Apprentissage Statistique Automatique I

Méthodes de régularisation (RIDGE - LASSO)

Andrés F. López-Lopera
Université Polytechnique Hauts-de-France (UPHF)

Thèmes

1. Régression Ridge
   - Point de vue par SVD
   - Point de vue par pénalisation – Ridge / Tikhonov
   - Analyse du biais, de la variance et du risque
   - Validation Croisée (CV)

2. Régression Lasso
   - Sélection de variables et parcimonie
   - Améliorations et extensions du Lasso

Régression linéaire

· Dans un cadre d'apprentissage, on cherche à créer une règle de régression dans la classe

$$\mathcal{M}_\beta := \left\{ f(x) = \beta_0 + \sum_{j=1}^{d} \beta_j x_j, \beta = (\beta_0, \ldots, \beta_d)^\top \in \mathbb{R}^{d+1} \right\} \tag{1}$$

avec $x = (x_1, \ldots, x_d)^\top \in \mathbb{R}^d$

· On suppose qu'il existe β_* (inconnu) t.q.

$$y = X\beta_* + \varepsilon \tag{2}$$

- $y \in \mathbb{R}^n$ est le vecteur des observations
- $X \in \mathbb{R}^{n \times p}$ ($p = d + 1$) est la matrice des variables explicatives
- $\beta_* \in \mathbb{R}^p$ est les vrais paramètres du modèle que l'on veut retrouver
- $\varepsilon \in \mathbb{R}^n$ est le vecteur avec des bruits additifs

Solution de moindres carrés

· La solution de moindres carrés (*Least Square, LS*) :

$$\widehat{\beta}^{\mathrm{LS}} = (X^\top X)^{-1} X^\top y$$

Limitations.

- $M = X^\top X$ doit être une matrice de plein rang, *i.e.*, $\mathrm{rg}(X) = p$
- Si les plus petites valeurs propres de M s'approchent de zéro alors la solution numérique de M^{-1} n'est pas stable !

Régression Ridge

Point de vue par SVD

Theorem (Décomposition SVD [Golub and Van Loan, 2013])

Pour toute matrice $X \in \mathbb{R}^{n\times p}$, il existe deux matrices orthogonales $U = [u_1, \ldots, u_n] \in \mathbb{R}^{n\times n}$ et $V = [v_1, \ldots, v_p] \in \mathbb{R}^{p\times p}$, telles que

$$U^\top X V = \Sigma \in \mathbb{R}^{n\times p},$$

avec $\Sigma = \mathrm{diag}(s_1, \ldots, s_r)$, $s_1 \geq \cdots \geq s_r > 0$, et $r = \mathrm{rang}(X)$.

$$X = U\Sigma V^\top = [u_1, \ldots, u_n] \left[\begin{array}{ccc|c} s_1 & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & s_r & \\ \hline & 0 & & 0 \end{array}\right] \begin{bmatrix} v_1^\top \\ \vdots \\ v_p^\top \end{bmatrix} = \sum_{i=1}^{r} s_i u_i v_i^\top$$

- Une solution des moindres carrés est alors :

$$\widehat{\beta}^{\mathrm{LS}} = (U\Sigma V^\top)^{-1}\, y = (V\Sigma^{-1}U^\top)\, y = \left(\sum_{i=1}^{r} \frac{1}{s_i} v_i u_i^\top\right) y$$

Rem : Si s_i s'approche de zéro, la solution de la SVD n'est pas stable !

Point de vue par SVD

- Une solution des moindres carrés doit vérifier :

$$\begin{aligned} X^\top X\beta = X^\top y &\Leftrightarrow (U\Sigma V^\top)^\top (U\Sigma V^\top)\beta = (U\Sigma V^\top)^\top y \\ &\Leftrightarrow V\Sigma^\top \Sigma V^\top \beta = V\Sigma^\top U^\top y \\ &\Leftrightarrow \Sigma^\top \Sigma V^\top \beta = \Sigma^\top U^\top y \end{aligned}$$

- $\Sigma^\top \Sigma \in \mathbb{R}^{p\times p}$ est diagonale avec r éléments non nuls qui sont $s_1^2, \ldots, s_r^2$

$$\Sigma^\top \Sigma = \left[\begin{array}{ccc|c} s_1^2 & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & s_r^2 & \\ \hline & 0 & & 0 \end{array}\right]$$

CERAMATHS Université Polytechnique HAUTS-DE-FRANCE

Point de vue par SVD

- Il est possible de considérer une alternative régularisée
- Pour $\lambda > 0$ (paramètre de régularisation),

$$\begin{array}{ccc}
\Sigma^\top \Sigma & \to & \Sigma^\top \Sigma + \lambda I_p \\
\left[\begin{array}{ccc|c} s_1^2 & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & s_r^2 & \\ \hline & 0 & & 0 \end{array}\right] & \to & \left[\begin{array}{ccc|c} s_1^2 + \lambda & \cdots & 0 & \\ \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & s_r^2 + \lambda & \\ \hline & 0 & & \lambda I_{p-r} \end{array}\right]
\end{array}$$

- Cela veut dire qu'on ajoute à toutes les valeurs propres de $X^\top X$ un terme $\lambda > 0$ "petit" afin de rassurer la inversion de la matrice
- De cette manière, on a $(\Sigma^\top \Sigma + \lambda I_p) V^\top \beta = \Sigma^\top U^\top y$, et

$$\beta = V(\Sigma^\top \Sigma + \lambda I_p)^{-1} \Sigma^\top U^\top y$$

Point de vue par SVD

- En sachant que $\Sigma = U^\top X V$ (définition SVD), on trouve ensuite **[exercice]**

$$\begin{aligned}\widehat{\beta}^{\text{Ridge}}_{\lambda} &= V(\Sigma^\top \Sigma + \lambda I_p)^{-1}\Sigma^\top U^\top y\\ &= (X^\top X + \lambda I_p)^{-1} X^\top y\end{aligned}$$

Rappel : sous l'hypothèse de plein rang : $\widehat{\beta}^{\text{LS}} = (X^\top X)^{-1} X^\top y$

Rem :

- $\lim\limits_{\lambda \to 0^+} \widehat{\beta}^{\text{Ridge}}_{\lambda} = \widehat{\beta}^{\text{LS}}$
- $\lim\limits_{\lambda \to +\infty} \widehat{\beta}^{\text{Ridge}}_{\lambda} = 0 \in \mathbb{R}^p$

Point de vue par SVD

- La prédiction associée s'obtient ainsi :

$$\widehat{y} = X\widehat{\beta}_{\lambda}^{\text{Ridge}} = \underbrace{X(X^\top X + \lambda I_p)^{-1}X^\top}_{H_\lambda} y$$

Rem :

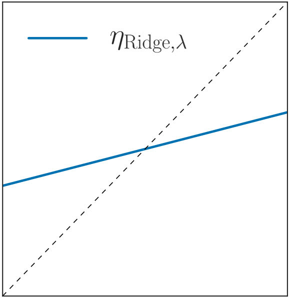
- L'estimateur $\widehat{y}$ est linéaire en y
- L'équivalent de la matrice chapeau (*hat matrix*) est

$$H_\lambda = X(X^\top X + \lambda I_p)^{-1}X^\top = \sum_{j=1}^{\text{rg}(X)} \frac{s_j^2}{s_j^2 + \lambda} u_j u_j^\top$$

Cas orthogonal

- Si X est une matrice orthogonal, *i.e.*, $X^\top X = I_p$:

$$
\begin{aligned}
\widehat{\beta}_{\lambda}^{\text{Ridge}} &= (X^\top X + \lambda I_p)^{-1} X^\top y \\
&= \frac{1}{1+\lambda} X^\top y \\
\widehat{y} &= \frac{1}{1+\lambda} y \\
&= [\eta_{rdg,\lambda}(y_1), \ldots, \eta_{rdg,\lambda}(y_n)]^\top
\end{aligned}
$$



Rem :

- Cas classique en traitement du signal (peu fréquent en statistique)
- La fonction réelle $\eta_{rdg,\lambda}$ est une contraction linéaire (*shrinkage*)

Astuce du noyau (*kernel trick*)

· En utilisant l'astuce du noyau, il est possible de démontrer **[exercice]**

$$X^\top (XX^\top + \lambda I_n)^{-1} y = (X^\top X + \lambda I_p)^{-1} X^\top y$$

Piste: utiliser l'identité matricielle de Woodbury

$$(Z + UWV^\top)^{-1} = Z^{-1} - Z^{-1}U(W^{-1} + V^\top Z^{-1}U)^{-1}V^\top Z^{-1}$$

· On note que

$\Leftarrow$ on doit résoudre un système $n \times n$

$\Rightarrow$ on doit résoudre un système $p \times p$

· Selon si $n > p$ ou $n \leq p$, une méthode qui cherche à trouver une solution de Ridge par inversion peut préférer l'une des deux formulations

Rem : Cette astuce est aussi utile pour les méthodes à noyaux (e.g., SVM)

Démonstration.

