MS BGD
MDI 720 : Statistiques

Joseph Salmon
http://josephsalmon.eu
Télécom Paristech, Institut Mines-Télécom

Plan

Retour sur le modèle

$$\mathbf{y} = X\boldsymbol{\theta}^* + \boldsymbol{\varepsilon} \in \mathbb{R}^n$$

$$X = [\mathbf{x}_1, \dots, \mathbf{x}_p] = \begin{pmatrix} x_{1,1} & \dots & x_{1,p} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \dots & x_{n,p} \end{pmatrix} \in \mathbb{R}^{n \times p}, \boldsymbol{\theta}^* \in \mathbb{R}^p$$

Motivation

Utilité des estimateurs $\hat{\boldsymbol{\theta}}$ avec beaucoup de coefficients nuls :

- pour l'interprétation
- pour l'efficacité computationnelle si p est énorme

L'idee sous-jacente : **sélectionner des variables**

Rem: aussi utile si $\boldsymbol{\theta}^*$ a peu de coefficients non nuls

Méthodes de sélection de variables

- Méthodes de type **écrémage** (: *screening*) : on supprime les $\mathbf{x}_j$ dont la corrélations sont faibles avec $\mathbf{y}$
  • avantages : rapide (+++), coût : p produits scalaires de taille n, intuitive (+++)
  • défauts : néglige les interactions entre variables $\mathbf{x}_j$, résultats théoriques faibles (- - -)

- Méthodes **gloutonnes** (: *greedy*) ou **pas à pas** (: *stagewise/stepwise*)
  • avantages : rapide (++), intuitive (++)
  • défauts : propagation mauvaises sélections de variables aux étapes suivantes ; résultats théoriques faibles (-)

- Méthodes **pénalisées** favorisant la parcimonie (*e.g.*, Lasso)
  • avantages : résultats théoriques bons (++)
  • défauts : encore lent (on y travaille !) (-),

La pseudo-norme ℓ_0

Définitions

Le **support** du vecteur $\boldsymbol{\theta}$ est l'ensemble des indices des coordonnées non nulles :

$$\mathrm{supp}(\boldsymbol{\theta}) = \{j \in [\![1, p]\!], \theta_j \neq 0\}$$

La **pseudo-norme** ℓ_0 d'un vecteur $\boldsymbol{\theta} \in \mathbb{R}^p$ est son nombre de coordonnées non-nulles :

$$\|\boldsymbol{\theta}\|_0 = \mathrm{card}\{j \in [\![1, p]\!], \theta_j \neq 0\}$$

Rem: $\|\cdot\|_0$ n'est pas une norme, $\forall t \in \mathbb{R}^*, \|t\boldsymbol{\theta}\|_0 = \|\boldsymbol{\theta}\|_0$

Rem: $\|\cdot\|_0$ n'est pas non plus convexe, $\boldsymbol{\theta}_1 = (1, 0, 1, \ldots, 0)$ $\boldsymbol{\theta}_2 = (0, 1, 1, \ldots, 0)$ et $3 = \|\frac{\boldsymbol{\theta}_1 + \boldsymbol{\theta}_2}{2}\|_0 \geqslant \frac{\|\boldsymbol{\theta}_1\|_0 + \|\boldsymbol{\theta}_2\|_0}{2} = 2$

Sommaire

La pénalisation ℓ_0

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

$$\boxed{\hat{\boldsymbol{\theta}}_\lambda = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_0}_{\text{régularisation}} \right)}$$

Problème combinatoire ! ! !

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 : $\approx 10^{10}$ moindre carrés

Rem: problème "NP-dur"

Sommaire

Le Lasso : la définition pénalisée

Lasso : *Least Absolute Shrinkage and Selection Operator*
Tibshirani (1996)

$$\boxed{\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_1}_{\textbf{régularisation}} \right)}$$

où $\|\boldsymbol{\theta}\|_1 = \sum_{j=1}^{p} |\theta_j|$ (somme des valeurs absolues des coefficients)

- On retrouve de nouveau les cas limites :

$$\lim_{\lambda\to 0} \hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}} = \hat{\boldsymbol{\theta}}^{\text{MCO}}$$
$$\lim_{\lambda\to +\infty} \hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}} = 0 \in \mathbb{R}^p$$

<u>Attention</u> : l'estimateur Lasso n'est pas toujours **unique** pour un λ fixé (prendre par exemple deux colonnes identiques)

Interprétation contrainte

Un problème de la forme :

$$\boxed{\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_1}_{\text{régularisation}} \right)}$$

admet la même solution qu'une version contrainte :

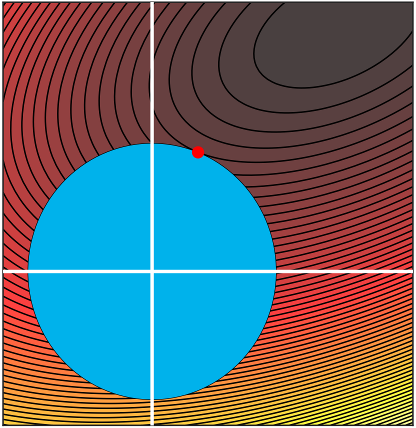
$$\begin{cases} \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \\ \text{t.q. } \|\boldsymbol{\theta}\|_1 \leqslant T \end{cases}$$

pour un certain $T > 0$.

<u>Rem</u>: hélas le lien $T \leftrightarrow \lambda$ n'est pas explicite

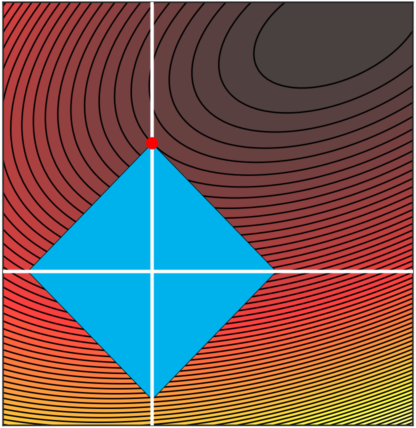
- Si $T \to 0$ on retrouve le vecteur nul : $0 \in \mathbb{R}^p$
- Si $T \to \infty$ on retrouve $\hat{\boldsymbol{\theta}}^{\text{MCO}}$ (non contraint)

Mise à zéro de certains coefficients



Optimisation sous contrainte ℓ_2 : solution non parcimonieuse

Mise à zéro de certains coefficients



Optimisation sous contrainte ℓ_1 : solution parcimonieuse

Sommaire

Sous-gradients / sous-différentielles

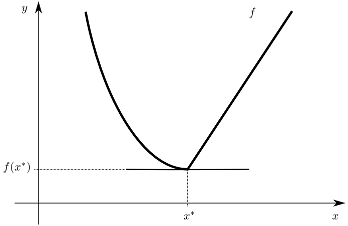
Définitions

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

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

La **sous-différentielle** est l'ensemble
$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle u, x - x^* \rangle\}$.

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



Sous-gradients / sous-différentielles

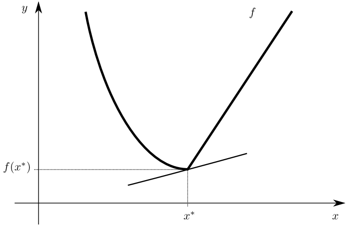
Définitions

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

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

La **sous-différentielle** est l'ensemble
$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle u, x - x^* \rangle\}$.

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



Sous-gradients / sous-différentielles

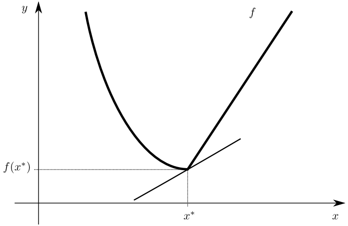
Définitions

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

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

La **sous-différentielle** est l'ensemble $\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle u, x - x^* \rangle\}$.

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



Sous-gradients / sous-différentielles

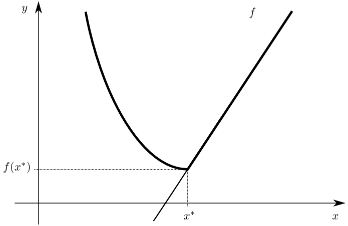
Définitions

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

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

La **sous-différentielle** est l'ensemble
$\partial f(x^*) = \{u \in \mathbb{R}^n : \forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle u, x - x^* \rangle\}$.

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



Règle de Fermat

Théorème

Un point x^* est un minimum d'une fonction convexe $f : \mathbb{R}^n \to \mathbb{R}$ 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 $\forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle 0, x - x^* \rangle$

Règle de Fermat

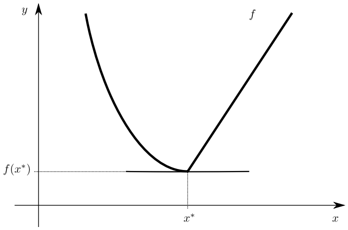
Théorème

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

<u>Preuve</u> : utiliser la définition des sous-gradients :

- 0 est un sous-gradient de f en x^* si et seulement si $\forall x \in \mathbb{R}^n, f(x) \geqslant f(x^*) + \langle 0, x - x^* \rangle$

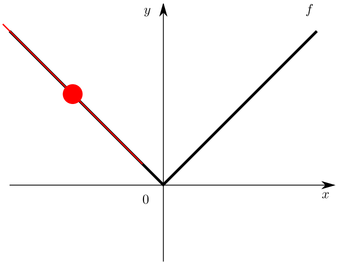
<u>Rem</u>: Visuellement cela correspond à une tangente horizontale



Sous-différentielle de la valeur absolue

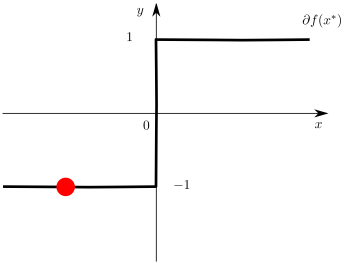
Fonction (abs) :

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



Sous-différentielle (sign)

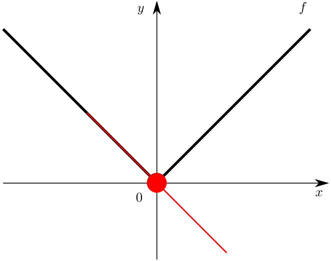
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Sous-différentielle de la valeur absolue

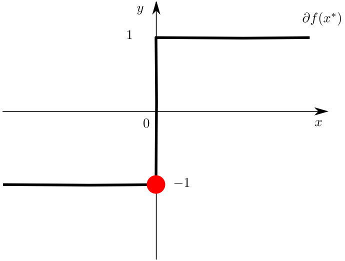
Fonction (abs) :

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



Sous-différentielle (sign)

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



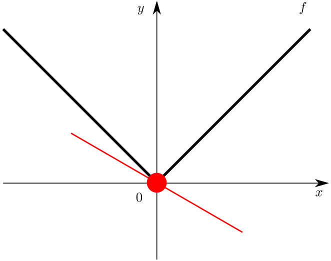
Sous-différentielle de la valeur absolue

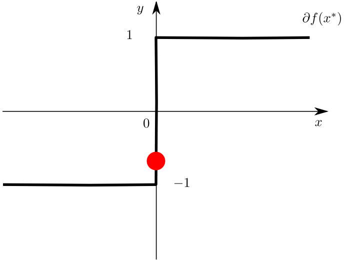
Fonction (abs) :

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

Sous-différentielle (sign)

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

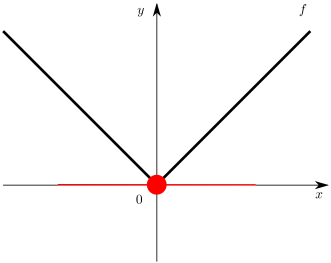




Sous-différentielle de la valeur absolue

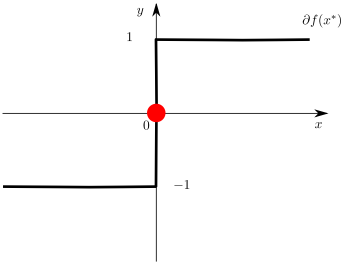
Fonction (abs) :

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



Sous-différentielle (sign)

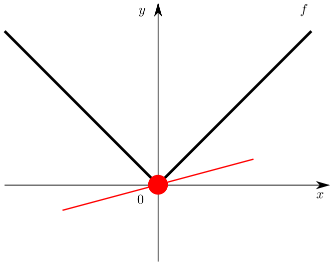
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Sous-différentielle de la valeur absolue

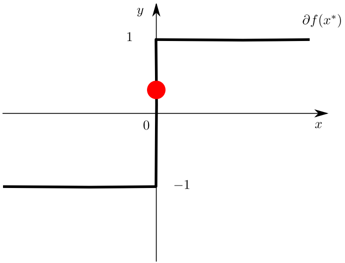
Fonction (abs) :

