

MS BGD

SD 204: Lasso

Joseph Salmon

<http://josephsalmon.eu>

Télécom Paristech, Institut Mines-Télécom

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure

- Stabilisation

- Least squares / Lasso extensions

Reminding the model

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

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

Motivation

Estimators of $\hat{\theta}$ with many zero coefficient are useful:

- for interpretation
- for computational efficiency if p is huge

Underlying idea: **variable selection**

Rem: also useful if θ^* has few non-zero coefficients

Variable selection overview

- ▶ “Screening” methods: one removes the \mathbf{x}_j ’s whose correlation with observation \mathbf{y} is weak
 - pros: fast (+++), cost: p inner product of size n , intuitive (+++)
 - cons: neglect variables interactions \mathbf{x}_j , weak theoretical results (- - -)
- ▶ **Greedy** methods aka stagewise / stepwise
 - pros: fast (++), intuitive (++)
 - cons: propagates wrong selection forward; weak theoretical results(-)
- ▶ Sparsity enforcing **penalized** method (e.g., Lasso)
 - pros: better theory (++)
 - cons: can be still slow (-),

The ℓ_0 pseudo-norm

Definition

The **support** of a vector θ is the set of indexes of non-zero coordinates:

$$\text{supp}(\theta) = \{j \in \llbracket 1, p \rrbracket, \theta_j \neq 0\}$$

The ℓ_0 **pseudo-norm** of a vector $\theta \in \mathbb{R}^p$ is the number of non-zero coordinate:

$$\|\theta\|_0 = \text{card}\{j \in \llbracket 1, p \rrbracket, \theta_j \neq 0\}$$

Rem: $\|\cdot\|_0$ is not a norm, $\forall t \in \mathbb{R}^*, \|t\theta\|_0 = \|\theta\|_0$

Rem: $\|\cdot\|_0$ it is not even convex, $\theta_1 = (1, 0, 1, \dots, 0)$

$\theta_2 = (0, 1, 1, \dots, 0)$ and $3 = \|\frac{\theta_1 + \theta_2}{2}\|_0 \geq \frac{\|\theta_1\|_0 + \|\theta_2\|_0}{2} = 2$

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure

- Stabilisation

- Least squares / Lasso extensions

The ℓ_0 penalty

First try to get a sparsity enforcing penalty: use ℓ_0 as a penalty (or regularization)

$$\hat{\boldsymbol{\theta}}_{\lambda} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_0}_{\text{regularization}} \right)$$

Combinatorial problem!!!

An exact resolution requires to consider all sub-models, *i.e.*, compute estimators for all possible support; there are 2^p , meaning one needs 2^p least squares computation!

Exemple :

$p = 10$ possible: $\approx 10^3$ least squares

$p = 30$ impossible: $\approx 10^{10}$ least squares

Rem: problem “NP-dur”, can be solved for small problems by mixed integer programming.

Syllabus

Reminders

Variable selection and sparsity

The ℓ_0 penalty and its limits

The ℓ_1 penalty

Sub-gradient / sub-differential

Improvement and extensions for the Lasso

LSLasso / Elastic-Net

Non-convex penalties / Adaptive Lasso

Support structure

Stabilisation

Least squares / Lasso extensions

Le Lasso: penalty point of view

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

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

où $\|\boldsymbol{\theta}\|_1 = \sum_{j=1}^p |\theta_j|$ (sum of absolute values of the coefficients)

- We recover the limiting cases:

$$\lim_{\lambda \rightarrow 0} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \hat{\boldsymbol{\theta}}^{\text{MCO}}$$

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

Beware: the Lasso estimator is not always **unique** for a fixed λ (consider cases with two equals columns in X)

Constraint point of view

The following problem:

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

share the same solutions as the constraints formulation:

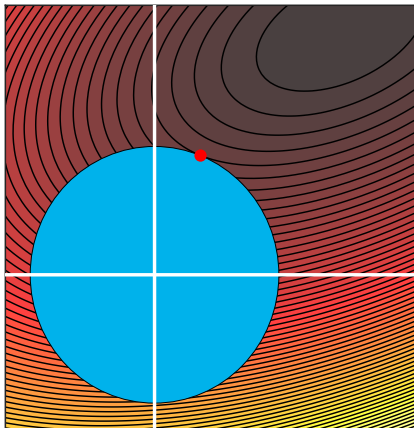
$$\begin{cases} \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 \\ \text{t.q. } \|\boldsymbol{\theta}\|_1 \leq T \end{cases}$$

for some $T > 0$.

Rem: unfortunately the link $T \leftrightarrow \lambda$ is not explicit

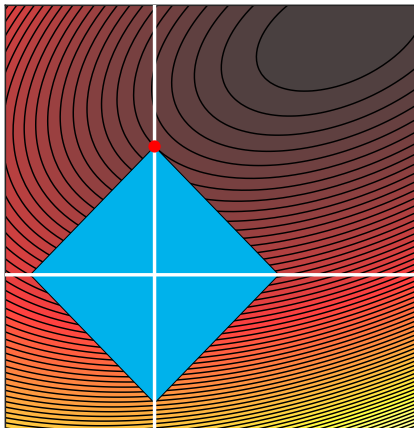
- ▶ If $T \rightarrow 0$ one recovers the null vector: $0 \in \mathbb{R}^p$
- ▶ If $T \rightarrow \infty$ one recovers $\hat{\boldsymbol{\theta}}^{\text{MCO}}$ (unconstrained)

Zeroing coefficients



Optimization under ℓ_2 constraint : non sparse solution

Zeroing coefficients



Optimization under ℓ_1 constraint : sparse solution

Syllabus

Reminders

Variable selection and sparsity

The ℓ_0 penalty and its limits

The ℓ_1 penalty

Sub-gradient / sub-differential

Improvement and extensions for the Lasso

LSLasso / Elastic-Net

Non-convex penalties / Adaptive Lasso

Support structure

Stabilisation

Least squares / Lasso extensions

Sub-gradients / sub-differential

Definitions

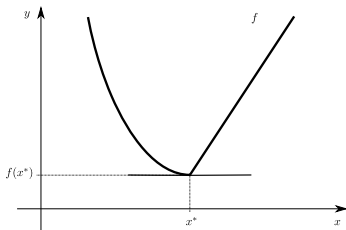
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a **sub-gradient** of f at x^* , if for any $x \in \mathbb{R}^n$,

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

The **sub-differential** is the set

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

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

Definitions

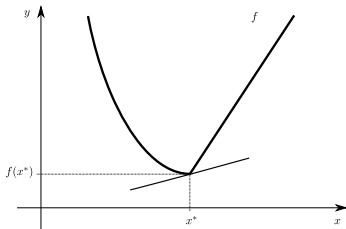
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a **sub-gradient** of f at x^* , if for any $x \in \mathbb{R}^n$,

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

The **sub-differential** is the set

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

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

Definitions

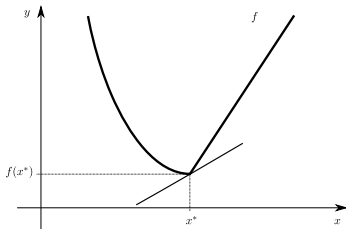
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a **sub-gradient** of f at x^* , if for any $x \in \mathbb{R}^n$,

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

The **sub-differential** is the set

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

Rem: if the sub-gradient is unique, one recovers the standard gradient



Sub-gradients / sub-differential

Definitions

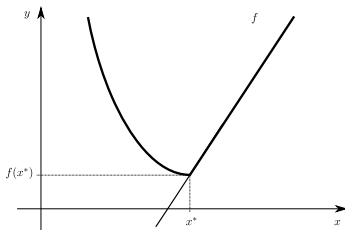
For a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $u \in \mathbb{R}^n$ is a **sub-gradient** of f at x^* , if for any $x \in \mathbb{R}^n$,