Identité matricielle de Woodbury

$$(Z + UWV^\top)^{-1} = Z^{-1} - Z^{-1}U(W^{-1} + V^\top Z^{-1}U)^{-1}V^\top Z^{-1}$$

- En utilisant l'identité matricielle de Woodbury, on a

$$(\lambda I_n + XX^\top)^{-1} = \frac{1}{\lambda} I_n - \frac{1}{\lambda^2} X \left(I_p + \frac{1}{\lambda} X^\top X\right)^{-1} X^\top$$

- En reportant le résultat sur l'expression initial, on obtient

$$\begin{aligned} X^\top (XX^\top + \lambda I_n)^{-1} y &= \frac{1}{\lambda} X^\top y - \frac{1}{\lambda^2} X^\top X \left(I_p + \frac{1}{\lambda} X^\top X\right)^{-1} X^\top y \\ &= \left[\frac{1}{\lambda} I_p - \frac{1}{\lambda} X^\top X \left(\lambda I_p + X^\top X\right)^{-1} \frac{\lambda}{\lambda}\right] X^\top y \\ &= \left(\lambda I_p + X^\top X\right)^{-1} X^\top y \end{aligned}$$

Point de vue par pénalisation – Ridge / Tikhonov

$$\widehat{\beta}_{\lambda}^{\text{Ridge}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \underbrace{\|y - X\beta\|_2^2}_{\text{attache aux données}} \quad + \quad \underbrace{\lambda \|\beta\|_2^2}_{\text{régularisation}} \tag{3}$$

· L'estimateur Ridge est unique pour un λ fixé

Rem : On retrouve de nouveau les cas limites :

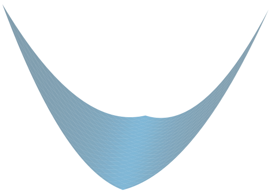
- $\lim\limits_{\lambda \to 0^+} \widehat{\beta}_{\lambda}^{\text{Ridge}} = \widehat{\beta}^{\text{LS}}$ (solution de norme $\|\cdot\|_2$ minimale)
- $\lim\limits_{\lambda \to +\infty} \widehat{\beta}_{\lambda}^{\text{Ridge}} = 0 \in \mathbb{R}^p$

· Lien avec les conditions nécessaires d'optimalité (CNO) :

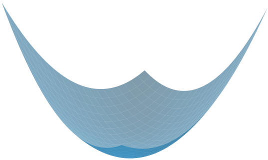
$$f(\beta) = \frac{\|y - X\beta\|_2^2}{2} + \frac{\lambda \|\beta\|_2^2}{2}$$

$$\nabla f(\beta) = X^\top (X\beta - y) + \lambda \beta = 0 \quad \Leftrightarrow \quad (X^\top X + \lambda I_p)\beta = X^\top y$$

Illustration avec $p = 2$



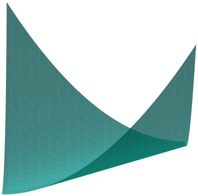
LS



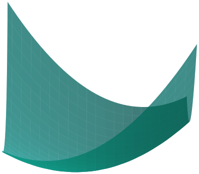
Ridge

Régulariser : simplifie le problème quand il est mal conditionné

Illustration avec $p = 2$



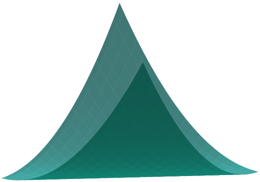
LS



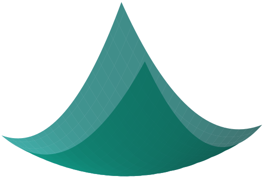
Ridge

Régulariser : simplifie le problème quand il est mal conditionné

Illustration avec $p = 2$



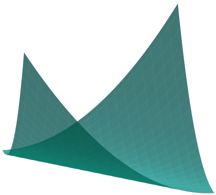
LS



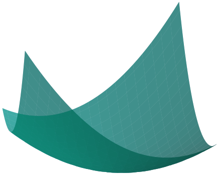
Ridge

Régulariser : simplifie le problème quand il est mal conditionné

Illustration avec $p = 2$

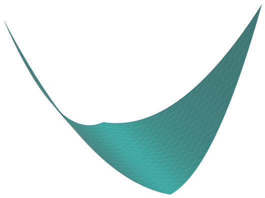


LS

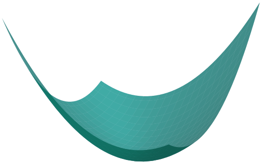


Ridge

Régulariser : simplifie le problème quand il est mal conditionné



LS



Ridge

Régulariser : simplifie le problème quand il est mal conditionné

Interprétation contrainte

· Un problème de la forme "Lagrangienne" suivante

$$\underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} + \underbrace{\frac{\lambda}{2}\|\beta\|_2^2}_{\text{régularisation}} \tag{4}$$

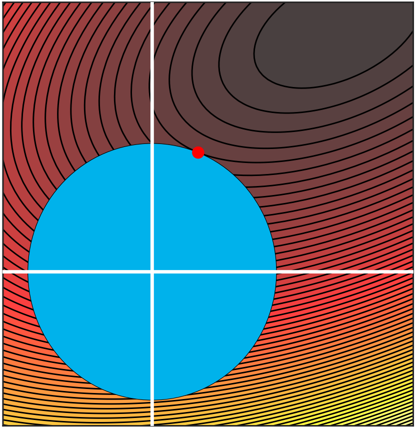
admet pour un certain $T > 0$ la même solution que :

$$\begin{cases} \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2 \\ \text{t.q. } \|\beta\|_2^2 \leq T \end{cases}$$

Rem : Le lien $T \leftrightarrow \lambda$ n'est pas explicite !

- Si $T \to 0^+$, on retrouve le vecteur nul $\widehat{\beta}_\lambda^{\text{Ridge}} = 0 \in \mathbb{R}^p$ (*i.e.*, $\lambda = +\infty$)
- Si $T \to +\infty$, on retrouve $\widehat{\beta}_\lambda^{\text{Ridge}} = \widehat{\beta}^{\text{LS}}$ (non contraint, *i.e.*, $\lambda = 0^+$)

Lignes de niveau et ensemble de contraintes



Optimisation sous contraintes ℓ_2

Ridge vs PCA

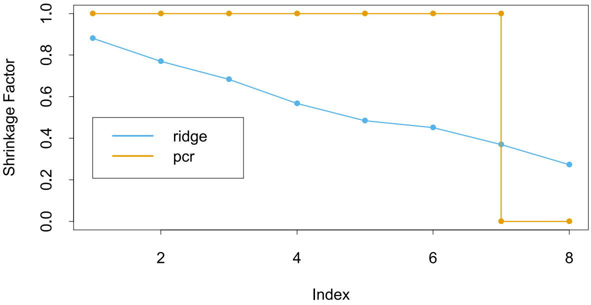


FIGURE 3.17. *Ridge regression shrinks the regression coefficients of the principal components, using shrinkage factors $d_j^2/(d_j^2 + \lambda)$ as in (3.47). Principal component regression truncates them. Shown are the shrinkage and truncation patterns corresponding to Figure 3.7, as a function of the principal component index.*

[Hastie et al., 2009]

Point normalisation et centrage

· Pour que la pénalisation contraigne de manière similaire toutes les variables, on peut normaliser les p variables

- Centrer l'observation et les variables explicatives $\Rightarrow$ pas de coefficient pour la variable constante (donc pas de contrainte)

$$\widehat{\beta}_\lambda^{\text{Ridge}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \frac{1}{2}\|y - X\beta - \beta_0 \mathbb{1}_n\|_2^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \tag{5}$$

- Ne pas centrer les variables explicatives $\Rightarrow$ ne pas mettre de contrainte sur la variable constante

Alternative. Si l'on ne normalise pas, on peut changer la pénalité en

$$\widehat{\beta}_\lambda^{\text{Ridge}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \frac{1}{2}\|y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} \alpha_j \beta_j^2 \qquad (\text{e.g., } \alpha_j = \|x_j\|^2) \tag{6}$$

Rem : Pour la validation croisée on utilisera plus naturellement $\frac{1}{2n}\|y - X\beta\|_2^2$ que $\frac{1}{2}\|y - X\beta\|_2^2$ pour conserver l'amplitude de λ

CERAMATHS Université Polytechnique HAUTS-DE-FRANCE

Analyse du biais par la SVD

- Hypothèse : bruit “blanc” $y = X\beta^\star + \varepsilon$ avec $\mathbb{E}(\varepsilon) = 0$:

$$\mathbb{E}(\widehat{\beta}_\lambda^{\text{Ridge}}) = \mathbb{E}\left((X^\top X + \lambda I_p)^{-1} X^\top y\right)$$

Analyse du biais par la SVD

- Hypothèse : bruit “blanc” $y = X\beta^\star + \varepsilon$ avec $\mathbb{E}(\varepsilon) = 0$:

$$
\begin{aligned}
\mathbb{E}(\widehat{\beta}_\lambda^{\text{Ridge}}) &= \mathbb{E}\left((X^\top X + \lambda I_p)^{-1} X^\top y\right) \\
&= \mathbb{E}\left((X^\top X + \lambda I_p)^{-1} X^\top [X\beta^\star + \varepsilon]\right) \\
&= (X^\top X + \lambda I_p)^{-1} X^\top X\beta^\star \\
&= \sum_{i=1}^{\mathrm{rg}(X)} \frac{s_i^2}{s_i^2 + \lambda} v_i v_i^\top \beta^\star
\end{aligned}
$$

Rem :

- On retrouve $\mathbb{E}(\widehat{\beta}^{\text{LS}}) = \sum_{i=1}^{\mathrm{rg}(X)} v_i v_i^\top \beta^\star$ quand $\lambda \to 0$
- Le biais vaut $b = -\lambda(X^\top X + \lambda I_p)^{-1}\beta^\star$:

$$
\mathbb{E}(\widehat{\beta}_\lambda^{\text{Ridge}}) - b = (X^\top X + \lambda I_p)^{-1} X^\top X \beta^\star + \lambda (X^\top X + \lambda I_p)^{-1}\beta^\star = \beta^\star
$$

Analyse de la variance par la SVD

- Hypothèse : bruit blanc et homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon) = 0$, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$V_\lambda^{\text{Ridge}} = \text{cov}(\widehat{\beta}_\lambda^{\text{Ridge}}) = \text{cov}((X^\top X + \lambda I_p)^{-1} X^\top [X\beta^\star + \varepsilon])$$

CERAMATHS | Université Polytechnique HAUTS-DE-FRANCE

Analyse de la variance par la SVD

· Hypothèse : bruit blanc et homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon) = 0$, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$\begin{aligned}
V_\lambda^{\text{Ridge}} = \operatorname{cov}(\widehat{\beta}_\lambda^{\text{Ridge}}) &= \operatorname{cov}((X^\top X + \lambda I_p)^{-1} X^\top [X\beta^\star + \varepsilon]) \\
&= \operatorname{cov}((X^\top X + \lambda I_p)^{-1} X^\top \varepsilon) \\
&= (X^\top X + \lambda I_p)^{-1} X^\top \mathbb{E}(\varepsilon\,\varepsilon^\top)\, X (X^\top X + \lambda I_p)^{-1} \\
&= \sigma^2 (X^\top X + \lambda I_p)^{-1} X^\top X (X^\top X + \lambda I_p)^{-1} \\
&= \sum_{i=1}^{\operatorname{rg}(X)} \frac{s_i^2 \sigma^2}{(s_i^2 + \lambda)^2} v_i v_i^\top
\end{aligned}$$

Rem :

