

ENSAI 3A – spécialité Sciences du vivant

Introduction aux méthodes régularisées et parcimonieuses

<http://julien.cremeriefamily.info>

-  Trevor Hastie, Robert Tibshirani, Jerome Friedman, 2009.
The Elements of Statistical Learning (2nd Edition).
-  Christophe Giraud,
High-dimensional Statistics, 2013.
-  J.C., manuscript d'habilitation
Contributions to sparse methods for complex data analysis

Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Plan

Prérequis

Dérivée par rapport à un vecteur

Vecteur aléatoire Gaussien

Projection orthogonale

Régression linéaire et OLS

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Plan

Prérequis

Dérivée par rapport à un vecteur

Vecteur aléatoire Gaussien

Projection orthogonale

Régression linéaire et OLS

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Gradient

Définition (gradient)

Soit f une application de \mathbb{R}^p dans \mathbb{R} . On appelle gradient de f le vecteur des dérivées partielles

$$\nabla f(\mathbf{x}) = \frac{\partial}{\partial \mathbf{x}} f(\mathbf{x}) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_p} \right)^T.$$

De cette définition, on déduit en particulier la dérivée par rapport à un vecteur d'une forme linéaire, d'une application linéaire et d'une forme quadratique.

Dérivée par rapport à un vecteur

Proposition (dérivée par rapport à un vecteur)

Soit $\mathbf{u}, \mathbf{x} \in \mathbb{R}^p$ et $\mathbf{A} \in \mathcal{M}_{mp}$ et $\mathbf{S} \in \mathcal{M}_{pp}$.

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{u}^\top \mathbf{x} = \mathbf{u}$$

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{A} \mathbf{x} = \mathbf{A}$$

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^\top \mathbf{S} \mathbf{x} = \mathbf{S} \mathbf{x} + \mathbf{S}^\top \mathbf{x}$$

Si de plus \mathbf{S} est symétrique, alors

$$\frac{\partial}{\partial \mathbf{x}} \mathbf{x}^\top \mathbf{S} \mathbf{x} = 2\mathbf{S}\mathbf{x}$$

Plan

Prérequis

Dérivée par rapport à un vecteur

Vecteur aléatoire Gaussien

Projection orthogonale

Régression linéaire et OLS

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Vecteur aléatoire, espérance et variance-covariance

Soit $X = (X_1, \dots, X_p)^\top$ un vecteur de variables aléatoires dont la distribution est définie par la densité jointe $f(\mathbf{x}) = f(x_1, \dots, x_p)$.

Définition (Espérance)

L'espérance de X est le vecteur d'espérance de chaque composant:

$$\mathbb{E}X = (\mathbb{E}(X_1), \dots, \mathbb{E}(X_p))^\top.$$

Définition (Variance)

La variance de X est la matrice (de variance-covariance) définie par

$$\mathbb{V}(X) = \mathbb{E}[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top]$$

Propriétés

Soit \mathbf{A} une matrice $m \times p$ de constantes, alors

$$\mathbb{E}(\mathbf{A}X) = \mathbf{A}\mathbb{E}(X), \quad \mathbb{V}(\mathbf{A}X) = \mathbf{A}\mathbb{V}(X)\mathbf{A}^\top$$

Vecteur aléatoire, espérance et variance-covariance

Soit $X = (X_1, \dots, X_p)^\top$ un vecteur de variables aléatoires dont la distribution est définie par la densité jointe $f(\mathbf{x}) = f(x_1, \dots, x_p)$.

Définition (Espérance)

L'espérance de X est le vecteur d'espérance de chaque composant:

$$\mathbb{E}X = (\mathbb{E}(X_1), \dots, \mathbb{E}(X_p))^\top.$$

Définition (Variance)

La variance de X est la matrice (de variance-covariance) définie par

$$\mathbb{V}(X) = \begin{pmatrix} \mathbb{V}(X_1) & \dots & \text{cov}(X_1, X_j) & \dots & \text{cov}(X_1, X_p) \\ \vdots & & \vdots & & \vdots \\ \text{cov}(X_1, X_j) & \dots & \mathbb{V}(X_j) & \dots & \text{cov}(X_j, X_p) \\ \vdots & & \vdots & & \vdots \\ \text{cov}(X_1, X_p) & \dots & \text{cov}(X_j, X_p) & \dots & \mathbb{V}(X_p) \end{pmatrix}$$

Vecteur aléatoire, espérance et variance-covariance

Soit $X = (X_1, \dots, X_p)^\top$ un vecteur de variables aléatoires dont la distribution est définie par la densité jointe $f(\mathbf{x}) = f(x_1, \dots, x_p)$.

Définition (Espérance)

L'espérance de X est le vecteur d'espérance de chaque composant:

$$\mathbb{E}X = (\mathbb{E}(X_1), \dots, \mathbb{E}(X_p))^\top.$$

Définition (Variance)

La variance de X est la matrice (de variance-covariance) définie par

$$\mathbb{V}(X) = \mathbb{E}[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top]$$

Propriétés

Soit \mathbf{A} une matrice $m \times p$ de constantes, alors

$$\mathbb{E}(\mathbf{A}X) = \mathbf{A}\mathbb{E}(X), \quad \mathbb{V}(\mathbf{A}X) = \mathbf{A}\mathbb{V}(X)\mathbf{A}^\top$$

Vecteur gaussien

Définition

Le vecteur $X \in \mathbb{R}^p$ suit une distribution multivariée de moyenne μ et de variance Σ si la fonction densité d'une réalisation de \mathbf{x} est données par

$$f(\mathbf{x}) = (2\pi)^{-p/2} |\Sigma|^{-1/2} \exp \left\{ -\frac{1}{2} (\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu}) \right\}.$$

On note $X \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ un vecteur gaussien de \mathbb{R}^p .

Log-vraisemblance

Soit \mathbf{X} la matrice $n \times p$ dont les lignes, notées \mathbf{x}_i , sont des réalisations indépendantes de X .

$$\log L(\boldsymbol{\mu}, \boldsymbol{\Sigma}; \mathbf{X}) = -\frac{np}{2} \log(2\pi) - \frac{n}{2} \log |\boldsymbol{\Sigma}| - \frac{1}{2} \sum_{i=1}^n (\mathbf{x}_i - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x}_i - \boldsymbol{\mu})$$

Vecteur gaussien

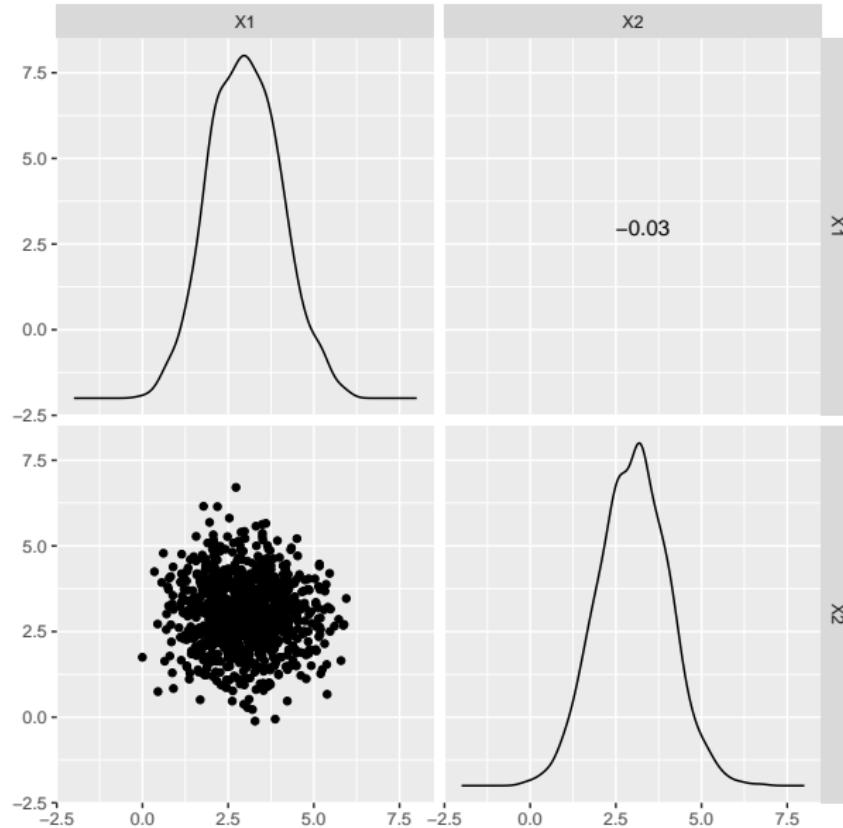
Exemples bivariés

```
library(mvtnorm)
mu <- c(3,3)
Sigma.id  <- matrix(c(1,0,0,1), 2, 2)
Sigma.diag <- matrix(c(.5,0,0,.5), 2, 2)
Sigma.cov1 <- matrix(c(1,0.5,0.5,1), 2, 2)
Sigma.cov2 <- matrix(c(.5,-0.75,-0.75,3), 2, 2)

X.id      <- rmvnorm(1000,mu,Sigma.id)
X.diag    <- rmvnorm(1000,mu,Sigma.diag)
X.cov1   <- rmvnorm(1000,mu,Sigma.cov1)
X.cov2   <- rmvnorm(1000,mu,Sigma.cov2)
```

