Proximal Gradient Methods and Iterative Soft Thresholding: A Comprehensive Analysis for Sparse Optimization

Lecture Notes

July 15, 2025

Contents

1 Introduction and Motivation

1.1 The Sparse Recovery Problem

The fundamental challenge in modern signal processing and machine learning involves recovering sparse representations from noisy observations. Consider the linear model:

$$\mathbf{y} = \mathbf{D}\mathbf{x} + \epsilon \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^m$ represents observed measurements, $\mathbf{D} \in \mathbb{R}^{m \times n}$ is a dictionary matrix (often overcomplete with $n > m$), $\mathbf{x} \in \mathbb{R}^n$ is the sparse coefficient vector we seek to recover, and $\epsilon \in \mathbb{R}^m$ denotes additive noise.

The sparsity constraint fundamentally transforms this inverse problem. Rather than seeking any solution satisfying the data fidelity constraint, we specifically desire solutions where most components of $\mathbf{x}$ equal zero. This preference leads to the basis pursuit denoising (BPDN) formulation:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \frac{1}{2} \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{x}\|_1 \tag{2}$$

1.2 Mathematical Challenges

The optimization problem (2) presents a fundamental mathematical challenge: while the data fidelity term $\frac{1}{2} \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2$ is smooth and convex with Lipschitz continuous gradient, the ℓ_1 regularization term $\lambda \|\mathbf{x}\|_1$ is non-differentiable at the origin. Specifically:

$$\nabla_{\mathbf{x}} \left[\frac{1}{2} \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 \right] = \mathbf{D}^T(\mathbf{D}\mathbf{x} - \mathbf{y}) \quad \text{(well-defined everywhere)} \tag{3}$$

$$\partial \|\mathbf{x}\|_1 = \begin{cases} \{\text{sign}(\mathbf{x})\} & \text{if } \mathbf{x} \neq \mathbf{0} \\ [-1, 1]^n & \text{if } \mathbf{x} = \mathbf{0} \end{cases} \tag{4}$$

This non-differentiability precludes direct application of gradient descent methods, necessitating more sophisticated optimization techniques.

2 Mathematical Foundations: Convex Analysis and Majorization

2.1 Convexity and Smoothness

Definition 2.1 (Convex Function)**.** A function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is *convex* if for all $\mathbf{x}, \mathbf{y} \in \text{dom}(f)$ and $\theta \in [0, 1]$:

$$f(\theta\mathbf{x} + (1-\theta)\mathbf{y}) \leq \theta f(\mathbf{x}) + (1-\theta)f(\mathbf{y}) \tag{5}$$

Definition 2.2 (L-Smooth Function)**.** A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is *L-smooth* if its gradient is Lipschitz continuous with constant $L > 0$:

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2 \leq L \|\mathbf{x} - \mathbf{y}\|_2 \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n \tag{6}$$

Theorem 2.1 (Descent Lemma)**.** If $f : \mathbb{R}^n \to \mathbb{R}$ is convex and L-smooth, then for all $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$:

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|_2^2 \tag{7}$$

Proof Sketch. The proof leverages the fundamental theorem of calculus and Lipschitz continuity:

$$f(\mathbf{y}) - f(\mathbf{x}) = \int_0^1 \langle \nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})), \mathbf{y} - \mathbf{x} \rangle dt \tag{8}$$

$$= \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \int_0^1 \langle \nabla f(\mathbf{x} + t(\mathbf{y} - \mathbf{x})) - \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle dt \tag{9}$$

Applying Cauchy-Schwarz and the L-smoothness property completes the proof. □

2.2 Majorization-Minimization Framework

The majorization-minimization (MM) principle provides a general framework for iterative optimization:

Definition 2.3 (Majorizer)**.** A function $Q_{\mathbf{x}}(\mathbf{z})$ is a *majorizer* of f at point $\mathbf{x}$ if:

1. $Q_{\mathbf{x}}(\mathbf{z}) \geq f(\mathbf{z})$ for all $\mathbf{z} \in \text{dom}(f)$ (global upper bound)
2. $Q_{\mathbf{x}}(\mathbf{x}) = f(\mathbf{x})$ (tangency condition)

For L-smooth convex functions, the descent lemma immediately provides a quadratic majorizer:

$$Q_{\mathbf{x}}(\mathbf{z}) = f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{z} - \mathbf{x}\|_2^2 \tag{10}$$

Proposition 2.2 (MM Algorithm Convergence)**.** The iterative scheme:

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{z}} Q_{\mathbf{x}_k}(\mathbf{z}) \tag{11}$$

generates a sequence $\{\mathbf{x}_k\}$ with monotonically decreasing function values: $f(\mathbf{x}_{k+1}) \leq f(\mathbf{x}_k)$.

