Structured Sparsity, Mixed Norms, and Statistical Connections: From Joint Sparsity to LASSO

Lecture Notes on Advanced Sparse Representations

July 17, 2025

Contents

1 Introduction to Structured Sparsity

1.1 Motivation and Overview

The classical sparse coding framework seeks representations with minimal non-zero coefficients, treating each coefficient independently. However, in many practical applications, the *locations* of non-zero coefficients exhibit inherent structure that standard sparsity models fail to exploit. Structured sparsity extends the sparse coding paradigm by incorporating prior knowledge about coefficient patterns, leading to more robust and interpretable representations.

Consider the fundamental sparse coding problem:

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda\|\mathbf{x}\|_0 \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^m$ represents the observed signal, $\mathbf{D} \in \mathbb{R}^{m \times n}$ denotes an overcomplete dictionary with $n > m$, and $\mathbf{x} \in \mathbb{R}^n$ contains the sparse coefficients.

This formulation promotes sparsity uniformly across all coefficients. However, structured sparsity recognizes that coefficients often exhibit dependencies or groupings that reflect the underlying data generation process.

1.2 Applications Motivating Structured Sparsity

(a) **Multi-channel Signal Processing**: When processing multiple signals acquired from the same source (e.g., multi-electrode recordings, hyperspectral imaging), the active atoms in the dictionary tend to be consistent across channels.

(b) **Texture Analysis**: Patches extracted from textured images share common structural elements, suggesting that their sparse representations should utilize similar dictionary atoms.

(c) **Statistical Variable Selection**: In high-dimensional regression problems, predictors often form natural groups (e.g., dummy variables for categorical features, measurements from the same instrument).

2 Joint Sparsity and Mixed Norms

2.1 Problem Formulation

Let $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_L] \in \mathbb{R}^{m \times L}$ represent a collection of L signals sharing common structural properties. The joint sparse coding problem seeks a coefficient matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_L] \in \mathbb{R}^{n \times L}$ such that:

$$\mathbf{Y} \approx \mathbf{D}\mathbf{X} \tag{2}$$

The key insight is that columns of $\mathbf{X}$ should not only be individually sparse but should also share common support patterns.

2.2 Mixed Norms for Matrices

To enforce joint sparsity, we introduce the (p, q)-mixed norm for matrices.

Definition 2.1 ($\ell_{p,q}$ Mixed Norm)**.** For a matrix $\mathbf{X} \in \mathbb{R}^{n \times L}$, the (p, q)-mixed norm is defined as:

$$\|\mathbf{X}\|_{p,q} = \left(\sum_{i=1}^{n} \left(\sum_{j=1}^{L} |X_{ij}|^p \right)^{q/p} \right)^{1/q} \tag{3}$$

This can be interpreted as:

$$\|\mathbf{X}\|_{p,q} = \left\| \left[\|\mathbf{x}^{(1)}\|_p, \|\mathbf{x}^{(2)}\|_p, \ldots, \|\mathbf{x}^{(n)}\|_p \right]^T \right\|_q \tag{4}$$

$$= \|\mathbf{v}\|_q \tag{5}$$

where $\mathbf{x}^{(i)}$ denotes the i-th row of $\mathbf{X}$, and $v_i = \|\mathbf{x}^{(i)}\|_p$.

2.2.1 Special Cases and Properties

1. **Frobenius Norm**: When $p = q = 2$:

$$\|\mathbf{X}\|_{2,2} = \sqrt{\sum_{i,j} |X_{ij}|^2} = \|\mathbf{X}\|_F \tag{6}$$

2. **Entry-wise Norms**: When $p = q$, the mixed norm reduces to the vectorized ℓ_p norm:

$$\|\mathbf{X}\|_{p,p} = \|\mathrm{vec}(\mathbf{X})\|_p \tag{7}$$

3. **Joint Sparsity Norm**: The $\ell_{2,1}$ norm:

$$\|\mathbf{X}\|_{2,1} = \sum_{i=1}^{n} \sqrt{\sum_{j=1}^{L} |X_{ij}|^2} = \sum_{i=1}^{n} \|\mathbf{x}^{(i)}\|_2 \tag{8}$$

Remark 2.1. The $\ell_{2,1}$ norm promotes row-sparsity in $\mathbf{X}$, meaning entire rows become zero. This corresponds to selecting the same dictionary atoms across all signals.

3 Optimization with Proximal Methods

3.1 Joint Sparse Coding Formulation

The joint sparse coding problem with the $\ell_{2,1}$ norm regularization is formulated as:

$$\min_{\mathbf{X}\in\mathbb{R}^{n\times L}} \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2 + \lambda\|\mathbf{X}\|_{2,1} \tag{9}$$

This optimization problem combines a smooth, convex data fidelity term with a non-smooth but convex regularizer, making it amenable to proximal gradient methods.