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



Sous-différentielle (sign)

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



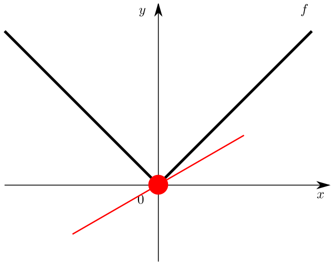
Sous-différentielle de la valeur absolue

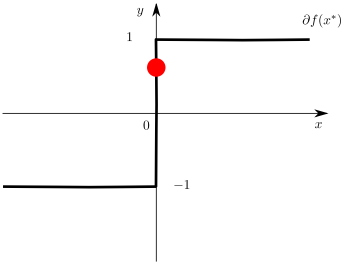
Fonction (abs) :

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

Sous-différentielle (sign)

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

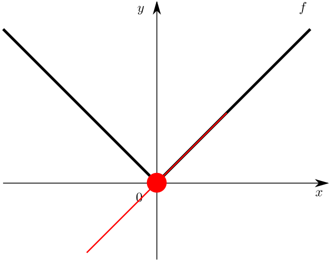




Sous-différentielle de la valeur absolue

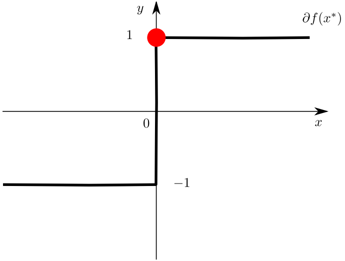
Fonction (abs) :

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



Sous-différentielle (sign)

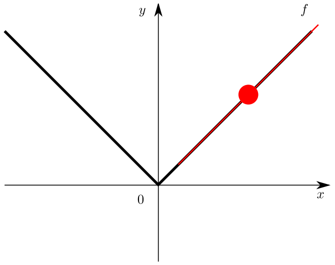
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Sous-différentielle de la valeur absolue

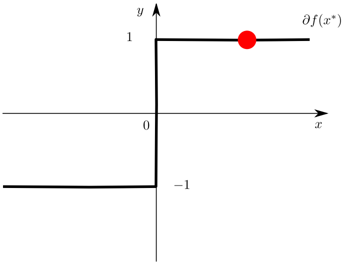
Fonction (abs) :

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



Sous-différentielle (sign)

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



Condition de Fermat pour le Lasso

$$\boxed{\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}} = \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_1}_{\text{régularisation}} \right)}$$

Conditions nécessaires et suffisantes d'optimalité (Fermat) :

$$\boxed{\forall j \in [p],\ \mathbf{x}_j^\top \left(\frac{y - X\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}}}{\lambda} \right) \in \begin{cases} \{\operatorname{sign}(\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}})_j\} & \text{si} \quad (\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}})_j \neq 0, \\ [-1, 1] & \text{si} \quad (\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}})_j = 0. \end{cases}}$$

Rem: si $\lambda > \lambda_{\max} := \max_{j\in[\![1,p]\!]} |\langle \mathbf{x}_j, \mathbf{y}\rangle|$, alors $\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso}} = 0$

Le cas orthogonal : le seuillage doux

Retour sur un cas simple (*design* orthogonal) : $X^\top X = \mathrm{Id}_p$

$$\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 = \|X^\top \mathbf{y} - X^\top X\boldsymbol{\theta}\|_2^2 = \|X^\top \mathbf{y} - \boldsymbol{\theta}\|_2^2$$

car X est une isométrie dans ce cas, l'objectif du lasso devient :

$$\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1 = \sum_{j=1}^{p}\left(\frac{1}{2}(\mathbf{x}_j^\top \mathbf{y} - \theta_j)^2 + \lambda|\theta_j|\right)$$

Problème séparable : problème qui revient à minimiser terme à terme en séparant les termes la somme

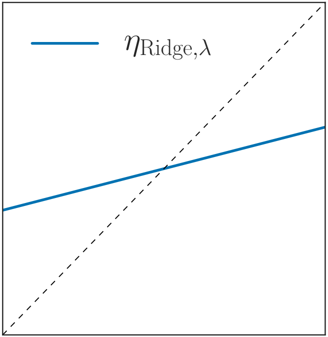
Il faut donc minimiser : $x \mapsto \frac{1}{2}(z - x)^2 + \lambda|x|$ pour $z = \mathbf{x}_j^\top \mathbf{y}$

Rem: on parle d'**opérateur proximal** en z de la fonction $x \mapsto \lambda|x|$ (*cf.* Parikh et Boyd (2013), pour les méthodes proximales)

Régularisation en 1D : Ridge

Solution du problème : $\eta_\lambda(z) = \arg\min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z-x)^2 + \frac{\lambda}{2}x^2$

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

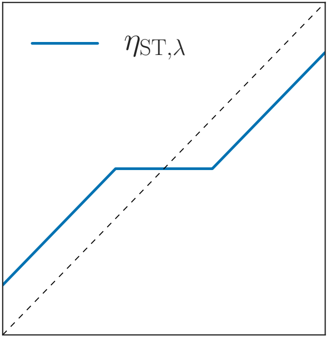


Contraction ℓ_2 : Ridge

Régularisation en 1D : Lasso

Solution du problème : $\eta_\lambda(z) = \arg\min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z-x)^2 + \lambda|x|$

$$\eta_\lambda(z) = \operatorname{sign}(z)(|z| - \lambda)_+$$

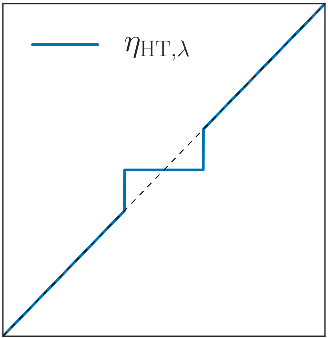


Contraction ℓ_1 : Seuillage doux (🇬🇧 : *soft thresholding*)

Régularisation en 1D : ℓ_0

Solution du problème : $\eta_\lambda(z) = \underset{x \in \mathbb{R}}{\arg\min}\, x \mapsto \frac{1}{2}(z-x)^2 + \lambda \mathbb{1}_{x \neq 0}$

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



Contraction ℓ_0 : Seuillage dur (: *hard thresholding*)

Exemple numérique : simulation

- $\boldsymbol{\theta}^* = (1, 1, 1, 1, 1, 0, \dots, 0) \in \mathbb{R}^p$ (5 coefficients non-nuls)
- $X \in \mathbb{R}^{n \times p}$ a des colonnes tirées selon une loi gaussienne
- $y = X\boldsymbol{\theta}^* + \boldsymbol{\varepsilon} \in \mathbb{R}^n$ avec $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma^2 \operatorname{Id}_n)$
- On utilise une grille de 50 valeurs de λ

Pour cet exemple les tailles sont : $n = 60, p = 40, \sigma = 1$

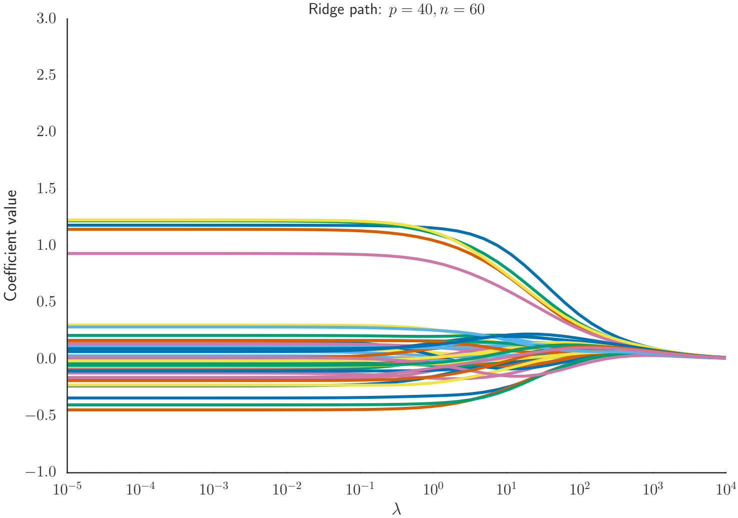
Seuillage doux : forme explicite

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

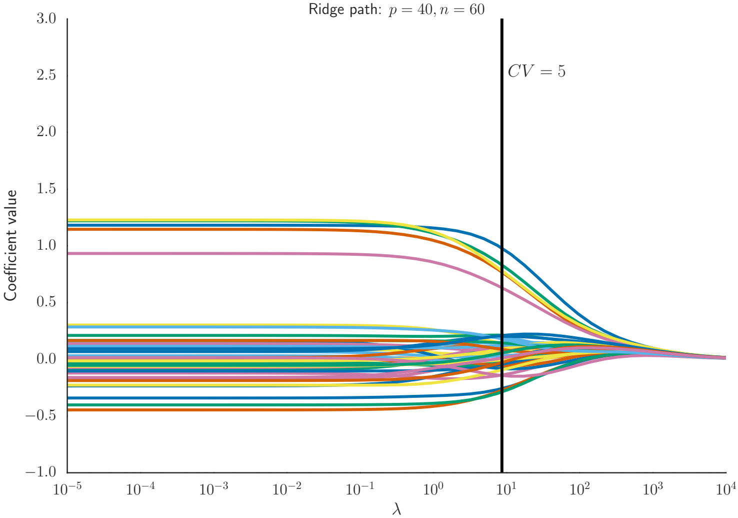
Exo: Prouver le résultat précédent en utilisant les sous-gradients

---

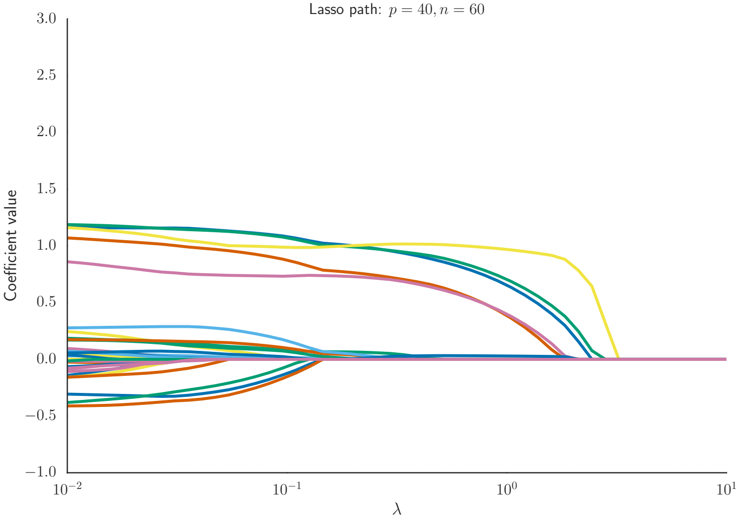
Lasso vs Ridge



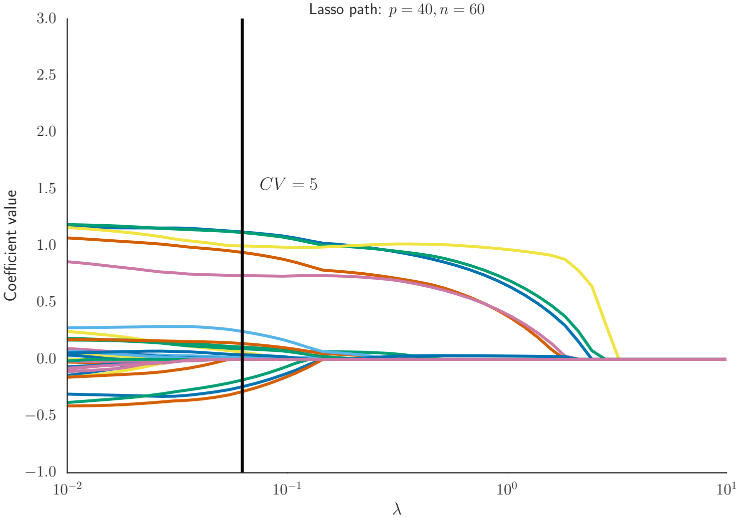
Lasso vs Ridge



Lasso vs Ridge



Lasso vs Ridge



Intérêt du Lasso

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

  Exemple : on obtient 17 coefficients non nuls pour `LassoCV` dans la simulation précédente

  Rem: RidgeCV n'a aucun coefficient nul

Analyse de l'estimateur dans le cas général

L'analyse théorique est nettement plus poussée que pour les moindres carrées ou que pour Ridge et peut-être trouvé dans des références récentes (*cf.* Buhlmann et van de Geer (2011) pour des résultats théoriques)