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

The **sub-differential** is the set

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

Rem: if the sub-gradient is unique, one recovers the standard gradient



Fermat's Rule

Theorem

A point x^* is a minimum of a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ if and only if $0 \in \partial f(x^*)$

Proof: use the sub-gradient definition:

- ▶ 0 is a sub-gradient of f at x^* if and only if
$$\forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle 0, x - x^* \rangle$$

Fermat's Rule

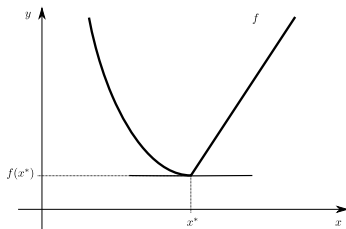
Theorem

A point x^* is a minimum of a convex function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ if and only if $0 \in \partial f(x^*)$

Proof: use the sub-gradient definition:

- ▶ 0 is a sub-gradient of f at x^* if and only if
$$\forall x \in \mathbb{R}^n, f(x) \geq f(x^*) + \langle 0, x - x^* \rangle$$

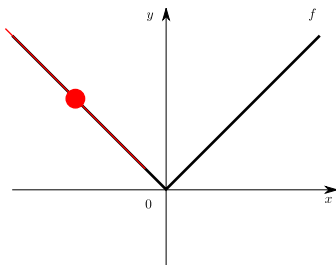
Rem: Visually, it corresponds to a horizontal tangent



Absolute value sub-differential

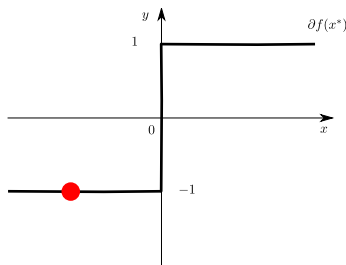
Function (abs):

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



Sub-differential (sign)

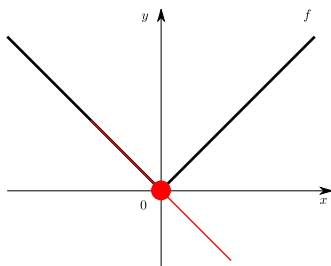
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

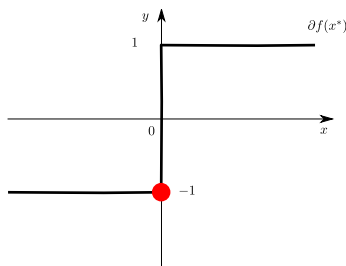
Function (abs):

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



Sub-differential (sign)

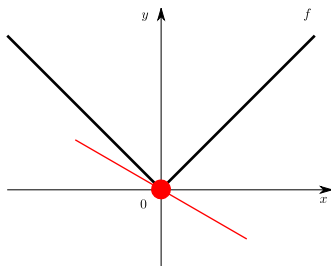
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

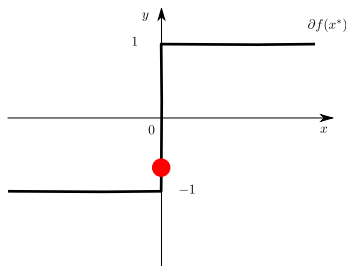
Function (abs):

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



Sub-differential (sign)

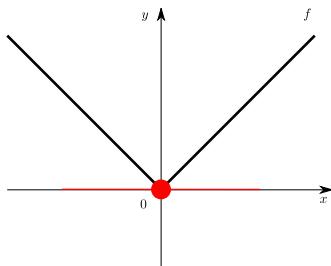
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

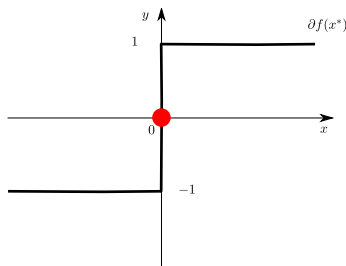
Function (abs):

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



Sub-differential (sign)

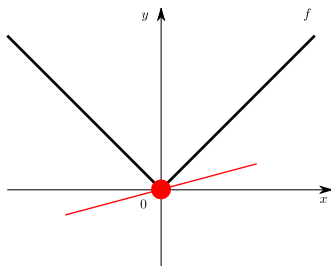
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

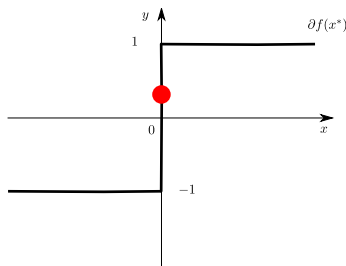
Function (abs):

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



Sub-differential (sign)

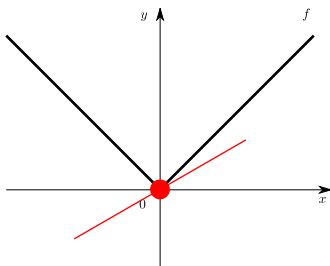
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

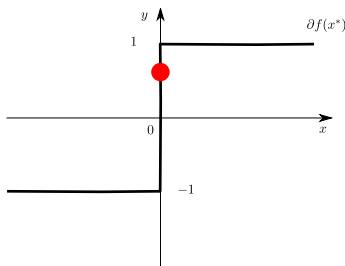
Function (abs):

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



Sub-differential (sign)

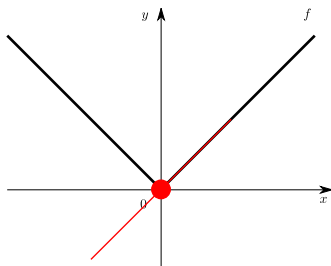
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

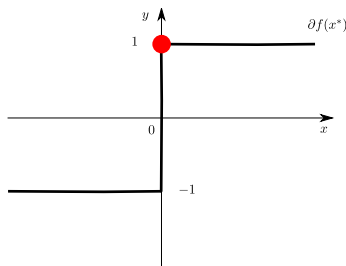
Function (abs):

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



Sub-differential (sign)

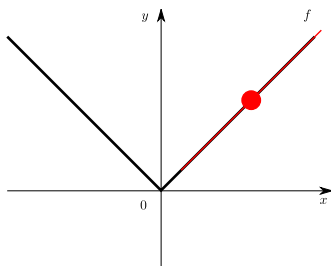
$$\partial f(x^*) = \begin{cases} \{-1\} & \text{if } x^* \in]-\infty, 0[\\ \{1\} & \text{if } x^* \in]0, \infty[\\ [-1, 1] & \text{if } x^* = 0 \end{cases}$$



Absolute value sub-differential

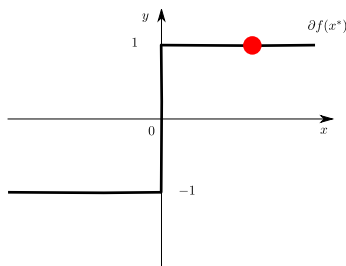
Function (abs):

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



Sub-differential (sign)

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



Fermat's rule for the Lasso

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\lambda \|\boldsymbol{\theta}\|_1}_{\text{regularization}} \right)$$

Necessary and sufficient optimality (Fermat):

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

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

Orthogonal case: soft thresholding

The simple case of orthogonal design: $X^\top X = \text{Id}_p$

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

cause X is an isometry in such a case, the Lasso objective become:

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

Separable problem: problem that can be reduced to minimizing coordinate by coordinate (independently)

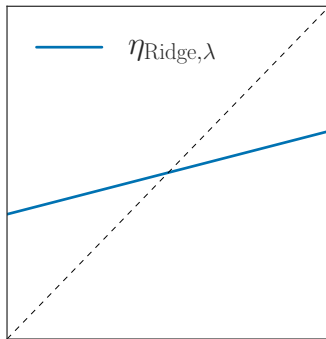
One needs to minimize: $x \mapsto \frac{1}{2}(z - x)^2 + \lambda|x|$ for $z = \mathbf{x}_j^\top \mathbf{y}$

