Principal Component Regression, LASSO, Group LASSO, and Fused LASSO (Objectives & Algorithms)

Supratim Das Swagato Das
MB2532 MB2534
M.Stat. I, Indian Statistical Institute, Kolkata

Course: Regression Techniques — Presentation Project (2025)

November 10, 2025

Key refs: [3], [4], [1], [2]

Outline

1. LASSO objective and coordinate-descent algorithm
2. Group LASSO objective and block-coordinate algorithm
3. Fused LASSO: objective and **Algorithm 1** & **Algorithm 2** from `fastgl`
4. Principal Component Analysis (PCA) and model selection for PCR

LASSO: objective

$$\hat{\beta}(\lambda) = \arg\min_{\beta \in \mathbb{R}^p} \ \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1,$$

where $\|\beta\|_1 = \sum_j |\beta_j|$.

- LASSO induces sparsity; λ controls the trade-off between fit and sparsity [3].

LASSO: Coordinate Descent (soft-threshold)

Standardize columns of X and center y. For coordinate j: fix β_{-j} and minimize over b

$$\min_b \frac{1}{2}\sum_i (r_i^{(j)} - x_{ij}b)^2 + \lambda|b|$$

where $r^{(j)} = y - X_{-j}\beta_{-j}$. The solution:

$$\beta_j \leftarrow S\left(\frac{\sum_i x_{ij} r_i^{(j)}}{\sum_i x_{ij}^2},\ \frac{\lambda}{\sum_i x_{ij}^2}\right),$$

where $S(z,\gamma) = \text{sign}(z)\max(0, |z| - \gamma)$.

LASSO: Coordinate Descent (pseudocode)

Algorithm 1: Coordinate Descent for LASSO (cyclic)

Standardize X, center y. Initialize $\beta \leftarrow 0$, residual $r \leftarrow y$;

for *iter* $= 1$ **to** *maxiter* **do**

  for $j = 1$ **to** p **do**

    $z_j \leftarrow x_j^\top r + \beta_j$;

    $\beta_j^{\text{new}} \leftarrow S(z_j, \lambda)$;

    $r \leftarrow r - x_j(\beta_j^{\text{new}} - \beta_j)$;

    $\beta_j \leftarrow \beta_j^{\text{new}}$;

  end

  stop if change in β small;

end

Group LASSO: objective

Let predictors be partitioned into groups $j = 1, \ldots, J$ with coefficients β_j.

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \ \frac{1}{2}\|y - \sum_{j=1}^{J} X_j \beta_j\|_2^2 + \lambda \sum_{j=1}^{J} w_j \|\beta_j\|_2,$$

where w_j often $= \sqrt{p_j}$ (group size) [4].

Group LASSO: block update (group soft-threshold)

If $X_j^\top X_j = I$ (orthonormal group), let

$$S_j = X_j^\top \Big(y - \sum_{k \neq j} X_k \beta_k\Big).$$

Then the closed-form update:

$$\beta_j \leftarrow \left(1 - \frac{\lambda w_j}{\|S_j\|_2}\right)_+ S_j.$$

Iterating over groups yields block coordinate descent.

Group LASSO: block-coordinate algorithm

Algorithm 2: Block-Coordinate Descent for Group LASSO

(Optional) orthonormalize each group X_j;

Initialize $\beta_j \leftarrow 0$ for all j;

for *iter* $= 1$ **to** *maxiter* **do**

  for *each group* $j = 1, \ldots, J$ **do**

    $r^{(j)} \leftarrow y - \sum_{k \neq j} X_k \beta_k$;

    $S_j \leftarrow X_j^\top r^{(j)}$;

    $$\beta_j \leftarrow \left(1 - \frac{\lambda w_j}{\|S_j\|_2}\right)_+ S_j;$$

  end

  stop if converged;

end

Fused LASSO / Generalized Lasso: objective

Generalized LASSO:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|D\beta\|_1,$$

where D is a penalty matrix encoding differences (e.g., adjacent differences for fused lasso). Fused LASSO encourages piecewise-constant structure (many entries of $D\beta$ equal zero). [1]

Algorithm 1: Dual path algorithm for the generalized lasso ($X = I$)

Algorithm 1 (Dual path algorithm for the generalized lasso, $X = I$)

Given $y \in \mathbb{R}^n$ and $D \in \mathbb{R}^{m \times n}$.