<u>En résumé</u> : on biaise l'estimateur des moindres carrés pour réduire la variance

Sommaire

Le biais du Lasso

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

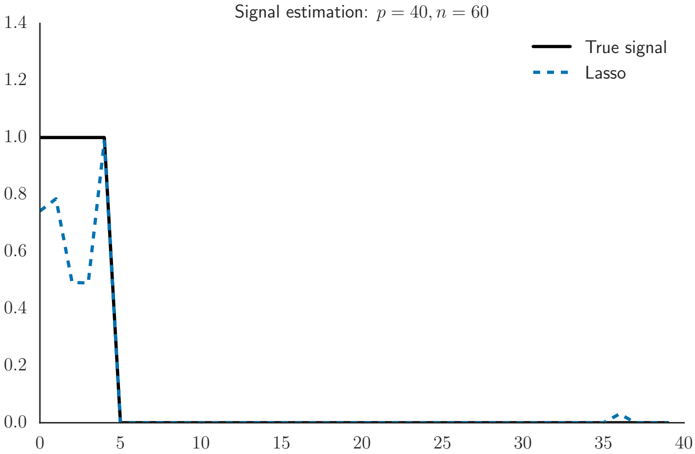


Illustration sur l'exemple

Le biais du Lasso

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

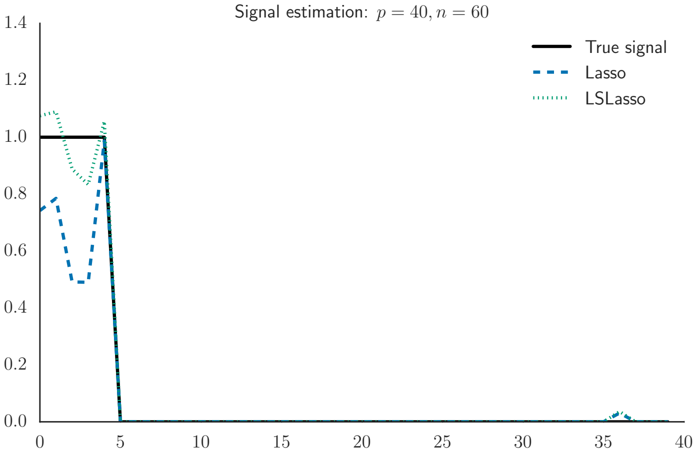


Illustration sur l'exemple

Le biais du Lasso : un remède simple

Comme les grands coefficients sont parfois contractés vers zéro, il est possible d'utiliser une procédure en deux étapes

LSLasso (Least Square Lasso)

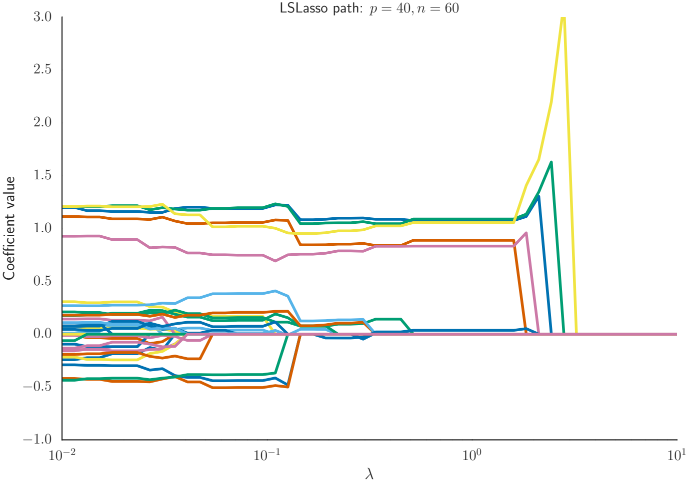
1. Lasso : obtenir $\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso}}$
2. Moindres-carrés sur les variables actives $\operatorname{supp}(\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso}})$

$$\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{LSLasso}} = \mathop{\arg\min}_{\substack{\boldsymbol{\theta}\in\mathbb{R}^p \\ \operatorname{supp}(\boldsymbol{\theta})=\operatorname{supp}(\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso}})}} \frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$$

Rem: il faut faire la CV sur la procédure entière ; choisir le λ du Lasso par CV puis faire un moindre carré conserve trop de variables

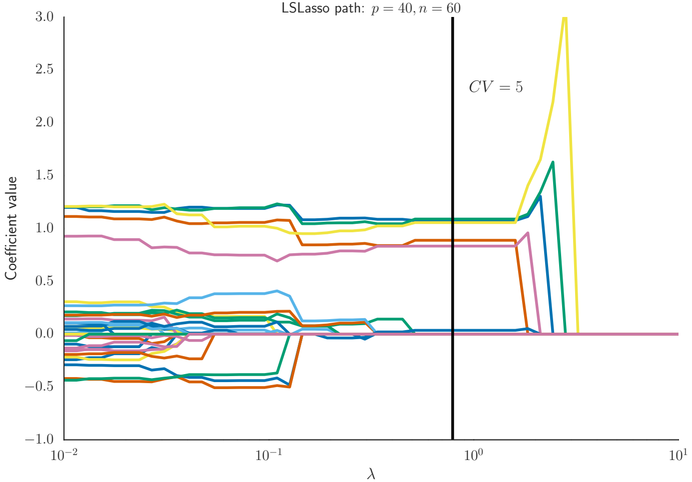
Rem: LSLasso pas forcément codé dans les packages usuels

Débiasage

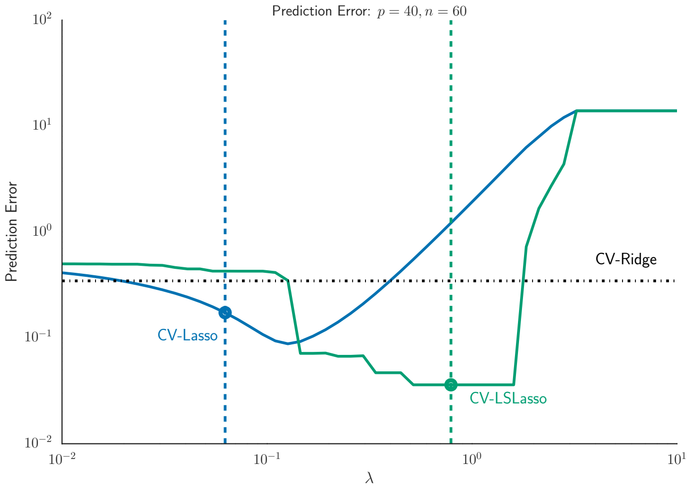


LSLasso path: $p = 40, n = 60$
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
Coefficient value
10^{-2}
10^{-1}
10^{0}
10^{1}
λ

Débiasage


LSLasso path: $p = 40, n = 60$
$CV = 5$
Coefficient value
λ
3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
10^{-2}
10^{-1}
10^{0}
10^{1}

Prédiction : Lasso vs. LSLasso



Bilan du LSLasso

Avantages

- les “vrais” grands coefficients sont moins atténués
- en faisant la CV on récupère moins de variables parasites (amélioration de l’interprétabilité)
  e.g., sur l’exemple précédent le LSLassoCV retrouve exactement les 5 “vraies” variables non nulles, plus un faux positif

LSLasso : utile pour <u>l’estimation</u>

Limites

- la différence en prédiction n’est pas toujours flagrante
- nécessite plus de calcul : re-calculer autant de moindre carrés que de paramètres λ, certes de dimension la taille des supports (on néglige les autres variables)

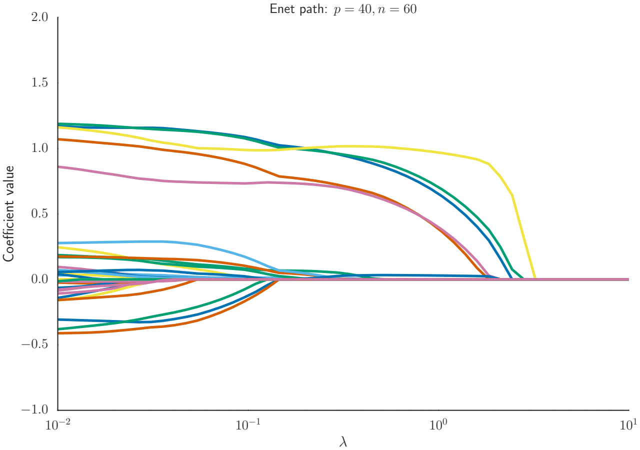
Elastic-net : régularisation ℓ_1/ℓ_2

L'Elastic-Net introduit par Zou et Hastie (2005) est solution de

$$\hat{\boldsymbol{\theta}}_\lambda = \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \mathbb{R}^p} \left[\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 + \lambda \left(\alpha \|\boldsymbol{\theta}\|_1 + (1-\alpha) \frac{\|\boldsymbol{\theta}\|_2^2}{2} \right) \right]$$

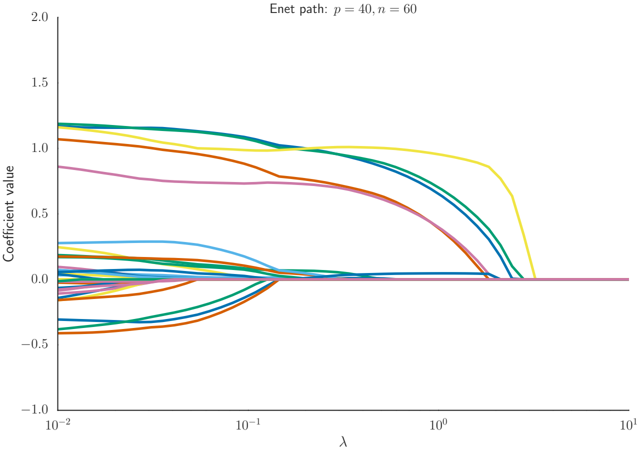
Rem: deux paramètres ici, un pour la régularisation globale, un qui balance la régularisation Ridge vs. Lasso
Rem: la solution est unique et la taille du support de l'Elastic-Net est plus petite que $\min(n, p)$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



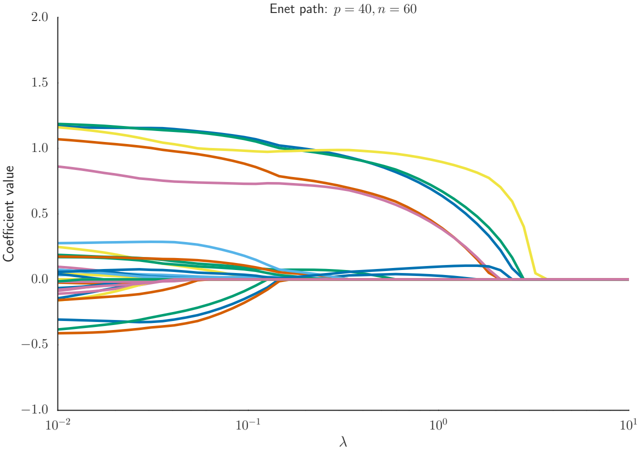
$$\alpha = 1.00$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



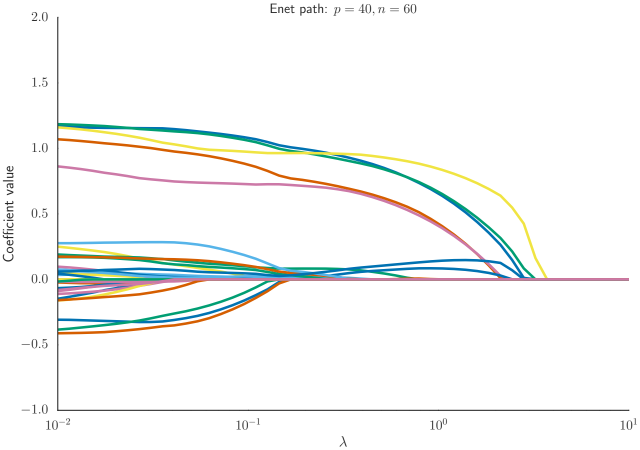
$$\alpha = 0.99$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



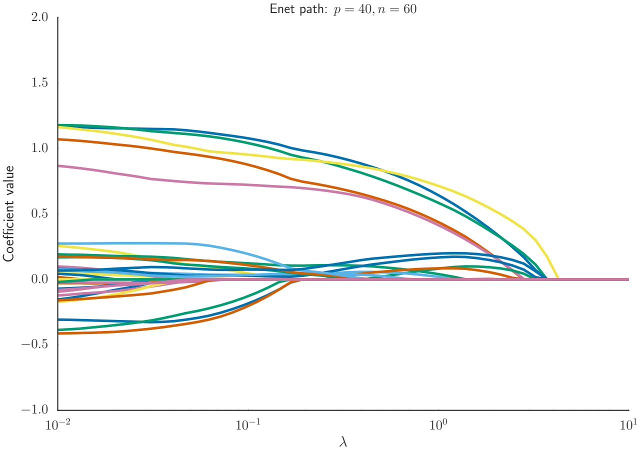
$$\alpha = 0.95$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