- On retrouve $V^{\text{LS}} = \sum_{i=1}^{\operatorname{rg}(X)} \frac{\sigma^2}{s_i^2} v_i v_i^\top$ quand $\lambda \to 0$
- On retrouve une variance nulle quand $\lambda \to \infty$

Analyse du risque par la SVD

- Hypothèse : modèle homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$R_{pred}(y, \widehat{y}) = \mathbb{E}\|X\beta^\star - X\widehat{\beta}_\lambda^{\text{Ridge}}\|_2^2$$

Analyse du risque par la SVD

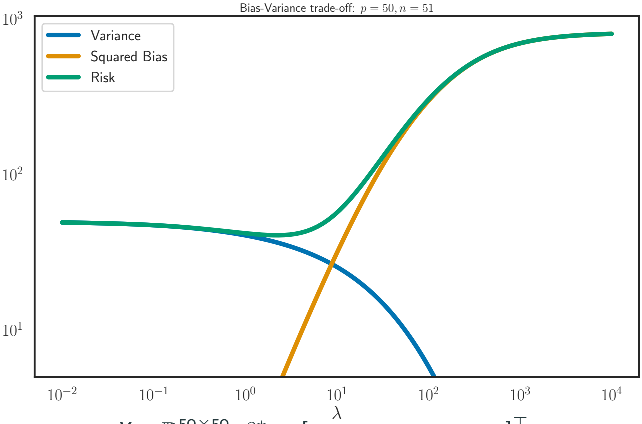
- Hypothèse : modèle homoscédastique (*i.e.*, $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 I_n$)

$$\begin{aligned}
R_{pred}(y, \widehat{y}) &= \mathbb{E}\|X\beta^\star - X\widehat{\beta}_\lambda^{\text{Ridge}}\|_2^2 \\
&= \mathbb{E}\left((\widehat{\beta}_\lambda^{\text{Ridge}} - \beta^\star)^\top X^\top X(\widehat{\beta}_\lambda^{\text{Ridge}} - \beta^\star)\right) \\
&= \mathbb{E}\Big(((X^\top X + \lambda I_p)^{-1} X^\top \varepsilon)^\top \, X^\top X \, (X^\top X + \lambda I_p)^{-1} X^\top \varepsilon\Big) \\
&\quad + \lambda^2 \beta^{\star\top}(X^\top X)(X^\top X + \lambda I_p)^{-2}\beta^\star \\
&= \sum_{i=1}^{\mathrm{rg}(X)} \frac{s_i^4 \sigma^2}{(s_i^2 + \lambda)^2} + \lambda^2 \beta^{\star\top}(X^\top X)(X^\top X + \lambda I_p)^{-2}\beta^\star
\end{aligned}$$

Rem :

- $\lim_{\lambda \to 0} R_{pred}(\beta^\star, \widehat{\beta}_\lambda^{\text{Ridge}}) = \mathrm{rg}(X)\sigma^2$
- $\lim_{\lambda \to \infty} R_{pred}(\beta^\star, \widehat{\beta}_\lambda^{\text{Ridge}}) = \|X\beta^\star\|_2^2$

Biais / Variance / Risque : exemple de simulation



$$X \in \mathbb{R}^{50\times 50}, \beta^\star = [2, 2, 2, 2, 2, 0, \dots, 0]^\top$$

Algorithmes pour la méthode *Ridge*

- 'svd' : méthode la plus stable, avantageuse pour calculer plusieurs λ car on ne "paye" la SVD qu'une fois
- 'cholesky' : décomposition matricielle proposant une formule fermée scipy.linalg.solve
- 'sparse_cg' : gradient conjugué utile dans les cas creux (*sparse*) et de grande dimension (baisser tol / max_iter)
- approche de type gradient stochastique si n est très grand

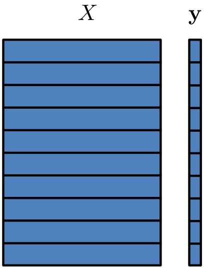
cf. le code des fonctions Ridge, ridge_path, RidgeCV dans le module linear_model de sklearn

Rem :

- On calcule rarement l'estimateur *Ridge* pour un λ, en général on en calcule plusieurs $(10, 100, \dots)$ et on cherche le meilleur
- Enjeu crucial de calculer des SVD de grandes tailles

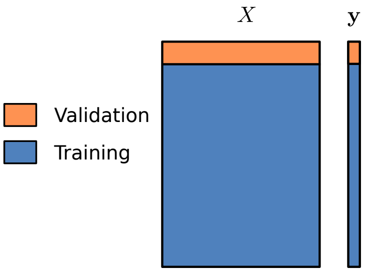
Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :



Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

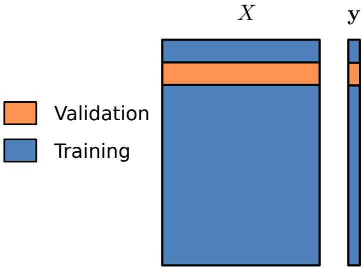


$$k = 1$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :



$k = 2$

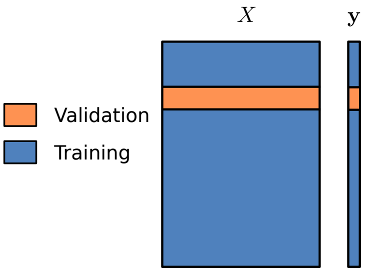
1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:

$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$

2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

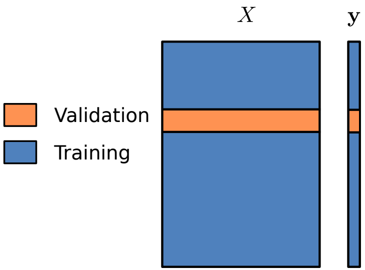
- Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :



$$k = 3$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \dots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \dots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \dots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

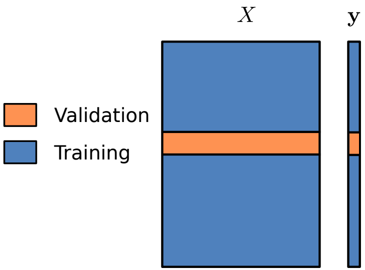


$k = 4$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

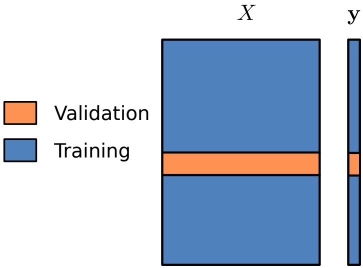


$$k = 5$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \dots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \dots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \dots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

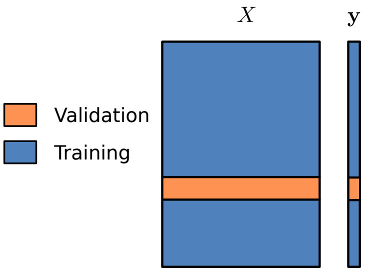


$$k = 6$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

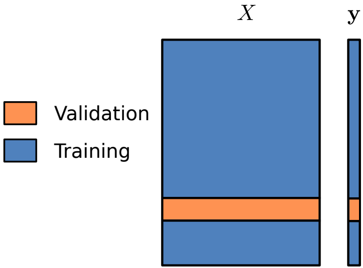


$$k = 7$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :

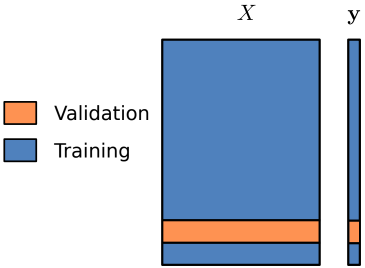


$k = 8$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Validation croisée K-fold ($K = 10$)

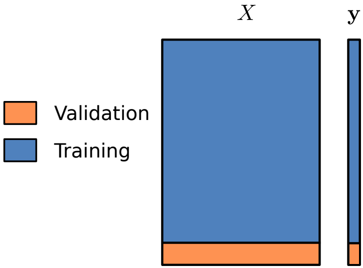
- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :



$$k = 9$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

- Choisir une grille de taille r de λ à tester : $\lambda_1, \ldots, \lambda_r$
- Diviser (X, y) selon les observations en K blocs (*fold*) :



$$k = 10$$

1. A partir des données d'entraînement, calculer les estimateurs pour $\lambda_1, \ldots, \lambda_r$:
$$\widehat{\beta}_{\lambda_1}, \ldots, \widehat{\beta}_{\lambda_r}$$
2. Calculer les erreurs de prédiction $\text{Erreur}_1^{(k)}, \ldots, \text{Erreur}_r^{(k)}$ sur l'ensemble de validation

Choix du paramètre. $\widehat{\lambda}_{CV} = \lambda_{\widehat{i}_{CV}}$ avec

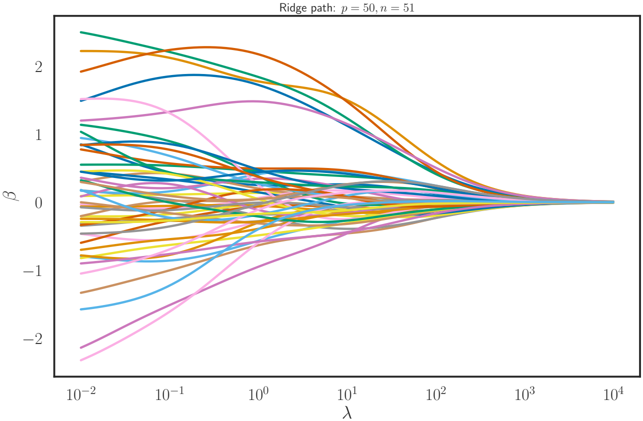
$$\widehat{i}_{CV} = \underset{i \in \{1, \ldots, r\}}{\text{argmin}} \, \overline{\text{Erreur}}_i$$

Re-calibration. Calculer $\widehat{\beta}_{\widehat{\lambda}_{CV}}$ sur toutes les observations (X, y)