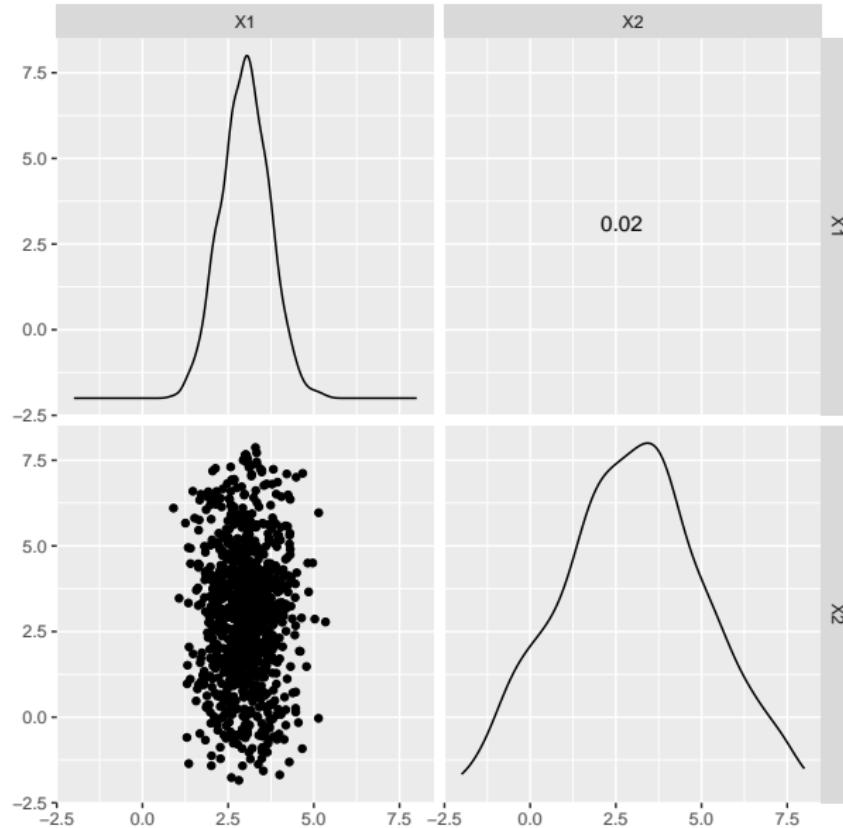
Vecteur gaussien

Exemples bivariés (I)



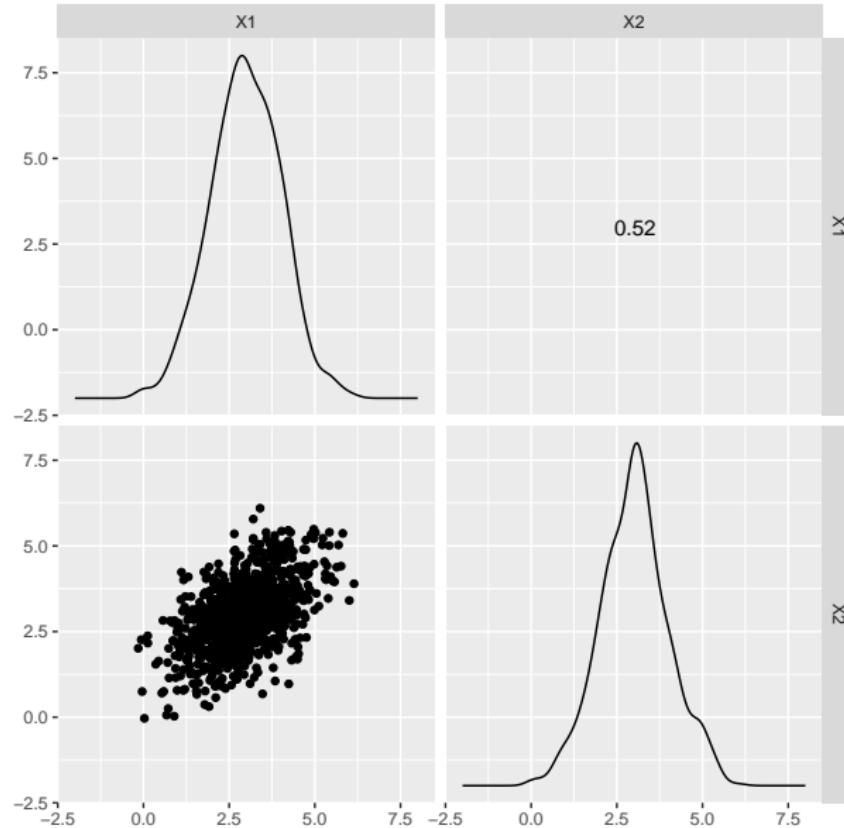
Vecteur gaussien

Exemples bivariés (II)



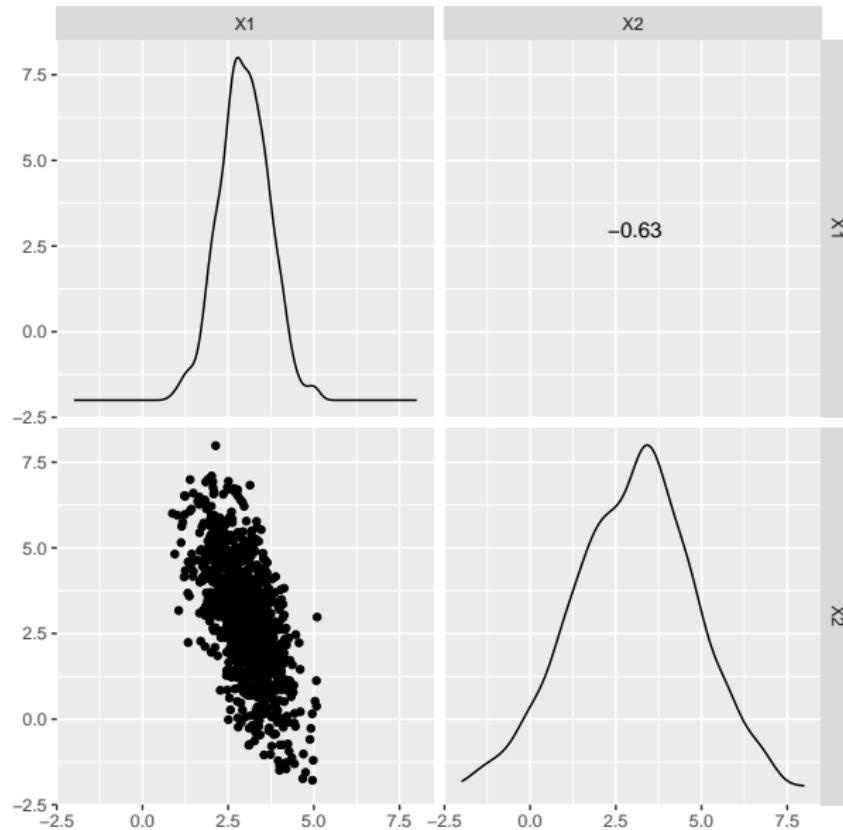
Vecteur gaussien

Exemples bivariés (III)



Vecteur gaussien

Exemples bivariés (IV)



Corrélations partielles et vecteur gaussien (I)

Indépendance conditionnelle: absence de **liens directs** entre variables
 X et Y sont indépendantes conditionnellement à Z (notée $X \perp\!\!\!\perp Y|Z$) ssi

$$\mathbb{P}(X, Y|Z) = \mathbb{P}(X|Z) \times \mathbb{P}(Y|Z).$$

Covariance/corrélation partielle

C'est la covariance/corrélation une fois ôté l'effet d'une autre variable.

$$\text{cov}(X, Y|Z) = \text{cov}(X, Y) - \text{cov}(X, Z)\text{cov}(Y, Z)/\mathbb{V}(Z),$$

$$\rho_{XY|Z} = \frac{\rho_{XY} - \rho_{XZ}\rho_{YZ}}{\sqrt{1 - \rho_{XZ}^2}\sqrt{1 - \rho_{YZ}^2}}.$$

Cas gaussien

Si X, Y, Z sont jointement gaussiens, alors

$$\text{cov}(X, Y|Z) = 0 \Leftrightarrow \text{cor}(X, Y|Z) = 0 \Leftrightarrow X \perp\!\!\!\perp Y|Z.$$

Plan

Prérequis

Dérivée par rapport à un vecteur

Vecteur aléatoire Gaussien

Projection orthogonale

Régression linéaire et OLS

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Sous espaces orthogonaux

Définition (Sous espaces vectoriels orthogonaux)

- ▶ *Les sous espaces V et W sont orthogonaux si tout les vecteurs de V sont orthogonaux à tous les vecteurs de W .*
- ▶ *L'ensemble de tous les vecteurs orthogonaux à V est appelé l'orthogonal de V et est noté V^\perp .*

Théorème

Soit V un sous-espace vectoriel de \mathbb{R}^n , alors tout vecteur de \mathbb{R}^n se décompose de manière unique en une somme de vecteurs de V et de V^\perp .

Projection orthogonale

Définition (Projection orthogonale)

Soit V un sous espace de \mathbb{R}^n , l'application linéaire qui à un vecteur $\mathbf{u} \in \mathbb{R}^n$ fait correspondre un vecteur $\mathbf{u}^* \in V$ tel que $\mathbf{u} - \mathbf{u}^*$ appartienne à V^\perp est appelée projection orthogonale de \mathbf{u} dans V .

Définition (Projecteur orthogonal et matrice)

Soit \mathbf{X} une matrice $n \times p$ de plein rang telle que $n < p$.



Projection orthogonale

Définition (Projection orthogonale)

Soit V un sous espace de \mathbb{R}^n , l'application linéaire qui à un vecteur $\mathbf{u} \in \mathbb{R}^n$ fait correspondre un vecteur $\mathbf{u}^* \in V$ tel que $\mathbf{u} - \mathbf{u}^*$ appartienne à V^\perp est appelée *projection orthogonale de \mathbf{u} dans V* .

Définition (Projecteur orthogonal et matrice)

Soit \mathbf{X} une matrice $n \times p$ de plein rang telle que $n < p$.

- La projection orthogonale de $\mathbf{u} \in \mathbb{R}^n$ dans l'image de \mathbf{X} vaut

$$\text{proj}_{\mathbf{X}}(\mathbf{u}) = \underbrace{\mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top}_{\mathbf{P}_\mathbf{X}} \mathbf{u}.$$

- La projection orthogonale de $\mathbf{u} \in \mathbb{R}^n$ dans le noyau de \mathbf{X} vaut

$$\text{proj}_{\mathbf{X}^\perp}(\mathbf{u}) = \underbrace{\left(\mathbf{I} - \mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \right)}_{\mathbf{I} - \mathbf{P}_\mathbf{X}} \mathbf{u}.$$

Projection orthogonale

Définition (Projection orthogonale)

Soit V un sous espace de \mathbb{R}^n , l'application linéaire qui à un vecteur $\mathbf{u} \in \mathbb{R}^n$ fait correspondre un vecteur $\mathbf{u}^* \in V$ tel que $\mathbf{u} - \mathbf{u}^*$ appartienne à V^\perp est appelée projection orthogonale de \mathbf{u} dans V .

