Fast Iterative Shrinkage-Thresholding Algorithm (FISTA): A Comprehensive Study of Accelerated Proximal Gradient Methods

Lecture Notes

July 17, 2025

Contents

1 Introduction and Motivation

1.1 Overview of Sparse Optimization

The Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) represents a significant advancement in solving composite optimization problems, particularly those involving sparsity-inducing regularizers. This document provides a comprehensive treatment of the theoretical foundations, algorithmic development, and practical implementation of FISTA.

The fundamental optimization problem we consider takes the form:

$$\min_{\mathbf{x}\in\mathbb{R}^n} F(\mathbf{x}) = f(\mathbf{x}) + g(\mathbf{x}) \tag{1}$$

where:

- $f : \mathbb{R}^n \to \mathbb{R}$ is a smooth, convex function with Lipschitz continuous gradient
- $g : \mathbb{R}^n \to \mathbb{R}$ is a convex, possibly non-smooth regularization term
- The composite function F captures both data fidelity and structural constraints

1.2 The ℓ_1-Regularized Least Squares Problem

A canonical instance of (1) arises in sparse signal recovery and compressed sensing:

$$\min_{\mathbf{x}\in\mathbb{R}^n} \left\{ \frac{1}{2} \left\|\mathbf{A}\mathbf{x} - \mathbf{b}\right\|_2^2 + \lambda \left\|\mathbf{x}\right\|_1 \right\} \tag{2}$$

Here, the objective function comprises:

1. **Data fidelity term**: $f(\mathbf{x}) = \frac{1}{2} \left\|\mathbf{A}\mathbf{x} - \mathbf{b}\right\|_2^2$
   - $\mathbf{A} \in \mathbb{R}^{m\times n}$ represents the measurement or design matrix
   - $\mathbf{b} \in \mathbb{R}^m$ denotes the observed data vector
   - This quadratic term quantifies the discrepancy between model predictions and observations
2. **Regularization term**: $g(\mathbf{x}) = \lambda \left\|\mathbf{x}\right\|_1 = \lambda \sum_{i=1}^n |x_i|$
   - $\lambda > 0$ is the regularization parameter controlling sparsity
   - The ℓ_1 norm promotes sparse solutions by encouraging many components to be exactly zero

1.3 Comparison of Regularization Norms

Norm	Definition	Properties	Optimization	Applications
ℓ_0	$\|\mathbf{x}\|_0 = \|\{i : x_i \neq 0\}\|$	Non-convex, discontinuous	NP-hard	Exact sparsity
ℓ_1	$\|\mathbf{x}\|_1 = \sum_i \|x_i\|$	Convex, non-smooth	Tractable	Convex relaxation
ℓ_2	$\|\mathbf{x}\|_2 = \sqrt{\sum_i x_i^2}$	Convex, smooth	Closed-form	Ridge regression

Table 1: Comparison of commonly used regularization norms in optimization

2 Proximal Gradient Methods

2.1 Limitations of Classical Gradient Descent

The non-smoothness of the ℓ_1 norm presents a fundamental challenge for classical optimization methods. Consider the subdifferential of the ℓ_1 norm at a point $\mathbf{x}$:

$$\partial \left\|\mathbf{x}\right\|_1 = \left\{ \mathbf{v} \in \mathbb{R}^n : v_i \in \begin{cases} \{1\} & \text{if } x_i > 0 \\ \{-1\} & \text{if } x_i < 0 \\ [-1, 1] & \text{if } x_i = 0 \end{cases} \right\} \tag{3}$$

The multi-valued nature of the subdifferential at $x_i = 0$ precludes the direct application of gradient descent, necessitating more sophisticated approaches.