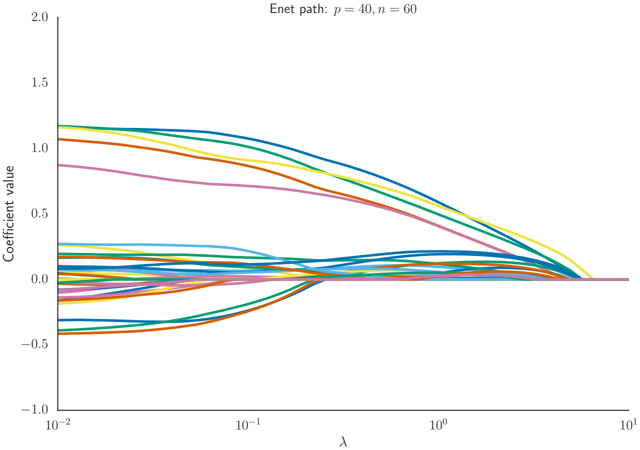
$$\alpha = 0.90$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



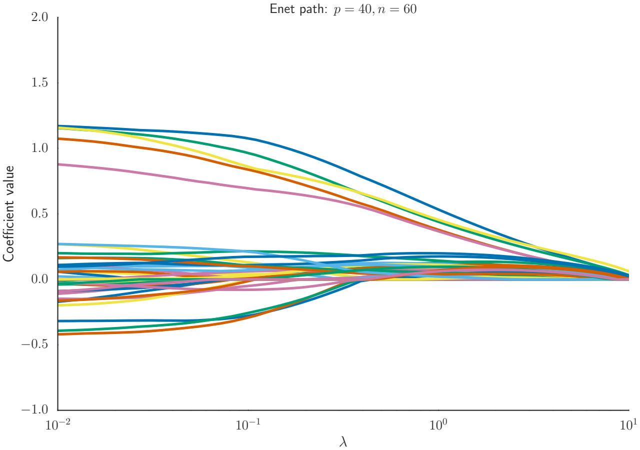
$$\alpha = 0.75$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



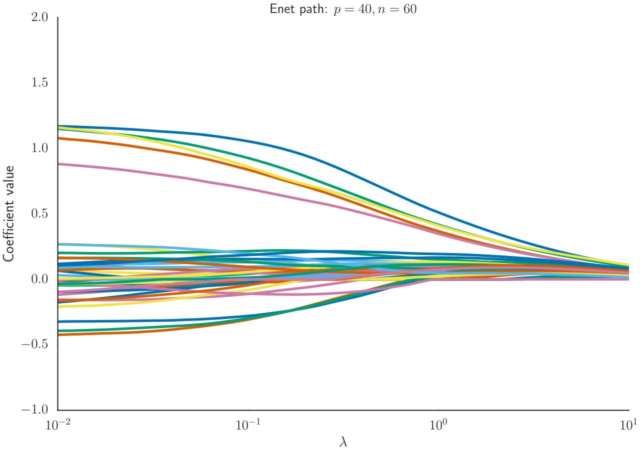
$$\alpha = 0.50$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



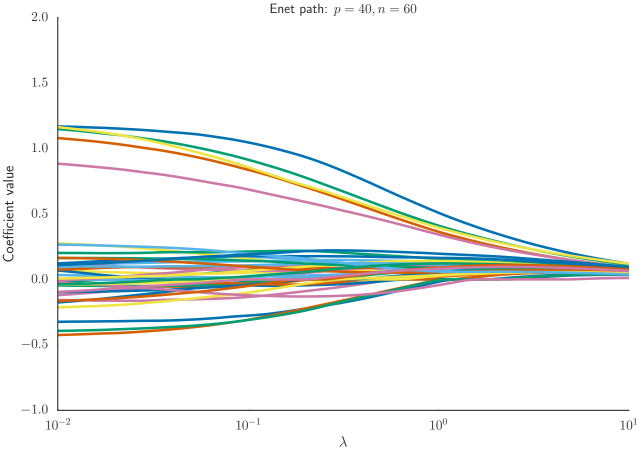
$$\alpha = 0.25$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



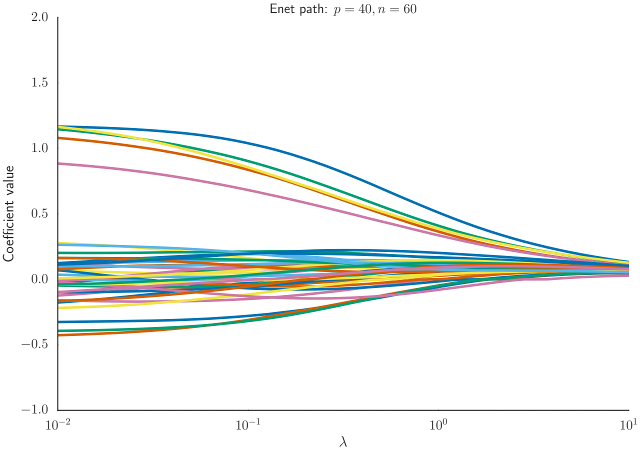
$$\alpha = 0.1$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



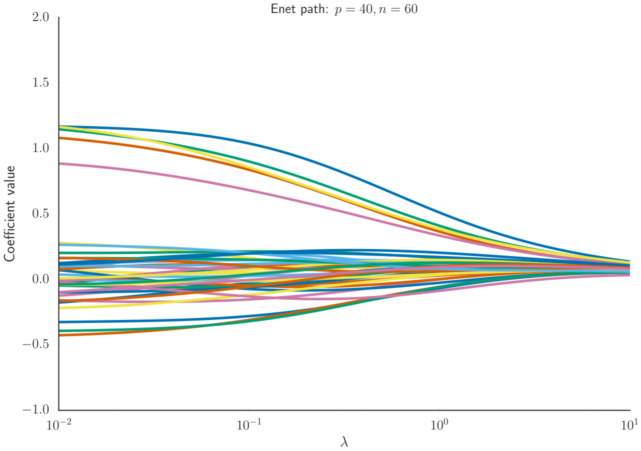
$$\alpha = 0.05$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



$$\alpha = 0.01$$

Elastic-Net : $\alpha\|\boldsymbol{\theta}\|_1 + (1-\alpha)\|\boldsymbol{\theta}\|_2^2/2$



$\alpha = 0.00$

Sommaire

Pénalités non-convexes

Utiliser une pénalité non-convexe approchant mieux $\|\cdot\|_0$, en choisissant $t \to \mathrm{pen}_{\lambda,\gamma}(t)$ non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{régularisation}} \right)}$$

Pénalités non-convexes

Utiliser une pénalité non-convexe approchant mieux $\|\cdot\|_0$, en choisissant $t \to \mathrm{pen}_{\lambda,\gamma}(t)$ non-convexe

$$\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{régularisation}} \right)$$

- Adaptive-Lasso Zou (2006) / ℓ_1 re-pondérés Candès *et al.* (2008)

$$\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q \text{ avec } 0 < q < 1$$

Pénalités non-convexes

Utiliser une pénalité non-convexe approchant mieux $\|\cdot\|_0$, en choisissant $t \to \mathrm{pen}_{\lambda,\gamma}(t)$ non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\textbf{régularisation}} \right)}$$

- ℓ_1 re-pondérés Candès *et al.* (2008)

$$\mathrm{pen}_{\lambda,\gamma}(t) = \lambda \log(1 + |t|/\gamma)$$

Pénalités non-convexes

Utiliser une pénalité non-convexe approchant mieux $\|\cdot\|_0$, en choisissant $t \to \mathrm{pen}_{\lambda,\gamma}(t)$ non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\textbf{régularisation}} \right)}$$

- MCP (*minimax concave penalty*) Zhang (2010) pour $\lambda > 0$ et $\gamma > 1$

$$\mathrm{pen}_{\lambda,\gamma}(t) = \begin{cases} \lambda|t| - \frac{t^2}{2\gamma}, & \text{if } |t| \leqslant \gamma\lambda \\ \frac{1}{2}\gamma\lambda^2, & \text{if } |t| > \gamma\lambda \end{cases}$$

Pénalités non-convexes

Utiliser une pénalité non-convexe approchant mieux $\|\cdot\|_0$, en choisissant $t \to \mathrm{pen}_{\lambda,\gamma}(t)$ non-convexe

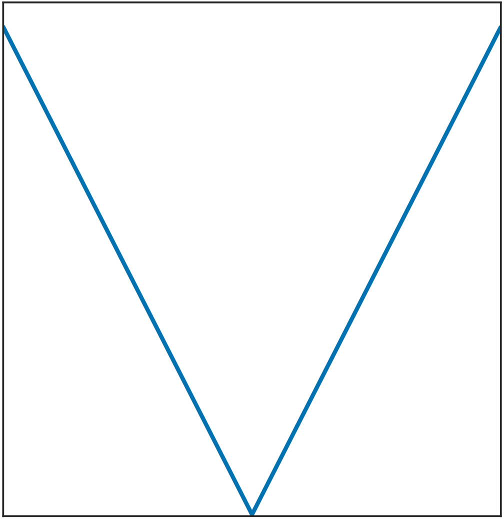
$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{régularisation}} \right)}$$

- SCAD (*Smoothly Clipped Absolute Deviation*) Fan et Li (2001) pour $\lambda > 0$ et $\gamma > 2$

$$\mathrm{pen}_{\lambda,\gamma}(t) = \begin{cases} \lambda|t|, & \text{if } |t| \leqslant \lambda \\ \frac{\gamma\lambda|t|-(t^2+\lambda^2)/2}{\gamma-1}, & \text{if } \lambda < |t| \leqslant \gamma\lambda \\ \frac{\lambda^2(\gamma^2-1)}{2(\gamma-1)}, & \text{if } |t| > \gamma\lambda \end{cases}$$

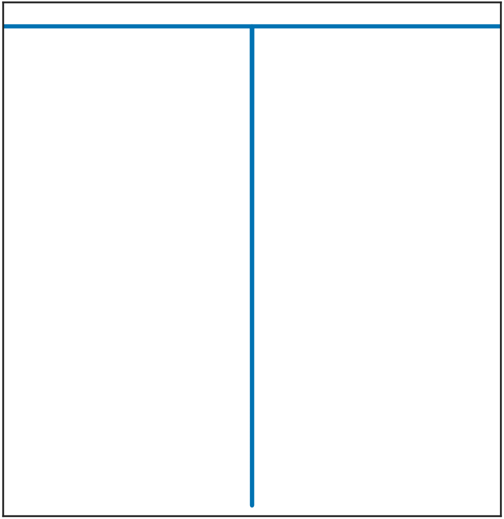
<u>Rem</u>: difficultés algorithmiques (arrêt, minima locaux, etc.) et théoriques