Rem: this function is called the **proximal operator** at z of the function $x \mapsto \lambda|x|$ (cf. Parikh and Boyd (2013), for more details on proximal methods)

1D Regularization: Ridge

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

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

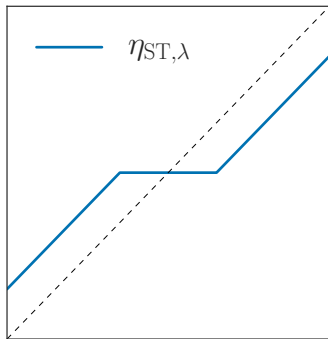


ℓ_2 shrinkage : Ridge

1D Regularization: Lasso

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

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

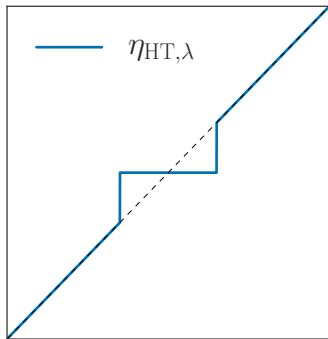


ℓ_1 shrinkage: soft thresholding

1D Regularization: ℓ_0

Solve: $\eta_\lambda(z) = \arg \min_{x \in \mathbb{R}} x \mapsto \frac{1}{2}(z - x)^2 + \lambda \mathbb{1}_{x \neq 0}$

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

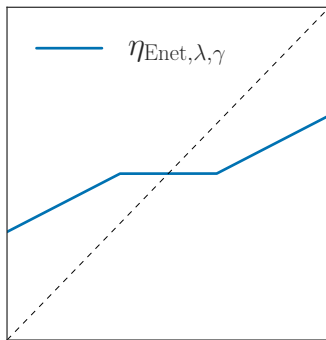


ℓ_0 shrinkage: hard thresholding

1D Regularization: enet

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

$\eta_\lambda(z)$ = Exercise



ℓ_1/ℓ_2

Soft thresholding: closed form solution

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

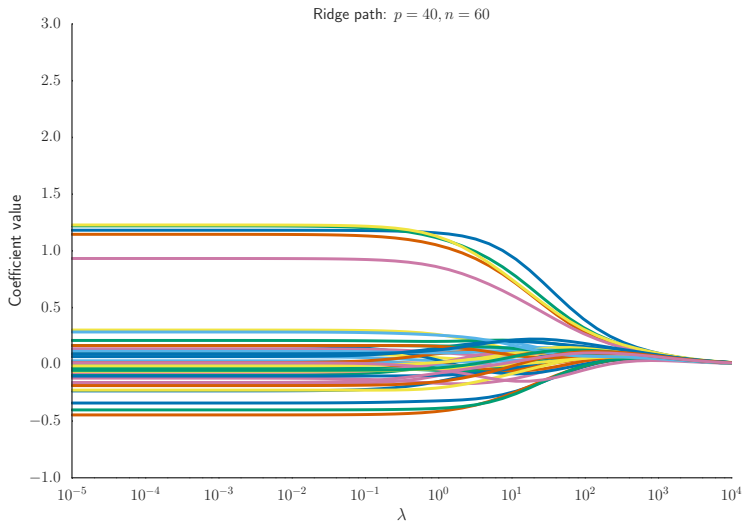
Exo: Use sub-gradients to prove the previous result

Numerical example on simulated data

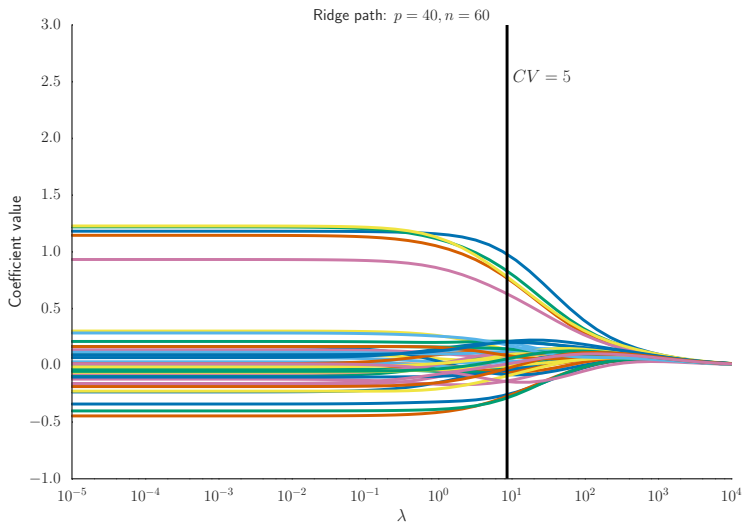
- ▶ $\theta^* = (1, 1, 1, 1, 1, 0, \dots, 0) \in \mathbb{R}^p$ (5 non-zero coefficients)
- ▶ $X \in \mathbb{R}^{n \times p}$ has columns drawn according to a Gaussian distribution
- ▶ $y = X\theta^* + \varepsilon \in \mathbb{R}^n$ with $\varepsilon \sim \mathcal{N}(0, \sigma^2 \text{Id}_n)$
- ▶ We use a grid of 50 λ values

For this example : $n = 60, p = 40, \sigma = 1$

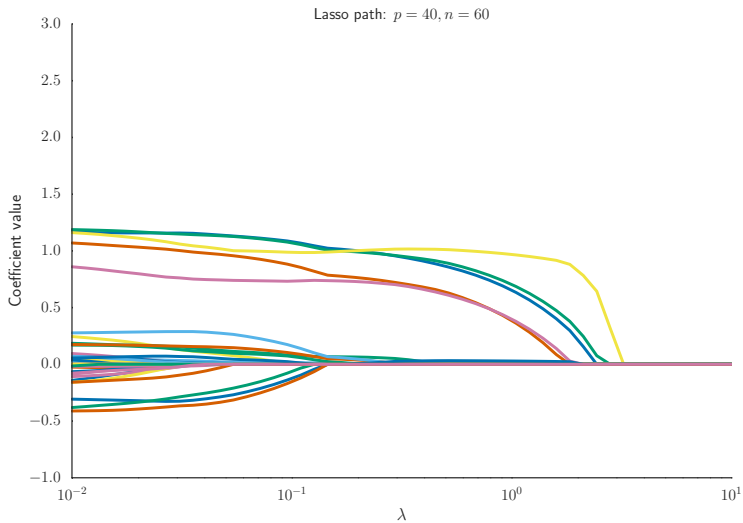
Lasso vs Ridge



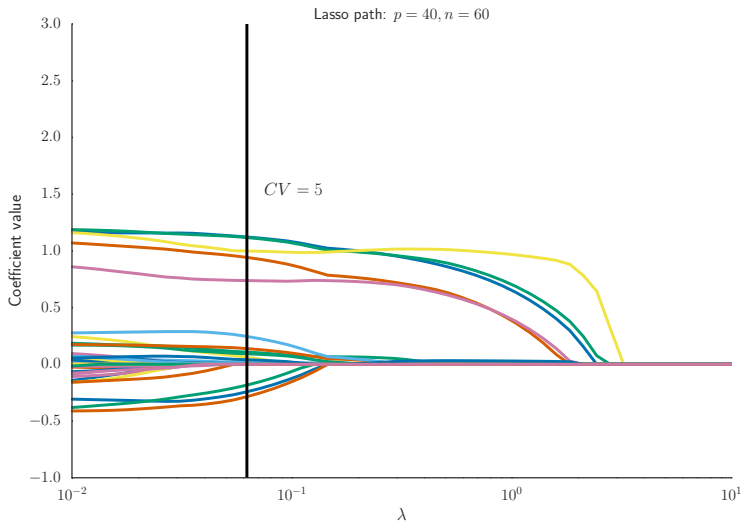
Lasso vs Ridge



Lasso vs Ridge



Lasso vs Ridge



Lasso properties

- ▶ Numerical aspect: the Lasso is a **convex** problem
- ▶ Variable selection / sparse solutions: $\hat{\theta}_{\lambda}^{\text{Lasso}}$ has potentially many zeroed coefficients. The λ parameter controls the sparsity level: if λ is large, solutions are very sparse.