2.2 The Proximal Mapping

Definition 2.1 (Proximal Operator). *For a convex function $h : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, the proximal operator is defined as:*

$$\operatorname{prox}_h(\mathbf{v}) = \arg\min_{\mathbf{u} \in \mathbb{R}^n} \left\{ h(\mathbf{u}) + \frac{1}{2} \left\|\mathbf{u} - \mathbf{v}\right\|_2^2 \right\} \tag{4}$$

The proximal operator can be interpreted as:

- A generalization of orthogonal projection onto convex sets
- A trade-off between minimizing h and staying close to $\mathbf{v}$
- An implicit gradient step that handles non-smoothness

2.3 Proximal Operator for ℓ_1 Regularization

Theorem 2.2 (Soft Thresholding). *The proximal operator of $h(\mathbf{x}) = \lambda \left\|\mathbf{x}\right\|_1$ is given component-wise by:*

$$[\operatorname{prox}_{\lambda\|\cdot\|_1}(\mathbf{v})]_i = S_\lambda(v_i) = \operatorname{sign}(v_i) \max\{|v_i| - \lambda, 0\} \tag{5}$$

where S_λ is the soft-thresholding operator.

Proof. For the scalar case, we need to solve:

$$\min_{u \in \mathbb{R}} \left\{ \lambda \left| u \right| + \frac{1}{2}(u - v)^2 \right\} \tag{6}$$

Taking the subdifferential and setting it to contain zero:

$$0 \in \lambda \cdot \partial \left| u \right| + (u - v) \tag{7}$$

$$v \in u + \lambda \cdot \partial \left| u \right| \tag{8}$$

Case analysis yields:

- If $v > \lambda$: $u = v - \lambda$
- If $v < -\lambda$: $u = v + \lambda$
- If $|v| \leq \lambda$: $u = 0$

Combining these cases gives the soft-thresholding formula. □

2.4 Visualization of Thresholding Operators

The soft-thresholding operator exhibits the following characteristics:

- **Shrinkage effect**: Non-zero coefficients are reduced by λ
- **Sparsification**: Coefficients with $|v_i| \leq \lambda$ are set to zero
- **Sign preservation**: The sign of large coefficients is maintained

Key Insight: Soft thresholding simultaneously promotes sparsity and shrinks large coefficients

3 The Proximal Gradient Algorithm

3.1 Algorithm Development

The proximal gradient method, also known as the Iterative Shrinkage-Thresholding Algorithm (ISTA), combines gradient descent for the smooth part with proximal operations for the non-smooth part.

Theorem 3.1 (Proximal Gradient Iteration)**.** *For problem* (1) *with f having L-Lipschitz gradient, the iteration:*

$$\mathbf{x}^{k+1} = \operatorname{prox}_{\alpha g}\left(\mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k)\right) \tag{9}$$

converges to the optimal solution when $0 < \alpha < 2/L$.

3.2 ISTA for ℓ_1-Regularized Least Squares

For the specific problem (2), the algorithm takes the form:

Algorithm 1 Iterative Shrinkage-Thresholding Algorithm (ISTA)

1. **Initialize**: Choose $\mathbf{x}^0 \in \mathbb{R}^n$, step size $\alpha > 0$
2. **For** $k = 0, 1, 2, \ldots$ **do**:
   (a) Compute gradient: $\mathbf{g}^k = \mathbf{A}^T(\mathbf{A}\mathbf{x}^k - \mathbf{b})$
   (b) Gradient step: $\mathbf{z}^k = \mathbf{x}^k - \alpha \mathbf{g}^k$
   (c) Soft threshold: $\mathbf{x}^{k+1} = S_{\alpha\lambda}(\mathbf{z}^k)$
3. **Until** convergence criterion is met

3.3 Step Size Selection

3.3.1 Lipschitz Constant Computation

Definition 3.2 (Lipschitz Continuity of Gradient)**.** *A function f has L-Lipschitz continuous gradient if:*

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L \|\mathbf{x} - \mathbf{y}\|_2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n \tag{10}$$

For the quadratic function $f(\mathbf{x}) = \frac{1}{2}\|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2$:

Lemma 3.3. *The Lipschitz constant of ∇f is $L = \left\|\mathbf{A}^T\mathbf{A}\right\|_2 = \lambda_{\max}(\mathbf{A}^T\mathbf{A})$, where $\lambda_{\max}$ denotes the largest eigenvalue.*