Définition (Projecteur orthogonal et matrice)

Soit \mathbf{X} une matrice $n \times p$ de plein rang telle que $n < p$.

- La projection orthogonale de $\mathbf{u} \in \mathbb{R}^n$ dans l'image de \mathbf{X} vaut

$$\text{proj}_{\mathbf{X}}(\mathbf{u}) = \underbrace{\mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top}_{\mathbf{P}_{\mathbf{X}}} \mathbf{u}.$$

- La projection orthogonale de $\mathbf{u} \in \mathbb{R}^n$ dans le noyau de \mathbf{X} vaut

$$\text{proj}_{\mathbf{X}}^\perp(\mathbf{u}) = \underbrace{\left(\mathbf{I} - \mathbf{X} (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \right)}_{\mathbf{I} - \mathbf{P}_{\mathbf{X}}} \mathbf{u}.$$

Plan

Prérequis

Dérivée par rapport à un vecteur

Vecteur aléatoire Gaussien

Projection orthogonale

Régression linéaire et OLS

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Régression linéaire multiple

Le modèle

On suppose que la vraie relation entre Y et x est linéaire:

$$Y = \beta_0 + \sum_{j=1}^p \beta_j X_j + \varepsilon,$$

- ▶ Y la variable aléatoire de réponse,
- ▶ $X = (X_1, \dots, X_p)$ un vecteur tel que X_j peut être
 - ▶ une covariable quantitative ou une transformation d'une covariable
 - ▶ une expansion de bases (polynomiale, haart, etc.),
 - ▶ une interaction entre covariables,
 - ▶ un design/codage associé à une variable qualitative.
- ▶ β_0 est la **constante (intercept)**
- ▶ β_j sont les **coefficients de régression**
- ▶ ε est le **résidu** (variable aléatoire)
 - ~~> erreur de mesure, variabilité individuelle, facteur(s) non expliqué(s)

Régression linéaire multiple

Échantillonnage et écriture matricielle

Collecte de données / échantillonnage aléatoire

Soit $\{(Y_i, x_i)\}_{i=1}^n$ un n -échantillon avec $Y_i \in \mathbb{R}$ et $x_i \in \mathbb{R}^p$. On a

$$Y_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} + \varepsilon_i,$$

avec $\{\varepsilon_i\}_{i=1}^n$ indépendants, identiquement distribués.

Les données

- ▶ $\mathcal{D} = \{i : (y_i, x_i) \in \text{l'ensemble d'entraînement}\},$
- ▶ $\mathcal{T} = \{i : (y_i, x_i) \in \text{l'ensemble de test}\},$
- ▶ $\mathbf{y} = (y_i)_{i \in \mathcal{D}}$, le vecteur de réponse dans $\mathbb{R}^{|\mathcal{D}|}$,
- ▶ $\mathbf{x}_j = (x_{ij})_{i \in \mathcal{D}}$ le vecteur de données pour le j^{e} prédicteur dans $\mathbb{R}^{|\mathcal{D}|}$,
- ▶ \mathbf{X} la matrice $n \times p$ dont la j^{e} ligne est \mathbf{x}_j ,
- ▶ $(\mathbf{y}_{\mathcal{T}}, \mathbf{X}_{\mathcal{T}})$ les données de test.

Régression linéaire multiple

Écriture matricielle

$$Y_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} + \varepsilon_i \quad i = 1, \dots, n$$

$$Y = \mathbf{1}_n \beta_0 + \sum_{j=1}^p \beta_j \mathbf{x}_j + \varepsilon$$

$$Y = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p) \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \varepsilon = \underbrace{\begin{pmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{pmatrix}}_{\mathbf{X}, \text{ une matrice } n \times (p+1)} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

En résumé,

$$Y = \mathbf{X}\boldsymbol{\beta} + \varepsilon.$$

Régression linéaire multiple

Écriture matricielle

$$Y_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} + \varepsilon_i \quad i = 1, \dots, n$$

$$Y = \mathbf{1}_n \beta_0 + \sum_{j=1}^p \beta_j \mathbf{x}_j + \boldsymbol{\varepsilon}$$

$$Y = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p) \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \boldsymbol{\varepsilon} = \underbrace{\begin{pmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{pmatrix}}_{\mathbf{X}, \text{ une matrice } n \times (p+1)} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

En résumé,

$$Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

Régression linéaire multiple

Écriture matricielle

$$Y_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} + \varepsilon_i \quad i = 1, \dots, n$$

$$Y = \mathbf{1}_n \beta_0 + \sum_{j=1}^p \beta_j \mathbf{x}_j + \boldsymbol{\varepsilon}$$

$$Y = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p) \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \boldsymbol{\varepsilon} = \underbrace{\begin{pmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{pmatrix}}_{\mathbf{X}, \text{ une matrice } n \times (p+1)} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

En résumé,

$$Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

Régression linéaire multiple

Écriture matricielle

$$Y_i = \beta_0 + \sum_{j=1}^p \beta_j x_{ij} + \varepsilon_i \quad i = 1, \dots, n$$

$$Y = \mathbf{1}_n \beta_0 + \sum_{j=1}^p \beta_j \mathbf{x}_j + \boldsymbol{\varepsilon}$$

$$Y = (\mathbf{1}_n, \mathbf{x}_1, \dots, \mathbf{x}_p) \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \boldsymbol{\varepsilon} = \underbrace{\begin{pmatrix} 1 & x_{11} & \dots & x_{1p} \\ 1 & x_{21} & \dots & x_{2p} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{n1} & \dots & x_{np} \end{pmatrix}}_{\mathbf{X}, \text{ une matrice } n \times (p+1)} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

En résumé,

$$Y = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}.$$

Estimateur des moindres carrés ordinaires

Géométrie

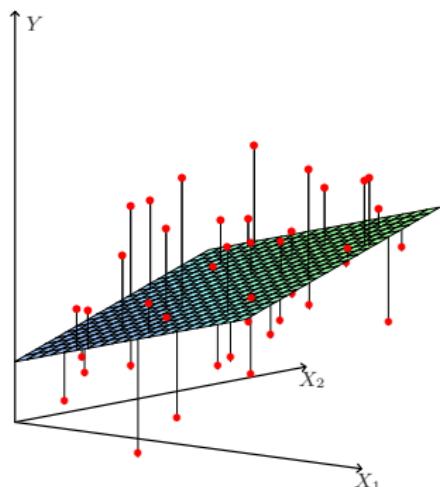


Figure: Dans l'espace des variables \mathbb{R}^{p+1}

Critère

L'estimateur des moindres carrés ordinaires minimise la somme des carrés des résidus (ou l'**erreur d'apprentissage** $\text{err}_{\mathcal{D}}(\boldsymbol{\beta})$.)

$$\begin{aligned}\hat{\boldsymbol{\beta}}^{\text{ols}} &= \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^{p+1}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \\ &= (\mathbf{X}^\top \mathbf{X})^{-1} \mathbf{X}^\top \mathbf{y},\end{aligned}$$

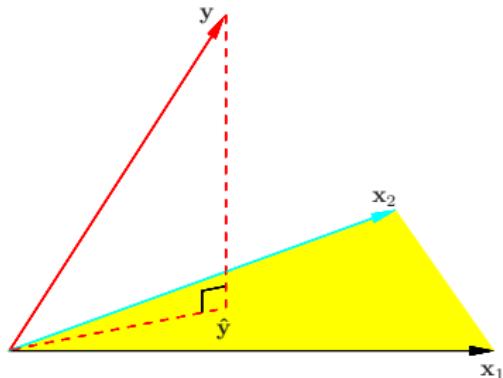
Si \mathbf{X} est de rang plein.

MCO: géométrie dans l'espace des observations

Valeurs ajustées

$$\hat{\mathbf{y}} = \mathbf{X}(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} = \mathbf{H}\mathbf{y},$$

où \mathbf{H} est la matrice de projection ou “matrice chapeau”.



Dans l'espace des colonnes de \mathbf{X} ,

- ▶ les \mathbf{x}_j engendrent un espace n -dimensionnel,
- ▶ $\hat{\mathbf{y}}$ est une combinaison linéaire des colonnes de \mathbf{x}_j ,
- ▶ $\hat{\mathbf{e}} = \mathbf{y} - \hat{\mathbf{y}}$ est orthogonal à ce sous-espace.

Estimation des paramètres

Propriétés des estimateurs

Cas général

$\hat{\beta}$ sont des estimateurs sans biais de β de variance

$$\mathbb{V}(\hat{\beta}) = \sigma^2(\mathbf{X}^\top \mathbf{X})^{-1}.$$

Cas gaussien

Si les résidus sont gaussien, i.e. $\varepsilon \sim \mathcal{N}(0, \sigma^2)$, alors

$$\hat{\beta} \sim \mathcal{N}(\beta, \sigma^2(\mathbf{X}^\top \mathbf{X})^{-1})$$

$$(\hat{\beta} - \beta)^\top \frac{\mathbf{X}^\top \mathbf{X}}{\sigma^2} (\hat{\beta} - \beta) \sim \chi_{p+1}^2$$

$$(n - p - 1)\hat{\sigma}^2 \sim \sigma^2 \chi_{n-p-1}^2$$

Estimation des paramètres

Propriétés des estimateurs (II)

Théorème de Gauss-Markov

- ▶ **Cas gaussien:** $\hat{\beta}^{\text{ols}}$ est le meilleur estimateur sans biais (i.e. de variance minimale).
 - ▶ **Cas non gaussien:** $\hat{\beta}^{\text{ols}}$ est le meilleur estimateur **linéaires** sans biais (i.e. de variance minimale).
- ~~ On dit que $\hat{\beta}^{\text{ols}}$ est le **BLUE** (best linear unbiased estimator)

Plan

Prérequis

Motivations: les limites du modèle linéaire

Qualité d'un modèle de régression

Colinéarité entre prédicteurs et OLS

Illustration: cancer de la prostate

High-dimensional setup

Sélection de variables

Régularisation

Plan

Prérequis

Motivations: les limites du modèle linéaire

Qualité d'un modèle de régression

Colinéarité entre prédicteurs et OLS

Illustration: cancer de la prostate

High-dimensional setup

Sélection de variables

Régularisation

Apprentissage statistique

Problème supervisé

1. une variable **réponse**

- ▶ soit quantitative (taille d'une tumeur, temps de survie, etc.)
- ▶ ou nominale (sous-type de cancer, degré d'avancement, etc.)