Exemple : We got 17 non-zero coefficients for LassoCV in the previous simulated example

Rem: RidgeCV has no zero coefficients

Lasso analysis

Theory is more involved for the Lasso than for least squares or Ridge, it can be found in recent references (cf. Bühlmann and van de Geer (2011) for an overview)

In a nutshell: we add bias to the standard least square to reduce the variance

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure

- Stabilisation

- Least squares / Lasso extensions

The Lasso bias

The Lasso is biased: it shrinks large coefficients towards 0

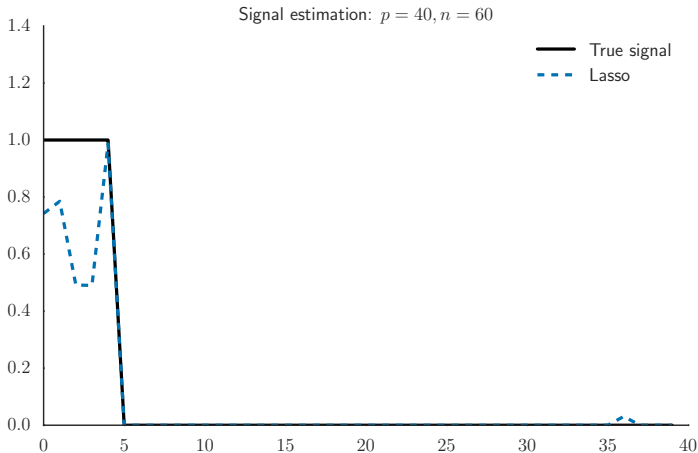


Illustration over the previous example

The Lasso bias

The Lasso is biased: it shrinks large coefficients towards 0

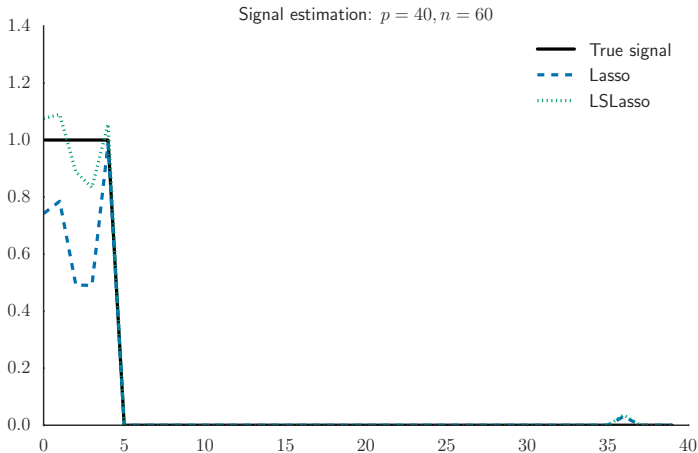


Illustration over the previous example

The Lasso bias: a simple remedy

As large coefficients are often shrunk towards zero, it is possible to use a simple two-stage procedure

LSLasso (Least Square Lasso)

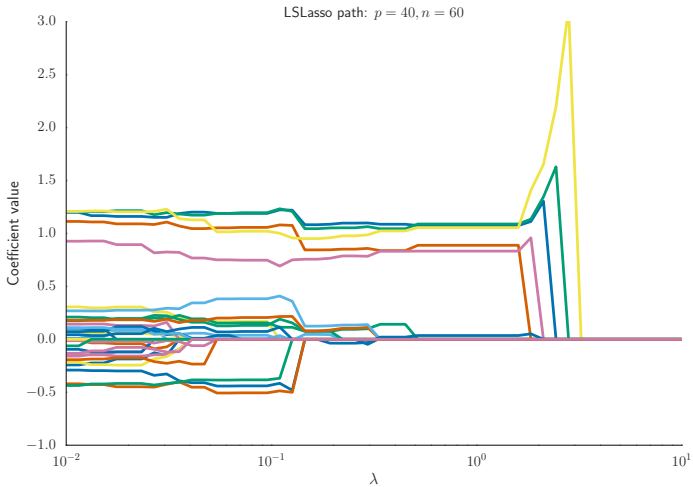
1. Lasso : get $\hat{\theta}_{\lambda}^{\text{Lasso}}$
2. Perform least squares over selected variables: $\text{supp}(\hat{\theta}_{\lambda}^{\text{Lasso}})$

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

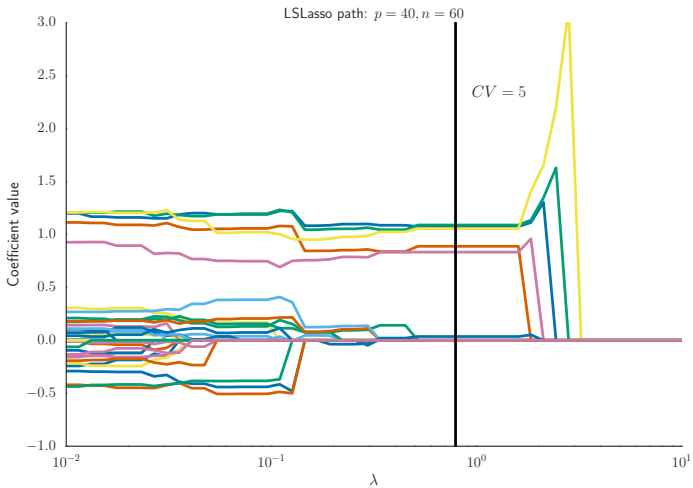
Rem: CV needs to be done over the whole procedure; choosing the Lasso λ by CV and then performing least squares keeps too many variables

Rem: LSLasso is not necessarily coded in standard packages...

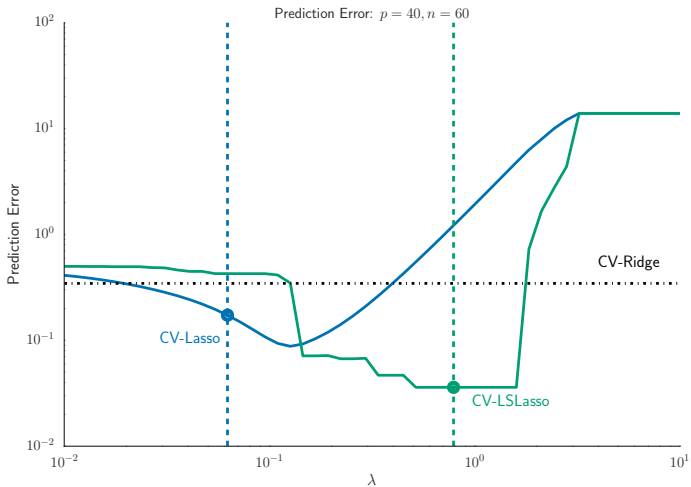
De-biasing



De-biasing



Prediction: Lasso vs. LSLasso



LSLasso evaluation

Pros

- ▶ the “true” large coefficients are less shrunk
- ▶ using CV we recover less “parasite” variables (improve interpretability)
e.g., in the previous example the LSLassoCV recovers exactly the 5 “true” non zero variables, up to a single false positive.