3.2 Proximal Gradient Algorithm

The iterative solution via proximal gradient descent follows:

$$\mathbf{X}^{(k+1)} = \text{prox}_{\gamma\lambda\|\cdot\|_{2,1}}\left(\mathbf{X}^{(k)} - \gamma\nabla f(\mathbf{X}^{(k)})\right) \tag{10}$$

where $f(\mathbf{X}) = \frac{1}{2}\|\mathbf{Y}-\mathbf{D}\mathbf{X}\|_F^2$ is the smooth part, and $\gamma > 0$ is the step size.

3.2.1 Gradient Computation

The gradient of the Frobenius norm term is:

$$\nabla f(\mathbf{X}) = \nabla_{\mathbf{X}}\left[\frac{1}{2}\,\text{trace}\left((\mathbf{Y}-\mathbf{D}\mathbf{X})^T(\mathbf{Y}-\mathbf{D}\mathbf{X})\right)\right] \tag{11}$$

$$= -\mathbf{D}^T(\mathbf{Y}-\mathbf{D}\mathbf{X}) \tag{12}$$

$$= \mathbf{D}^T\mathbf{D}\mathbf{X} - \mathbf{D}^T\mathbf{Y} \tag{13}$$

3.3 Proximal Mapping of the $\ell_{2,1}$ Norm

The key computational challenge lies in evaluating the proximal mapping:

$$\text{prox}_{\tau\|\cdot\|_{2,1}}(\mathbf{Z}) = \arg\min_{\mathbf{X}}\left\{\tau\|\mathbf{X}\|_{2,1} + \frac{1}{2}\|\mathbf{X}-\mathbf{Z}\|_F^2\right\} \tag{14}$$

Theorem 3.1 (Row-wise Separability of $\ell_{2,1}$ Proximal Mapping)**.** The proximal mapping of the $\ell_{2,1}$ norm can be computed row-wise:

$$[\text{prox}_{\tau\|\cdot\|_{2,1}}(\mathbf{Z})]_i = \text{shrink}_\tau^{(2)}(\mathbf{z}^{(i)}) \tag{15}$$

where $\mathbf{z}^{(i)}$ is the i-th row of $\mathbf{Z}$, and $\text{shrink}_\tau^{(2)}$ is the multivariate soft-thresholding operator.

Proof Sketch. The objective function in (14) can be rewritten as:

$$\tau\|\mathbf{X}\|_{2,1} + \frac{1}{2}\|\mathbf{X}-\mathbf{Z}\|_F^2 = \tau\sum_{i=1}^{n}\|\mathbf{x}^{(i)}\|_2 + \frac{1}{2}\sum_{i=1}^{n}\|\mathbf{x}^{(i)}-\mathbf{z}^{(i)}\|_2^2 \tag{16}$$

$$= \sum_{i=1}^{n}\left[\tau\|\mathbf{x}^{(i)}\|_2 + \frac{1}{2}\|\mathbf{x}^{(i)}-\mathbf{z}^{(i)}\|_2^2\right] \tag{17}$$

Since the objective decomposes into independent row-wise problems, the minimization can be performed separately for each row. □

3.3.1 Multivariate Soft-Thresholding Operator

Definition 3.1 (Multivariate Soft-Thresholding)**.** For a vector $\mathbf{v} \in \mathbb{R}^L$ and threshold $\tau > 0$:

$$\text{shrink}_{\tau}^{(2)}(\mathbf{v}) = \begin{cases} \frac{\mathbf{v}}{\|\mathbf{v}\|_2} \cdot \max(0, \|\mathbf{v}\|_2 - \tau) & \text{if } \mathbf{v} \neq \mathbf{0} \\ \mathbf{0} & \text{if } \mathbf{v} = \mathbf{0} \end{cases} \tag{18}$$

This operator exhibits two key behaviors:

1. **Nullification**: If $\|\mathbf{v}\|_2 \leq \tau$, the entire vector is set to zero.
2. **Shrinkage**: If $\|\mathbf{v}\|_2 > \tau$, the vector is scaled down while preserving its direction.

4 Group Sparsity and Extensions

4.1 Group Structure in Dictionaries

Consider a dictionary $\mathbf{D}$ partitioned into G groups:

$$\mathbf{D} = [\mathbf{D}_1 | \mathbf{D}_2 | \cdots | \mathbf{D}_G] \tag{19}$$

where $\mathbf{D}_g \in \mathbb{R}^{m \times n_g}$ contains atoms corresponding to group g, and $\sum_{g=1}^{G} n_g = n$.

