Le Lasso
La solution component wise

S.Canu
MLA ITI /INSA de Rouen Normandie

19 octobre 2026

Le Lasso

- Données
  - $X \in \mathbb{R}^{n \times p}$ les p variables observées n fois
  - $y \in \mathbb{R}^n$ la variable réponse
- Inconnues
  - $\beta \in \mathbb{R}^p$
- hypothèses : pas de biais (pas de terme constant)
  - $mean(X) = 0$
  - $\|X_j\|^2 = 1$
  - $mean(y) = 0$

le cout pénalisé

$$J_\lambda(\beta) \;=\; \tfrac{1}{2}\|X\beta - y\|^2 + \lambda\|\beta\|_1 \;,$$

Les conditions d'optimalité

$$0 \in \partial J_\lambda(\beta) \;=\; X^\top(X\beta - y) + \lambda\partial(\|\beta\|_1) \;,$$

Algo CW pour le Lasso

$$J_\lambda(\beta) \;=\; \tfrac{1}{2}\|X\beta - y\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| \;,$$

Algo CW pour le Lasso

itérer, tant que on n'a pas convergé :

Pour $j = 1, p$
  fixer toutes les variables sauf β_j
  résoudre le Lasso en une dimension :
$$\hat{\beta}_j \;=\; \arg\min_{\beta \in \mathbb{R}} J_\lambda^{(j)}(\beta) \tag{1}$$

Le cout du Lasso en une dimension

$$J_\lambda^{(j)}(\beta) = \tfrac{1}{2}\|X_j\beta - \left(y - X\beta^{(-j)}\right)\|^2 + \lambda|\beta|$$

Le Lasso en 1d

$$\min_{\beta\in\mathbb{R}} J_\lambda^{(1d)}(\beta) \qquad \text{avec} \quad J_\lambda^{(1d)}(\beta) = \tfrac{1}{2}\|\boldsymbol{x}\beta - \boldsymbol{r}\|^2 + \lambda|\beta|$$

$$\partial_\beta J_\lambda^{(1d)}(\beta) = \boldsymbol{x}^\top(\boldsymbol{x}\beta - \boldsymbol{r}) + \begin{cases} \lambda\alpha & \text{if } \beta = 0 \\ \lambda\text{sign}(\beta) & \text{else.} \end{cases}$$

$$0 \in \partial_\beta J_\lambda^{(1d)}(\beta) \quad \Leftrightarrow \quad \begin{cases} \exists\alpha \in [-1,1], -\boldsymbol{x}^\top\boldsymbol{r} + \lambda\alpha = 0 & \text{if } \beta = 0 \\ \|\boldsymbol{x}\|^2\beta - \boldsymbol{x}^\top\boldsymbol{r} + \lambda\text{sign}(\beta) = 0 & \text{else.} \end{cases}$$

The differential part :

$$\begin{cases} \text{if } \beta > 0 \quad \beta = \frac{\boldsymbol{x}^\top\boldsymbol{r} - \lambda}{\|\boldsymbol{x}\|^2} & \text{it works when } \boldsymbol{x}^\top\boldsymbol{r} > \lambda \\ \text{if } \beta < 0 \quad \beta = \frac{\boldsymbol{x}^\top\boldsymbol{r} + \lambda}{\|\boldsymbol{x}\|^2} & \text{it works when } \boldsymbol{x}^\top\boldsymbol{r} < -\lambda \end{cases}$$

It remains the non differential part : when $-\lambda < \boldsymbol{x}^\top\boldsymbol{r} < \lambda$ then $\beta = 0$

Le Lasso en 1d

The differential part :

$$\begin{cases} \text{if } \beta > 0 \quad \beta = \frac{\boldsymbol{x}^\top \boldsymbol{r} - \lambda}{\|\boldsymbol{x}\|^2} & \text{it works when } \boldsymbol{x}^\top \boldsymbol{r} > \lambda \\ \text{if } \beta < 0 \quad \beta = \frac{\boldsymbol{x}^\top \boldsymbol{r} + \lambda}{\|\boldsymbol{x}\|^2} & \text{it works when } \boldsymbol{x}^\top \boldsymbol{r} < -\lambda \end{cases}$$