LSLasso: especially useful for estimation

Cons

- ▶ the difference in term of prediction is not always obvious
- ▶ requires more computation: needs to compute as many least squares than λ parameters (though with smaller dimension, neglecting useless variables)

Elastic-net : ℓ_1/ℓ_2 regularization

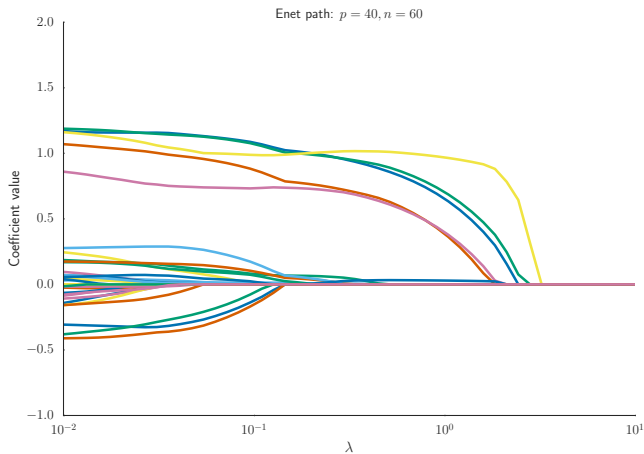
The Elastic-Net, introduced by Zou and Hastie (2005) is the (unique) solution of

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

Rem: it has two parameters one for the global regularization level, one trading-off Ridge vs. Lasso

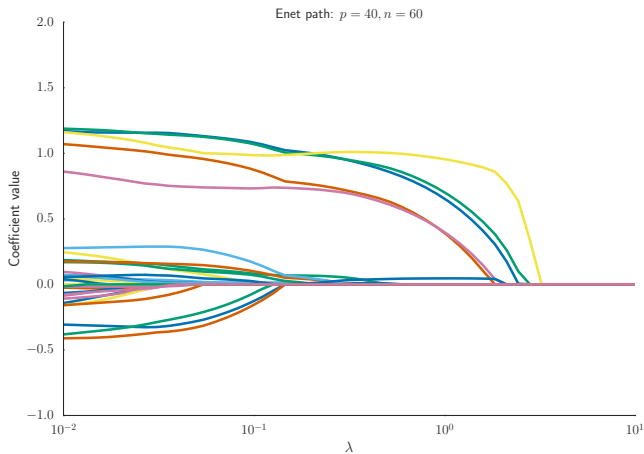
Rem: the solution is unique the size of the Elastic-Net support is smaller than $\min(n, p)$