The coefficient vector $\mathbf{x}$ is correspondingly partitioned:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{[1]} \\ \mathbf{x}_{[2]} \\ \vdots \\ \mathbf{x}_{[G]} \end{bmatrix} \tag{20}$$

where $\mathbf{x}_{[g]} \in \mathbb{R}^{n_g}$ contains coefficients for group g.

4.2 Group LASSO Formulation

The group sparse coding problem seeks representations that activate entire groups rather than individual atoms:

$$\min_{\mathbf{x}} \frac{1}{2} \|\mathbf{y} - \mathbf{D}\mathbf{x}\|_2^2 + \lambda \sum_{g=1}^{G} w_g \|\mathbf{x}_{[g]}\|_2 \tag{21}$$

where $w_g > 0$ are group-specific weights, typically set as $w_g = \sqrt{n_g}$ to account for group size differences.

Remark 4.1. The group LASSO penalty induces sparsity at the group level: either all coefficients within a group are zero, or the group is active with potentially multiple non-zero coefficients.

4.3 Proximal Operator for Group LASSO

The proximal mapping for the group LASSO penalty decomposes into group-wise operations:

$$[\text{prox}_{\tau \sum_g w_g \|\cdot\|_2}(\mathbf{z})]_{[g]} = \text{shrink}^{(2)}_{\tau w_g}(\mathbf{z}_{[g]}) \tag{22}$$

This allows efficient computation by applying multivariate soft-thresholding to each group independently.

5 Statistical Perspective: The LASSO

5.1 Linear Regression Framework

In the statistical setting, we observe m samples with response variable y_i and n predictors $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{in}]^T$. The linear model assumes:

$$y_i = \mathbf{x}_i^T \beta + \epsilon_i, \quad i = 1, 2, \ldots, m \tag{23}$$

In matrix notation:

$$\mathbf{y} = \mathbf{X}\beta + \epsilon \tag{24}$$

where $\mathbf{y} \in \mathbb{R}^m$, $\mathbf{X} \in \mathbb{R}^{m \times n}$ is the design matrix, $\beta \in \mathbb{R}^n$ contains regression coefficients, and $\epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I})$.

5.2 From Least Squares to LASSO

5.2.1 Ordinary Least Squares

The classical least squares estimator:

$$\hat{\beta}_{LS} = \arg\min_\beta \|\mathbf{y} - \mathbf{X}\beta\|_2^2 = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y} \tag{25}$$

This estimator is unbiased ($\mathbb{E}[\hat{\beta}_{LS}] = \beta$) but may have high variance, especially when predictors are correlated or n is large relative to m.

5.2.2 The LASSO Estimator

The Least Absolute Shrinkage and Selection Operator (LASSO) adds ℓ_1 regularization:

$$\hat{\beta}_{LASSO} = \arg\min_\beta \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda\|\beta\|_1 \right\} \tag{26}$$

Theorem 5.1 (Variable Selection Property)**.** For sufficiently large λ, the LASSO estimator $\hat{\beta}_{LASSO}$ contains exact zeros, performing automatic variable selection.

5.3 Bias-Variance Trade-off

The LASSO introduces bias to reduce variance:

$$\begin{aligned} \text{MSE}(\hat{\beta}) &= \mathbb{E}\left[\|\hat{\beta} - \beta\|_2^2\right] && (27)\\ &= \|\mathbb{E}[\hat{\beta}] - \beta\|_2^2 + \text{trace}(\text{Var}(\hat{\beta})) && (28)\\ &= \text{Bias}^2 + \text{Variance} && (29) \end{aligned}$$

While OLS minimizes bias, LASSO accepts some bias in exchange for substantially reduced variance through sparsity.

5.4 High-Dimensional Regime

Proposition 5.2 (LASSO in High Dimensions)**.** When $n > m$ (more predictors than observations), OLS is not unique. However, LASSO provides a unique sparse solution for appropriate $\lambda > 0$.

This property makes LASSO particularly valuable in modern applications like genomics, where the number of features vastly exceeds the sample size.

6 Elastic Net: Combining ℓ_1 and ℓ_2 Penalties

6.1 Motivation and Formulation

The LASSO has two notable limitations:

1. In the $n > m$ setting, it selects at most m variables
2. When predictors are highly correlated, LASSO tends to arbitrarily select one from each group

The Elastic Net addresses these issues by combining ℓ_1 and ℓ_2 penalties:

$$\hat{\beta}_{EN} = \arg\min_\beta \left\{ \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2 \right\} \tag{30}$$

6.2 Geometric Interpretation

The constraint region for Elastic Net is:

$$\mathcal{C}_{EN} = \left\{ \beta : \alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2 \leq t \right\} \tag{31}$$

This creates a compromise between the diamond-shaped ℓ_1 ball and the spherical ℓ_2 ball, maintaining sparsity-inducing corners while allowing smoother boundaries.