It remains the non differential part : when $-\lambda < \boldsymbol{x}^\top \boldsymbol{r} < \lambda$ then $\beta = 0$

if $\|\boldsymbol{x}\|^2 = 1$

$$\hat{\beta}_{Lasso1d} = \begin{cases} 0 & \text{if } |\boldsymbol{x}^\top \boldsymbol{r}| \leq \lambda \\ \text{sign}(\boldsymbol{x}^\top \boldsymbol{r})(|\boldsymbol{x}^\top \boldsymbol{r}| - \lambda) & \text{else.} \end{cases}$$

Or in one line :

$$\text{sign}(\boldsymbol{x}^\top \boldsymbol{r}) \max\big((|\boldsymbol{x}^\top \boldsymbol{r}| - \lambda), 0\big)$$

Algo CW pour le Lasso

Pour $j = 1, p$: $\hat{\beta}_j = \arg\min_{\beta \in \mathbb{R}} J^{(j)}(\beta)$

$$\begin{aligned} J^{(j)}(\beta) &= \frac{1}{2}\|y - X\beta^{(-j)} - X_j\beta\|^2 + \lambda|\beta| \\ &= \frac{1}{2}\|\boldsymbol{r} - X_j\beta\|^2 + \lambda|\beta| \end{aligned} \tag{2}$$

avec $\boldsymbol{r} = y - X\beta^{(-j)}$ et $\beta^{(-j)} = (\beta_1, \ldots, \beta_{j-1}, 0, \beta_{j+1}, \ldots, \beta_p)$

Maths	Info
Pour $j = 1, p$	for j in range(p) :
$\beta^{(-j)} = (\beta_1, .., \beta_{j-1}, 0, \beta_{j+1}, .., \beta_p)$	$bj = beta$; $bj[j] = 0$;
$r = y - X\beta^{(-j)}$	$r = y - X@bj$
$\hat{\beta}_j = \arg\min_{\beta \in \mathbb{R}} J^{(j)}(\beta, X_j, r)$	$beta[j] = \text{sign}(\boldsymbol{x}^\top \boldsymbol{r}) \max\big((\lvert\boldsymbol{x}^\top \boldsymbol{r}\rvert - \lambda), 0\big)$
fin de pour	end

Codage

1. let's begin with $\beta = 0$
2. in that case $r \leftarrow y$ since $X\beta = 0$
3. fit a single (the jth) component of vector β :
   $\beta[j] = \text{sign}(\boldsymbol{x}^\top \boldsymbol{r}) \max\big((|\boldsymbol{x}^\top \boldsymbol{r}| - \lambda), 0\big)$
4. update r : $r \leftarrow r - X[:, j]\beta[j]$
5. choose another component j'
6. $r \leftarrow r + X[:, j']\beta[j']$
7. fit the other component j' : $\beta[j'] = \text{sign}(\boldsymbol{x}^\top \boldsymbol{r}) \max\big((|\boldsymbol{x}^\top \boldsymbol{r}| - \lambda), 0\big)$
8. update r : $r \leftarrow r - X[:, j']\beta[j']$
9. . . .iterate (go to 5)

LLM ?

Expliquer l'algorithme component-wise pour résoudre le problème de la régression Lasso Réécrire le problème d'optimisation du lasso comme une fonction d'une seule variable

Sources

[PDF] Le LASSO - Université de Lille
math · 1

[PDF] Régression pénalisée : le Lasso
pbil · 2

Lasso (statistiques) - Wikipédia
fr.wikipedia · 3

View 5 more

Perplexity

Pour expliquer l'algorithme component-wise (ou par coordonnées) pour résoudre le problème de la régression Lasso, commençons par réécrire le problème d'optimisation du Lasso comme une fonction d'une seule variable.

Réécriture du problème Lasso

Le problème d'optimisation du Lasso s'écrit généralement sous la forme suivante :

$$\min_\beta \frac{1}{2n}\|Y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

où Y est le vecteur des observations, X la matrice des variables explicatives, β le vecteur des coefficients à estimer, n le nombre d'observations, et λ le paramètre de régularisation [1].

