f_{g_{lin,x_k}}}(x_k) \\
&= \underset{x}{\text{argmin}} \left(g(x_k) + \nabla g(x_k)^T (x - x_k) + \frac{1}{2\alpha_k} \|x - x_k\|_2^2 + h(x) \right) \\
&= \underset{x}{\text{argmin}} \left(-\frac{\alpha_k}{2} \|\nabla g(x_k)\|_2^2 + \frac{1}{2\alpha_k} \|x - x_k + \alpha_k \nabla g(x_k)\|_2^2 + h(x) \right) \\
&= \underset{x}{\text{argmin}} \left(\frac{1}{2\alpha_k} \|x - x_k + \alpha_k \nabla g(x_k)\|_2^2 + h(x) \right) \\
&= \text{prox}_{\alpha_k h}(x_k - \alpha_k \nabla g(x_k))
\end{aligned}
$$

1 Introduction & Motivation

2 Proximal Algorithms
Proximal Map Definition
Proximal Gradient Descent
Convergence Analysis

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

Proximal Method Convergence Analysis

Theorem (Proximal Convergence)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex. Using fixed step size $\alpha_k = \frac{1}{L}$, and denoting $x^\star$ as the minimizer of f:

$$f(x_k) - f(x^\star) \leq \frac{L}{2k} \|x_0 - x^\star\|_2^2$$

Proximal Method Convergence Analysis

Theorem (Proximal Convergence)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex. Using fixed step size $\alpha_k = \frac{1}{L}$, and denoting $x^\star$ as the minimizer of f:

$$f(x_k) - f(x^\star) \leq \frac{L}{2k}\|x_0 - x^\star\|_2^2$$

- Shining result is that the non-smooth optimization has similar behavior like gradient descent for smooth function. (Much faster than subgradient method!)

Proximal Method Convergence Analysis

Theorem (Proximal Convergence)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex. Using fixed step size $\alpha_k = \frac{1}{L}$, and denoting $x^\star$ as the minimizer of f:

$$f(x_k) - f(x^\star) \leq \frac{L}{2k}\|x_0 - x^\star\|_2^2$$

- Shining result is that the non-smooth optimization has similar behavior like gradient descent for smooth function. (Much faster than subgradient method!)
- Easy computation of prox_h is assumed.

Question to be Answered

Can we accelerate proximal gradient descent?

1 Introduction & Motivation

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

BPDN - Problem Definition

$$\min_{x \in \mathbb{R}^n} \quad \|y - Ax\|_2^2 + \lambda \|x\|_1$$

where $y \in \mathbb{R}^m$ is the measurement signal, $x \in \mathbb{R}^n$ is the signal of interest to be recovered from the measurement y, the matrix $A \in \mathbb{R}^{m \times n}$ is a known sensing matrix (usually $m < n$) and $\lambda \geq 0$.

$$\min_{x \in \mathbb{R}^n} \quad \|y - Ax\|_2^2 + \lambda \|x\|_1$$

where $y \in \mathbb{R}^m$ is the measurement signal, $x \in \mathbb{R}^n$ is the signal of interest to be recovered from the measurement y, the matrix $A \in \mathbb{R}^{m \times n}$ is a known sensing matrix (usually $m < n$) and $\lambda \geq 0$.

- $g := \|.\|_2$ is L-smooth and convex.

BPDN - Problem Definition

$$\min_{x \in \mathbb{R}^n} \quad \|y - Ax\|_2^2 + \lambda \|x\|_1$$

where $y \in \mathbb{R}^m$ is the measurement signal, $x \in \mathbb{R}^n$ is the signal of interest to be recovered from the measurement y, the matrix $A \in \mathbb{R}^{m \times n}$ is a known sensing matrix (usually $m < n$) and $\lambda \geq 0$.

- $g := \|.\|_2$ is L-smooth and convex.
- $h := \lambda\|.\|_1$ is not differentiable but convex.

BPDN - Problem Definition

$$\min_{x \in \mathbb{R}^n} \quad \|y - Ax\|_2^2 + \lambda \|x\|_1$$

where $y \in \mathbb{R}^m$ is the measurement signal, $x \in \mathbb{R}^n$ is the signal of interest to be recovered from the measurement y, the matrix $A \in \mathbb{R}^{m \times n}$ is a known sensing matrix (usually $m < n$) and $\lambda \geq 0$.