Forme des pénalités classiques



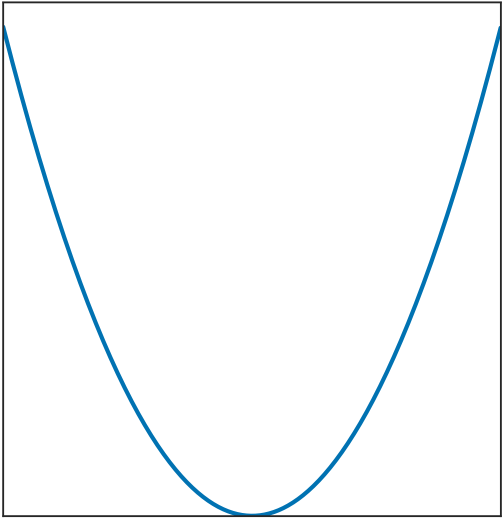
l1

Forme des pénalités classiques



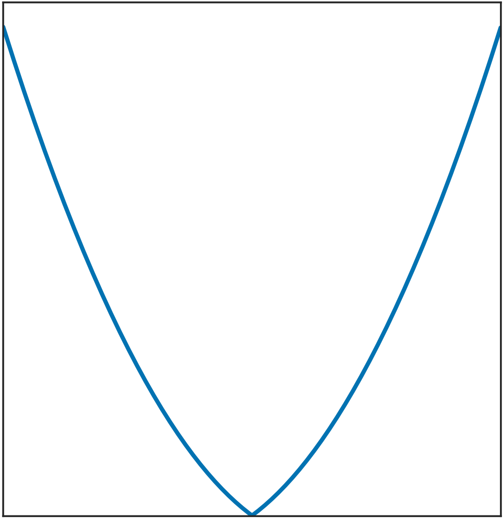
l0

Forme des pénalités classiques



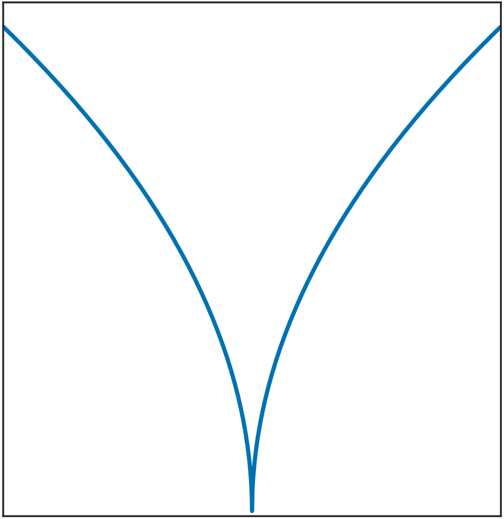
l22

Forme des pénalités classiques



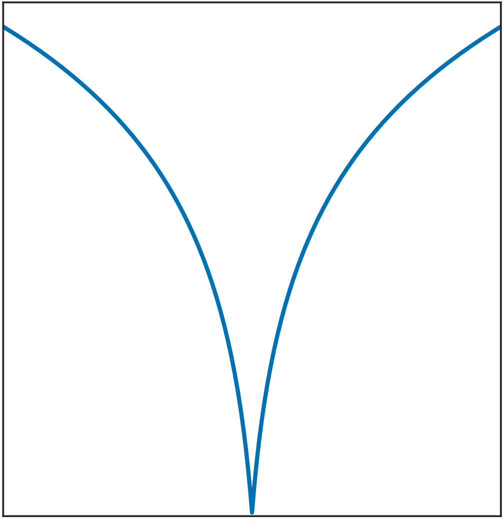
enet

Forme des pénalités classiques



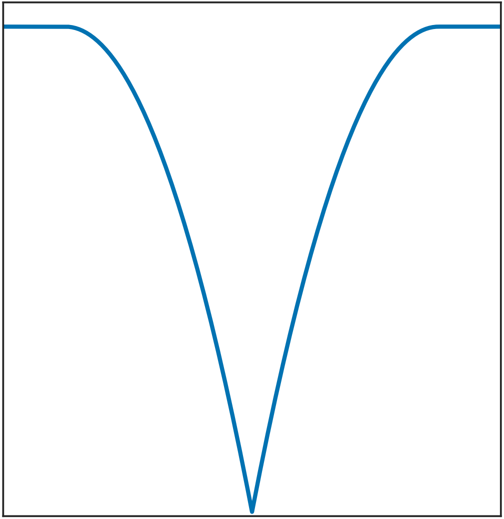
sqrt

Forme des pénalités classiques



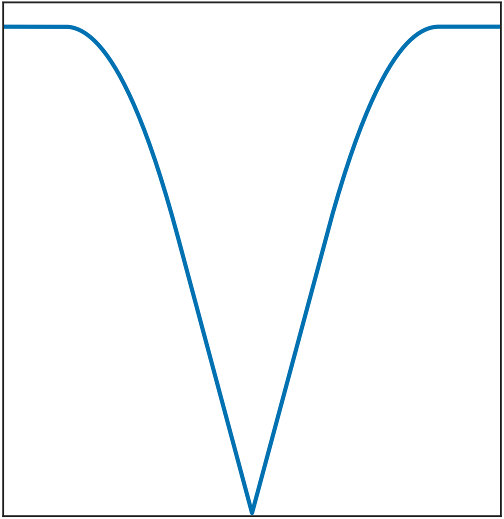
log

Forme des pénalités classiques



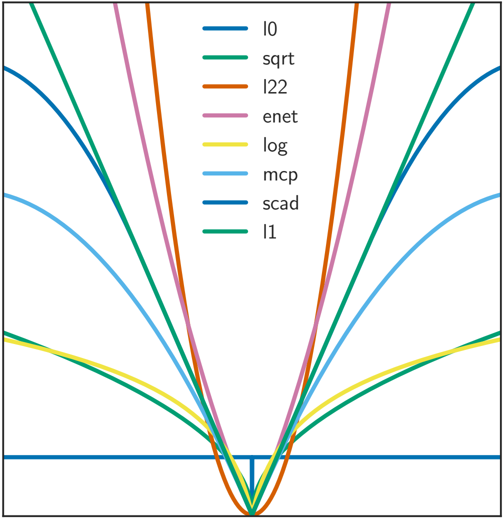
mcp

Forme des pénalités classiques



scad

Forme des pénalités classiques



Adaptive-Lasso

Plusieurs noms pour une même idée :

- Adaptive-Lasso Zou (2006)
- ℓ_1 re-pondérés Candès *et al.* (2008)
- approche DC-programming (pour *Difference of Convex Programming*) Gasso *et al.* (2008)

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ

Initialisation : $\hat{w} \leftarrow (1, \dots, 1)^\top$

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ
Initialisation : $\hat{w} \leftarrow (1, \dots, 1)^\top$
pour $k = 1, \dots, K$ **faire**

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda |t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ

Initialisation : $\hat{w} \leftarrow (1, \dots, 1)^\top$

pour $k = 1, \dots, K$ **faire**

$$\hat{\boldsymbol{\theta}} \leftarrow \arg\min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^{p} \hat{w}_j |\theta_j| \right)$$

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ

Initialisation : $\hat{w} \leftarrow (1, \dots, 1)^\top$

pour $k = 1, \dots, K$ **faire**

$$\hat{\boldsymbol{\theta}} \leftarrow \arg\min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^{p} \hat{w}_j |\theta_j| \right)$$

$$\hat{w}_j \leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}}, \ \forall j \in [\![1, p]\!]$$

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ

Initialisation : $\hat{w} \leftarrow (1, \dots, 1)^\top$

pour $k = 1, \dots, K$ **faire**

$$\hat{\boldsymbol{\theta}} \leftarrow \underset{\boldsymbol{\theta}}{\arg\min} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^{p} \hat{w}_j |\theta_j| \right)$$

$$\hat{w}_j \leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}}, \ \forall j \in [\![1, p]\!]$$

Rem: en pratique pas besoin d'itérer beaucoup (2 / 3 itérations)

Adaptive Lasso

Exemple : prendre $\mathrm{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ avec $q = 1/2$

Algorithme : Adaptive Lasso (cas $q = 1/2$)

Entrées : $X, \mathbf{y}$, nombre d'itérations K, régularisation λ

Initialisation : $\hat{w} \leftarrow (1, \ldots, 1)^\top$

pour $k = 1, \ldots, K$ **faire**

$$\hat{\boldsymbol{\theta}} \leftarrow \arg\min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^{p} \hat{w}_j |\theta_j| \right)$$

$$\hat{w}_j \leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}},\ \forall j \in [\![1, p]\!]$$

Rem: en pratique pas besoin d'itérer beaucoup (2 / 3 itérations)

Rem: utiliser un solveur Lasso pour mettre à jour $\hat{\boldsymbol{\theta}}$

Structure du support

On suppose ici que l'on connaît une structure de groupes sur les variables au préalable de l'étude : $[\![1, p]\!] = \bigcup_{g \in \mathcal{G}} g$

Vecteur et ses coordonnées actives (en orange) :

Support creux : quelconque

Pénalité envisagée : Lasso

$\|\theta\|_1 = \sum_{j=1}^{p} |\theta_j|$

Structure du support

On suppose ici que l'on connaît une structure de groupes sur les variables au préalable de l'étude : $[\![1, p]\!] = \bigcup_{g \in \mathcal{G}} g$

Vecteur et ses coordonnées actives (en orange) :

Support creux : groupes

Pénalité envisagée : Groupe-Lasso

$\|\theta\|_{2,1} = \sum_{g \in G} \|\theta_g\|_2$

Structure du support

On suppose ici que l'on connaît une structure de groupes sur les variables au préalable de l'étude : $[\![1, p]\!] = \bigcup_{g \in \mathcal{G}} g$

Vecteur et ses coordonnées actives (en orange) :

Support creux : groupes + sous groupes

Pénalité envisagée : Sparse-Groupe-Lasso

$$\alpha \|\theta\|_1 + (1-\alpha)\|\theta\|_{2,1} = \alpha \sum_{j=1}^{p} |\theta_j| + (1-\alpha) \sum_{g \in G} \|\theta_g\|_2$$

Sommaire

Groupe-Lasso

La pénalisation par la norme ℓ_1 assure que peu de coefficients sont actifs, mais aucune autre structure sur le support n'est utilisée

On peut chercher à avoir :

- Parcimonie par groupe/bloc : Groupe-Lasso Yuan et Lin (2006)
- Parcimonie individuelle et par groupe : Sparse Groupe-Lasso Simon, Friedman, Hastie et Tibshirani (2012)
- Structures hiérarchiques (par exemple avec les interactions d'ordre supérieur) Bien,Taylor et Tibshirani (2013)
- Structures sur des graphes, des gradients, etc.

Sommaire

Stabilisation du Lasso

Le Lasso peut être **instable** : quand il n'y a pas unicité de la solution (*e.g.*, quand $p > n$) selon le solveur numérique et la précision demandée, il peut y avoir des erreurs sur les variables sélectionnées.
On peur limiter ce genre de défauts en utilisant des techniques de ré-échantillonnage :

- Bolasso Bach (2008)
- Stability Selection Meinshausen et Buhlmann (2010)

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

  Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$

  Calculer le Lasso sur cet échantillon : $\hat{\boldsymbol{\theta}}_{\lambda}^{\mathrm{Lasso},(k)}$

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

- Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$
- Calculer le Lasso sur cet échantillon : $\hat{\boldsymbol{\theta}}_{\lambda}^{\mathrm{Lasso},(k)}$
- Calculer le support associé : $S_k = \operatorname{supp}\left(\hat{\boldsymbol{\theta}}_{\lambda}^{\mathrm{Lasso},(k)}\right)$

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$

Calculer le Lasso sur cet échantillon : $\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso},(k)}$

Calculer le support associé : $S_k = \operatorname{supp}\left(\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso},(k)}\right)$

Calculer : $S := \displaystyle\bigcap_{k=1}^{B} S_k$

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \dots, B$ **faire**

- Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$
- Calculer le Lasso sur cet échantillon : $\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso},(k)}$
- Calculer le support associé : $S_k = \operatorname{supp}\left(\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso},(k)}\right)$

Calculer : $S := \displaystyle\bigcap_{k=1}^{B} S_k$

Calculer : $\hat{\boldsymbol{\theta}}_\lambda^{\text{Bolasso}} \in \displaystyle\mathop{\arg\min}_{\substack{\boldsymbol{\theta}\in\mathbb{R}^p \\ \operatorname{supp}(\boldsymbol{\theta})=S}} \frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$

Exo: coder le Bolasso avec `Python` et `sklearn`

Bolasso Bach (2008)

Algorithme : Bootstrap Lasso

Entrées : $X, \mathbf{y}$, nombre de réplications B, régularisation λ

pour $k = 1, \ldots, B$ **faire**

Générer un échantillon *bootstrap* : $X^{(k)}, y^{(k)}$

Calculer le Lasso sur cet échantillon : $\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso},(k)}$

Calculer le support associé : $S_k = \mathrm{supp}\left(\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Lasso},(k)}\right)$

Calculer : $S := \bigcap_{k=1}^{B} S_k$

Calculer : $\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Bolasso}} \in \mathop{\arg\min}_{\substack{\boldsymbol{\theta}\in\mathbb{R}^p \\ \mathrm{supp}(\boldsymbol{\theta})=S}} \frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2$

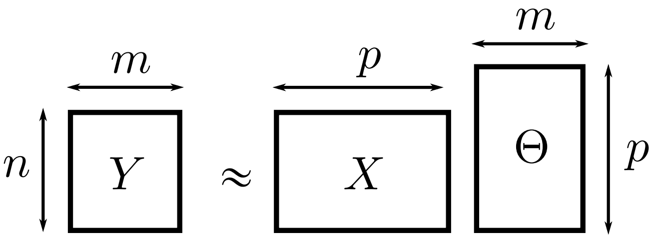
Sorties : un support S, et un vecteur $\hat{\boldsymbol{\theta}}_\lambda^{\mathrm{Bolasso}}$

Exo: coder le Bolasso avec `Python` et `sklearn`

Sommaire

Régression multi-tâches

On veut résoudre m régressions linéaires conjointement : $Y \approx X\Theta$



avec

- $Y \in \mathbb{R}^{n \times m}$: matrice des observations
- $X \in \mathbb{R}^{n \times p}$: matrice de design (commune)
- $\Theta \in \mathbb{R}^{p \times m}$: matrice des coefficients