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



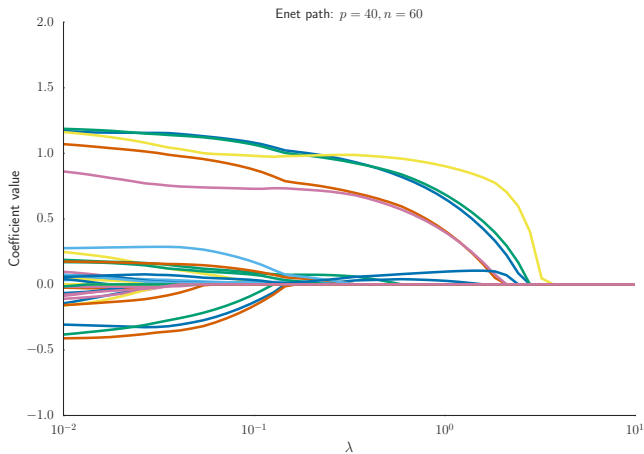
$$\gamma = 1.00$$

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



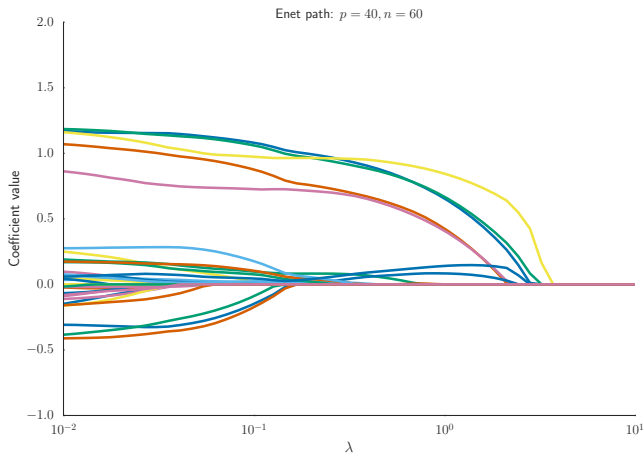
$$\gamma = 0.99$$

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



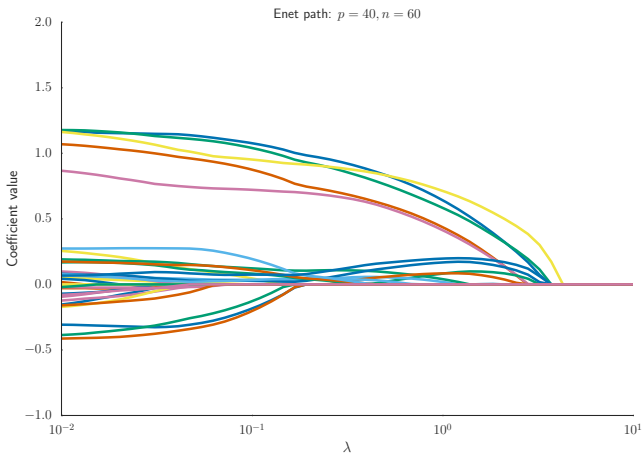
$$\gamma = 0.95$$

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



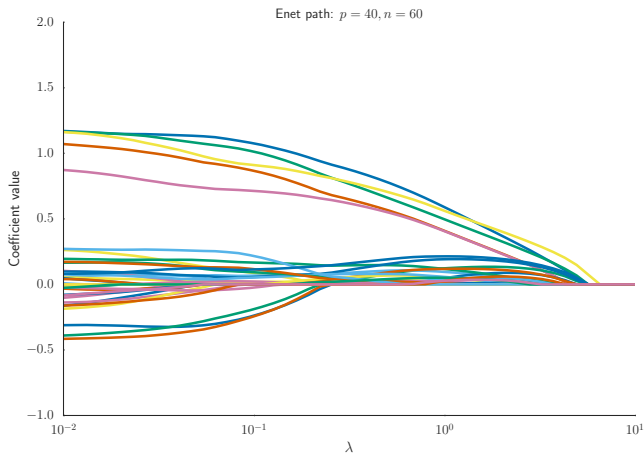
$\gamma = 0.90$

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



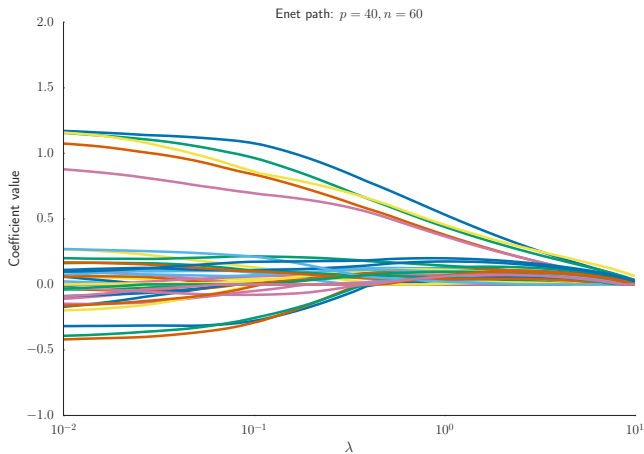
$$\gamma = 0.75$$

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



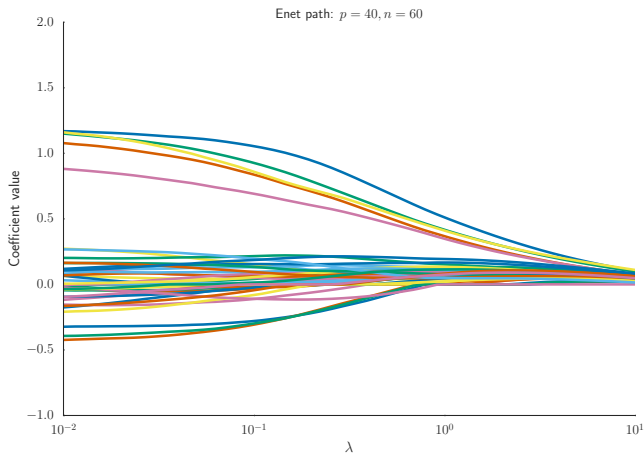
$$\gamma = 0.50$$

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



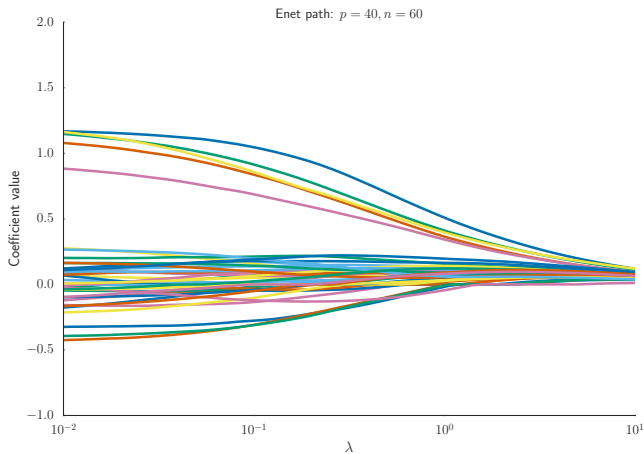
$$\gamma = 0.25$$

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



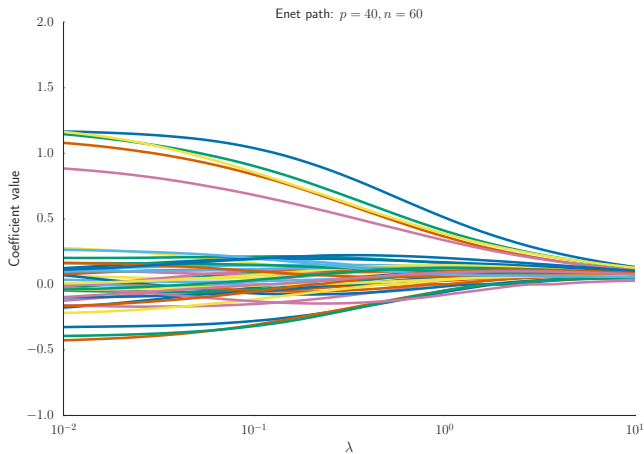
$$\gamma = 0.1$$

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



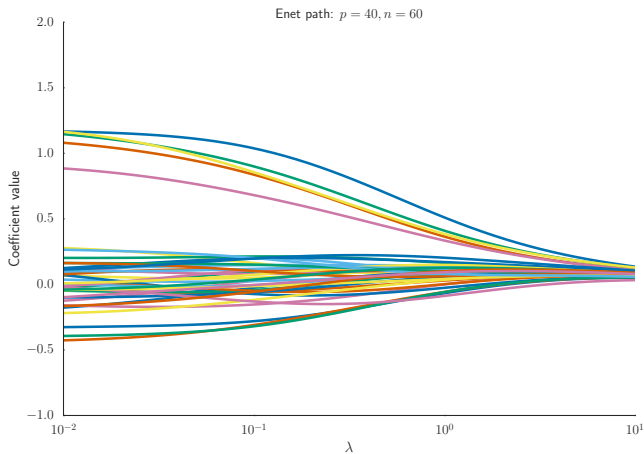
$$\gamma = 0.05$$

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



$$\gamma = 0.01$$

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



$$\gamma = 0.00$$

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure

- Stabilisation

- Least squares / Lasso extensions

Non-convex penalties

Use a (smooth) penalty approximating better $\|\cdot\|_0$, choosing a non-convex $t \rightarrow \text{pen}_{\lambda,\gamma}(t)$

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

Non-convex penalties

Use a (smooth) penalty approximating better $\|\cdot\|_0$, choosing a non-convex $t \rightarrow \text{pen}_{\lambda,\gamma}(t)$

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

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

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

Non-convex penalties

Use a (smooth) penalty approximating better $\|\cdot\|_0$, choosing a non-convex $t \rightarrow \text{pen}_{\lambda,\gamma}(t)$

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

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

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

Non-convex penalties

Use a (smooth) penalty approximating better $\|\cdot\|_0$, choosing a non-convex $t \rightarrow \text{pen}_{\lambda,\gamma}(t)$

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

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

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

Non-convex penalties

Use a (smooth) penalty approximating better $\|\cdot\|_0$, choosing a non-convex $t \rightarrow \text{pen}_{\lambda,\gamma}(t)$

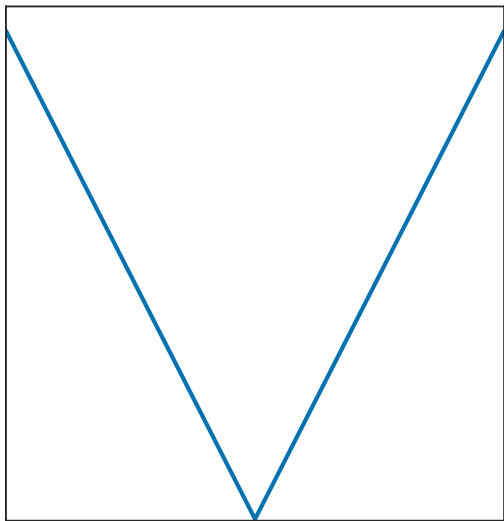
$$\hat{\boldsymbol{\theta}}_{\lambda,\gamma}^{\text{pen}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \left(\underbrace{\frac{1}{2} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}_{\text{data fitting}} + \underbrace{\sum_{j=1}^p \text{pen}_{\lambda,\gamma}(|\theta_j|)}_{\text{regularization}} \right)$$

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

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

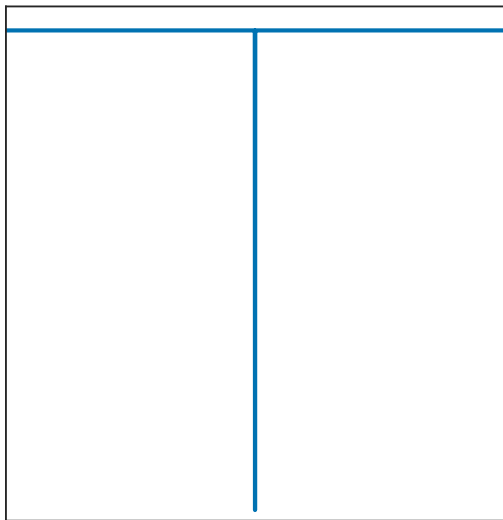
Rem: algorithmic difficulties (stopping, local minimum, etc.) and theoretical