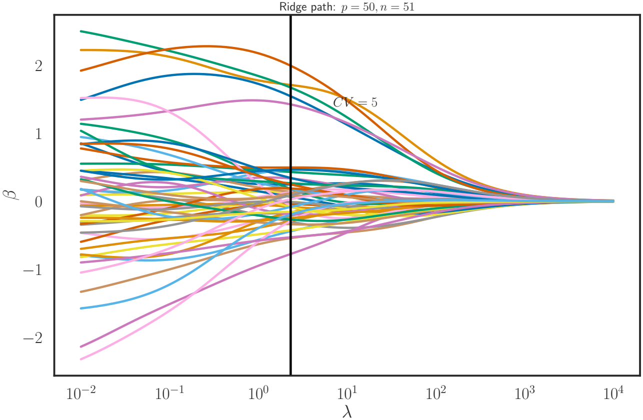
Conseils pratiques

- $K = n$ (stratégie "*leave-one-out*") : calcul efficace pour Ridge mais assez instable
- Choix habituels : $K = 5, 10$
- "Randomiser les observations" : observations dans un ordre aléatoire afin éviter des blocs de données trop similaires (chaque sous-bloc doit être représentatif de l'ensemble)
- Prédiction : moyenner les meilleurs estimateurs obtenus plutôt que de re-calibrer sur toutes les données

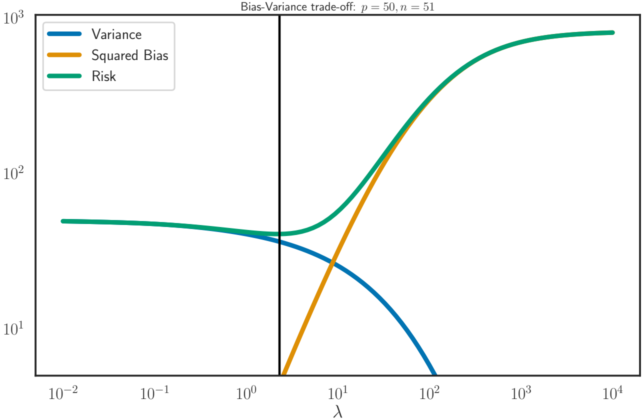
Choix de λ : exemple avec $CV = 5$



Choix de λ : exemple avec $CV = 5$



Choix de λ : exemple avec $CV = 5$



Régression Lasso

Rappel point de vue par pénalisation – Ridge / Tikhonov

$$\widehat{\beta}_{\lambda}^{\text{Ridge}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \; \underbrace{\|y - X\beta\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda \|\beta\|_2^2}_{\text{régularisation}} \tag{7}$$

- Est-il possible de considérer d'autres normes pour la régularisation ?

La pseudo-norme ℓ_0

· Le **support** du vecteur $\beta \in \mathbb{R}^p$ est l'ensemble des indices des coordonnées non nulles :

$$\text{supp}(\beta) = \{j \in [\![1, p]\!], \theta_j \neq 0\}$$

· La **pseudo-norme** ℓ_0 du vecteur β est son nombre de coordonnées non-nulles :

$$\|\beta\|_0 = \text{card}\{j \in [\![1, p]\!], \theta_j \neq 0\}$$

Rem :

- $\|\cdot\|_0$ n'est pas une norme, $\forall t \in \mathbb{R}^*, \|t\beta\|_0 = \|\beta\|_0$
- $\|\cdot\|_0$ n'est pas non plus convexe, $\beta_1 = (1, 0, 1, \ldots, 0)$ $\beta_2 = (0, 1, 1, \ldots, 0)$ et $3 = \|\frac{\beta_1 + \beta_2}{2}\|_0 \geq \frac{\|\beta_1\|_0 + \|\beta_2\|_0}{2} = 2$

La régularisation ℓ_0

· Première tentative de méthode pénalisée pour introduire de la parcimonie : utiliser ℓ_0 pour la pénalisation / régularisation

$$\widehat{\beta}_{\lambda,\ell_0} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \ \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\beta\|_0}_{\text{régularisation}} \tag{8}$$

Problème combinatoire !!! (problème “NP-dur”)

· La résolution exacte nécessite de considérer tous les sous-modèles, *i.e.*, calculer les estimateurs pour tous les supports possibles; il y en a 2^p, ce qui requiert le calcul de 2^p moindres carrés !

Exemple.

- $p = 10$ possible : $\approx 10^3$ moindres carrés
- $p = 30$ impossible : $\approx 10^{10}$ moindres carrés

Rem : avancées récentes en MIP [Bertsimas et al., 2016]

La régularisation ℓ_1 (Lasso)

$$\widehat{\beta}_{\lambda,\ell_1} = \underset{\beta\in\mathbb{R}^p}{\operatorname{argmin}} \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\beta\|_1}_{\text{régularisation}} \tag{9}$$

où $\|\beta\|_1 = \sum_{j=1}^{p}|\beta_j|$

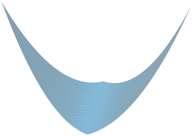
· On retrouve de nouveau les cas limites :

- $\lim_{\lambda\to 0^+} \widehat{\beta}_{\lambda,\ell_0} = \widehat{\beta}^{\text{LS}}$
- $\lim_{\lambda\to +\infty} \widehat{\beta}_{\lambda,\ell_0} = 0 \in \mathbb{R}^p$

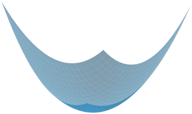
· $\widehat{\beta}_{\lambda,\ell_1}$ est connu comme l'estimateur **Lasso** (*Least Absolute Shrinkage and Selection Operator*) [Tibshirani, 1996]

Rem : $\widehat{\beta}_{\lambda}^{\text{Lasso}} := \widehat{\beta}_{\lambda,\ell_1}$ n'est pas toujours **unique** pour un λ fixé ; prendre par exemple deux colonnes identiques

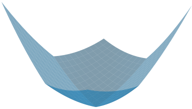
Illustration avec $p = 2$



LS

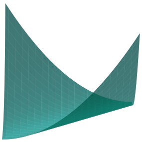


Ridge

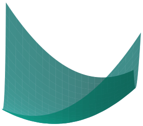


Lasso

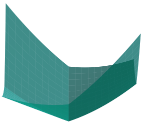
Illustration avec $p = 2$



LS

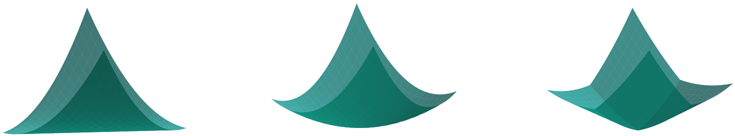


Ridge



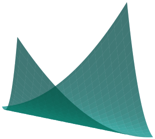
Lasso

Illustration avec $p = 2$

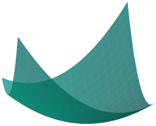


LS Ridge Lasso

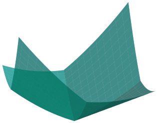
Illustration avec $p = 2$



LS

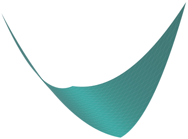


Ridge

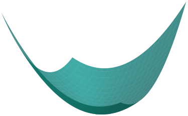


Lasso

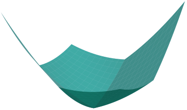
Illustration avec $p = 2$



LS



Ridge



Lasso

Interprétation contrainte

· Un problème de la forme "Lagrangienne" suivante

$$\widehat{\beta}_\lambda^{\text{Lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} \quad + \quad \underbrace{\frac{\lambda}{2}\|\beta\|_1}_{\text{régularisation}} \tag{10}$$

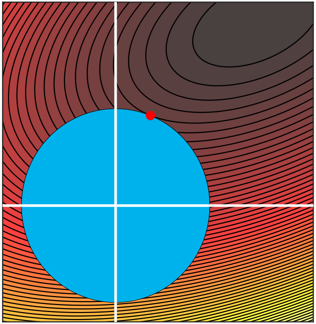
admet pour un certain $T > 0$ la même solution que :

$$\begin{cases} \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \|y - X\beta\|_2^2 \\ \text{t.q. } \|\beta\|_1 \leq T \end{cases}$$

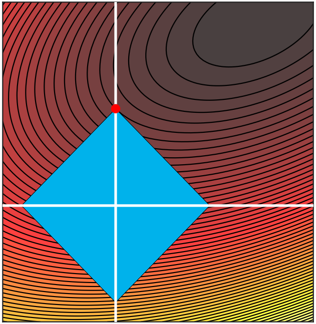
Rem : Le lien $T \leftrightarrow \lambda$ n'est pas explicite !

- Si $T \to 0^+$, on retrouve le vecteur nul $\widehat{\beta}_\lambda^{\text{Lasso}} = 0 \in \mathbb{R}^p$ (*i.e.*, $\lambda = +\infty$)
- Si $T \to +\infty$, on retrouve $\widehat{\beta}_\lambda^{\text{Lasso}} = \widehat{\beta}^{\text{LS}}$ (non contraint, *i.e.*, $\lambda = 0^+$)

Lignes de niveau et ensemble de contraintes



Optimisation sous contraintes ℓ_2



Optimisation sous contraintes ℓ_1

- Optimisation sous contrainte ℓ_1 : solution parcimonieuse !