1. Compute $\hat{u}$, the minimum ℓ_2 norm solution of
$$\min_{u \in \mathbb{R}^m} \|y - D^T u\|_2^2.$$
2. Compute the first hitting time λ_1, and the hitting coordinate i_1. Record the solution $\hat{u}(\lambda) = \hat{u}$ for $\lambda \in [\lambda_1, \infty)$. Initialize $B = \{i_1\}$, $s = \text{sign}(\hat{u}_{i_1})$, and $k = 1$.
3. While $\lambda_k > 0$:
    1. Compute $\hat{a}$ and $\hat{b}$, the minimum ℓ_2 norm solutions of
    $$\min_{a \in \mathbb{R}^{m-|B|}} \|y - D_{-B}^T a\|_2^2 \quad \text{and} \quad \min_{b \in \mathbb{R}^{m-|B|}} \|D_B^T s - D_{-B}^T b\|_2^2,$$
    respectively.
    2. Compute the next hitting time and the next leaving time.
    Let λ_{k+1} denote the larger of the two; if the hitting time is larger, then add the hitting coordinate to B and its sign to s, otherwise remove the leaving coordinate from B and its sign from s.
    3. Record the solution $\hat{u}(\lambda) = \hat{a} - \lambda \hat{b}$ for $\lambda \in [\lambda_{k+1}, \lambda_k]$, and update $k \leftarrow k + 1$.

Algorithm 2: Dual path algorithm for the generalized lasso (general X)

Algorithm 2 (Dual path algorithm for the generalized lasso, general X)

Given $y \in \mathbb{R}^n$, $D \in \mathbb{R}^{m \times p}$, and $X \in \mathbb{R}^{n \times p}$ with $\text{rank}(X) = p$.

1. Compute $\tilde{y} = XX^+ y \in \mathbb{R}^n$ X^+ is the Moore Penrose generalized inverse of X and $\tilde{D} = DX^+ \in \mathbb{R}^{m \times n}$.
2. Run Algorithm 1 on $\tilde{y}$ and $\tilde{D}$.

Notes on Algorithms 1 & 2

- Algorithm 1 (dual path) produces the entire solution path by tracking dual variables and boundary events.
- Algorithm 2 provides the crucial speedups: incremental QR/Cholesky updates and exploiting structure (1D/graph Laplacian) to keep per-event costs low.
- The `genlasso` and `fastgl` implementations use combinations of these strategies for robust, scalable solvers. [1]

Principal Component Analysis (PCA): idea

- PCA finds orthogonal directions capturing maximal variance in predictors $X \in \mathbb{R}^{n \times p}$.
- Let X be centered. The PCA directions (loadings) are the eigenvectors of $S = X^\top X$ (or singular vectors from SVD: $X = U\Sigma V^\top$).
- Project onto first K components: $Z = XV_K$ where V_K contains first K right-singular vectors.
- **Principal Component Regression (PCR):** regress y on Z (first K components).

Selecting number of components (cross-validation)

- Use K-fold CV: for each candidate K, fit PCR on training folds and compute validation MSE on held-out fold; average MSE across folds.
- Choose K that minimizes CV MSE (or use 1-SE rule).
- Alternative diagnostics: scree plot (variance explained), cumulative explained variance threshold (e.g., 95%), or adjusted PRESS statistic.

PCR: practical tips

- Standardize predictors before PCA when scales differ.
- Use `pcr(..., validation='CV')` or nested CV if PCR selection leads into a larger model selection pipeline.

SVD Concept

- For a matrix $A_{m\times n}$, the SVD is $A = U\Sigma V^T$.
- U and V are orthogonal matrices.
- Σ is a diagonal matrix of singular values.

Pseudocode for SVD Computation

Goal: Compute U, Σ, and V such that $A = U\Sigma V^T$.

Algorithm Steps

1. Input: matrix A of size $m \times n$.
2. Compute $A^T A$ and AA^T.
3. Find eigenvalues and eigenvectors of $A^T A$:
   - Eigenvectors of $A^T A$ form columns of V.
   - Eigenvalues (square roots) give singular values.
4. Compute $U = AV\Sigma^{-1}$.
5. Output: U, Σ, and V.

Summary

- LASSO: simple coordinate descent (soft threshold); fast and widely used.
- Group LASSO: block updates selecting entire groups (use group weights).
- Fused LASSO: generalized lasso; dual-path algorithm yields path; fast implementations use QR updates and specialized linear solvers.
- PCA/PCR: reduce dimension and avoid multicollinearity; select number of components by CV.

References I

Timothy B. Arnold and Ryan J. Tibshirani.
Efficient implementations of the generalized lasso dual path algorithm.
Technical report, Stanford University / authors' tech report (fastgl), 2014.
Available as `fastgl.pdf` in project files.

Ian T. Jolliffe.
Principal Component Analysis.
Springer, New York, 2nd edition, 2002.

Robert Tibshirani.
Regression shrinkage and selection via the lasso.
Journal of the Royal Statistical Society. Series B (Methodological), 58(1):267–288, 1996.

References II


Ming Yuan and Yi Lin.
Model selection and estimation in regression with grouped variables.
Journal of the Royal Statistical Society. Series B (Statistical Methodology), 68(1):49–67, 2006.

Thank You

Questions?

1

[1] For R codes refer to the Github Repo