2. un ensemble de **prédicteurs**

- ▶ mesures cliniques (niveau d'expression,)
- ▶ âge, fumeur/non fumeur, taille, poids, etc.

Stratégie

Pour un ensemble de données d'entraînement, on cherche à

1. proposer un modèle,
2. apprendre ce modèle sur l'ensemble d'entraînement,
3. tester ce modèle sur de nouvelles observations.

~~> **Un bon modèle doit prédire correctement de nouvelles réponses.**

Modèle de régression

On cherche une fonction f qui prédise Y via X .

Proposition

Le modèle $f(X) = \mathbb{E}[Y|X]$ minimise la perte quadratique, c'est-à-dire

$$f(X) = \arg \min_{\varphi} \mathbb{E}[(Y - \varphi(X))^2].$$

\rightsquigarrow La meilleure prédition de Y à tout point $X = x$ en terme d'espérance de l'erreur quadratique est l'espérance conditionnelle.

Cette remarque est à l'origine des modèles de régression

$$Y = f(X) + \varepsilon,$$

où

- ε est un terme d'erreur additif tel que $\mathbb{E}[\varepsilon] = 0$, $\mathbb{V}[\varepsilon] = \sigma^2$,
- $f(x) = \mathbb{E}[Y|X = x]$ est la **fonction de régression**.

Stratégie d'apprentissage

Problème

Les distributions $\mathbb{P}(Y|X)$ et $\mathbb{P}(X)$ sont inconnues donc $\mathbb{E}(Y|X)$, $\text{err}(f(X))$ inaccessibles: il faut les **estimer**.

Stratégie

1. On se donne une famille \mathcal{F} de modèles

Pour la régression linéaire, $\mathcal{F} = \{X^T \beta, \beta \in \mathbb{R}^p\}$.

2. On ajuste $\hat{f} \in \mathcal{F}$ sur des données d'entraînement \mathcal{D}

On calcule l'estimateur des moindre carré $\hat{\beta}^{\text{ols}}$ et $\hat{f} = \hat{Y} = \mathbf{X}\hat{\beta}^{\text{ols}}$

3. On estime l'erreur de prédiction err à l'aide des données test \mathcal{T} .

Par exemple, $\text{err}(\mathbf{X}_{\mathcal{T}}\hat{\beta}^{\text{ols}}) = \frac{1}{n} \|\mathbf{y}_{\mathcal{T}} - \mathbf{X}_{\mathcal{T}}\hat{\beta}_D^{\text{ols}}\|^2$.

Stratégie d'apprentissage

Problème

Les distributions $\mathbb{P}(Y|X)$ et $\mathbb{P}(X)$ sont inconnues donc $\mathbb{E}(Y|X)$, $\text{err}(f(X))$ inaccessibles: il faut les **estimer**.

Stratégie

1. On se donne une famille \mathcal{F} de modèles

Pour la régression linéaire, $\mathcal{F} = \{X^T \beta, \beta \in \mathbb{R}^p\}$.

2. On ajuste $\hat{f} \in \mathcal{F}$ sur des données d'entraînement \mathcal{D}

On calcule l'estimateur des moindre carré $\hat{\beta}^{\text{ols}}$ et $\hat{f} = \hat{Y} = \mathbf{X}\hat{\beta}^{\text{ols}}$

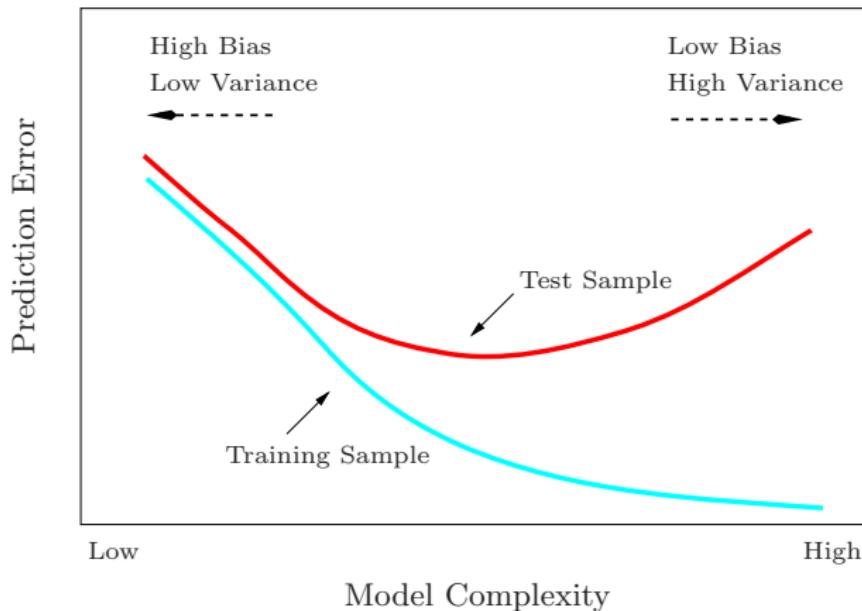
3. On estime l'erreur de prédiction err à l'aide des données test \mathcal{T} .

Par exemple, $\text{err}(\mathbf{X}_{\mathcal{T}} \hat{\beta}^{\text{ols}}) = \frac{1}{n} \left\| \mathbf{y}_{\mathcal{T}} - \mathbf{X}_{\mathcal{T}} \hat{\beta}_{\mathcal{D}}^{\text{ols}} \right\|^2$.

Compromis Biais/Variance

À un nouveau point $X = x$,

$$\text{err}(\hat{f}(x)) = \underbrace{\sigma^2}_{\text{incompressible error}} + \underbrace{\text{bias}^2(\hat{f}(x)) + \mathbb{V}(\hat{f}(x))}_{\text{MSE}(\hat{f}(x))}.$$



Cas de la régression linéaire

Erreur de prédition

On peut montrer pour \mathbf{X} fixé que

$$\hat{\text{err}}(\mathbf{X}\hat{\boldsymbol{\beta}}^{\text{ols}}) = \sigma^2 \frac{(p+1)}{n} + \sigma^2.$$

Théorème de Gauss-Markov

$\hat{Y} = X^\top \hat{\boldsymbol{\beta}}^{\text{ols}}$ est le meilleur modèle (i.e. de plus faible variance) pour les estimateurs sans biais de $\boldsymbol{\beta}$.

~~ Y a-t-il des situations où l'on a intérêt à utiliser un **estimateur biaisé de plus faible variance** ?

Plan

Prérequis

Motivations: les limites du modèle linéaire

Qualité d'un modèle de régression

Colinéarité entre prédicteurs et OLS

Illustration: cancer de la prostate

High-dimensional setup

Sélection de variables

Régularisation

OLS et colinéarité: Gram-Schmidt (I)

Régression par orthogonalisations successives

Algorithme de Gram-Schmidt

s0 Initialisation

$$\mathbf{z}_0 \leftarrow \mathbf{x}_0 (= \mathbf{1}_p);$$

s2 Régression sur une base orthonormale de $\text{vect}(\mathbf{X})$

for $j = 1, \dots, p$ do

 for $k = 1, \dots, j - 1$ do

 Régression de \mathbf{x}_j sur \mathbf{z}_k

$$\gamma_{kj} \leftarrow \frac{\mathbf{z}_k^T \mathbf{x}_j}{\mathbf{z}_k^T \mathbf{z}_k}$$

 Mis à jour des résidus \mathbf{z}_j

$$\mathbf{z}_j \leftarrow \mathbf{x}_j - \sum_{\ell=0}^{j-1} \gamma_{\ell k} \mathbf{z}_{\ell-1}$$

s3 Calcul de l'estimation $\hat{\beta}_p$

$$\hat{\beta}_p \leftarrow \frac{\mathbf{z}_p^T \mathbf{y}}{\mathbf{z}_p^T \mathbf{z}_p}.$$

L'étape 2 peut s'écrire (avec \mathbf{D} diagonale telle que $\mathbf{D}_{jj} = \mathbf{z}_j^T \mathbf{z}_j$)

$$\mathbf{X} = \mathbf{Z}\Gamma = \mathbf{Z}\mathbf{D}^{-1}\mathbf{D}\Gamma = \mathbf{Q}\mathbf{R} = (\mathbf{Q}_1 \quad \mathbf{Q}_2) \begin{pmatrix} \mathbf{R}_1 \\ \mathbf{0} \end{pmatrix} = \mathbf{Q}_1 \mathbf{R}_1,$$

où \mathbf{Q}_1 est orthogonale et \mathbf{R}_1 triangulaire supérieure.