Sous-gradients, sous-différentielles

Définitions

Pour $f : \mathbb{R}^d \to \mathbb{R}$ une fonction convexe, $u \in \mathbb{R}^d$ est un **sous-gradient** de f en x^*, si pour tout $x \in \mathbb{R}^d$ on a

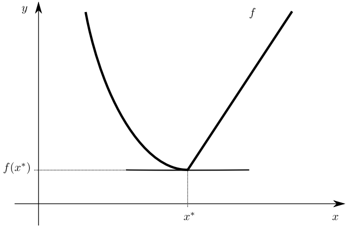
$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

La **sous-différentielle** est l'ensemble des sous-gradients :

$$\partial f(x^*) = \{u \in \mathbb{R}^d : \forall x \in \mathbb{R}^d, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem : si le sous-gradient est unique, on retrouve le gradient

· Visuellement cela correspond à des droites tangentes sur x^*



Sous-gradients, sous-différentielles

Définitions

Pour $f : \mathbb{R}^d \to \mathbb{R}$ une fonction convexe, $u \in \mathbb{R}^d$ est un **sous-gradient** de f en x^*, si pour tout $x \in \mathbb{R}^d$ on a

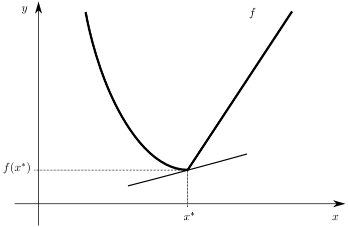
$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

La **sous-différentielle** est l'ensemble des sous-gradients :

$$\partial f(x^*) = \{u \in \mathbb{R}^d : \forall x \in \mathbb{R}^d, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem : si le sous-gradient est unique, on retrouve le gradient

· Visuellement cela correspond à des droites tangentes sur x^*



Sous-gradients, sous-différentielles

Définitions

Pour $f : \mathbb{R}^d \to \mathbb{R}$ une fonction convexe, $u \in \mathbb{R}^d$ est un **sous-gradient** de f en x^*, si pour tout $x \in \mathbb{R}^d$ on a

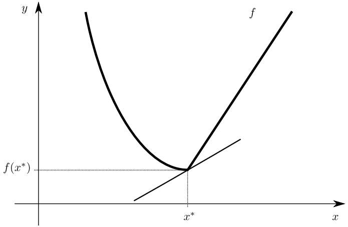
$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

La **sous-différentielle** est l'ensemble des sous-gradients :

$$\partial f(x^*) = \{u \in \mathbb{R}^d : \forall x \in \mathbb{R}^d, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem : si le sous-gradient est unique, on retrouve le gradient

· Visuellement cela correspond à des droites tangentes sur x^*



Sous-gradients, sous-différentielles

Définitions

Pour $f : \mathbb{R}^d \to \mathbb{R}$ une fonction convexe, $u \in \mathbb{R}^d$ est un **sous-gradient** de f en x^*, si pour tout $x \in \mathbb{R}^d$ on a

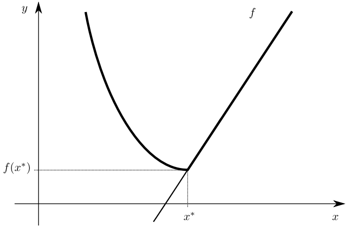
$$f(x) \geq f(x^*) + \langle u, x - x^* \rangle$$

La **sous-différentielle** est l'ensemble des sous-gradients :

$$\partial f(x^*) = \{u \in \mathbb{R}^d : \forall x \in \mathbb{R}^d, f(x) \geq f(x^*) + \langle u, x - x^* \rangle\}.$$

Rem : si le sous-gradient est unique, on retrouve le gradient

- Visuellement cela correspond à des droites tangentes sur x^*



Théorème

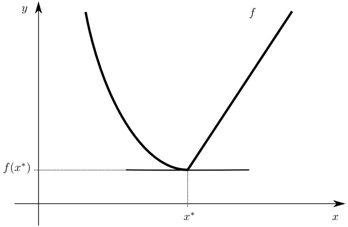
Un point x^* est un minimum d'une fonction convexe $f : \mathbb{R}^d \to \mathbb{R}$ si et seulement si $0 \in \partial f(x^*)$

Preuve : utiliser la définition des sous-gradients :

- 0 est un sous-gradient de f en x^* si et seulement si

$$\forall x \in \mathbb{R}^d, f(x) \geq f(x^*) + \langle 0, x - x^* \rangle$$

· V visuellement cela correspond à une tangente horizontale



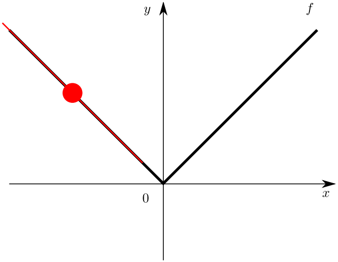
Sous-différentielle de la valeur absolue

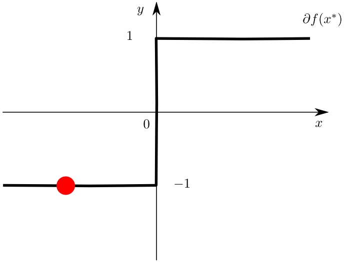
Fonction abs($\cdot$)

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign($\cdot$)

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$



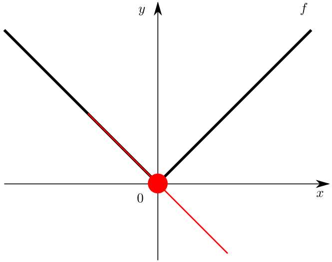


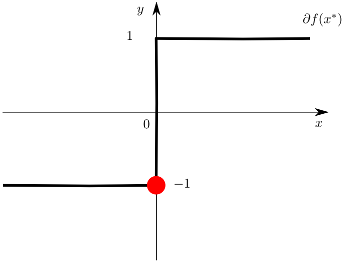
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$



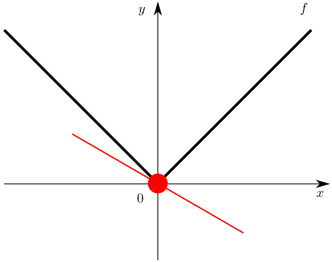


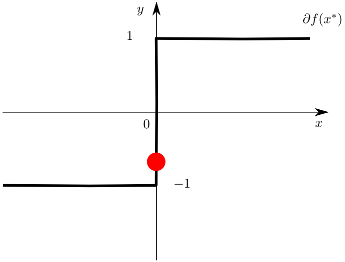
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$

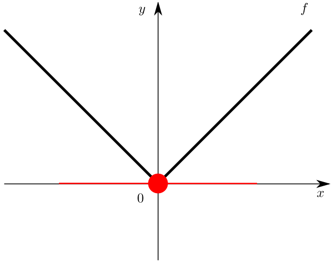




Sous-différentielle de la valeur absolue

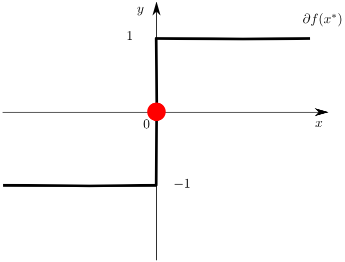
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



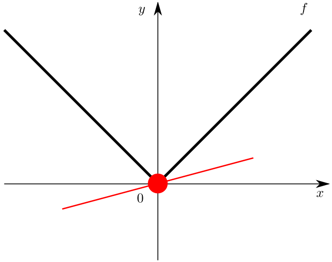
Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$



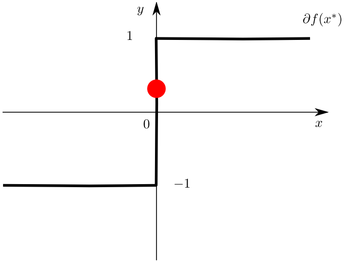
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$



Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$



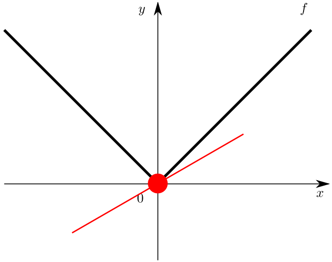
Sous-différentielle de la valeur absolue

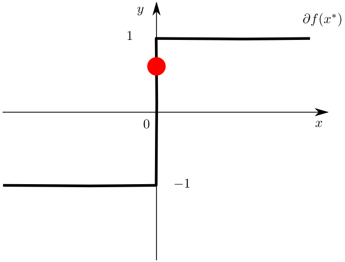
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$





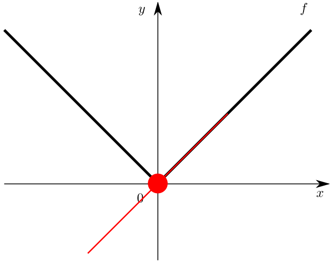
Sous-différentielle de la valeur absolue

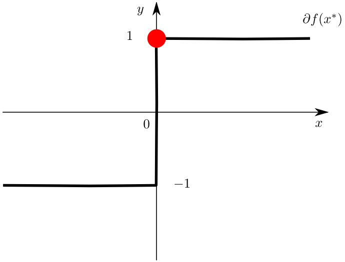
Fonction abs(·)

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign(·)

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$





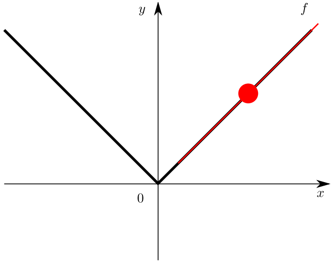
Sous-différentielle de la valeur absolue

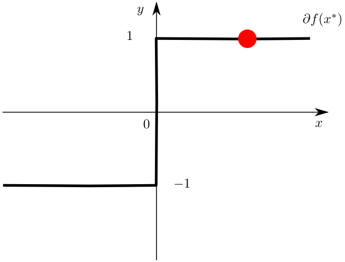
Fonction abs$(\cdot)$

$$f : \begin{cases} \mathbb{R} & \to \mathbb{R} \\ x & \mapsto |x| \end{cases}$$

Sous-différentielle sign$(\cdot)$

$$\partial f(x^*) = \begin{cases} \{-1\} & \text{si } x^* \in]-\infty, 0[\\ [-1, 1] & \text{si } x^* = 0 \\ \{1\} & \text{si } x^* \in]0, \infty[\end{cases}$$





Condition de Fermat pour le Lasso

$$\widehat{\beta}_\lambda^{\text{Lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} \quad + \quad \underbrace{\lambda\|\beta\|_1}_{\text{régularisation}}$$

- Conditions nécessaires et suffisantes d'optimalité (Fermat) :

$$\forall j \in [\![1, p]\!], \; x_j^\top \left(\frac{y - X\widehat{\beta}_\lambda^{\text{Lasso}}}{\lambda} \right) \in \begin{cases} \{\operatorname{sign}(\widehat{\beta}_\lambda^{\text{Lasso}})_j\} & \text{si} \quad (\widehat{\beta}_\lambda^{\text{Lasso}})_j \neq 0, \\ [-1, 1] & \text{si} \quad (\widehat{\beta}_\lambda^{\text{Lasso}})_j = 0. \end{cases}$$

Condition de Fermat pour le Lasso

$$\widehat{\beta}_\lambda^{\text{Lasso}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \quad \underbrace{\frac{1}{2}\|y - X\beta\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\beta\|_1}_{\text{régularisation}}$$