3 Proximal Gradient Methods

3.1 The Composite Optimization Problem

Consider the general composite optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^n} F(\mathbf{x}) := f(\mathbf{x}) + g(\mathbf{x}) \tag{12}$$

where:

- $f : \mathbb{R}^n \to \mathbb{R}$ is convex and L-smooth (differentiable component)
- $g : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ is convex but possibly non-smooth (regularizer)

3.2 Proximal Mapping: The Key Innovation

Definition 3.1 (Proximal Mapping)**.** The *proximal mapping* of a convex function g at point $\mathbf{v}$ with parameter $\gamma > 0$ is:

$$\text{prox}_{\gamma g}(\mathbf{v}) := \arg\min_{\mathbf{z}\in\mathbb{R}^n} \left\{ g(\mathbf{z}) + \frac{1}{2\gamma} \|\mathbf{z} - \mathbf{v}\|_2^2 \right\} \tag{13}$$

Remark 3.1. The proximal mapping balances two objectives:

1. Minimizing the function g (the original objective)
2. Staying close to the reference point $\mathbf{v}$ (stability/regularization)

The parameter γ controls this trade-off: larger γ emphasizes minimizing g, while smaller γ keeps the solution closer to $\mathbf{v}$.

3.3 Derivation of the Proximal Gradient Algorithm

Starting from the majorizer (10) for the smooth component f:

$$F(\mathbf{z}) = f(\mathbf{z}) + g(\mathbf{z}) \tag{14}$$

$$\leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{z} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{z} - \mathbf{x}\|_2^2 + g(\mathbf{z}) \tag{15}$$

Setting $\gamma = 1/L$ and minimizing the right-hand side:

$$\mathbf{x}_{k+1} = \arg\min_{\mathbf{z}} \left\{ f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{z} - \mathbf{x}_k \rangle + \frac{1}{2\gamma} \|\mathbf{z} - \mathbf{x}_k\|_2^2 + g(\mathbf{z}) \right\} \tag{16}$$

$$= \arg\min_{\mathbf{z}} \left\{ \frac{1}{2\gamma} \|\mathbf{z} - \mathbf{x}_k\|_2^2 + \langle \nabla f(\mathbf{x}_k), \mathbf{z} \rangle + g(\mathbf{z}) \right\} \tag{17}$$

Completing the square:

$$= \arg\min_{\mathbf{z}} \left\{ \frac{1}{2\gamma} \|\mathbf{z} - (\mathbf{x}_k - \gamma \nabla f(\mathbf{x}_k))\|_2^2 + g(\mathbf{z}) \right\} \tag{18}$$

$$= \text{prox}_{\gamma g}(\mathbf{x}_k - \gamma \nabla f(\mathbf{x}_k)) \tag{19}$$

Theorem 3.1 (Proximal Gradient Algorithm)**.** The iterative scheme:

$$\mathbf{x}_{k+1} = \text{prox}_{\gamma g}(\mathbf{x}_k - \gamma \nabla f(\mathbf{x}_k)) \tag{20}$$

converges to a minimizer of $F = f + g$ when f is convex and L-smooth, g is convex, and $0 < \gamma \leq 1/L$.

4 Computing the Proximal Mapping of the ℓ_1 Norm

4.1 Problem Formulation

For the ℓ_1 regularization term $g(\mathbf{x}) = \lambda \|\mathbf{x}\|_1$, we need to compute:

$$\text{prox}_{\gamma\lambda\|\cdot\|_1}(\mathbf{v}) = \arg\min_{\mathbf{z}} \left\{ \lambda \|\mathbf{z}\|_1 + \frac{1}{2\gamma} \|\mathbf{z} - \mathbf{v}\|_2^2 \right\} \tag{21}$$

4.2 Separability and Component-wise Solution

A crucial observation is that both terms in (21) are separable:

$$\lambda \|\mathbf{z}\|_1 = \lambda \sum_{i=1}^{n} |z_i| \tag{22}$$

$$\frac{1}{2\gamma} \|\mathbf{z} - \mathbf{v}\|_2^2 = \frac{1}{2\gamma} \sum_{i=1}^{n} (z_i - v_i)^2 \tag{23}$$

This separability allows us to solve for each component independently:

$$[\text{prox}_{\gamma\lambda\|\cdot\|_1}(\mathbf{v})]_i = \arg\min_{z_i \in \mathbb{R}} \left\{ \lambda |z_i| + \frac{1}{2\gamma} (z_i - v_i)^2 \right\} \tag{24}$$

4.3 Solving the Scalar Problem

For the scalar optimization problem (24), we consider three cases:

4.3.1 Case 1: $z_i > 0$

The objective becomes:

$$h(z_i) = \lambda z_i + \frac{1}{2\gamma}(z_i - v_i)^2 \tag{25}$$

Taking the derivative and setting to zero:

$$\frac{\partial h}{\partial z_i} = \lambda + \frac{1}{\gamma}(z_i - v_i) = 0 \tag{26}$$

$$\Rightarrow z_i^* = v_i - \gamma\lambda \tag{27}$$

This solution is valid only if $z_i^* > 0$, i.e., $v_i > \gamma\lambda$.