- $g := \|.\|_2$ is L-smooth and convex.
- $h := \lambda \|.\|_1$ is not differentiable but convex.
- Proximal map of h is easy to compute (soft thresholding):

$$\text{prox}_h(x) = \operatorname{sign}(x) \max\{|x| - \lambda, 0\}$$

$$\min_{x \in \mathbb{R}^n} \quad \|y - Ax\|_2^2 + \lambda \|x\|_1$$

where $y \in \mathbb{R}^m$ is the measurement signal, $x \in \mathbb{R}^n$ is the signal of interest to be recovered from the measurement y, the matrix $A \in \mathbb{R}^{m \times n}$ is a known sensing matrix (usually $m < n$) and $\lambda \geq 0$.

- $g := \|.\|_2$ is L-smooth and convex.
- $h := \lambda \|.\|_1$ is not differentiable but convex.
- Proximal map of h is easy to compute (soft thresholding):

$$\text{prox}_h(x) = \operatorname{sign}(x) \max\{|x| - \lambda, 0\}$$

- Minimizing this problem using proximal gradient descent is called Iterative Shrinkage-Thresholding Algorithm (ISTA) in the literature.

1 Introduction & Motivation

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods
    FISTA
    SpaRSA

5 Numerical Experiments

6 Conclusions

7 References

1. Introduction & Motivation
2. Proximal Algorithms
3. Basis Pursuit DeNoising (BPDN)
4. Accelerated Proximal Methods
    - FISTA
    - SpaRSA
5. Numerical Experiments
6. Conclusions
7. References

Fast ISTA - Core Idea

- The idea stems from momentum-based acceleration we saw for gradient descent method.
- Current estimates are updated based on history of two previous updates.
- Convergence rate has been derived analytically and shows its proven advantage over the regular proximal gradient descent method.

FISTA - Algorithm

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

given $y_1 = x_0 \in \mathbb{R}^n, t_1 = 1$;
repeat

$$x_k = \text{prox}_{\frac{1}{L}h}(y_k - \tfrac{1}{L}\nabla g(y_k));$$

$$t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2};$$

$$y_{k+1} = x_k + \frac{t_k - 1}{t_{k+1}}(x_k - x_{k-1});$$

until *stopping criterion is satisfied*;

FISTA - Algorithm

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

given $y_1 = x_0 \in \mathbb{R}^n, t_1 = 1$;
repeat

$$x_k = \text{prox}_{\frac{1}{L}h}(y_k - \tfrac{1}{L}\nabla g(y_k));$$
$$t_{k+1} = \tfrac{1+\sqrt{1+4t_k^2}}{2};$$
$$y_{k+1} = x_k + \tfrac{t_k - 1}{t_{k+1}}(x_k - x_{k-1});$$

until *stopping criterion is satisfied*;

- Main difference to ISTA is that the proximal operator is evaluated at a linear combination of two previous points instead of only the last one.

Theorem (FISTA Convergence)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex. Using fixed step size $\alpha_k = \frac{1}{L}$, and denoting $x^\star$ as the minimizer of f:

$$f(x_k) - f(x^\star) \leq \frac{2L}{(k+1)^2} \|x_0 - x^\star\|_2^2$$

Theorem (FISTA Convergence)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex. Using fixed step size $\alpha_k = \frac{1}{L}$, and denoting $x^\star$ as the minimizer of f:

$$f(x_k) - f(x^\star) \leq \frac{2L}{(k+1)^2}\|x_0 - x^\star\|_2^2$$

- For regular proximal gradient (*e.g.*, ISTA) we had:

$$f(x_k) - f(x^\star) \leq \frac{L}{2k}\|x_0 - x^\star\|_2^2$$

1 Introduction & Motivation

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods
- FISTA
- SpaRSA

5 Numerical Experiments

6 Conclusions

7 References

SpaRSA - Core Idea

- The ISTA algorithm is modified heuristically to speed up convergence.
- The step size α_k is chosen heuristically based on a Barzilai-Borwein Spectral Method.
- As SpaRSA tends to slow down extremely when λ is small, a method, called "continuation" is used to avoid that, by warm-starting the from the solution of the problem for a larger value of λ.