Standard non-convex penalties



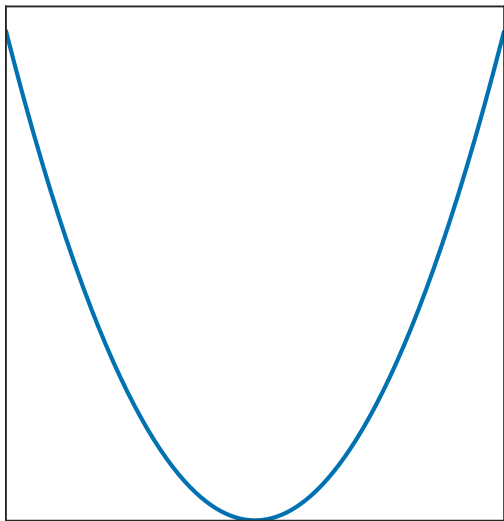
l_1

Standard non-convex penalties



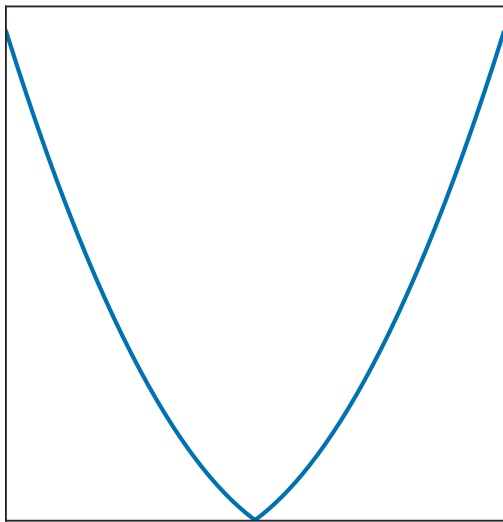
10

Standard non-convex penalties



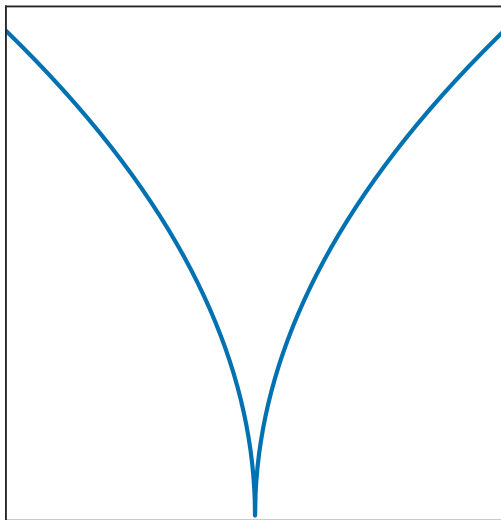
l22

Standard non-convex penalties



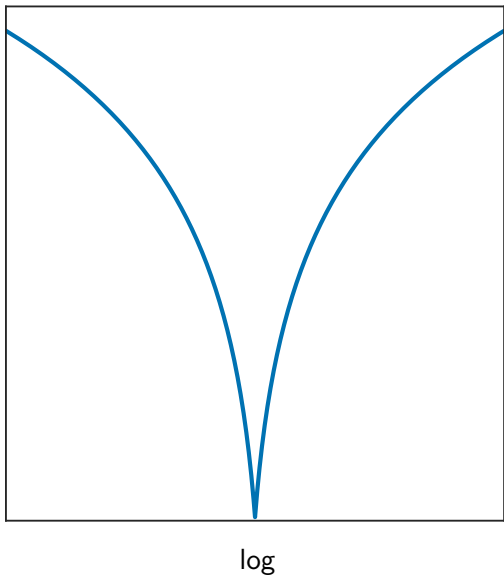
enet

Standard non-convex penalties

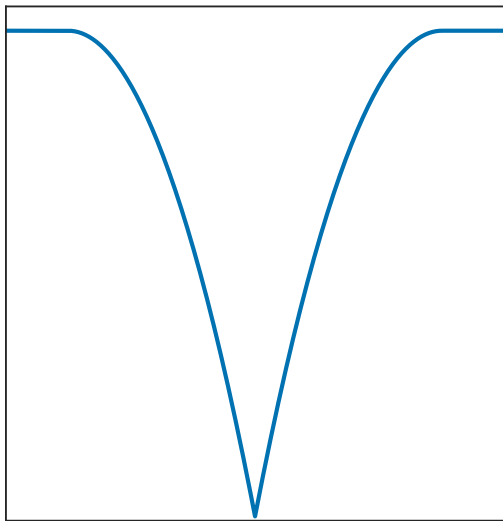


sqrt

Standard non-convex penalties

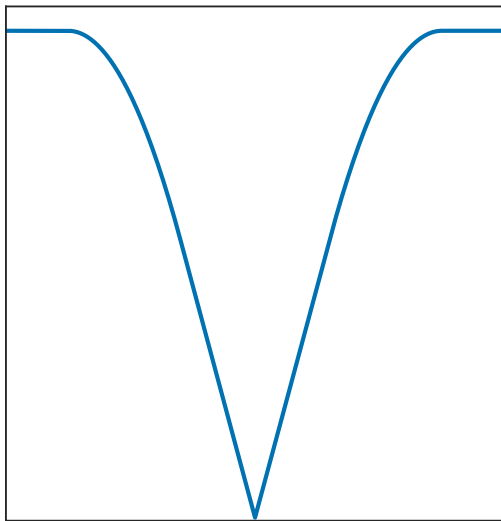


Standard non-convex penalties



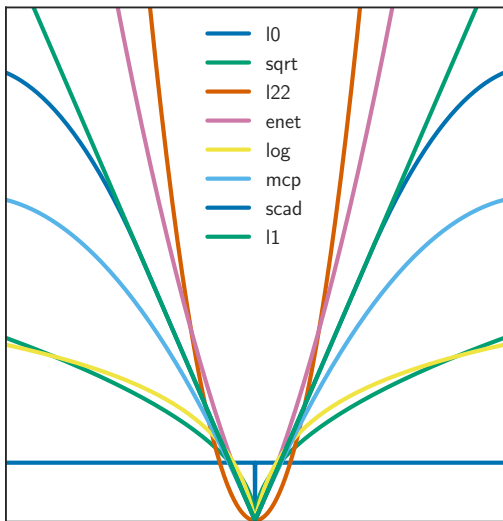
mcp

Standard non-convex penalties



scad

Standard non-convex penalties



Adaptive-Lasso

Several names for the same idea:

- ▶ Adaptive-Lasso Zou (2006)
- ▶ re-weighted ℓ_1 Candès *et al.* (2008)
- ▶ DC-programming approach (for *Difference of Convex Programming*) Gasso *et al.* (2008)

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

for $k = 1, \dots, K$ **do**

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

for $k = 1, \dots, K$ **do**

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

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

for $k = 1, \dots, K$ **do**

$$\begin{aligned} \hat{\boldsymbol{\theta}} &\leftarrow \arg \min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^p \hat{w}_j |\theta_j| \right) \\ \hat{w}_j &\leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}}, \forall j \in \llbracket 1, p \rrbracket \end{aligned}$$

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

for $k = 1, \dots, K$ **do**

$$\begin{aligned} \hat{\boldsymbol{\theta}} &\leftarrow \arg \min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^p \hat{w}_j |\theta_j| \right) \\ \hat{w}_j &\leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}}, \forall j \in \llbracket 1, p \rrbracket \end{aligned}$$

Rem: in practice few iterations need (about 5/10)

Adaptive-Lasso

Exemple : take $\text{pen}_{\lambda,\gamma}(t) = \lambda|t|^q$ with $q = 1/2$

Algorithm: Adaptive Lasso ($q = 1/2$ case)

Input : X, \mathbf{y} , maximum number of iterations K , λ (regularization)

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

for $k = 1, \dots, K$ **do**

$$\begin{aligned} \hat{\boldsymbol{\theta}} &\leftarrow \arg \min_{\boldsymbol{\theta}} \left(\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2} + \lambda \sum_{j=1}^p \hat{w}_j |\theta_j| \right) \\ \hat{w}_j &\leftarrow \frac{1}{|\hat{\theta}_j|^{\frac{1}{2}}}, \forall j \in \llbracket 1, p \rrbracket \end{aligned}$$

Rem: in practice few iterations need (about 5/10)

Rem: re-use a Lasso solver to update $\hat{\boldsymbol{\theta}}$

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure**

- Stabilisation

- Least squares / Lasso extensions

Support structure

Here we suppose that we have a known group structure on the variables (prior the experiment) : $\llbracket 1, p \rrbracket = \bigcup_{g \in \mathcal{G}} g$