- Conditions nécessaires et suffisantes d'optimalité (Fermat) :

$$\forall j \in [\![1, p]\!], \; x_j^\top \left(\frac{y - X\widehat{\beta}_\lambda^{\text{Lasso}}}{\lambda} \right) \in \begin{cases} \{\operatorname{sign}(\widehat{\beta}_\lambda^{\text{Lasso}})_j\} & \text{si} \quad (\widehat{\beta}_\lambda^{\text{Lasso}})_j \neq 0, \\ [-1, 1] & \text{si} \quad (\widehat{\beta}_\lambda^{\text{Lasso}})_j = 0. \end{cases}$$

Démonstration.

$$\frac{\partial}{\partial \beta_j}\left(\frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1\right) = -x_j^\top (y - X\beta) + \lambda \frac{\partial}{\partial \beta_j}|\beta_j|$$

Si $\beta = \widehat{\beta}_\lambda^{\text{Lasso}}$, alors $-x_j^\top (y - X\beta) + \lambda \frac{\partial}{\partial \beta_j}|\beta_j| = 0$, et on peut en déduire

$$\frac{\partial}{\partial \beta_j}|(\widehat{\beta}_\lambda^{\text{Lasso}})_j| = x_j^\top \left(\frac{y - X\widehat{\beta}_\lambda^{\text{Lasso}}}{\lambda} \right)$$

Le cas orthogonal : le seuillage doux

- Retour sur un cas simple (*design* orthogonal) : $X^\top X = I_p$

$$\|y - X\beta\|_2^2 = \|X^\top y - X^\top X\beta\|_2^2 = \|X^\top y - \beta\|_2^2$$

car X est une isométrie dans ce cas

- L'objectif du lasso devient :

$$\frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1 = \sum_{j=1}^{p} \frac{1}{2}(x_j^\top y - \beta_j)^2 + \lambda|\beta_j|$$

Problème séparable : problème qui revient à minimiser terme à terme en séparant les termes la somme. Il faut donc minimiser :

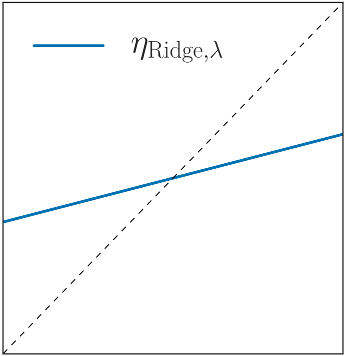
$$x \mapsto \frac{1}{2}(z - x)^2 + \lambda|x|,$$

pour $z = x_j^\top y$

Régularisation en 1D : ℓ_2 (Ridge)

Résoudre : $\eta_\lambda(z) = \underset{x \in \mathbb{R}}{\operatorname{argmin}}\, x \mapsto \frac{1}{2}(z-x)^2 + \frac{\lambda}{2}x^2$

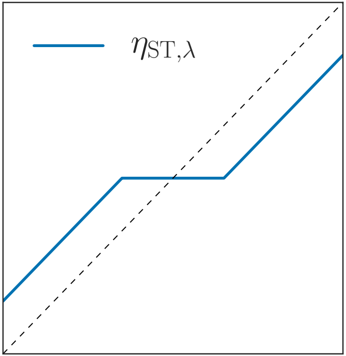
$$\eta_\lambda(z) = \frac{z}{1+\lambda}$$



Contraction ℓ_2 : Ridge

Résoudre : $\eta_\lambda(z) = \underset{x \in \mathbb{R}}{\operatorname{argmin}}\ x \mapsto \frac{1}{2}(z-x)^2 + \lambda|x|$

$$\eta_\lambda(z) = \operatorname{sign}(z)(|z| - \lambda)_+ \quad \textbf{[exercice]}$$

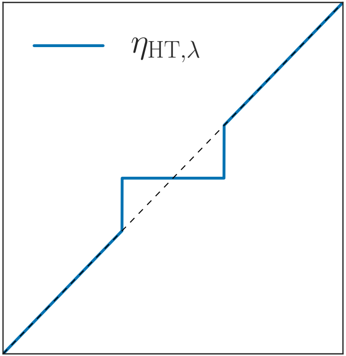


Contraction ℓ_1 : Seuillage doux (*soft thresholding*)

Régularisation en 1D : ℓ_0

Résoudre : $\eta_\lambda(z) = \underset{x\in\mathbb{R}}{\operatorname{argmin}}\ x \mapsto \frac{1}{2}(z-x)^2 + \lambda \mathbb{1}_{x\neq 0}$

$$\eta_\lambda(z) = z\mathbb{1}_{|z|\geq\sqrt{2\lambda}}$$



Contraction ℓ_0 : Seuillage dur (*hard thresholding*)

Seuillage doux : forme explicite

Solution.

- Si $x < 0$:

$$\frac{d}{dx}\left(\frac{1}{2}(z-x)^2 - \lambda x\right) = x - z - \lambda = 0 \quad \Rightarrow \quad x = z + \lambda < 0 \quad (x < 0 \text{ si } z < -\lambda)$$

- Si $x > 0$:

$$\frac{d}{dx}\left(\frac{1}{2}(z-x)^2 + \lambda x\right) = x - z + \lambda = 0 \quad \Rightarrow \quad x = z - \lambda \quad (x > 0 \text{ si } z > \lambda)$$

- Si $x = 0$:

$$\partial f(x) \in [x - z - \lambda, x - z + \lambda] = [-z - \lambda, -z + \lambda]$$

Le sous-gradient doit vérifier $\partial f(x) = 0$, alors $|z| \leq \lambda$

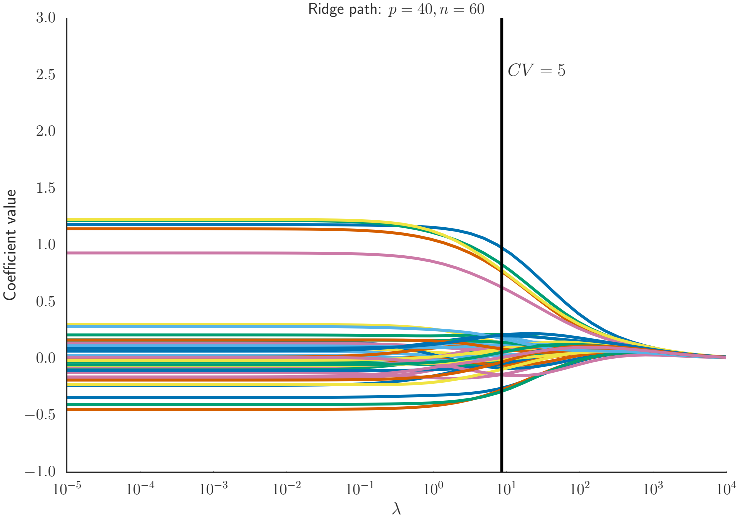
- On obtient finalement

$$\eta_{\text{Lasso},\lambda}(z) = \begin{cases} z + \lambda & \text{si } z < -\lambda \\ 0 & \text{si } |z| \leq \lambda \\ z - \lambda & \text{si } z > \lambda \end{cases}$$

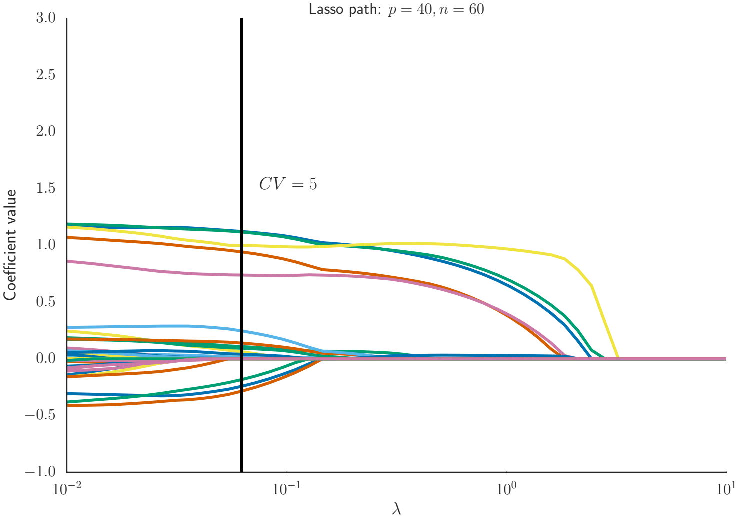
Exemple numérique : simulation

- Pour cet exemple les tailles sont : $n = 60, p = 40, \sigma^2 = 1$
  - $\beta^\star = (1, 1, 1, 1, 1, 0, \dots, 0) \in \mathbb{R}^p$ (5 coefficients non-nuls)
  - $X \in \mathbb{R}^{n \times p}$ a des colonnes tirées selon une loi gaussienne
  - $y = X\beta^\star + \varepsilon \in \mathbb{R}^n$ avec $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_n)$
  - On utilise une grille de 50 valeurs de λ

Choix de λ : Lasso vs Ridge (exemple avec $CV = 5$)



Choix de λ : Lasso vs Ridge (exemple avec $CV = 5$)



Intérêt du Lasso

- Enjeu numérique : le Lasso est un problème **convexe**
- Sélection de variables / solutions parcimonieuses (sparse) : $\widehat{\beta}_\lambda^{\text{Lasso}}$ a potentiellement de nombreux coefficients nuls
- Le paramètre λ contrôle le niveau de parcimonie : si λ est grand, les solutions sont très creuses

Exemple. Dans la simulation précédente :

Méthode	$\lVert\widehat{\beta}_{\widehat{\lambda}_{CV}}\rVert_0$ ($\lVert\widehat{\beta}^\star\rVert_0 = 5$, $p = 40$)
RidgeCV	40
LassoCV	17

Analyse théorique. Plus poussée que pour LS ou que pour Ridge [voir, *e.g.*, Bühlmann and van de Geer, 2011, pour des résultats théoriques]