- It is a gradient method with step sizes inspired by Newton's method but without involving the Hessian.

Barzilai-Borwein Spectral Method

- It is a gradient method with step sizes inspired by Newton's method but without involving the Hessian.
- Barzilai-Borwein approach chooses α'_k where $\alpha'_k I_{n\times n}$ be closest to the Hessian of f over the last step.

- It is a gradient method with step sizes inspired by Newton's method but without involving the Hessian.
- Barzilai-Borwein approach chooses α'_k where $\alpha'_k I_{n\times n}$ be closest to the Hessian of f over the last step.
- With $r_k := \nabla f(x_k) - \nabla f(x_{k-1})$ and $s_k := x_k - x_{k-1}$, we find α'_k such that $\alpha'_k s_k \approx r_k$, *i.e.*,

$$\alpha'_k = \underset{\alpha'}{\operatorname{argmin}} \|\alpha' s_k - r_k\|_2^2 = \frac{s_k^T r_k}{s_k^T s_k}$$

(Notation: α'_k is playing the role of $\frac{1}{\alpha_k}$ in previous slides)

Consider $f(x) = g(x) + h(x)$ where g is L-smooth and convex and h is convex.

Given

$\eta > 1, [0 < \alpha'_{min} < \alpha'_{max}], x_0 \in \mathbb{R}^n$

repeat

  $\alpha'_k \in [\alpha'_{min}, \alpha'_{max}]$; (Safeguarded Barzilai-Borwein)

  repeat

    $x_{k+1} = \text{prox}_{\frac{1}{\alpha'_k} h}(x_k - \frac{1}{\alpha'_k} \nabla g(x_k))$;

    $\alpha'_{k+1} \leftarrow \eta \alpha'_k$;

  until x_{k+1} *satisfies an acceptance criterion*;

  $k \leftarrow k + 1$;

until *stopping criterion is satisfied*;

SpaRSA - Cont'd

- Acceptence Criterion of x_{k+1}

SpaRSA - Cont'd

- Acceptence Criterion of x_{k+1}
  - Naive solution: to accept any solution of the proximal mapping. No monotone convergence is guaranteed.

SpaRSA - Cont'd

- Acceptence Criterion of x_{k+1}
  - Naive solution: to accept any solution of the proximal mapping. No monotone convergence is guaranteed.
  - Accept if objective value is slightly smaller than the maximum of M past iterations:

$$f(x_{k+1}) \leq \max_{i \in \{\text{M-last iterations}\}} f(x_i) - \frac{\sigma}{2} \alpha'_k \|x_{k+1} - x_k\|_2^2$$

where $\sigma \in (0, 1)$. Although again no monotone convergence is guaranteed, but in practice it performs well.

- Acceptence Criterion of x_{k+1}
  - Naive solution: to accept any solution of the proximal mapping. No monotone convergence is guaranteed.
  - Accept if objective value is slightly smaller than the maximum of M past iterations:

$$f(x_{k+1}) \leq \max_{i \in \{\text{M-last iterations}\}} f(x_i) - \frac{\sigma}{2}\alpha'_k \|x_{k+1} - x_k\|_2^2$$

    where $\sigma \in (0, 1)$. Although again no monotone convergence is guaranteed, but in practice it performs well.
- Stopping Criterion of Algorithm

- Acceptence Criterion of x_{k+1}
  - Naive solution: to accept any solution of the proximal mapping. No monotone convergence is guaranteed.
  - Accept if objective value is slightly smaller than the maximum of M past iterations:

$$f(x_{k+1}) \leq \max_{i \in \{\text{M-last iterations}\}} f(x_i) - \frac{\sigma}{2}\alpha'_k \|x_{k+1} - x_k\|_2^2$$

    where $\sigma \in (0, 1)$. Although again no monotone convergence is guaranteed, but in practice it performs well.
- Stopping Criterion of Algorithm
  - As simple as comparing the relative change in the objective function with a small fixed value $\mathrm{tol} > 0$:

$$\frac{|f(x_{k+1}) - f(x_k)|}{f(x_k)} \leq \mathrm{tol}$$

1 Introduction & Motivation

2 Proximal Algorithms

3 Basis Pursuit DeNoising (BPDN)

4 Accelerated Proximal Methods

5 Numerical Experiments

6 Conclusions

7 References

- BPDN problem where signal of interest x is the wavelet coefficients of the image.
- Image passed through Gaussian blur with variance of 3

Original

Blurred

Figure 1: Original and Blurred Cameraman Image.