Exemple : plusieurs signaux sont observés au cours du temps (*e.g.*, divers capteurs d'un même phénomène)

Rem: *cf.* `MultiTaskLasso` dans `sklearn` pour le numérique

Moindre carres pénalisées

Dans le contexte de la régression multi-tâches on peut résoudre les moindres carrés pénalisés :

$$\boxed{\hat{\Theta}_\lambda = \underset{\Theta \in \mathbb{R}^{p\times m}}{\arg\min} \left(\underbrace{\frac{1}{2}\| Y - X\Theta\|_F^2}_{\textbf{attache aux données}} + \underbrace{\lambda\Omega(\Theta)}_{\textbf{régularisation}} \right)}$$

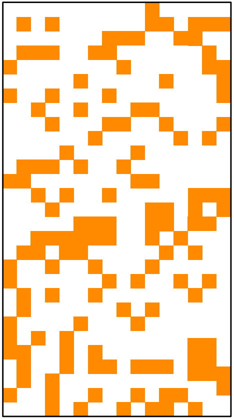
où Ω est une pénalité / régularisation à préciser

Rem: la norme de Frobenius $\|\cdot\|_F$ est définie pour toute matrice $A \in \mathbb{R}^{n_1 \times n_2}$ par

$$\|A\|_F^2 = \sum_{j_1=1}^{n_1} \sum_{j_2=1}^{n_2} A_{j_1,j_2}^2$$

Pénalisation pour le cas multi-tâches

On doit adapter les pénalisations vectorielles rencontrées :



Paramètre $\Theta \in \mathbb{R}^{p \times m}$

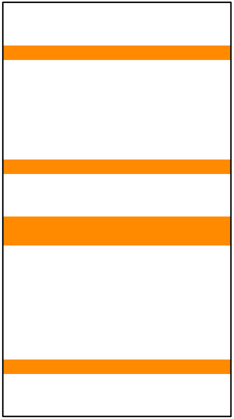
Support creux :
quelconque

Pénalité Lasso :

$$\|\Theta\|_1 = \sum_{j=1}^{p} \sum_{k=1}^{m} |\Theta_{j,k}|$$

Pénalisation pour le cas multi-tâches

On doit adapter les pénalisations vectorielles rencontrées :



Paramètre $\Theta \in \mathbb{R}^{p \times m}$

Support creux :
groupes

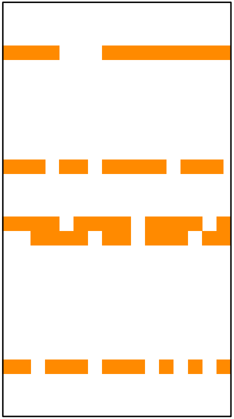
Pénalité Groupe-Lasso :

$$\|\Theta\|_{2,1} = \sum_{j=1}^{p} \|\Theta_{j:}\|_2$$

Rem: on note $\Theta_{j,:}$ la j^{e} ligne de Θ

Pénalisation pour le cas multi-tâches

On doit adapter les pénalisations vectorielles rencontrées :



Paramètre $\Theta \in \mathbb{R}^{p \times m}$

Support creux :
groupes + sous groupes

Pénalité Sparse-Groupe-Lasso :

$$\alpha \|\Theta\|_1 + (1 - \alpha) \|\Theta\|_{2,1}$$

Sommaire

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$\theta_1^{(k)} \leftarrow \underset{\theta_1 \in \mathbb{R}}{\arg\min} f(\theta_1 \quad, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min}\, f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \underset{\theta_1\in\mathbb{R}}{\arg\min}\, f(\theta_1\ , \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \underset{\theta_2\in\mathbb{R}}{\arg\min}\, f(\theta_1^{(k)}, \theta_2\ , \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta} \in \mathbb{R}^p}{\arg\min} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \ldots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \underset{\theta_1 \in \mathbb{R}}{\arg\min} f(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \ldots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \underset{\theta_2 \in \mathbb{R}}{\arg\min} f(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \ldots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_3^{(k)} \leftarrow \underset{\theta_3 \in \mathbb{R}}{\arg\min} f(\theta^{(k)}, \theta_2^{(k)}, \theta_3, \ldots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min}\, f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \underset{\theta_1\in\mathbb{R}}{\arg\min}\, f(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \underset{\theta_2\in\mathbb{R}}{\arg\min}\, f(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_3^{(k)} \leftarrow \underset{\theta_3\in\mathbb{R}}{\arg\min}\, f(\theta^{(k)}, \theta_2^{(k)}, \theta_3, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\vdots$$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \arg\min_{\theta_1\in\mathbb{R}} f(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \arg\min_{\theta_2\in\mathbb{R}} f(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_3^{(k)} \leftarrow \arg\min_{\theta_3\in\mathbb{R}} f(\theta^{(k)}, \theta_2^{(k)}, \theta_3, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\vdots$$

$$\theta_p^{(k)} \leftarrow \arg\min_{\theta_p\in\mathbb{R}} f(\theta_1^{(k)}, \theta_2^{(k)}, \theta_3^{(k)}, \dots, \theta_{p-1}^{(k)}, \theta_p)$$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\displaystyle\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \arg\min_{\theta_1\in\mathbb{R}} f(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \arg\min_{\theta_2\in\mathbb{R}} f(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_3^{(k)} \leftarrow \arg\min_{\theta_3\in\mathbb{R}} f(\theta^{(k)}, \theta_2^{(k)}, \theta_3, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\vdots$$

$$\theta_p^{(k)} \leftarrow \arg\min_{\theta_p\in\mathbb{R}} f(\theta_1^{(k)}, \theta_2^{(k)}, \theta_3^{(k)}, \dots, \theta_{p-1}^{(k)}, \theta_p)$$

Sorties : $\boldsymbol{\theta}^{(K)}$

Descente par coordonnée

Objectif : trouver une solution (approchée !) de $\quad \underset{\boldsymbol{\theta}\in\mathbb{R}^p}{\arg\min} f(\boldsymbol{\theta})$

Algorithme : Descente par coordonnée

Entrées : f, nombre d'« époques » K (ou de « passes »)

Initialisation : $k = 0$ et $\boldsymbol{\theta}^{(k)} = 0 \in \mathbb{R}^p$

pour $k = 1, \dots, K$ **faire**

$$\theta_1^{(k)} \leftarrow \underset{\theta_1\in\mathbb{R}}{\arg\min} f(\theta_1, \theta_2^{(k-1)}, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_2^{(k)} \leftarrow \underset{\theta_2\in\mathbb{R}}{\arg\min} f(\theta_1^{(k)}, \theta_2, \theta_3^{(k-1)}, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\theta_3^{(k)} \leftarrow \underset{\theta_3\in\mathbb{R}}{\arg\min} f(\theta^{(k)}, \theta_2^{(k)}, \theta_3, \dots, \theta_{p-1}^{(k-1)}, \theta_p^{(k-1)})$$

$$\vdots$$

$$\theta_p^{(k)} \leftarrow \underset{\theta_p\in\mathbb{R}}{\arg\min} f(\theta_1^{(k)}, \theta_2^{(k)}, \theta_3^{(k)}, \dots, \theta_{p-1}^{(k)}, \theta_p)$$

Sorties : $\boldsymbol{\theta}^{(K)}$

Autre critères d'arrêts : itéré stable, objectif stable, saut de dualité

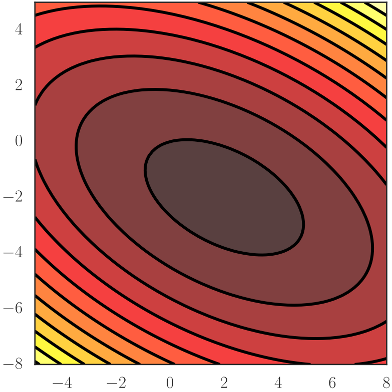
...

Intérêt

- la descente par coordonnée peut être est utile quand p est grand particulièrement
- mathématiquement cet algorithme converge vers un minimum (sous certaines conditions : fonction lisse, ou bien non lisse mais séparable *cf.* Tseng (2001))
- parcours possible : cyclique, aléatoire, avec/sans remise, etc.
- on peut faire le même raisonnement par bloc : on ne met plus à jour une coordonnée, mais tout un groupe/bloc (: *Block Coordinate Descent*)

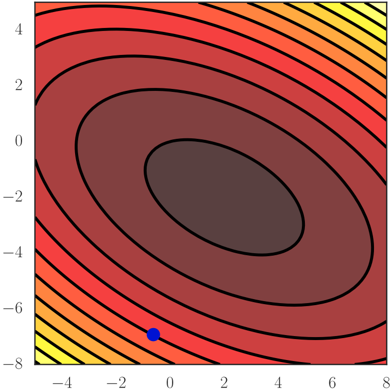
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



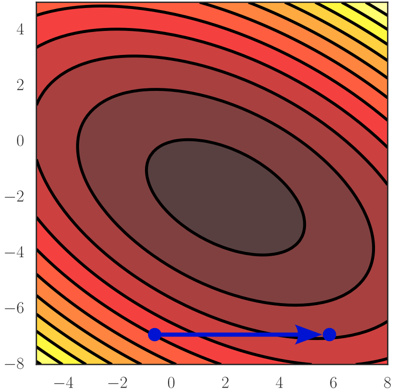
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



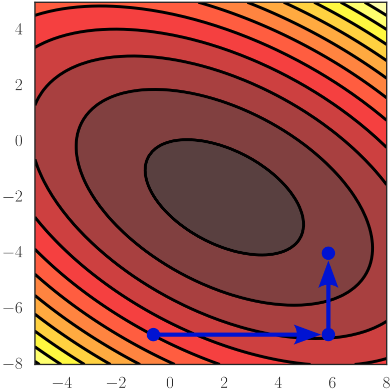
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



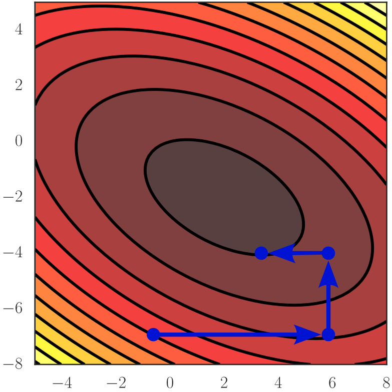
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



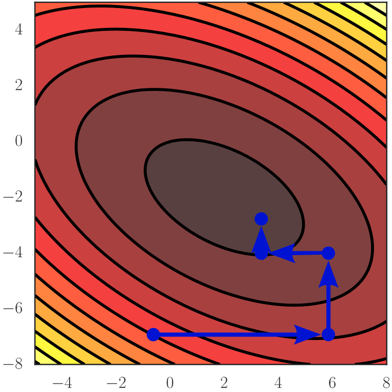
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



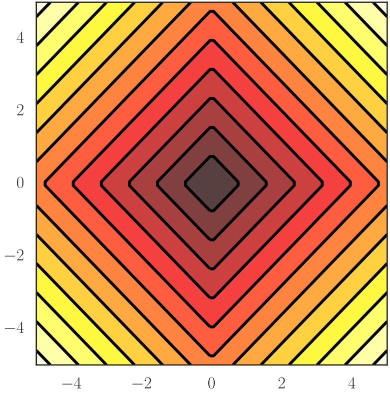
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions lisses
  cf. Tseng (2001)



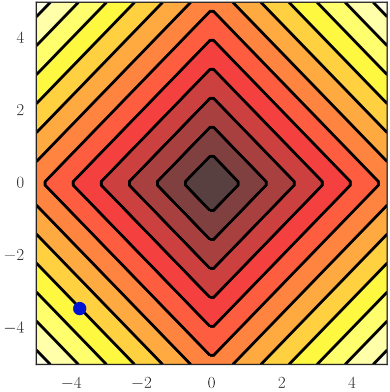
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions non-lisses séparables *cf.* Tseng (2001)



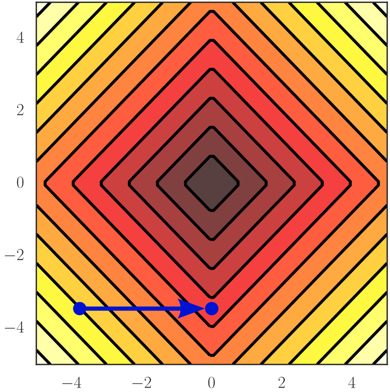
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions non-lisses séparables *cf.* Tseng (2001)