Le biais du Lasso

- Le lasso est biaisé : il contracte les grands coefficients vers 0

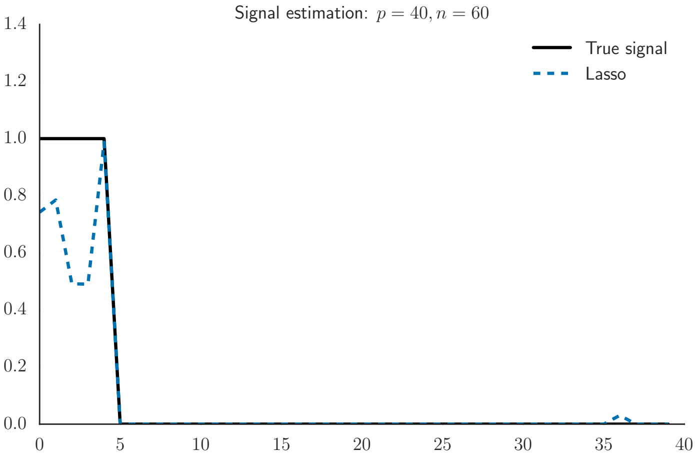


Illustration sur l'exemple

Le biais du Lasso

- Le lasso est biaisé : il contracte les grands coefficients vers 0

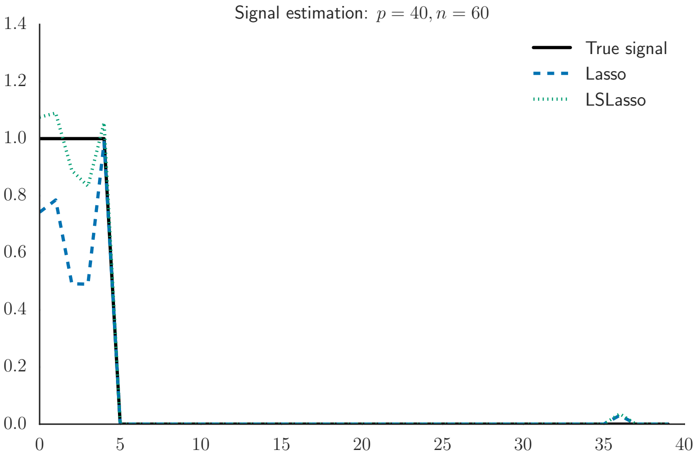


Illustration sur l'exemple

Le biais du Lasso : un remède simple

· Il est possible d'utiliser une procédure en deux étapes

Algorithm LSLasso (Least Square Lasso)

Données d'entré : (X, y), paramètres pour le Lasso (*e.g.*, λ, nb d'itérations, tolérance)

1: Lasso pour obtenir $\widehat{\beta}_\lambda^{\text{Lasso}}$ (sélection du modèle)

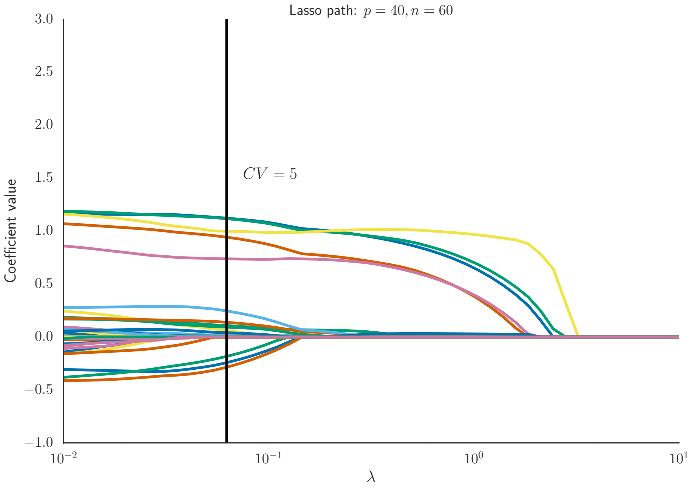
2: LS sur les variables actives (correction du biais)

$$\widehat{\beta}_\lambda^{\text{LSLasso}} = \underset{\substack{\beta \in \mathbb{R}^p \\ \text{supp}(\beta) = \text{supp}(\widehat{\beta}_\lambda^{\text{Lasso}})}}{\text{argmin}} \frac{1}{2} \|y - X\beta\|_2^2$$

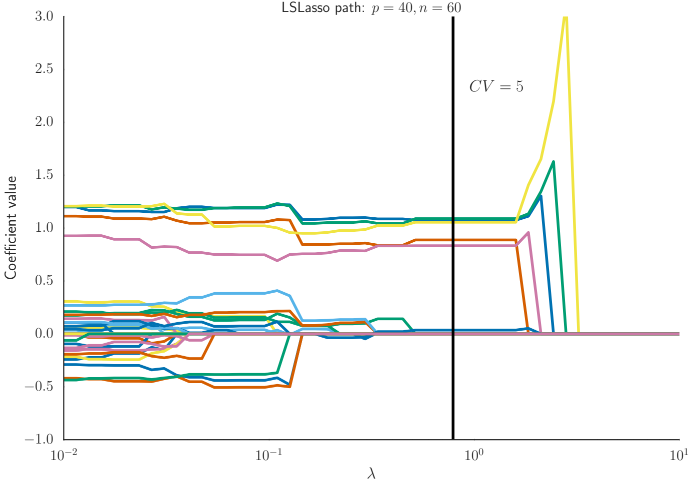
Rem :

- Il faut faire la CV sur la procédure entière
  - Choisir λ par LassoCV puis faire les LS garde trop de variables !
- LSLasso n'est pas forcément codé dans les packages usuels

Choix de λ : LSLasso vs Lasso (exemple avec $CV = 5$)



Choix de λ : LSLasso vs Lasso (exemple avec $CV = 5$)



Avantages.

- Les “vrais” grands coefficients sont moins atténués
- En faisant la CV on récupère moins de variables parasites (amélioration de l’interprétabilité)

Méthode	$\|\widehat{\beta}_{\widehat{\lambda}_{CV}}\|_0$ ($\|\widehat{\beta}^\star\|_0 = 5$, $p = 40$)
RidgeCV	40
LassoCV	17
LSLassoCV	6 (1 faux positif)

LSLasso : utile pour l’estimation

Limites.

- La différence en prédiction n’est pas toujours flagrante
- Nécessite plus de calcul : re-calculer autant de LS que de paramètres λ
- Non packagé

La régularisation ℓ_1/ℓ_2 (Elastic Net)

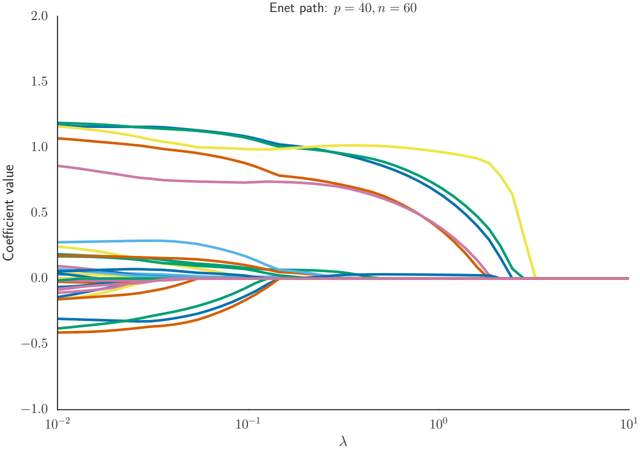
- L'Elastic Net introduit par [Zou and Hastie, 2005] est solution de

$$\widehat{\beta}_{\lambda,\gamma}^{\text{E-Net}} = \underset{\beta \in \mathbb{R}^p}{\operatorname{argmin}} \left[\frac{1}{2} \|y - X\beta\|_2^2 + \lambda \left(\gamma \|\beta\|_1 + (1-\gamma) \frac{\|\beta\|_2^2}{2} \right) \right]$$

Rem :

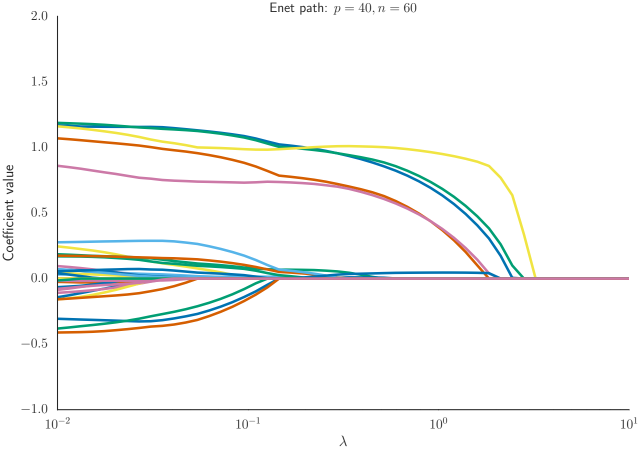
- Deux paramètres de régularisation : λ pour une régularisation globale, γ pour un contrôle de l'influence Ridge vs. Lasso
- La solution est unique et la taille du support de $\widehat{\beta}_{\lambda,\gamma}^{\text{E-Net}}$ est plus petite que $\min(n, p)$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



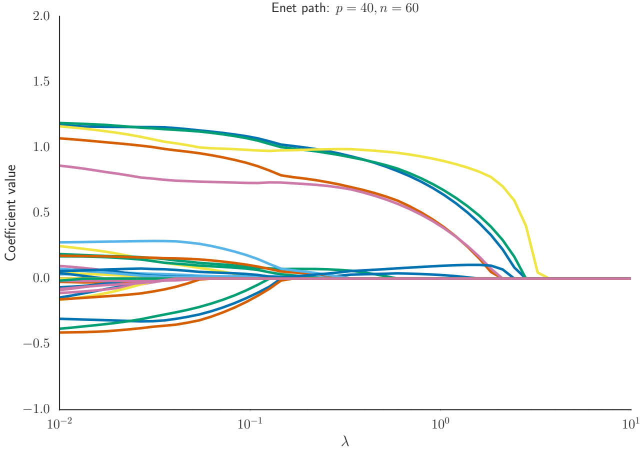
$\gamma = 1.00$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