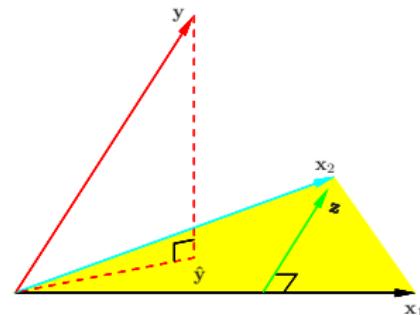


Figure: Exemple avec deux prédicteurs

OLS et colinéarité: Gram-Schmidt (II)

Apportée par la factorisation QR

Estimateur et prédiction en fonction de la factorisation QR

$$\hat{\beta}^{\text{ols}} = \mathbf{R}_1^{-1} \mathbf{Q}_1^T \mathbf{y}, \quad \hat{\mathbf{y}} = \mathbf{Q}_1 \mathbf{Q}_1^T \mathbf{y}.$$

On peut permuter les colonnes de \mathbf{X} dans Gram-Schmidt, ainsi

- ▶ $\hat{\beta}_j$ est la contribution additionnelle de \mathbf{x}_j sur \mathbf{y} une fois que \mathbf{x}_j a été ajusté sur les autres prédicteurs,
- ▶ La variance de $\hat{\beta}_p$ peut s'écrire

$$\mathbb{V}(\hat{\beta}_p) = \frac{\sigma^2}{\|\mathbf{z}_p\|_2^2}.$$

↔ prédicteurs colinéaires ⇒ **mauvaise estimation de β .**

OLS et colinéarité: limite de l'interprétabilité

Supposons que (X, Y) est un vecteur gaussien dans le modèle linéaire

$$Y = X^\top \beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Alors on peut montrer que

$$Y = \sum_{j=1}^p X_j \text{cor}(X_j, Y | X_k, k \neq j) \frac{\sigma}{\sqrt{\mathbb{V}(X_j)}} + \varepsilon.$$

~ β_j est proportionnel à la corrélation partielle entre X_j et Y
i.e. l'effet de X_j sur Y une fois les autres effets ôtés.

$$\text{cov}(\hat{\beta}_i^{\text{ols}}, \hat{\beta}_j^{\text{ols}}) \propto -\text{cor}(X_i, X_j | X_k, k \neq i, j),$$

~ Les prédicteurs fortement liés impliquent des covariances négatives entre les coefficients de régression!

OLS et colinéarité: limite de l'interprétabilité

Supposons que (X, Y) est un vecteur gaussien dans le modèle linéaire

$$Y = X^\top \beta + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2).$$

Alors on peut montrer que

$$Y = \sum_{j=1}^p X_j \text{cor}(X_j, Y | X_k, k \neq j) \frac{\sigma}{\sqrt{\mathbb{V}(X_j)}} + \varepsilon.$$

~ β_j est proportionnel à la corrélation partielle entre X_j et Y
i.e. l'effet de X_j sur Y une fois les autres effets ôtés.

$$\text{cov}(\hat{\beta}_i^{\text{ols}}, \hat{\beta}_j^{\text{ols}}) \propto -\text{cor}(X_i, X_j | X_k, k \neq i, j),$$

~Les prédicteurs fortement liés impliquent des covariances négatives entre les coefficients de régression!

Plan

Prérequis

Motivations: les limites du modèle linéaire

Qualité d'un modèle de régression

Colinéarité entre prédicteurs et OLS

Illustration: cancer de la prostate

High-dimensional setup

Sélection de variables

Régularisation

Exemple: données cancer de la prostate I

97 patients atteints d'un cancer de la prostate

Déterminer les liens entre le niveau d'un antigène spécifique au cancer (y) et diverses mesures cliniques.

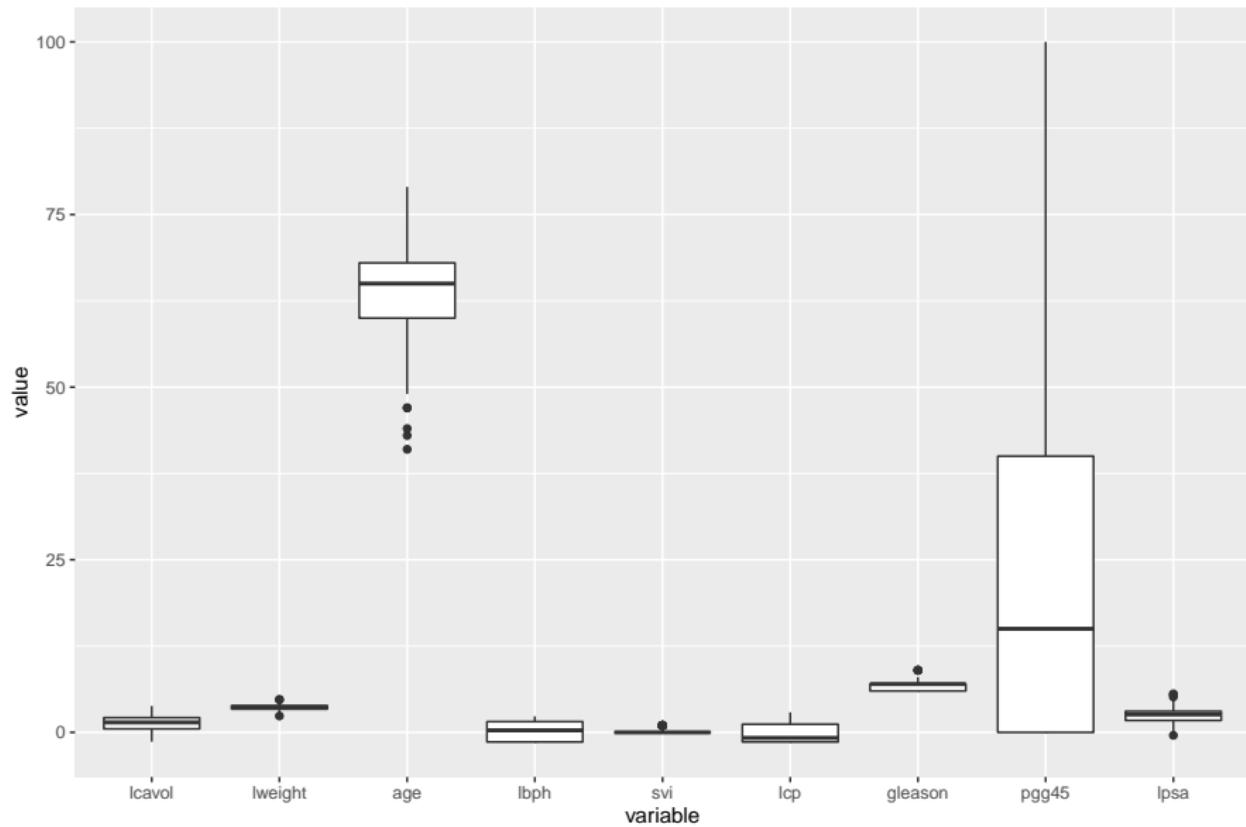
```
load("prostate.rda")
dim(prostate)

## [1] 97 10

print(head(prostate), digits=3)

##   lcavol lweight age  lbph svi    lcp gleason pgg45    lpsa train
## 1 -0.580    2.77  50 -1.39    0 -1.39       6      0 -0.431 TRUE
## 2 -0.994    3.32  58 -1.39    0 -1.39       6      0 -0.163 TRUE
## 3 -0.511    2.69  74 -1.39    0 -1.39       7     20 -0.163 TRUE
## 4 -1.204    3.28  58 -1.39    0 -1.39       6      0 -0.163 TRUE
## 5  0.751    3.43  62 -1.39    0 -1.39       6      0  0.372 TRUE
## 6 -1.050    3.23  50 -1.39    0 -1.39       6      0  0.765 TRUE
```

Exemple: données cancer de la prostate II



Corrélations entre prédicteurs I

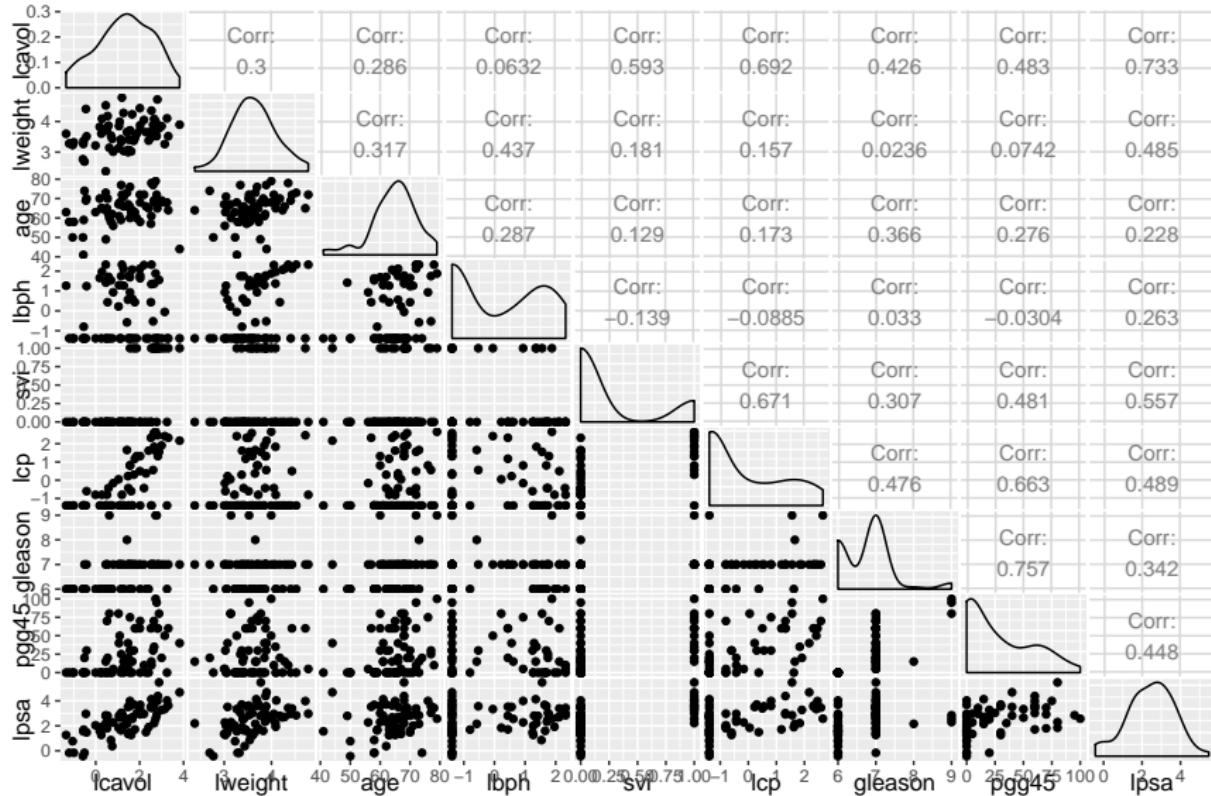
```
print(as.dist(var(prostate[train,1:8])),digits=1)

##          lcavol lweight      age     lbph      svi      lcp gleason
## lweight   0.178
## age       2.669   1.132
## lbph      0.115   0.305   3.155
## svi       0.309   0.036   0.406 -0.086
## lcp        1.205   0.105   1.817 -0.182   0.395
## gleason   0.376   0.008   1.946   0.034   0.091   0.473
## pgg45     17.592  1.036 60.630 -1.304   5.924 27.193  15.725

print(as.dist(cor(prostate[train,1:8])),digits=1)

##          lcavol lweight      age     lbph      svi      lcp gleason
## lweight   0.30
## age       0.29   0.32
## lbph      0.06   0.44   0.29
## svi       0.59   0.18   0.13 -0.14
## lcp        0.69   0.16   0.17 -0.09   0.67
## gleason   0.43   0.02   0.37   0.03   0.31   0.48
## pgg45     0.48   0.07   0.28 -0.03   0.48   0.66   0.76
```

