Sparsity and smoothness via the fused lasso

Tibshirani, et. al (2005)

Daniel Cowley

STA315, Winter 2025

Table of Contents

LASSO Overview

Standard linear model:

$$y_i = \sum_j x_{ij}\beta_j + \varepsilon_i$$

Lasso finds coefficients $\hat{\beta} = \left(\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_p\right)$ satisfying:

$$\hat{\beta} = \arg\min \left\{ \sum_i \left(y_i - \sum_j x_{ij}\beta_j \right)^2 \right\} \quad \text{subject to} \quad \sum_j |\beta_j| \leq s$$

Note:

- As $s \to \infty$, we obtain the least squares solution
- When $p > N$, we obtain one of the many least squares solutions

Introducing the Fusion Penalty

Standard linear model:

$$y_i = \sum_j x_{ij}\beta_j + \varepsilon_i$$

Fusion finds coefficients $\hat{\beta} = \left(\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_p\right)$ satisfying:

$$\hat{\beta} = \text{arg min}\left\{\sum_i \left(y_i - \sum_j x_{ij}\beta_j\right)^2\right\} \quad \text{subject to} \quad \sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq s$$

Note:

- Encourages sparsity in differences

Introducing the Fused LASSO

Standard linear model:

$$y_i = \sum_j x_{ij}\beta_j + \varepsilon_i$$

Fusion finds coefficients $\hat{\beta} = \left(\hat{\beta}_1, \hat{\beta}_2, \ldots, \hat{\beta}_p\right)$ satisfying:

$$\hat{\beta} = \arg\min \sum_i \left(y_i - \sum_j x_{ij}\beta_j\right)^2$$

subject to $\sum_{j=1}^{p} |\beta_j| \leq s_1$ and $\sum_{j=2}^{p} |\beta_j - \beta_{j-1}| \leq s_2$

Computational Approach

Use **quadratic programming** to approximate the solution to the minimization:

$$\hat{\beta} = \arg\min\{(y - X\beta)^T S(y - X\beta)\}$$

$$\begin{pmatrix} -a_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \underbrace{\begin{pmatrix} L & 0 & 0 & -I & I \\ I & -I & I & 0 & 0 \\ 0 & e^T & e^T & 0 & 0 \\ 0 & 0 & 0 & e_0^T & e_0^T \end{pmatrix}}_{(2p+2)\times 5p} \begin{pmatrix} \beta \\ \beta^+ \\ \beta^- \\ \theta^+ \\ \theta^- \end{pmatrix} \leq \begin{pmatrix} a_0 \\ 0 \\ s_1 \\ s_2 \end{pmatrix}$$

Computational Approach

Use **quadratic programming** to approximate the solution to the minimization:

$$\hat{\beta} = \arg\min\{(y - X\beta)^T S(y - X\beta)\}$$

$$\begin{pmatrix} -a_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \underbrace{\begin{pmatrix} L & 0 & 0 & -I & I \\ I & -I & I & 0 & 0 \\ 0 & e^T & e^T & 0 & 0 \\ 0 & 0 & 0 & e_0^T & e_0^T \end{pmatrix}}_{(2p+2)\times 5p} \begin{pmatrix} \beta \\ \beta^+ \\ \beta^- \\ \theta^+ \\ \theta^- \end{pmatrix} \leq \begin{pmatrix} a_0 \\ 0 \\ s_1 \\ s_2 \end{pmatrix}$$

Fusion Constraint: $L \leq L\beta - \theta^+ + \theta^- \leq U,$ where $\theta = L\beta$

$$L = \underbrace{\begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{pmatrix}}_{p\times p}, \qquad \theta_j = \beta_j - \beta_{j-1}, \qquad a_0 = (\infty, 0, 0, \ldots, 0)$$

Computational Approach

Use **quadratic programming** to approximate the solution to the minimization:

$$\hat{\beta} = \arg\min\{(y - X\beta)^T S(y - X\beta)\}$$

$$\begin{pmatrix} -a_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \underbrace{\begin{pmatrix} L & 0 & 0 & -I & I \\ I & -I & I & 0 & 0 \\ 0 & e^T & e^T & 0 & 0 \\ 0 & 0 & 0 & e_0^T & e_0^T \end{pmatrix}}_{(2p+2)\times 5p} \begin{pmatrix} \beta \\ \beta^+ \\ \beta^- \\ \theta^+ \\ \theta^- \end{pmatrix} \leq \begin{pmatrix} a_0 \\ 0 \\ s_1 \\ s_2 \end{pmatrix}$$

Lasso Constraint: $L \leq \beta - \beta^+ + \beta^- \leq U$

Computational Approach

Use **quadratic programming** to approximate the solution to the minimization:

$$\hat{\beta} = \arg\min\{(y - X\beta)^T S(y - X\beta)\}$$

$$\begin{pmatrix} -a_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \underbrace{\begin{pmatrix} L & 0 & 0 & -I & I \\ I & -I & I & 0 & 0 \\ 0 & e^T & e^T & 0 & 0 \\ 0 & 0 & 0 & e_0^T & e_0^T \end{pmatrix}}_{(2p+2)\times 5p} \begin{pmatrix} \beta \\ \beta^+ \\ \beta^- \\ \theta^+ \\ \theta^- \end{pmatrix} \leq \begin{pmatrix} a_0 \\ 0 \\ s_1 \\ s_2 \end{pmatrix}$$