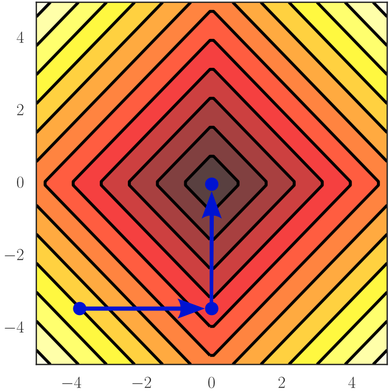
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions non-lisses séparables *cf.* Tseng (2001)



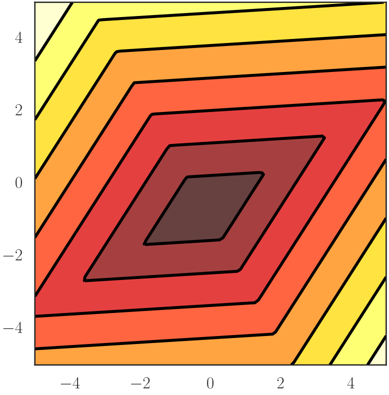
Motivation (cas convexe)

- Convergence vers un minimum pour des fonctions non-lisses séparables *cf.* Tseng (2001)



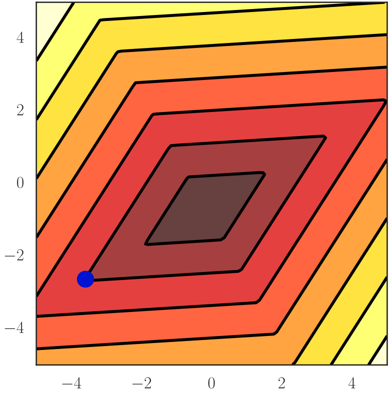
Motivation (cas convexe)

- Attention : pas (toujours) convergence vers un minimum pour des fonctions non-séparables/non-lisses



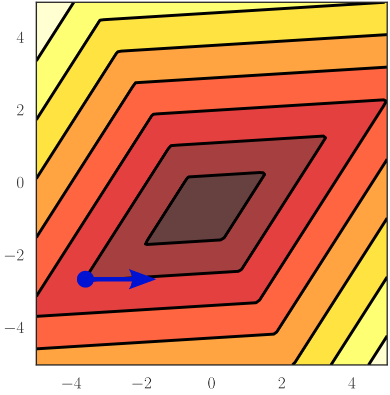
Motivation (cas convexe)

- Attention : pas (toujours) convergence vers un minimum pour des fonctions non-séparables/non-lisses



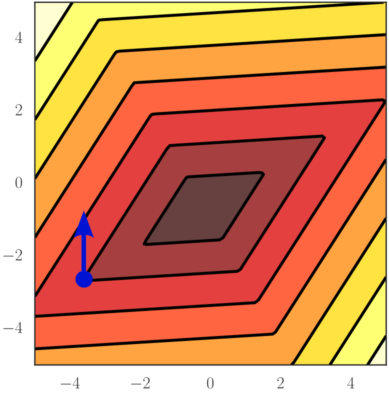
Motivation (cas convexe)

- Attention : pas (toujours) convergence vers un minimum pour des fonctions non-séparables/non-lisses



Motivation (cas convexe)

- Attention : pas (toujours) convergence vers un minimum pour des fonctions non-séparables/non-lisses



Moindre carrés

$$\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} f(\boldsymbol{\theta}) \text{ pour } f(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 = \frac{1}{2}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p}\theta_j X_{i,j})^2$$

Rappel : $\nabla f(\boldsymbol{\theta}) = X^\top(X\boldsymbol{\theta} - \mathbf{y}) = \begin{pmatrix} \mathbf{x}_1^\top(X\boldsymbol{\theta} - \mathbf{y}) \\ \vdots \\ \mathbf{x}_p^\top(X\boldsymbol{\theta} - \mathbf{y}) \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial \theta_1}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial f}{\partial \theta_p}(\boldsymbol{\theta}) \end{pmatrix}$

Minimiser en θ_j avec les autres θ_k, pour $k \neq j$, fixes :

$$0 = \frac{\partial f}{\partial \theta_j}(\boldsymbol{\theta}) = \mathbf{x}_j^\top(X\boldsymbol{\theta} - \mathbf{y}) = \mathbf{x}_j^\top\left(\mathbf{x}_j\theta_j + \sum_{k\neq j}\mathbf{x}_k\theta_k - \mathbf{y}\right)$$

$$\Leftrightarrow \theta_j = \frac{\mathbf{x}_j^\top\left(\mathbf{y} - \sum_{k\neq j}\mathbf{x}_k\theta_k\right)}{\mathbf{x}_j^\top\mathbf{x}_j} = \frac{\mathbf{x}_j^\top\left(\mathbf{y} - \sum_{k=1}^{p}\mathbf{x}_k\theta_k + \mathbf{x}_j\theta_j\right)}{\|\mathbf{x}_j\|_2^2}$$

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Pour chaque $j \in [\![1, p]\!]$, faire :

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$r^{\text{int}} \leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)}$$

Pour chaque $j \in [\![1, p]\!]$, faire :

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$
\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\text{int}} / \|\mathbf{x}_j\|_2^2 \end{aligned}
$$

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$
\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : }
\begin{aligned}
r^{\mathrm{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\
\theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\mathrm{int}} / \|\mathbf{x}_j\|_2^2 \\
r^{(k+1)} &\leftarrow r^{\mathrm{int}} - \mathbf{x}_j \theta_j^{(k+1)}
\end{aligned}
$$

CD pour les moindres carrés

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\text{int}} / \|\mathbf{x}_j\|_2^2 \\ r^{(k+1)} &\leftarrow r^{\text{int}} - \mathbf{x}_j \theta_j^{(k+1)} \end{aligned}$$

Impact mémoire faible :

- stocker un vecteur de résidus de taille n
- stocker un vecteur d'estimation de taille p

Rem: $\|\mathbf{x}_j\|_2^2 = 1$ utile en optimisation ($\neq$ en statistique)

Ridge : descente par coordonnée

$$\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} f(\boldsymbol{\theta}) \text{ pour } f(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p}\theta_j X_{i,j})^2 + \frac{\lambda}{2}\sum_{j=1}^{p}\theta_j^2$$

$$\nabla f(\boldsymbol{\theta}) = X^\top(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\boldsymbol{\theta} = \begin{pmatrix} \mathbf{x}_1^\top(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\theta_1 \\ \vdots \\ \mathbf{x}_p^\top(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\theta_p \end{pmatrix} = \begin{pmatrix} \frac{\partial f}{\partial\theta_1}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial f}{\partial\theta_p}(\boldsymbol{\theta}) \end{pmatrix}$$

Minimise en θ_j avec les autres θ_k $(k \neq j)$ fixes

$$0 = \frac{\partial f}{\partial\theta_j}(\boldsymbol{\theta}) = \mathbf{x}_j^\top(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\theta_j = \mathbf{x}_j^\top\left(\mathbf{x}_j\theta_j + \sum_{k\neq j}\mathbf{x}_k\theta_k - \mathbf{y}\right) + \lambda\theta_j$$

$$\Leftrightarrow \theta_j = \frac{\mathbf{x}_j^\top\left(\mathbf{y} - \sum_{k\neq j}\mathbf{x}_k\theta_k\right)}{\mathbf{x}_j^\top\mathbf{x}_j + \lambda} = \frac{\mathbf{x}_j^\top\left(\mathbf{y} - \sum_{k=1}^{p}\mathbf{x}_k\theta_k + \mathbf{x}_j\theta_j\right)}{\|\mathbf{x}_j\|_2^2 + \lambda}$$

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Pour chaque $j \in [\![1, p]\!]$, faire :

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$r^{\text{int}} \leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)}$$

Pour chaque $j \in [\![1, p]\!]$, faire :

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\text{int}} / (\|\mathbf{x}_j\|_2^2 + \lambda) \end{aligned}$$

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\text{int}} / (\|\mathbf{x}_j\|_2^2 + \lambda) \\ r^{(k+1)} &\leftarrow r^{\text{int}} - \mathbf{x}_j \theta_j^{(k+1)} \end{aligned}$$

Ridge descente par coordonnée (II)

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \mathbf{x}_j^\top r^{\text{int}} / (\|\mathbf{x}_j\|_2^2 + \lambda) \\ r^{(k+1)} &\leftarrow r^{\text{int}} - \mathbf{x}_j \theta_j^{(k+1)} \end{aligned}$$

Impact mémoire faible :

- stocker un vecteur de résidus de taille n
- stocker un vecteur d'estimation de taille p

Rem: $\|\mathbf{x}_j\|_2^2 = 1$ utile en optimisation ($\neq$ en statistique)

Lasso : descente par coordonnée

$$\arg\min_{\boldsymbol{\theta}\in\mathbb{R}^p} f(\boldsymbol{\theta}) \text{ pour } f(\boldsymbol{\theta}) = \frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|^2 + \lambda \sum_{j=1}^{p} |\theta_j|$$

Minimise en θ_j avec les autres θ_k $(k \neq j)$ fixes

$$\begin{aligned}
\hat{\theta}_j &= \arg\min_{\theta_j\in\mathbb{R}} f(\theta_1, \ldots, \theta_p) \\
&= \arg\min_{\theta_j\in\mathbb{R}} \frac{1}{2}\|\mathbf{y} - \sum_{k\neq j} \theta_k \mathbf{x}_k - \mathbf{x}_j\theta_j\|^2 + \lambda \sum_{k\neq j} |\theta_j| + \lambda|\theta_j| \\
&= \arg\min_{\theta_j\in\mathbb{R}} \frac{1}{2}\|\mathbf{x}_j\|^2\theta_j^2 - \langle y - \sum_{k\neq j} \theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\theta_j + \lambda|\theta_j| \\
&= \arg\min_{\theta_j\in\mathbb{R}} \|\mathbf{x}_j\|^2 \left[\frac{1}{2}\left(\theta_j - \|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j} \theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)^2 + \frac{\lambda}{\|\mathbf{x}_j\|^2}|\theta_j|\right]
\end{aligned}$$

<u>Rappel</u> : $\quad \eta_{\mathrm{ST},\lambda}(z) = \arg\min_{t\in\mathbb{R}} \frac{1}{2}(z-t)^2 + \lambda|t|$

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2}\left(\|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j}\theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)$$

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2} \left(\|\mathbf{x}_j\|^{-2} \langle y - \sum_{k \neq j} \theta_k \mathbf{x}_k, \mathbf{x}_j \rangle \right)$$

<u>Mise à jour intelligente</u> : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Pour chaque $j \in [\![1, p]\!]$, faire :

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x}_\mathrm{j}\|^2}\left(\|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j}\theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)$$

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$r^{\mathrm{int}} \leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)}$$

Pour chaque $j \in [\![1, p]\!]$, faire :

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2}\left(\|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j}\theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)$$

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$r^{\mathrm{int}} \leftarrow r^{(k)} + \mathbf{x}_j\theta_j^{(k)}$$

Pour chaque $j \in [\![1, p]\!]$, faire : $\theta_j^{(k+1)} \leftarrow \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2}\left(\mathbf{x}_j^\top r^{\mathrm{int}}/\|\mathbf{x}_j\|^2\right)$

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x}_\mathrm{j}\|^2}\left(\|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j}\theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)$$

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\mathrm{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \eta_{\mathrm{ST},\lambda/\|\mathbf{x}_\mathrm{j}\|^2}\left(\mathbf{x}_j^\top r^{\mathrm{int}} / \|\mathbf{x}_j\|^2\right) \\ r^{(k+1)} &\leftarrow r^{\mathrm{int}} - \mathbf{x}_j \theta_j^{(k+1)} \end{aligned}$$

Lasso : descente par coordonnée (II)

Solution :

$$\hat{\theta}_j = \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2}\left(\|\mathbf{x}_j\|^{-2}\langle y - \sum_{k\neq j}\theta_k \mathbf{x}_k, \mathbf{x}_j\rangle\right)$$

Mise à jour intelligente : stocker les résidus courants $r^{(k)}$ et les coefficients dans $\boldsymbol{\theta}^{(k)}$

Pour chaque $j \in [\![1,p]\!]$, faire :
$$\begin{aligned} r^{\mathrm{int}} &\leftarrow r^{(k)} + \mathbf{x}_j\theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \eta_{\mathrm{ST},\lambda/\|\mathbf{x_j}\|^2}\left(\mathbf{x}_j^\top r^{\mathrm{int}}/\|\mathbf{x}_j\|^2\right) \\ r^{(k+1)} &\leftarrow r^{\mathrm{int}} - \mathbf{x}_j\theta_j^{(k+1)} \end{aligned}$$

Impact mémoire faible :