Figure 2: Recovered Images from each Algorithm.

Image Deblurring

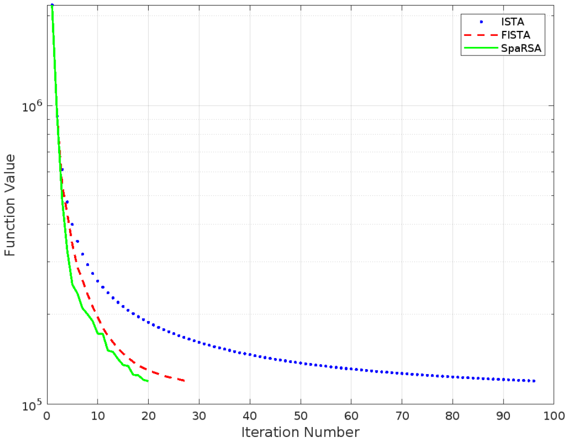

Figure 3: Objective Value vs Iteration Number of ISTA, FISTA, and SpaRSA

1. Introduction & Motivation
2. Proximal Algorithms
3. Basis Pursuit DeNoising (BPDN)
4. Accelerated Proximal Methods
5. Numerical Experiments
6. Conclusions
7. References

Conclusions & Learning Outcomes

- Exciting thing: the type of analysis we learned over the semester are so common over these very technical publications.

Conclusions & Learning Outcomes

- Exciting thing: the type of analysis we learned over the semester are so common over these very technical publications.
- Proximal methods are like the gradient method for non-smooth objective functions, although with having in mind to have a simple proximal map.

Conclusions & Learning Outcomes

- Exciting thing: the type of analysis we learned over the semester are so common over these very technical publications.
- Proximal methods are like the gradient method for non-smooth objective functions, although with having in mind to have a simple proximal map.
- Although FISTA provides a very well-established convergence analysis, SpaRSA presents better results in practice.

Conclusions & Learning Outcomes

- Exciting thing: the type of analysis we learned over the semester are so common over these very technical publications.
- Proximal methods are like the gradient method for non-smooth objective functions, although with having in mind to have a simple proximal map.
- Although FISTA provides a very well-established convergence analysis, SpaRSA presents better results in practice.
- Both FISTA and SpaRSA are general accelerated proximal gradient descent methods, not limited to BPDN problem.

1. Introduction & Motivation

2. Proximal Algorithms

3. Basis Pursuit DeNoising (BPDN)

4. Accelerated Proximal Methods

5. Numerical Experiments

6. Conclusions

7. References

[1] N. Parikh, S. Boyd *et al.*, "Proximal algorithms," *Foundations and trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[2] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM journal on imaging sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[3] S. J. Wright, R. D. Nowak, and M. A. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Transactions on signal processing*, vol. 57, no. 7, pp. 2479–2493, 2009.

[4] S. P. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[5] S. Boyd, L. Xiao, and A. Mutapcic, "Subgradient methods," *lecture notes of EE392o, Stanford University, Autumn Quarter*, vol. 2004, no. 01, 2003.

[6] J. Romberg and M. Davenport, "Lecture notes of ece6270," *Georgia Institute of Technology, Fall Semester*, 2022.

[7] S. Becker, J. Bobin, and E. J. Candès, "Nesta: A fast and accurate first-order method for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 1–39, 2011.

[8] W. W. Hager, D. T. Phan, and H. Zhang, "Gradient-based methods for sparse recovery," *SIAM Journal on Imaging Sciences*, vol. 4, no. 1, pp. 146–165, 2011.

[9] W. Yin, "Math 164: Optimization barzilai-borwein method," *University of California, Los Angeles, Spring Semester*, 2015.