Pour réécrire ce problème en fonction d'une seule variable, nous allons nous concentrer sur un coefficient β_j à la fois, en considérant tous les autres coefficients comme fixes. Cela nous donne :

$$\min_{\beta_j} \frac{1}{2n}\|r_j - x_j\beta_j\|_2^2 + \lambda|\beta_j|$$

où $r_j = Y - \sum_{k \neq j} x_k\beta_k$ est le résidu partiel (c'est-à-dire le résidu sans prendre en compte la j-ème variable), et x_j est la j-ème colonne de X [2].

Algorithme component-wise

L'algorithme component-wise pour résoudre le problème Lasso fonctionne comme suit :

1. **Initialisation** : Commencer avec un vecteur initial $\beta^{(0)}$, généralement tous les coefficients sont mis à zéro.
2. **Itération** : Pour chaque itération t et pour chaque coefficient j :

Lasso et Elastic Net

Le Lasso

$$J_\lambda(\beta) \;=\; \tfrac{1}{2}\|X'\beta - y'\|^2 + \lambda\|\beta\|_1 \;,$$

Elastic Net

$$J_{\mathsf{el}}(\beta) \;=\; \tfrac{1}{2}\|X\beta - y\|^2 + \lambda\|\beta\|_1 + \gamma\|\beta\|_2^2 \;,$$

Lasso = Elastic Net with $\gamma = 0, X = X', y = y'$

Elastic Net = Lasso with $X' = (X^\top, \sqrt{\gamma} I_p)^\top, y' = (y^\top, 0_p)^\top$

$$\|X\beta - y\|_2^2 + \underbrace{\gamma\,\|\beta\|_2^2}_{\|\sqrt{\gamma}\beta\|_2^2} = \Bigg\| \underbrace{\begin{bmatrix} X \\ \sqrt{\gamma} I_p \end{bmatrix}}_{=:X'} \beta - \underbrace{\begin{bmatrix} y \\ 0_p \end{bmatrix}}_{=:\underline{y}'} \Bigg\|_2^2 = \|X'\beta - y'\|_2^2 \,.$$

Regularization and variable selection via the elastic net H Zou, T Hastie, 2005

Reweigthed least square (Lasso and Ridge)

$$J_\lambda(\beta) \;=\; \tfrac{1}{2}\|X\beta - y\|^2 + \lambda \sum_{j=1}^{p} |\beta_j| = \frac{\beta_j^2}{|\beta_j|} \;,$$

the idea : iterate the weighted rigde towards a fixed point

$$\beta^{(k+1)} \;=\; \arg\min_{\beta} \tfrac{1}{2}\|X\beta - y\|^2 + \lambda \sum_{j=1}^{p} \frac{\beta_j^2}{|\beta_j^{(k)}|} = w_j\ \beta_j^2 \;,$$

with $w_j = 1/|\beta_j^{(k)}|$, that is

$$\beta^{(k+1)} \;=\; (X^\top X + \lambda W)^{-1} X^\top y \qquad \text{with } diag(W) = \frac{1}{|\beta_j^{(k)}|}$$

W = Identité
Tantque on n'a pas convergé
  $\beta = \arg\min \frac{1}{2}\|X\beta - y\|^2 + \lambda \beta^\top W \beta$
  $W = diag(1/|\beta|)$
fin de tantque

The adaptive Lasso

$$J_\lambda(\beta) \;=\; \tfrac{1}{2}\|X\beta - y\|^2 + \lambda \sum_{j=1}^{p} w_j |\beta_j| \;,$$

$$w_j = \frac{1}{|\hat{\beta}_j^{(\mathsf{ls})}|^\gamma}$$

```
w = np.linalg.solve(X.T@X,X.T@y)
w = 1/np.sqrt(np.abs(w))
X_c = X/w
c = mon_lasso(X_c,y,lambda)
beta = c/w
```

The Adaptive Lasso and Its Oracle Properties, H. Zou, JASA, 2012