$\gamma = 0.99$

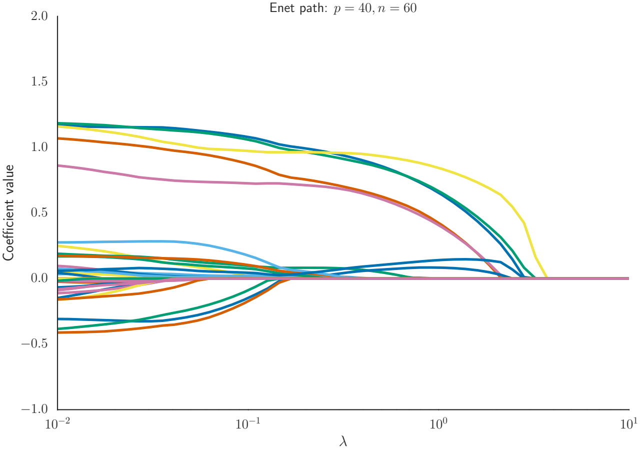
Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



$$\gamma = 0.95$$

CERAMATHS Université Polytechnique HAUTS-DE-FRANCE

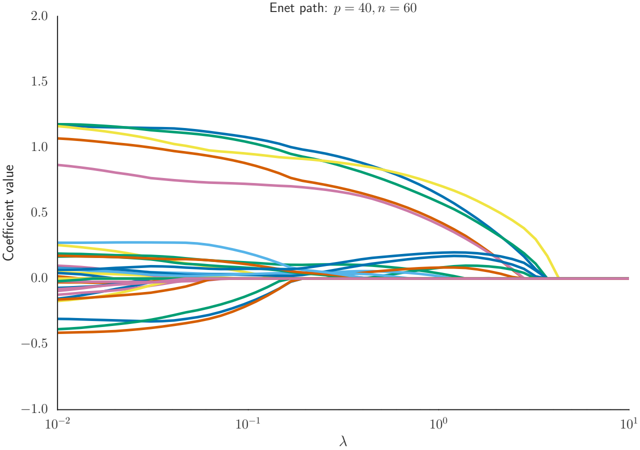
Elastic-Net : $\gamma\|\beta\|_1+(1-\gamma)\|\beta\|_2^2/2$



$\gamma = 0.90$

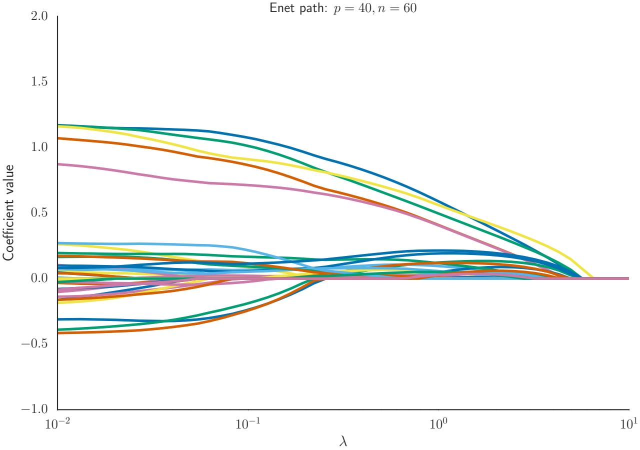
CERAMATHS Université Polytechnique HAUTS-DE-FRANCE

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



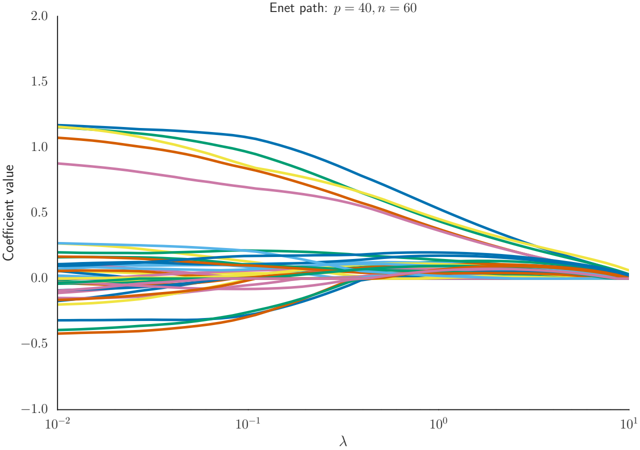
$\gamma = 0.75$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



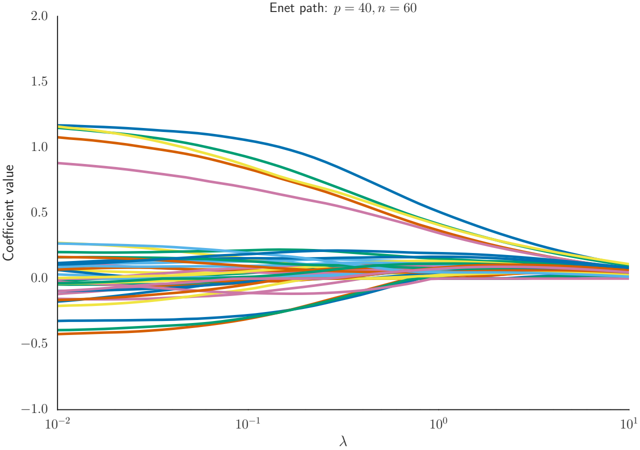
$\gamma = 0.50$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



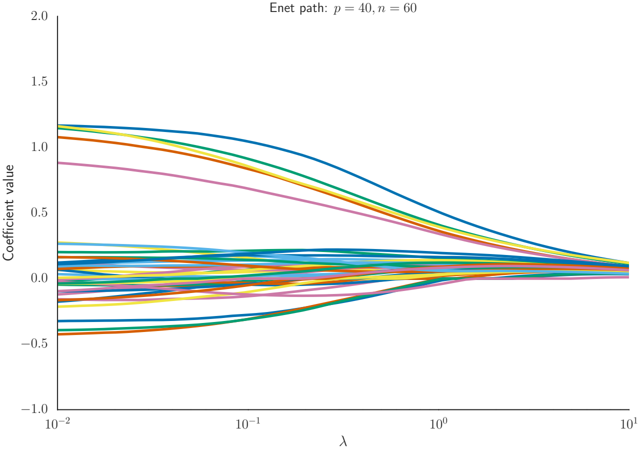
$\gamma = 0.25$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



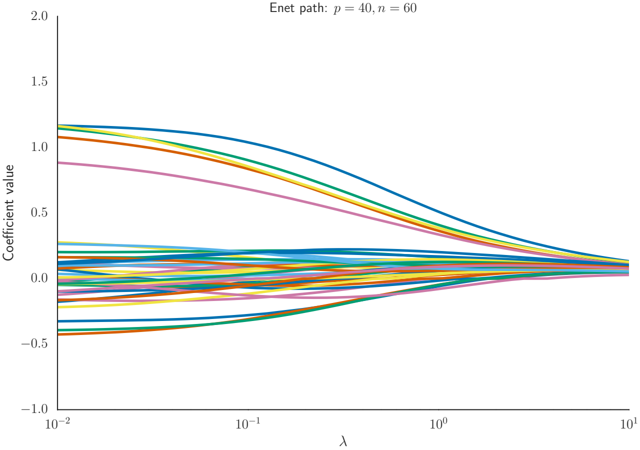
$\gamma = 0.1$

Elastic-Net : $\gamma\|\beta\|_1+(1-\gamma)\|\beta\|_2^2/2$



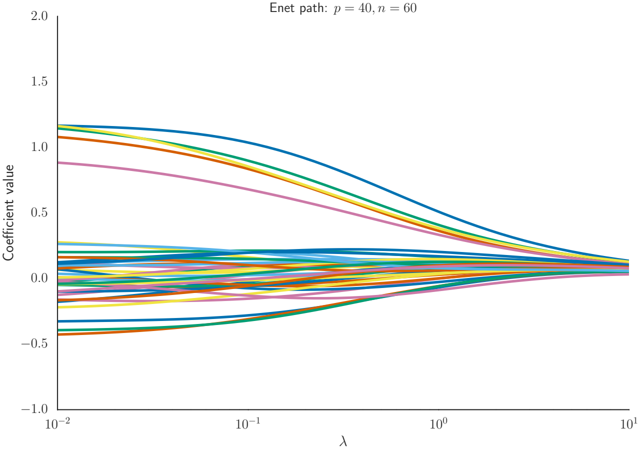
$\gamma = 0.05$

Elastic-Net : $\gamma\|\beta\|_1+(1-\gamma)\|\beta\|_2^2/2$



$\gamma = 0.01$

Elastic-Net : $\gamma\|\beta\|_1 + (1-\gamma)\|\beta\|_2^2/2$



$\gamma = 0.00$

D'autres extensions

1. Pénalités non-convexes
   - *Adaptive Lasso* [Candès et al., 2008, Zou, 2006] :

$$\mathrm{pen}_\lambda(\beta) = \lambda \sum_{j=1}^{p} |\beta_j|^q \text{ avec } 0 < q < 1$$

2. Pénalités par groupes
   - *Group Lasso* [Meier et al., 2008, Yuan and Lin, 2006] :

$$\mathrm{pen}_\lambda(\beta) = \lambda \sum_{g=1}^{G} \|\beta_{\mathcal{I}_g}\|_2,$$

   avec $\mathcal{I}_g$ l'ensemble d'indices appartenant au g-ème groupe de variables

3. Cadre bayésien
   - *Bayesian Lasso* [Park and Casella, 2008] : pour $\sigma > 0$ et $\tau > 0$,

$$y|\beta, \sigma^2 \sim \mathcal{N}(y|X\beta, \sigma^2 I_n),$$
$$\beta|\tau \sim \left(\frac{\tau}{2}\right)^2 \exp(-\tau\|\beta\|_1)$$

   · Le mode $\widehat{\beta} = \underset{\beta \in \mathbb{R}^p}{\mathrm{argmax}}\ p(\beta|y, \sigma^2, \tau)$ converge vers $\widehat{\beta}_\lambda^{\mathrm{Lasso}}$ avec $\lambda = 2\tau\sigma^2$

Références

D. Bertsimas, A. King, and R. Mazumder. Best subset selection via a modern optimization lens. *Annals of Statistics*, 44(2):813–852, 2016.

P. Bühlmann and S. van de Geer. *Statistics for high-dimensional data*. Springer Series in Statistics. Springer, Heidelberg, 2011. Methods, theory and applications.

E. J. Candès, M. B. Wakin, and S. P. Boyd. Enhancing sparsity by reweighted l_1 minimization. *J. Fourier Anal. Applicat.*, 14(5-6):877–905, 2008.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 4 edition, 2013.

Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer, New York, second edition, 2009.

L. Meier, S. Van De Geer, and P. Bühlmann. The group Lasso for logistic regression. *Journal of the Royal Statistical Society. Ser. B.*, 70(1):53–71, 01 2008.

T. Park and G. Casella. The Bayesian Lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Ser. B. Methodological*, 58(1):267–288, 1996.

M. Yuan and Y. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society. Ser. B.*, 68(1):49–67, 2006.

H. Zou. The adaptive Lasso and its oracle properties. 101(476):1418–1429, 2006.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. 67(2): 301–320, 2005.