6.3 Proximal Gradient Solution

The Elastic Net optimization can be solved efficiently using proximal gradient methods. The key insight is that the ℓ_2 penalty can be absorbed into the smooth part:

$$f(\beta) = \frac{1}{2}\|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_2\|\beta\|_2^2 \tag{32}$$

with gradient:

$$\nabla f(\beta) = \mathbf{X}^T(\mathbf{X}\beta - \mathbf{y}) + 2\lambda_2\beta = (\mathbf{X}^T\mathbf{X} + 2\lambda_2\mathbf{I})\beta - \mathbf{X}^T\mathbf{y} \tag{33}$$

The proximal gradient update becomes:

$$\beta^{(k+1)} = \text{shrink}_{\gamma\lambda_1}\left(\beta^{(k)} - \gamma\nabla f(\beta^{(k)})\right) \tag{34}$$

where shrink_τ is the element-wise soft-thresholding operator.

7 Computational Considerations and Algorithms

7.1 Algorithm Summary

Problem	Regularizer	Proximal Operator
Standard Sparsity	$\|\mathbf{x}\|_1$	$\text{shrink}_\tau(x_i) = \text{sign}(x_i)\max(0, \|x_i\| - \tau)$
Joint Sparsity	$\|\mathbf{X}\|_{2,1}$	Row-wise multivariate soft-thresholding
Group Sparsity	$\sum_g w_g \|\mathbf{x}_{[g]}\|_2$	Group-wise multivariate soft-thresholding
Elastic Net	$\lambda_1\|\mathbf{x}\|_1 + \lambda_2\|\mathbf{x}\|_2^2$	Modified soft-thresholding with ℓ_2 in gradient

Table 1: Summary of sparsity-inducing regularizers and their proximal operators

7.2 Convergence Analysis

For the proximal gradient method applied to problems of the form:

$$\min_{\mathbf{x}} f(\mathbf{x}) + g(\mathbf{x}) \tag{35}$$

where f is L-smooth and g is convex, the convergence rate is:

$$F(\mathbf{x}^{(k)}) - F(\mathbf{x}^*) \leq \frac{2L\|\mathbf{x}^{(0)} - \mathbf{x}^*\|_2^2}{k} \tag{36}$$

when using step size $\gamma = 1/L$.

7.3 Practical Implementation Tips

1. **Step Size Selection**: Use backtracking line search or set $\gamma = 1/\|\mathbf{D}^T\mathbf{D}\|_2$ for guaranteed convergence

2. **Warm Starts**: When solving for multiple λ values, use the solution from λ_{i-1} to initialize λ_i

3. **Active Set Strategies**: Maintain and update only non-zero coefficients to reduce computational cost

4. **Stopping Criteria**: Monitor relative change in objective value or coefficient updates:

$$\frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2}{\|\mathbf{x}^{(k)}\|_2 + \epsilon} < \text{tol} \tag{37}$$

8 Conclusions and Future Directions

8.1 Key Takeaways

1. **Structured sparsity** extends classical sparse coding by incorporating prior knowledge about coefficient patterns through mixed norms
2. **Proximal methods** provide a unified framework for solving various structured sparsity problems, with the key computational challenge being the evaluation of proximal operators
3. **Statistical connections** reveal that signal processing techniques (basis pursuit denoising) and statistical methods (LASSO) are fundamentally addressing the same optimization problem in different contexts
4. **Group structures** enable more interpretable models by enforcing sparsity at the group level rather than individual coefficients

8.2 Open Research Questions

1. **Overlapping Groups**: Developing efficient algorithms for group sparsity with overlapping groups remains challenging
2. **Adaptive Regularization**: Learning the group structure or regularization parameters from data
3. **Non-convex Extensions**: Exploring non-convex penalties that better approximate the ℓ_0 norm while maintaining computational tractability
4. **Dictionary Learning**: Jointly learning dictionaries and sparse codes with structured sparsity constraints

8.3 Applications Beyond Linear Models

The principles of structured sparsity extend to:

- Deep neural networks (structured pruning)
- Graphical models (structure learning)
- Matrix completion (low-rank plus sparse decomposition)
- Time series analysis (detecting change points)

Remark 8.1 (Final Thought)**.** The marriage of signal processing insights with statistical methodology, exemplified by the connection between basis pursuit and LASSO, continues to drive innovation in high-dimensional data analysis. As data complexity grows, structured sparsity provides a principled framework for incorporating domain knowledge into learning algorithms.