Corrélations entre prédicteurs II



Corrélation entre prédicteurs III

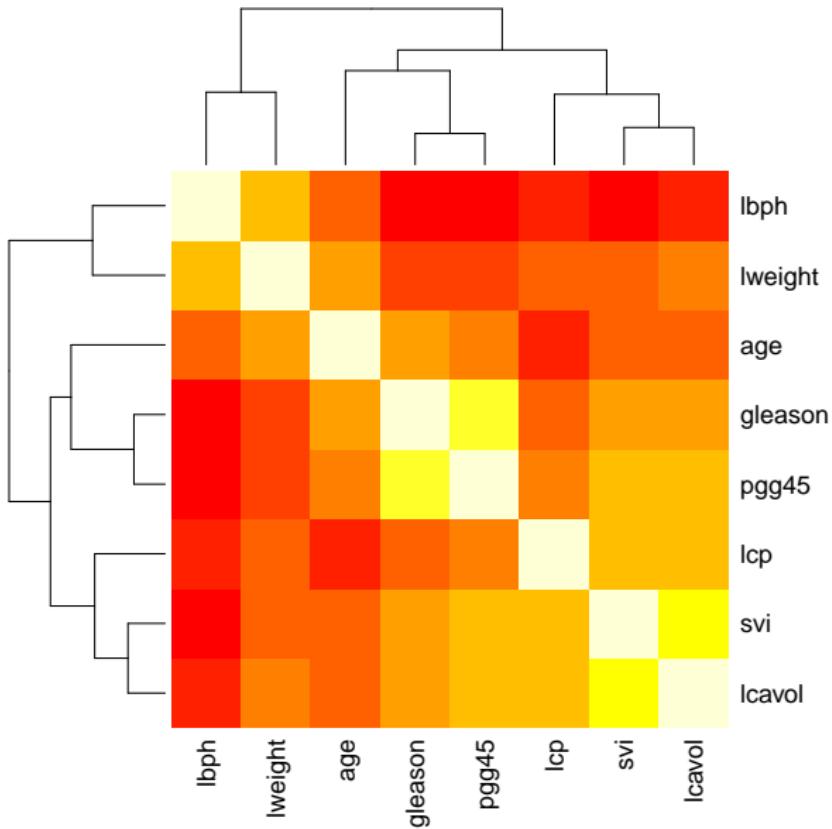


Illustration des limites de l'estimateur OLS I

Pour étudier l'effet des corrélations, on ajuste un modèle avec des prédicteurs de variances comparables (normalisées).

```
prostate.train <- subset(prostate, train==TRUE, -train)
prostate.train[, 1:8] <- scale(prostate.train[, 1:8], FALSE, TRUE)
prostate.test <- subset(prostate, train==FALSE, -train)
prostate.test[, 1:8] <- scale(prostate.test[, 1:8], FALSE, TRUE)
model.full <- lm(lpsa~.,prostate.train)
```

Estimation de l'erreur de prédiction

```
y.hat <- predict(model.full, newdata=prostate.test)
y.test <- prostate.test$lpsa
err.ols <- mean((y.test-y.hat)^2)
print(err.ols)

## [1] 0.5221043
```

Illustration des limites de l'estimateur OLS II

```
summary(model.full)

##
## Call:
## lm(formula = lpsa ~ ., data = prostate.train)
##
## Residuals:
##       Min     1Q Median     3Q    Max 
## -1.64870 -0.34147 -0.05424  0.44941  1.48675 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  0.4292    1.5536   0.276  0.78334    
## lcavol       1.0466    0.1950   5.366 1.47e-06 ***  
## lweight      2.2623    0.8224   2.751  0.00792 **   
## age          -1.2477   0.8938  -1.396  0.16806    
## lbph         0.2123    0.1032   2.056  0.04431 *    
## svi          0.3515    0.1423   2.469  0.01651 *    
## lcp          -0.2924   0.1566  -1.867  0.06697    
## gleason     -0.2012   1.3716  -0.147  0.88389    
## pgg45        0.3737    0.2151   1.738  0.08755    
## ---        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 
##
## Residual standard error: 0.7123 on 58 degrees of freedom
## Multiple R-squared:  0.6944, Adjusted R-squared:  0.6522 
## F-statistic: 16.47 on 8 and 58 DF,  p-value: 2.042e-12
```

Effet de la présence de colinéarité/redondance

Certains coefficients ont une grande variance (e.g. 'pgg45' et 'gleason')

Considération d'ordre statistique

Les variables corrélées ne sont pas bien estimées,

- ▶ Elles portent la même information vis à vis de la réponse.
- ▶ Rappel: $\text{cov}(\hat{\beta}_i, \hat{\beta}_j) \propto -\text{cor}(X_i, X_j | X_k, k \neq i, j)$.

Considération d'ordre numérique

Les variables corrélées induisent un mauvais conditionnement de $\mathbf{X}^T \mathbf{X}$,

- ▶ Rappel: $\mathbb{V}(\hat{\beta}_p^{\text{ols}}) = \frac{\sigma^2}{\|\mathbf{z}_p\|}$ dans la procédure de Gram-Schmidt.
- ▶ l'OLS ne peut pas être calculé en présence de variables redondantes dans \mathbf{X} ou quand $n < p$.

~~~ L'interprétation devient problématique

# Plan

## Prérequis

## Motivations: les limites du modèle linéaire

Qualité d'un modèle de régression

Colinéarité entre prédicteurs et OLS

Illustration: cancer de la prostate

**High-dimensional setup**

## Sélection de variables

## Régularisation

# Definition

## Definition

We are in a high-dimensional setup when the number of observations  $n$  is smaller, and sometimes much smaller, than the number of predictors to be estimated.

Thanks to new technologies, it is now possible to monitor a huge number of features for a single individual, e.g.

## Example

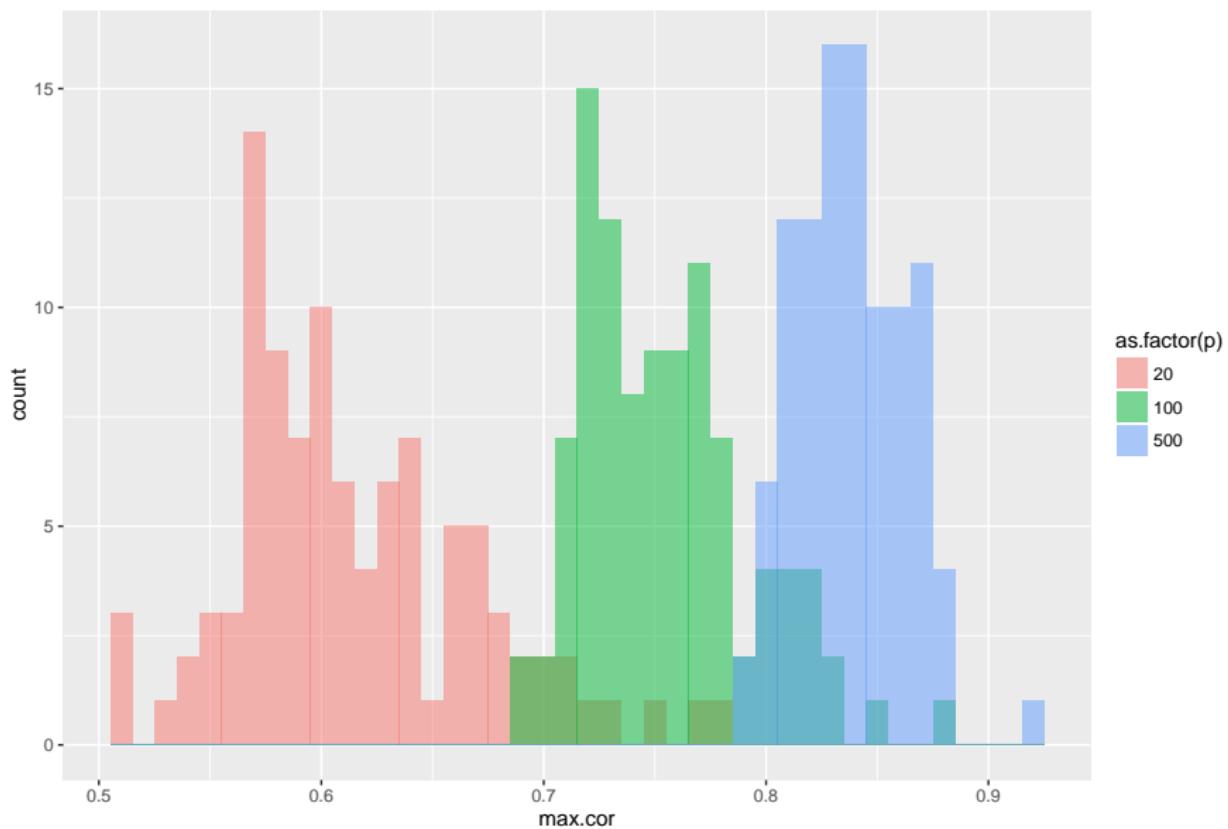
- ▶ Genomic data,
- ▶ Image processing,
- ▶ Networks,
- ▶ What about your data set?

# Consequence I

1. High level of correlation between variables become structural
2. Inverting the empirical correlation matrix is very problematic
  - ▶ The smaller  $n$ , The harder the inversion.
  - ▶ As soon as  $n < p$ ,  $\mathbf{X}^T \mathbf{X}$  is singular.