Proof. The gradient is $\nabla f(\mathbf{x}) = \mathbf{A}^T(\mathbf{A}\mathbf{x} - \mathbf{b})$. Thus:

$$\begin{aligned}
\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 &= \left\|\mathbf{A}^T\mathbf{A}(\mathbf{x} - \mathbf{y})\right\|_2 &(11)\\
&\leq \left\|\mathbf{A}^T\mathbf{A}\right\|_2 \|\mathbf{x} - \mathbf{y}\|_2 &(12)\\
&= \lambda_{\max}(\mathbf{A}^T\mathbf{A}) \|\mathbf{x} - \mathbf{y}\|_2 &(13)
\end{aligned}$$

where we used the fact that the spectral norm equals the largest eigenvalue for symmetric positive semidefinite matrices. □

3.3.2 Backtracking Line Search

When computing eigenvalues is impractical, adaptive step size selection via backtracking provides a robust alternative:

Algorithm 2 Backtracking Line Search for Proximal Gradient

---

1. **Parameters**: $\beta \in (0, 1)$ (typically $\beta = 0.5$), $\eta \in (0, 1)$ (typically $\eta = 0.9$)
2. **Initialize**: $\alpha = \alpha_0$ (initial guess, e.g., $\alpha_0 = 1$)
3. **Repeat**:
   (a) Compute: $\mathbf{x}^+ = \text{prox}_{\alpha g}(\mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k))$
   (b) **While** $F(\mathbf{x}^+) > F(\mathbf{x}^k)$:
   - Set $\alpha \leftarrow \beta\alpha$
   - Recompute $\mathbf{x}^+$
4. **Set**: $\mathbf{x}^{k+1} = \mathbf{x}^+$

3.4 Convergence Analysis

Theorem 3.4 (ISTA Convergence Rate)**.** *For the proximal gradient method with constant step size $\alpha = 1/L$, we have:*

$$F(\mathbf{x}^k) - F(\mathbf{x}^*) \leq \frac{L \left\|\mathbf{x}^0 - \mathbf{x}^*\right\|_2^2}{2k} \tag{14}$$

where $\mathbf{x}^*$ is an optimal solution.

Important: ISTA achieves $O(1/k)$ convergence

4 Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)

4.1 Motivation for Acceleration

While ISTA achieves $O(1/k)$ convergence, Nesterov's acceleration technique can improve this to $O(1/k^2)$ without additional computational cost per iteration. This acceleration is achieved through a momentum-like mechanism that exploits the history of iterates.

4.2 The FISTA Algorithm

Algorithm 3 Fast Iterative Shrinkage-Thresholding Algorithm (FISTA)

1. **Initialize**:
   - Choose $\mathbf{x}^0 = \mathbf{y}^1 \in \mathbb{R}^n$
   - Set $t_1 = 1$
   - Choose step size $\alpha \leq 1/L$
2. **For** $k = 1, 2, 3, \ldots$ **do**:

   (a) Proximal gradient step:

   $$\mathbf{x}^k = \text{prox}_{\alpha g}(\mathbf{y}^k - \alpha \nabla f(\mathbf{y}^k)) \tag{15}$$

   (b) Update momentum parameter:

   $$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \tag{16}$$

   (c) Compute extrapolated point:

   $$\mathbf{y}^{k+1} = \mathbf{x}^k + \frac{t_k - 1}{t_{k+1}}(\mathbf{x}^k - \mathbf{x}^{k-1}) \tag{17}$$

3. **Until** convergence

4.3 Key Innovations in FISTA

4.3.1 The Momentum Sequence

The sequence $\{t_k\}$ satisfies the recurrence relation:

$$t_{k+1}^2 - t_{k+1} - t_k^2 = 0 \tag{18}$$

This yields the closed-form expression:

$$t_k = \frac{k+1}{2} + O(1) \approx \frac{k}{2} \text{ for large } k \tag{19}$$

4.3.2 The Extrapolation Step

The extrapolation coefficient:

$$\beta_k = \frac{t_k - 1}{t_{k+1}} \approx \frac{k-2}{k+1} \to 1 \text{ as } k \to \infty \tag{20}$$

This creates an "overshoot" effect that accelerates convergence by anticipating the trajectory of the iterates.