4.3.2 Case 2: $z_i < 0$

The objective becomes:

$$h(z_i) = -\lambda z_i + \frac{1}{2\gamma}(z_i - v_i)^2 \tag{28}$$

Similarly:

$$\frac{\partial h}{\partial z_i} = -\lambda + \frac{1}{\gamma}(z_i - v_i) = 0 \tag{29}$$

$$\Rightarrow z_i^* = v_i + \gamma\lambda \tag{30}$$

Valid when $z_i^* < 0$, i.e., $v_i < -\gamma\lambda$.

4.3.3 Case 3: $z_i = 0$

By the subdifferential optimality condition, $0 \in \partial h(0)$:

$$0 \in \lambda[-1, 1] + \frac{1}{\gamma}(0 - v_i) \tag{31}$$

This holds when $|v_i| \leq \gamma\lambda$.

4.4 The Soft Thresholding Operator

Combining all cases yields the *soft thresholding operator*:

Theorem 4.1 (Soft Thresholding)**.** The proximal mapping of the ℓ_1 norm is given componentwise by:

$$[\text{prox}_{\gamma\lambda\|\cdot\|_1}(\mathbf{v})]_i = \text{soft}_{\gamma\lambda}(v_i) := \text{sign}(v_i)\max\{|v_i| - \gamma\lambda, 0\} \tag{32}$$

Equivalently:

$$\text{soft}_{\gamma\lambda}(v_i) = \begin{cases} v_i - \gamma\lambda & \text{if } v_i > \gamma\lambda \\ 0 & \text{if } |v_i| \leq \gamma\lambda \\ v_i + \gamma\lambda & \text{if } v_i < -\gamma\lambda \end{cases} \tag{33}$$

Remark 4.1 (Comparison with Hard Thresholding)**.** The soft thresholding operator differs fundamentally from hard thresholding:

- **Hard thresholding**: $\mathcal{H}_\tau(v) = v \cdot \mathbb{1}_{|v|>\tau}$ (discontinuous, preserves large values)
- **Soft thresholding**: $\text{soft}_\tau(v) = \text{sign}(v)\max\{|v| - \tau, 0\}$ (continuous, shrinks all values)

5 The ISTA Algorithm: Iterative Soft Thresholding

5.1 Algorithm Formulation

Combining the proximal gradient framework with the soft thresholding operator yields:

Theorem 5.1 (ISTA for Basis Pursuit Denoising)**.** The Iterative Soft Thresholding Algorithm (ISTA) for solving:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda \|\mathbf{x}\|_1 \tag{34}$$

is given by:

$$\boxed{\mathbf{x}_{k+1} = \text{soft}_{\gamma\lambda}(\mathbf{x}_k - \gamma \mathbf{D}^T(\mathbf{D}\mathbf{x}_k - \mathbf{y}))} \tag{35}$$

where $\gamma \in (0, 1/\left\|\mathbf{D}^T\mathbf{D}\right\|_2]$ is the step size.

5.2 Algorithm Implementation

Example 5.1 (ISTA Implementation)**.** The algorithm can be implemented efficiently as:

1. **Precomputation**:
   - Compute $\mathbf{D}^T\mathbf{D}$ and $\mathbf{D}^T\mathbf{y}$ (one-time cost)
   - Estimate $L = \left\|\mathbf{D}^T\mathbf{D}\right\|_2$ via power iteration
2. **Iteration**: For $k = 0, 1, 2, \ldots$

$$\mathbf{g}_k = \mathbf{D}^T(\mathbf{D}\mathbf{x}_k - \mathbf{y}) \quad \text{(gradient computation)} \tag{36}$$

$$\mathbf{z}_k = \mathbf{x}_k - \gamma \mathbf{g}_k \quad \text{(gradient descent step)} \tag{37}$$

$$\mathbf{x}_{k+1} = \text{soft}_{\gamma\lambda}(\mathbf{z}_k) \quad \text{(soft thresholding)} \tag{38}$$

5.3 Convergence Analysis

Theorem 5.2 (ISTA Convergence Rate)**.** Under the conditions:

- $f(\mathbf{x}) = \frac{1}{2} \|\mathbf{D}\mathbf{x} - \mathbf{y}\|_2^2$ is convex and L-smooth
- $g(\mathbf{x}) = \lambda \|\mathbf{x}\|_1$ is convex
- Step size $\gamma \in (0, 1/L]$

ISTA converges with rate:

$$F(\mathbf{x}_k) - F(\mathbf{x}^*) \leq \frac{\|\mathbf{x}_0 - \mathbf{x}^*\|_2^2}{2\gamma k} \tag{39}$$

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

5.4 Extensions and Generalizations

The proximal gradient framework extends naturally to other regularizers:

Example 5.2 (Group Lasso)**.** For group sparsity with $g(\mathbf{x}) = \sum_{g \in \mathcal{G}} \lambda_g \|\mathbf{x}_g\|_2$:

$$[\text{prox}_{\gamma g}(\mathbf{v})]_g = \max\left\{1 - \frac{\gamma \lambda_g}{\|\mathbf{v}_g\|_2}, 0\right\} \mathbf{v}_g \tag{40}$$

Example 5.3 (Nuclear Norm)**.** For low-rank matrix recovery with $g(\mathbf{X}) = \lambda \|\mathbf{X}\|_*$:

$$\text{prox}_{\gamma\lambda\|\cdot\|_*}(\mathbf{V}) = \mathbf{U}\, \text{soft}_{\gamma\lambda}(\mathbf{\Sigma})\mathbf{V}^T \tag{41}$$

where $\mathbf{V} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ is the SVD.

6 Practical Considerations and Advanced Topics

6.1 Acceleration: FISTA

The Fast Iterative Soft Thresholding Algorithm (FISTA) improves convergence from $O(1/k)$ to $O(1/k^2)$:

$$\mathbf{y}_{k+1} = \mathbf{x}_k + \frac{t_k - 1}{t_{k+1}}(\mathbf{x}_k - \mathbf{x}_{k-1}) \tag{42}$$

$$\mathbf{x}_{k+1} = \text{soft}_{\gamma\lambda}(\mathbf{y}_{k+1} - \gamma\mathbf{D}^T(\mathbf{D}\mathbf{y}_{k+1} - \mathbf{y})) \tag{43}$$

$$t_{k+1} = \frac{1 + \sqrt{1 + 4t_k^2}}{2} \tag{44}$$

6.2 Adaptive Step Size Selection

In practice, the Lipschitz constant L may be unknown or conservative. Backtracking line search provides an adaptive solution:

1. Start with $\gamma = \gamma_0$ (e.g., $\gamma_0 = 1$)
2. While the descent condition is violated:

$$f(\mathbf{x}_{k+1}) > f(\mathbf{x}_k) + \langle \nabla f(\mathbf{x}_k), \mathbf{x}_{k+1} - \mathbf{x}_k \rangle + \frac{1}{2\gamma} \|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^2 \tag{45}$$

   reduce $\gamma \leftarrow \beta\gamma$ (typically $\beta = 0.5$)

6.3 Warm Starting and Continuation

For solving a sequence of related problems (e.g., regularization path):

- **Warm starting**: Initialize $\mathbf{x}_0$ for λ_i using solution from λ_{i-1}
- **Continuation**: Solve for decreasing sequence $\lambda_1 > \lambda_2 > \cdots > \lambda_{\text{target}}$

6.4 Stopping Criteria

Practical termination conditions include:

1. **Relative change**: $\frac{\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2}{\|\mathbf{x}_k\|_2} < \epsilon_{\text{tol}}$
2. **Gradient magnitude**: $\left\|\mathbf{x}_k - \text{prox}_{\gamma g}(\mathbf{x}_k - \gamma\nabla f(\mathbf{x}_k))\right\|_2 < \epsilon_{\text{tol}}$
3. **Duality gap**: For problems with known dual formulation

7 Conclusion

The proximal gradient method elegantly handles composite optimization problems by:

1. Separating smooth and non-smooth components
2. Applying gradient descent to the smooth part
3. Using proximal mappings for the non-smooth part

For ℓ_1-regularized problems, this yields the efficient ISTA algorithm, where each iteration consists of:

- A gradient descent step (handling data fidelity)
- A soft thresholding operation (inducing sparsity)

The framework extends naturally to numerous other regularizers and has become a cornerstone of modern optimization in machine learning and signal processing.

Remark 7.1 (Historical Note)**.** The development of proximal methods traces back to Moreau (1962) and Rockafellar (1976), but their application to sparse recovery problems emerged prominently with the work of Daubechies, Defrise, and De Mol (2004) on iterative thresholding, followed by Beck and Teboulle's FISTA (2009).