```
r.maxcor <- function(n,p) {  
  C <- cor(matrix(rnorm(n*p),n,p))  
  return(max(abs(C[upper.tri(C)])))  
}  
nsimu <- 100  
n <- 20  
p20 <- replicate(nsimu,r.maxcor(n,20))  
p100 <- replicate(nsimu,r.maxcor(n,100))  
p500 <- replicate(nsimu,r.maxcor(n,500))  
res <- data.frame(p = rep(c(20,100,500),each=nsimu),  
                   max.cor = c(p20,p100,p500))
```

## Consequence II



# Solutions

## Sélection de variable

Si le vrai modèle sous-jacent ne contient que peu de prédicteurs réellement liés à la variable réponse, on peut vouloir **sélectionner** ceux ayant un grand pouvoir prédictif. On vise ainsi

- ▶ de meilleures performances prédictives,
- ▶ une meilleure interprétabilité du modèle.

## Régularisation

Si tous les prédicteurs ont des effet similaires sur la réponse, la sélection de prédicteurs interprétables est très difficile.

Une solution est de **régulariser** le problème en **constrignant** les paramètres  $\beta$  à vivre dans un espace approprié, de sorte à

- ▶ rendre  $\mathbf{X}^T \mathbf{X}$  inversible,
- ▶ faciliter l'interprétabilité.

# Solutions

## Sélection de variable

Si le vrai modèle sous-jacent ne contient que peu de prédicteurs réellement liés à la variable réponse, on peut vouloir **sélectionner** ceux ayant un grand pouvoir prédictif. On vise ainsi

- ▶ de meilleures performances prédictives,
- ▶ une meilleure interprétabilité du modèle.

## Régularisation

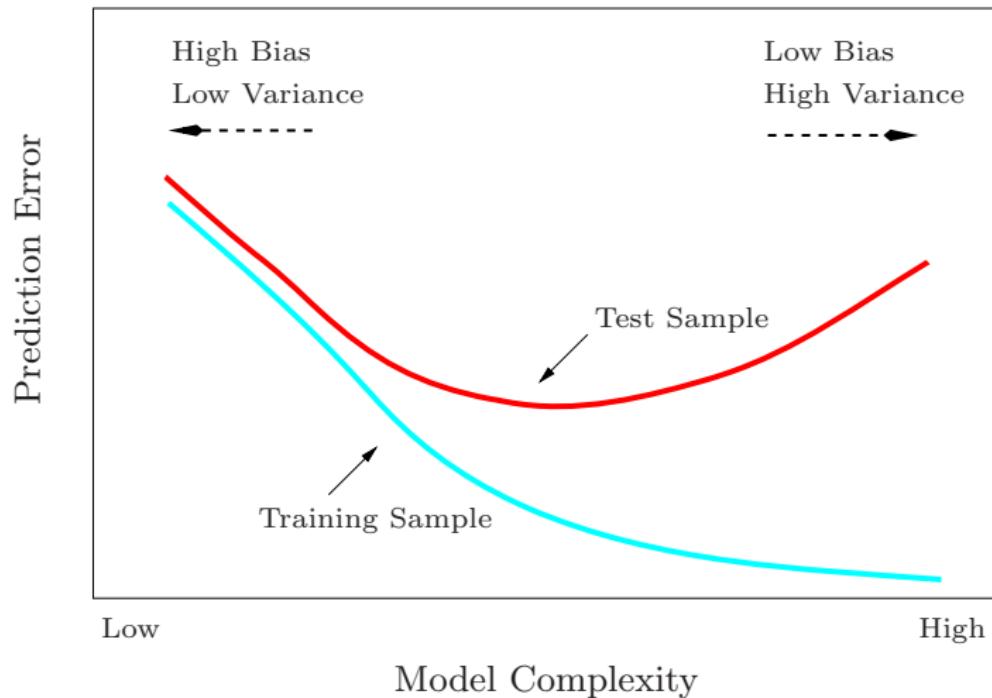
Si tous les prédicteurs ont des effet similaires sur la réponse, la sélection de prédicteurs interprétables est très difficile.

Une solution est de **régulariser** le problème en **constrignant** les paramètres  $\beta$  à vivre dans un espace approprié, de sorte à

- ▶ rendre  $\mathbf{X}^T \mathbf{X}$  inversible,
- ▶ faciliter l'interprétabilité.

# Compromis Biais/Variance, toujours

Dans les deux cas, une meilleure erreur de prédiction est attendue en ajustant le **compromis biais/variance** dans le modèle.



# Plan

Prérequis

Motivations: les limites du modèle linéaire

## Sélection de variables

Critères de choix/comparaison de modèles

Algorithmes de sélection de sous-ensembles

Illustration: cancer de la prostate

Régularisation

# Sélection de variable

## Problématique

En augmentant le nombre de variables

- ▶ on intègre de plus en plus d'information dans le modèle ;
- ▶ on augmente le nombre de paramètres à estimer et  $\mathbb{V}(\hat{Y}_i) \nearrow$ .

## Idée

On recherche un (petit) ensemble  $\mathcal{S}$  de  $k$  variables parmi  $p$  telles que

$$Y \approx X_{\mathcal{S}}^T \hat{\beta}_{\mathcal{S}}.$$

## Ingrédients

Pour trouver un compromis, on a besoin

1. d'un critère pour évaluer la qualité du modèle;
2. d'un algorithme pour déterminer les  $k$  variables optimisant le critère.

# Sélection de variable

## Problématique

En augmentant le nombre de variables

- ▶ on intègre de plus en plus d'information dans le modèle ;
- ▶ on augmente le nombre de paramètres à estimer et  $\mathbb{V}(\hat{Y}_i) \nearrow$ .

## Idée

On recherche un (petit) ensemble  $\mathcal{S}$  de  $k$  variables parmi  $p$  telles que

$$Y \approx X_{\mathcal{S}}^T \hat{\beta}_{\mathcal{S}}.$$

## Ingrédients

Pour trouver un compromis, on a besoin

1. d'un **critère** pour évaluer la qualité du modèle;
2. d'un **algorithme** pour déterminer les  $k$  variables optimisant le critère.

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Critères de choix/comparaison de modèles

Algorithmes de sélection de sous-ensembles

Illustration: cancer de la prostate

Régularisation

# Critères pénalisés

## Principe général

### Idée

Plutôt que d'estimer l'erreur de prédiction par l'erreur de test, on estime de combien l'erreur d'entraînement sous-estime la vraie erreur.

### Forme générique des critères

Sans ajuster d'autres modèles, on calcule

$$\hat{\text{err}} = \text{err}_{\mathcal{D}} + \text{"optimisme"}$$
.

### Remarques

- ▶ beaucoup moins coûteux que la validation croisée
- ▶ revient à "pénaliser" les modèles trop complexes.

# Critères pénalisés

## Principe général

### Idée

Plutôt que d'estimer l'erreur de prédiction par l'erreur de test, on estime de combien l'erreur d'entraînement sous-estime la vraie erreur.

### Forme générique des critères

Sans ajuster d'autres modèles, on calcule

$$\hat{\text{err}} = \text{err}_{\mathcal{D}} + \text{"optimisme"}$$
.

### Remarques

- ▶ beaucoup moins coûteux que la validation croisée
- ▶ revient à "pénaliser" les modèles trop complexes.

# Critères pénalisés

Les plus populaires en régression

Soit  $k$  la dimension du modèle (le nombre de prédicteurs utilisés).

Critères pour le modèle de régression linéaire  $\sigma$  connue

On choisit le modèle de taille  $k$  minimisant un des critères suivants.

- ▶  **$C_p$  de Mallows**

$$C_p = \frac{\text{err}_{\mathcal{D}}}{\sigma^2} - n + 2 \frac{k}{n}$$

- ▶ **Akaike Information Criteria** équivalent au  $C_p$  quand  $\sigma$  est connue

$$\text{AIC} = -2\text{loglik} + 2k = \frac{n}{\sigma^2} \text{err}_{\mathcal{D}} + 2k.$$

- ▶ **Bayesian Information Criterion**

$$\text{BIC} = -2\text{loglik} + k \log(n) = \frac{n}{\sigma^2} \text{err}_{\mathcal{D}} + k \log(n).$$

# Critères pénalisés

Les plus populaires en régression

Soit  $k$  la dimension du modèle (le nombre de prédicteurs utilisés).

Critères pour le modèle de régression linéaire  $\sigma$  inconnue

On choisit le modèle de taille  $k$  minimisant un des critères suivants.

- **$C_p$  de Mallows**  $\sigma$  estimée par l'estimateur sans biais  $\hat{\sigma}$

$$C_p = \frac{\text{err}_{\mathcal{D}}}{\hat{\sigma}^2} - n + 2 \frac{k}{n}$$

- **Akaike Information Criteria**  $\sigma^2$  estimée par  $\text{err}_{\mathcal{D}}/n$

$$\text{AIC} = -2\text{loglik} + 2k = n \log(\text{err}_{\mathcal{D}}) + 2k.$$

- **Bayesian Information Criterion**  $\sigma^2$  estimée par  $\text{err}_{\mathcal{D}}/n$

$$\text{BIC} = -2\text{loglik} + k \log(n) = n \log(\text{err}_{\mathcal{D}}) + k \log(n).$$

## $C_p$ /AIC: preuve

L'idéal serait de minimiser l'espérance de la distance entre le vrai modèle  $\mathbf{X}\beta = \mu$  et celui de l'OLS. La distance se décompose comme suit:

$$\begin{aligned}\|\mu - \mathbf{X}\hat{\beta}^{\text{ols}}\|^2 &= \|\mathbf{y} - \varepsilon - \mathbf{P}_X\mathbf{y}\|^2 \\&= \|\mathbf{y} - \hat{\mathbf{y}}\|^2 + \|\varepsilon\|^2 - 2\varepsilon^\top(\mathbf{y} - \mathbf{P}_X\mathbf{y}) \\&= n\text{err}_{\mathcal{D}} + \|\varepsilon\|^2 - 2\varepsilon^\top(\mathbf{I} - \mathbf{P}_X)(\mu + \varepsilon) \\&= n\text{err}_{\mathcal{D}} - \|\varepsilon\|^2 + 2\varepsilon^\top\mathbf{P}_X\varepsilon - 2\varepsilon^\top(\mathbf{I} - \mathbf{P}_X)\mu\end{aligned}$$

En espérance, on a

- ▶  $\mathbb{E}[\|\varepsilon\|^2] = n\sigma^2$
- ▶  $\mathbb{E}[\varepsilon^\top(\mathbf{I} - \mathbf{P}_X)\mu] = 0$
- ▶  $\mathbb{E}[2\varepsilon^\top\mathbf{P}_X\varepsilon] = 2\mathbb{E}[\text{trace}(\varepsilon^\top\mathbf{P}_X\varepsilon)] = 2\text{trace}(\mathbf{P}_X)\sigma^2$

Si  $k$  est la dimension de l'espace où l'on projette, on trouve

$$\mathbb{E}\|\mu - \mathbf{X}\hat{\beta}^{\text{ols}}\|^2 = n\text{err}_{\mathcal{D}} - n\sigma^2 + 2k\sigma^2$$

Il suffit alors de diviser par  $n\sigma^2$ .

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Critères de choix/comparaison de modèles

Algorithmes de sélection de sous-ensembles

Illustration: cancer de la prostate

Régularisation

# Recherche exhaustive (best-subset)

## Algorithme

Pour  $k = 0, \dots, p$ , trouver le sous-ensemble de  $k$  variables qui donne le plus petit  $SCR$  parmi les  $2^k$  modèles.

## Propriétés

- ▶ Peut être généralisé à d'autres critères ( $R^2$ , AIC, BIC...)
- ▶ Existence d'un algorithme efficace ("Leaps and Bound")
- ▶ impossible dès que  $p > 30$ .

# Sélection avant (Forward regression)

## Algorithme

1. Commencer avec  $\mathcal{S} = \emptyset$
2. À l'étape  $k$  trouver la variable qui ajoutée à  $\mathcal{S}$  donne le meilleur modèle
- 2'. À l'étape  $k$  trouver le meilleur modèle lorsqu'une variable est ajoutée ou enlevée.
- 3 etc. jusqu'au modèle à  $p$  variables

## Propriétés

- ▶ le meilleur modèle est compris en terme de SCR ou  $R^2$ , AIC, BIC...
- ▶ approprié lorsque  $p$  est grand
- ▶ biais important, mais variance/complexité contrôlée.
- ▶ algorithme dit “glouton” (greedy)

## Sélection avant Pas à pas (Forward-stepwise)

### Algorithme

1. Commencer avec  $\mathcal{S} = \emptyset$
2. À l'étape  $k$  trouver la variable qui ajoutée à  $\mathcal{S}$  donne le meilleur modèle
- 2'. À l'étape  $k$  trouver le meilleur modèle lorsqu'une variable est ajoutée ou enlevée.
- 3 etc. jusqu'au modèle à  $p$  variables

### Propriétés

- ▶ le meilleur modèle est compris en terme de SCR ou  $R^2$ , AIC, BIC...
- ▶ approprié lorsque  $p$  est grand
- ▶ biais important, mais variance/complexité contrôlée.
- ▶ algorithme dit “glouton” (greedy)

## Sélection arrière

### Algorithm

- 1 Commencer avec le modèle plein  $\mathcal{S} = \{1, \dots, p\}$
- 2 À l'étape  $k$ , enlever la variable ayant le moins d'influence sur l'ajustement.
- 3 etc. jusqu'au modèle nul.

### Propriétés

- ▶ le meilleur modèle est compris en terme de SCR ou  $R^2$ , AIC, BIC...
- ▶ ne fonctionne pas si  $n < p$
- ▶ algorithme dit “glouton” (greedy)

# Plan

## Prérequis

Motivations: les limites du modèle linéaire

## Sélection de variables

Critères de choix/comparaison de modèles

Algorithmes de sélection de sous-ensembles

**Illustration: cancer de la prostate**

## Régularisation

# Recherche exhaustive I