Lasso Absolute Difference: Ensures the absolute sum of β does not exceed s_1

$$e = \underbrace{(1, 1, ..., 1)^T}_{1\times p}$$

UNIVERSITY OF TORONTO

Computational Approach

Use **quadratic programming** to approximate the solution to the minimization:

$$\hat{\beta} = \arg\min\{(y - X\beta)^T S(y - X\beta)\}$$

$$\begin{pmatrix} -a_0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leq \underbrace{\begin{pmatrix} L & 0 & 0 & -I & I \\ I & -I & I & 0 & 0 \\ 0 & e^T & e^T & 0 & 0 \\ 0 & 0 & 0 & e_0^T & e_0^T \end{pmatrix}}_{(2p+2)\times 5p} \begin{pmatrix} \beta \\ \beta^+ \\ \beta^- \\ \theta^+ \\ \theta^- \end{pmatrix} \leq \begin{pmatrix} a_0 \\ 0 \\ s_1 \\ s_2 \end{pmatrix}$$

Fusion Absolute Difference: Ensures the absolute sum of θ does not exceed s_2

$$e_0 = \underbrace{(0, 1, 1, \ldots, 1)^T}_{1\times p}$$

Finding s_1 and s_2

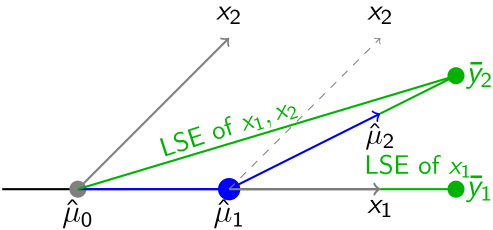
Small/Medium Problems: ($p < 1000$)

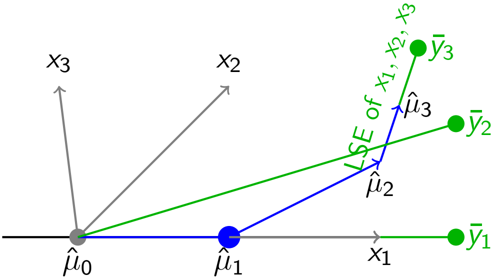
- Grid Search

Larger Problems: ($p \geq 1000$)

- LARS Procedure
  1. Calculate residual: $r = y - \bar{y}$ and let $\beta_1, \beta_2, \ldots, \beta_p = 0$
  2. Find a feature X_j most correlated with residual r
  3. Incrementally update coefficient β_j towards least squares coefficient $\langle x_j, r \rangle$ until another feature X_k has the same correlation as X_j
  4. Now, move (β_j, β_k) in the direction of their joint least squares coefficients of current residual (x_j, x_k), until another feature X_k has the same correlation with r
  5. Repeat for all p features, arriving after p steps at the full least squares solution

LARS Visualization





Model Assumptions and Properties

Assumptions:

- Normalization of features $N(E[X_j] = 0,\ Var[X_j] = 1)$

Properties:

- Selects at most N features
- Arbitrarily selects one feature from a correlated group of features

Applications:

- Time Series
- Spatial

Simulation 1 - Analysis of Prostate Cancer Data

- $N = 324$ patients and $p = 48538$ measurements total
- Split data into blocks of $N = 20$, $p = 2181$

Method	Validation errors/108	Degrees of freedom	Number of sites	s_1	s_2
Nearest shrunken centroids	30		227		
Lasso	16	60	40	83	164
Fusion	18	102	2171	16	32
Fused lasso	16	103	218	113	103

UNIVERSITY OF TORONTO

Simulation 2 - Leukemia Classification by using Microarrays

- $N = 38$ samples and $p = 7129$
- No prespecified order of features
- Optimized tuning parameters using cross-validation

Method	**10-fold cross-validation error**	**Test error**	**Number of genes**
(1) Golub *et al.* (1999) (50 genes)	3/38	4/34	50
(2) Nearest shrunken centroid (21 genes)	1/38	2/34	21
(3) Lasso, 37 degrees of freedom ($s_1 = 0.65, s_2 = 1.32$)	1/38	1/34	37
(4) Fused lasso, 38 degrees of freedom ($s_1 = 1.08, s_2 = 0.71$)	1/38	2/34	135
(5) Fused lasso, 20 degrees of freedom ($s_1 = 1.35, s_2 = 1.01$)	1/38	4/34	737
(6) Fusion, 1 degree of freedom	1/38	12/34	975

UNIVERSITY OF TORONTO

Conclusion

Pros:

- Identifies more true non-zero coefficients than lasso
- Less reliant on having ordered features than fusion

Cons:

- Computationally expensive

Q & A

Q&A