4.4 Convergence Theory

Theorem 4.1 (FISTA Convergence Rate)**.** *For FISTA with step size $\alpha = 1/L$, the following bound holds:*

$$F(\mathbf{x}^k) - F(\mathbf{x}^*) \leq \frac{2L \left\| \mathbf{x}^0 - \mathbf{x}^* \right\|_2^2}{(k+1)^2} \tag{21}$$

Remark 4.2. *The $O(1/k^2)$ rate is optimal for first-order methods on the class of convex functions with Lipschitz continuous gradients.*

4.5 Geometric Interpretation

FISTA can be viewed as performing gradient descent on an auxiliary sequence $\{\mathbf{y}^k\}$ that is constructed to have favorable properties:

- The sequence $\{\mathbf{y}^k\}$ exhibits less oscillation than $\{\mathbf{x}^k\}$
- The extrapolation step creates a "look-ahead" effect
- The momentum builds up over iterations, accelerating convergence in consistent directions

5 Implementation Considerations and Extensions

5.1 Practical Implementation Details

5.1.1 Stopping Criteria

Common convergence criteria for FISTA include:

1. **Relative change in objective**:

$$\frac{\left|F(\mathbf{x}^k) - F(\mathbf{x}^{k-1})\right|}{|F(\mathbf{x}^{k-1})|} < \epsilon_{\text{obj}} \tag{22}$$

2. **Relative change in iterates**:

$$\frac{\left\|\mathbf{x}^k - \mathbf{x}^{k-1}\right\|_2}{\left\|\mathbf{x}^{k-1}\right\|_2} < \epsilon_{\text{sol}} \tag{23}$$

3. **Optimality conditions**:

$$\text{dist}(0, \partial F(\mathbf{x}^k)) < \epsilon_{\text{opt}} \tag{24}$$

5.1.2 Computational Complexity

Per iteration, FISTA requires:

- One gradient evaluation: $O(mn)$ for matrix-vector products
- One soft-thresholding operation: $O(n)$
- Vector operations: $O(n)$

Total complexity: $O(mn)$ per iteration, same as ISTA but with faster convergence.

5.2 Extensions and Variants

5.2.1 Adaptive Restart

Adaptive restart strategies can further improve practical performance:

$$\text{Restart if: } \langle \mathbf{y}^k - \mathbf{x}^k, \mathbf{x}^k - \mathbf{x}^{k-1} \rangle > 0 \tag{25}$$

This condition detects when the momentum is counterproductive.

5.2.2 Strong Convexity

When f is μ-strongly convex, linear convergence can be achieved:

$$F(\mathbf{x}^k) - F(\mathbf{x}^*) \leq \left(1 - \sqrt{\frac{\mu}{L}}\right)^k [F(\mathbf{x}^0) - F(\mathbf{x}^*)] \tag{26}$$

Regularizer	Proximal Operator	Application
$\|\mathbf{x}\|_1$	Soft thresholding	Sparse recovery
$\|\mathbf{x}\|_2$	Scaling	Group sparsity
$\delta_C(\mathbf{x})$	Projection onto C	Constrained optimization
$\|\mathbf{X}\|_*$	Singular value thresholding	Low-rank matrix recovery

Table 2: Common regularizers and their proximal operators

5.3 Applications Beyond ℓ_1 Regularization

FISTA's framework extends to various proximal operators:

5.4 Numerical Experiments and Convergence Behavior

In practice, FISTA exhibits several characteristic behaviors:

1. **Initial phase**: Rapid decrease in objective value
2. **Middle phase**: Steady convergence with momentum benefits
3. **Final phase**: Oscillations may occur near the solution

Comparison with ISTA Empirical studies consistently show FISTA requiring 5-10Œ fewer iterations than ISTA for the same accuracy, validating the theoretical acceleration.

Conclusion

The Fast Iterative Shrinkage-Thresholding Algorithm represents a fundamental advancement in composite convex optimization, combining:

- Elegant handling of non-smooth regularizers via proximal operators
- Optimal convergence rates through Nesterov's acceleration
- Practical efficiency and broad applicability

FISTA's success has inspired numerous extensions and remains a cornerstone algorithm in machine learning, signal processing, and computational statistics.