- stocker un vecteur de résidus de taille n
- stocker un vecteur d'estimation de taille p

Rem: $\|\mathbf{x}_j\|_2^2 = 1$ utile en optimisation ($\neq$ en statistique)

Descente par coordonnée : cas non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\textbf{régularisation}} \right)}$$

Même approche (mais sans garantie de convergence), si $\|\mathbf{x}_j\|_2^2 = 1$

Descente par coordonnée : cas non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\text{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\sum_{j=1}^{p} \text{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{régularisation}} \right)}$$

Même approche (mais sans garantie de convergence), si $\|\mathbf{x}_j\|_2^2 = 1$

Pour chaque $j \in [\![1, p]\!]$, faire :

Descente par coordonnée : cas non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\text{pen}}_{\lambda,\gamma} = \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\sum_{j=1}^{p} \text{pen}_{\lambda,\gamma}(|\theta_j|)}_{\textbf{régularisation}} \right)}$$

Même approche (mais sans garantie de convergence), si $\|\mathbf{x}_j\|_2^2 = 1$

$$r^{\text{int}} \leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)}$$

Pour chaque $j \in [\![1, p]\!]$, faire :

Descente par coordonnée : cas non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\text{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\sum_{j=1}^{p} \text{pen}_{\lambda,\gamma}(|\theta_j|)}_{\textbf{régularisation}} \right)}$$

Même approche (mais sans garantie de convergence), si $\|\mathbf{x}_j\|_2^2 = 1$

$$\text{Pour chaque } j \in [\![1, p]\!], \text{ faire : } \begin{aligned} r^{\text{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \eta_{\text{pen}_{\lambda,\gamma}}\left(\mathbf{x}_j^\top r^{\text{int}}\right) \end{aligned}$$

Descente par coordonnée : cas non-convexe

$$\boxed{\hat{\boldsymbol{\theta}}^{\mathrm{pen}}_{\lambda,\gamma} = \mathop{\arg\min}_{\boldsymbol{\theta}\in\mathbb{R}^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{attache aux données}} + \underbrace{\sum_{j=1}^{p} \mathrm{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{régularisation}} \right)}$$

Même approche (mais sans garantie de convergence), si $\|\mathbf{x}_j\|_2^2 = 1$

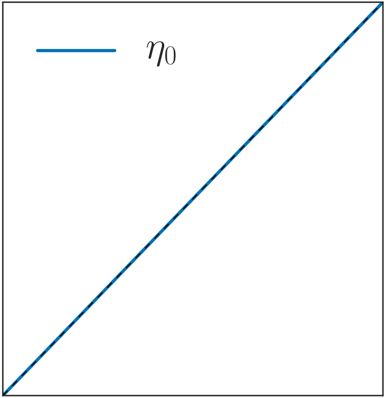
$$\text{Pour chaque } j \in [\![1, p]\!], \ \text{faire :} \quad \begin{aligned} r^{\mathrm{int}} &\leftarrow r^{(k)} + \mathbf{x}_j \theta_j^{(k)} \\ \theta_j^{(k+1)} &\leftarrow \eta_{\mathrm{pen}_{\lambda,\gamma}}\left(\mathbf{x}_j^\top r^{\mathrm{int}}\right) \\ r^{(k+1)} &\leftarrow r^{\mathrm{int}} - \mathbf{x}_j \theta_j^{(k+1)} \end{aligned}$$

$$\text{où} \quad \eta_{\mathrm{pen}_{\lambda,\gamma}}(z) = \mathop{\arg\min}_{t\in\mathbb{R}} \frac{1}{2}(z-t)^2 + \mathrm{pen}_{\lambda,\gamma}(t)$$

voir par exemple Breheny et Huang (2011)

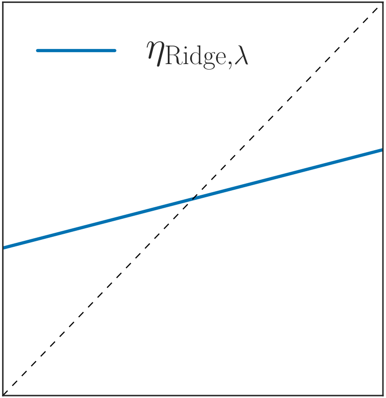
Régularisation en 1D : Aucune

$\eta_0(z) = z$



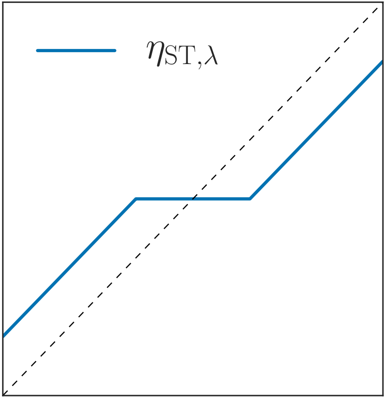
Régularisation en 1D : Ridge

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



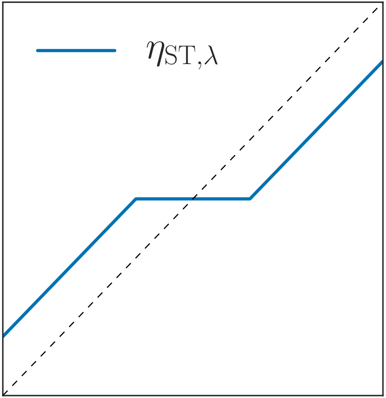
Régularisation en 1D : Lasso

$$\eta_{\mathrm{ST},\lambda}(z) = \mathrm{sign}(z)(|z| - \lambda)_+$$



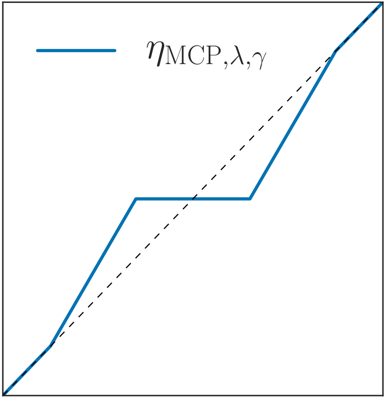
Régularisation en 1D : ℓ_0

$$\eta_{\mathrm{HT},\lambda}(z) = z\mathbb{1}_{|z|\geqslant\sqrt{2\lambda}}$$



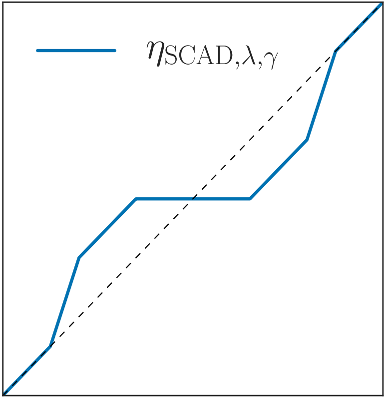
Régularisation en 1D : MCP

$$\eta_{\mathrm{MCP},\lambda,\gamma}(z) = \begin{cases} \mathrm{sign}(z)(|z| - \lambda)_+/(1 - 1/\gamma) & \text{si } |z| \leqslant \gamma\lambda \\ z & \text{si } |z| > \gamma\lambda \end{cases}$$



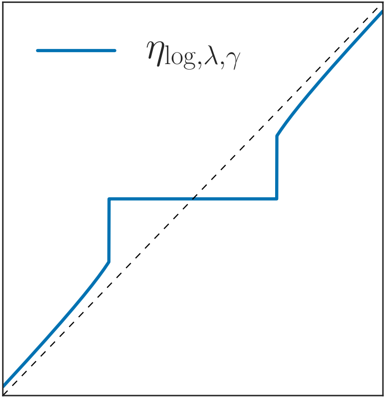
Régularisation en 1D : SCAD

$$\eta_{\mathrm{SCAD},\lambda,\gamma}(z) = \begin{cases} \mathrm{sign}(z)(|z|-\lambda)_+/(1-1/\gamma) & \text{si } |z| \leqslant 2\lambda \\ ([\gamma-1)z - \mathrm{sign}(z)\gamma\lambda]/(\gamma-2) & \text{si } 2\lambda \leqslant |z| \leqslant \gamma\lambda \\ z & \text{si } |z| > \gamma\lambda \end{cases}$$



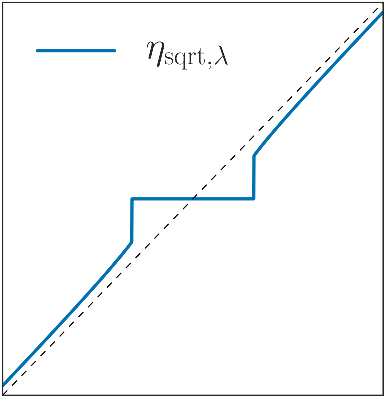
Régularisation en 1D : $\log$

$\eta_{\log,\lambda}(z) = \ldots$



Régularisation en 1D : sqrt

$\eta_{\mathrm{sqrt},\lambda}(z) = \ldots$



Lasso-Positif

Exo: Proposer une manière de résoudre le problème Lasso avec une contrainte de positivité sur les coefficients

$$\boxed{\hat{\boldsymbol{\theta}}_\lambda^{\text{Lasso+}} = \mathop{\arg\min}_{\boldsymbol{\theta} \in \mathbb{R}_+^p} \left(\underbrace{\frac{1}{2}\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\textbf{attache aux données}} + \underbrace{\lambda\|\boldsymbol{\theta}\|_1}_{\textbf{régularisation}} \right)}$$

Sommaire

Optimisation : autres méthodes

D'autres algorithmes peuvent être utilisés pour construire une solution approchée du Lasso :

- LARS Efron *et al.* (2004) pour le chemin entier
- méthodes de gradient proximal, Forward-Backward, de type Seuillage Doux Itératif (: *ISTA, FISTA*), *cf.* Beck et Teboulle(2009)

Ces dernières méthodes seront vues en INFMDI 341

Références I

- F. Bach.

  Bolasso : model consistent Lasso estimation through the bootstrap.

  In *ICML*, 2008.

- P. Breheny and J. Huang.

  Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection.

  5(1) :232, 2011.

- A. Beck and M. Teboulle.

  A fast iterative shrinkage-thresholding algorithm for linear inverse problems.

  SIAM J. Imaging Sci., 2(1) :183–202, 2009.

- P. Bühlmann and S. van de Geer.

  Statistics for high-dimensional data.

  Springer Series in Statistics. Springer, Heidelberg, 2011.

  Methods, theory and applications.

Références II

- 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.
- B. Efron, T. Hastie, I. M. Johnstone, and R. Tibshirani.
  Least angle regression.
  Ann. Statist., 32(2) :407–499, 2004.
  With discussion, and a rejoinder by the authors.
- J. Fan and R. Li.
  Variable selection via nonconcave penalized likelihood and its oracle properties.
  J. Amer. Statist. Assoc., 96(456) :1348–1360, 2001.
- G. Gasso, A. Rakotomamonjy, and S. Canu.
  Recovering sparse signals with non-convex penalties and DC programming.
  57(12) :4686–4698, 2009.

Références III

- Bien J, J. Taylor, and R. Tibshirani.
  A lasso for hierarchical interactions.
  Ann. Statist., 41(3) :1111–1141, 2013.

- N. Meinshausen and P. Bühlmann.
  Stability selection.
  Journal of the Royal Statistical Society : Series B (Statistical Methodology), 72(4) :417–473, 2010.

- N. Parikh, S. Boyd, E. Chu, B. Peleato, and J. Eckstein.
  Proximal algorithms.
  Foundations and Trends in Machine Learning, 1(3) :1–108, 2013.

- N. Simon, J. Friedman, T. Hastie, and R. Tibshirani.
  A sparse-group lasso.
  J. Comput. Graph. Statist., 22(2) :231–245, 2013.

- R. Tibshirani.
  Regression shrinkage and selection via the lasso.
  J. Roy. Statist. Soc. Ser. B, 58(1) :267–288, 1996.

Références IV

- P. Tseng.

  Convergence of a block coordinate descent method for nondifferentiable minimization.

  J. Optim. Theory Appl., 109(3) :475–494, 2001.

- M. Yuan and Y. Lin.

  Model selection and estimation in regression with grouped variables.

  J. Roy. Statist. Soc. Ser. B, 68(1) :49–67, 2006.

- H. Zou and T. Hastie.

  Regularization and variable selection via the elastic net.

  J. Roy. Statist. Soc. Ser. B, 67(2) :301–320, 2005.

- C.-H Zhang.

  Nearly unbiased variable selection under minimax concave penalty.

  Ann. Statist., 38(2) :894–942, 2010.

- H. Zou.

  The adaptive lasso and its oracle properties.

  J. Am. Statist. Assoc., 101(476) :1418–1429, 2006.