Vector and active coordinate (in orange):



Sparse support: any

Possible penalties: Lasso

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

Support structure

Here we suppose that we have a known group structure on the variables (prior the experiment) : $\llbracket 1, p \rrbracket = \bigcup_{g \in \mathcal{G}} g$

Vector and active coordinate (in orange):



Sparse support: group

Possible penalties: Group-Lasso

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

Support structure

Here we suppose that we have a known group structure on the variables (prior the experiment) : $\llbracket 1, p \rrbracket = \bigcup_{g \in \mathcal{G}} g$

Vector and active coordinate (in orange):



Sparse support: group + sub-groups

Possible penalties: Sparse-Group-Lasso

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

Group-Lasso

The ℓ_1 norm penalty ensures that few coefficients are active, but no other structure is enforced

One can aim at:

- ▶ group/block wise sparsity: Group-Lasso [Yuan and Lin \(2006\)](#)
- ▶ individual and group wise : Sparse Group-Lasso [Simon, Friedman, Hastie and Tibshirani \(2012\)](#)
- ▶ hierarchical structures (for instance with higher order interactions) [Bien, Taylor and Tibshirani \(2013\)](#)
- ▶ graph structures, gradients structures, etc.

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

- Support structure

Stabilisation

- Least squares / Lasso extensions

Lasso stability

The Lasso can be **instable**: when non-unique solutions (e.g., when $p > n$) depending on the numerical solver and the required precision, there might be errors in the variable selection process. One can limit such drawbacks using re-sampling techniques:

- ▶ Bolasso [Bach \(2008\)](#)
- ▶ Stability Selection [Meinshausen and Bühlmann \(2010\)](#)

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, y , replications number B , λ regularization

Exo: code the Bolasso in Python using sklearn

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, \mathbf{y} , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

|

Exo: code the Bolasso in Python using `sklearn`

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, y , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

Exo: code the Bolasso in Python using sklearn

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, y , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

 Compute the Lasso for this sample: $\hat{\theta}_{\lambda}^{\text{Lasso},(k)}$

Exo: code the Bolasso in Python using `sklearn`

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, y , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

 Compute the Lasso for this sample: $\hat{\theta}_{\lambda}^{\text{Lasso},(k)}$

 Compute the associated support: $S_k = \text{supp} \left(\hat{\theta}_{\lambda}^{\text{Lasso},(k)} \right)$

Exo: code the Bolasso in Python using `sklearn`

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, y , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

 Compute the Lasso for this sample: $\hat{\theta}_{\lambda}^{\text{Lasso},(k)}$

 Compute the associated support: $S_k = \text{supp} \left(\hat{\theta}_{\lambda}^{\text{Lasso},(k)} \right)$

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

Exo: code the Bolasso in Python using sklearn

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, \mathbf{y} , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

 Compute the Lasso for this sample: $\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso},(k)}$

 Compute the associated support: $S_k = \text{supp} \left(\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso},(k)} \right)$

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

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

Exo: code the Bolasso in Python using sklearn

Bolasso Bach (2008)

Algorithm: Bootstrap Lasso

Input : X, \mathbf{y} , replications number B , λ regularization

for $k = 1, \dots, B$ **do**

 Draw a *bootstrap* sample: $X^{(k)}, y^{(k)}$

 Compute the Lasso for this sample: $\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso},(k)}$

 Compute the associated support: $S_k = \text{supp} \left(\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Lasso},(k)} \right)$

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

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

Output: support S , and a vector $\hat{\boldsymbol{\theta}}_{\lambda}^{\text{Bolasso}}$

Exo: code the Bolasso in Python using `sklearn`

Syllabus

Reminders

Variable selection and sparsity

- The ℓ_0 penalty and its limits

- The ℓ_1 penalty

- Sub-gradient / sub-differential

Improvement and extensions for the Lasso

- LSLasso / Elastic-Net

- Non-convex penalties / Adaptive Lasso

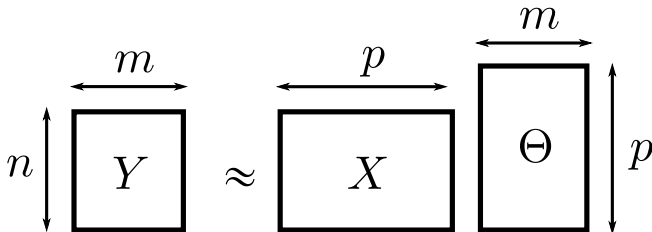
- Support structure

- Stabilisation

- Least squares / Lasso extensions

Multi-task regression

One aims at jointly solving m linear regression: $Y \approx X\Theta$



with

- ▶ $Y \in \mathbb{R}^{n \times m}$: observation matrix
- ▶ $X \in \mathbb{R}^{n \times p}$: design matrix (known)
- ▶ $\Theta \in \mathbb{R}^{p \times m}$: coefficient matrix (unknown)

Example : several observed signals through time (e.g., several captors for the same phenomenon)

Rem: cf. MultiTaskLasso in sklearn for a solver

Multi-task and regularization

In multi-task settings penalties can also be helpful:

$$\hat{\Theta}_{\lambda} = \arg \min_{\Theta \in \mathbb{R}^{p \times m}} \left(\underbrace{\frac{1}{2} \|Y - X\Theta\|_F^2}_{\text{data fitting}} + \underbrace{\lambda \Omega(\Theta)}_{\text{regularization}} \right)$$

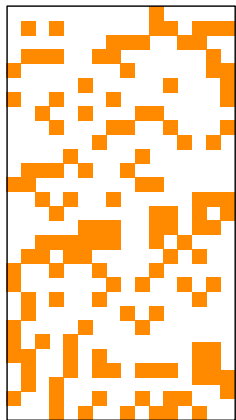
where Ω is a penalty / regularization

Rem: the Frobenius norm $\|\cdot\|_F$ is defined for any matrix $A \in \mathbb{R}^{n_1 \times n_2}$ by

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

Multi-tasks penalties

Vectorial penalties need to be adapted:



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

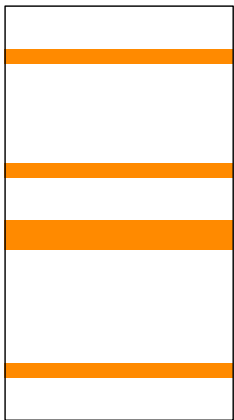
Sparse support:
any

Penalty: Lasso

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

Multi-tasks penalties

Vectorial penalties need to be adapted:



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

Sparse support:
group

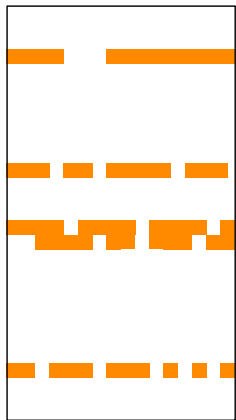
Penalty: Group-Lasso

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

where $\Theta_{j,:}$: the j -th line of Θ

Multi-tasks penalties

Vectorial penalties need to be adapted:



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

Sparse support:
group + sub-groups

Penalty: Sparse-Group-Lasso

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

References I

- ▶ F. Bach.
Bolasso: model consistent Lasso estimation through the bootstrap.
In *ICML*, 2008.
- ▶ 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.
- ▶ J. Fan and R. Li.
Variable selection via nonconcave penalized likelihood and its oracle properties.
J. Amer. Statist. Assoc., 96(456):1348–1360, 2001.

References II

- ▶ G. Gasso, A. Rakotomamonjy, and S. Canu.
Recovering sparse signals with non-convex penalties and DC programming.
IEEE Trans. Sig. Process., 57(12):4686–4698, 2009.
- ▶ 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.

References III

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