```
library(leaps)
```

On calcule tous les modèles possibles

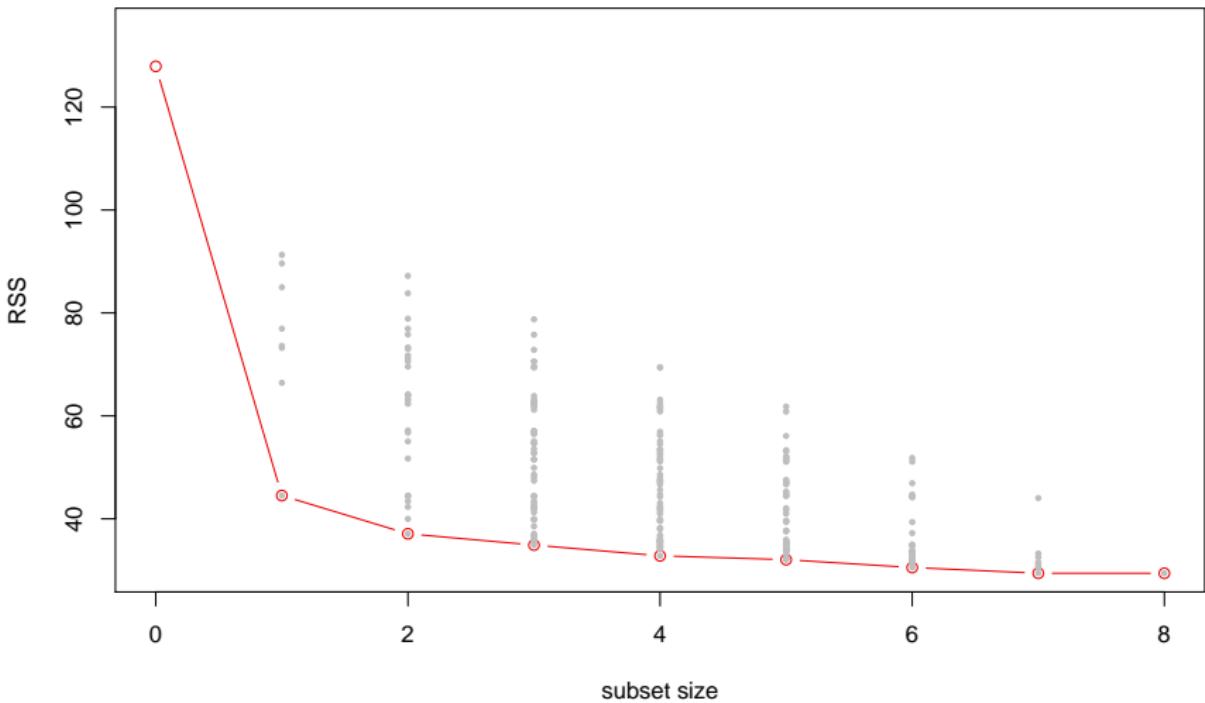
```
out <- regsubsets(lpsa ~ . , data=prostate.train,
                   nbest=100, really.big=TRUE)
bss <- summary(out)
```

Extraction de la taille et des SCR. Ajout du modèle nul (juste l'intercept)

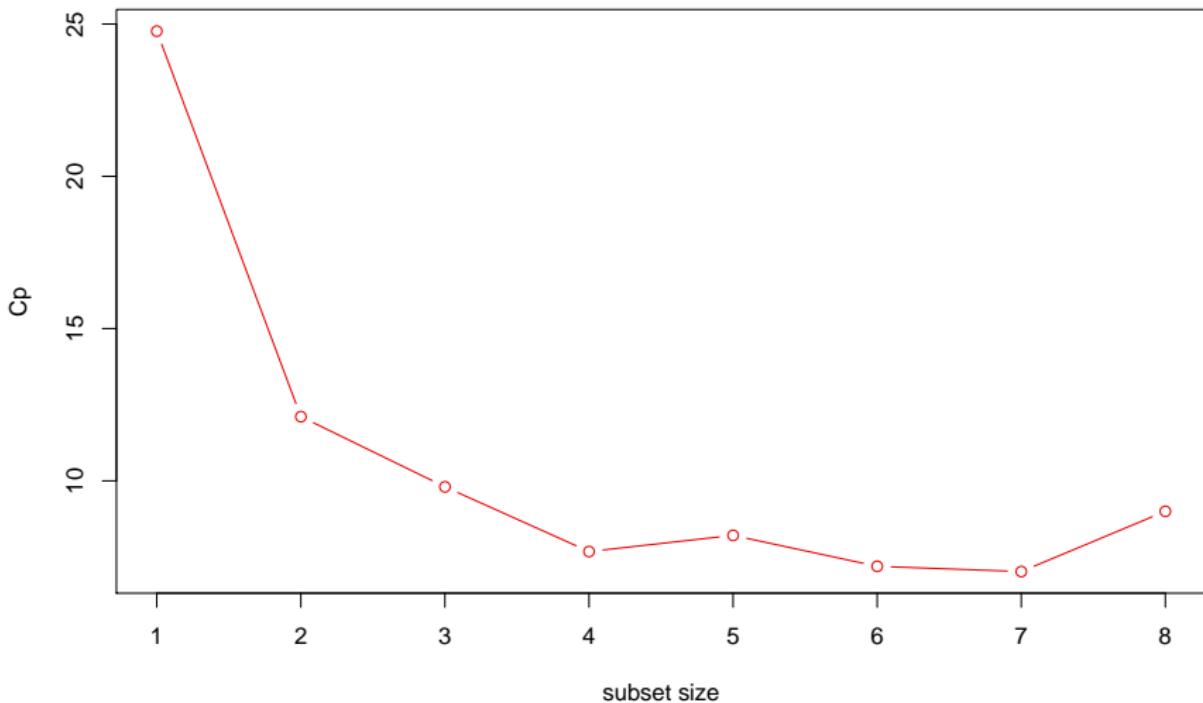
```
bss.size <- as.numeric(rownames(bss$which))
intercept <- lm(lpsa ~ 1, data=prostate)
bss.best.rss <- c(sum(resid(intercept)^2), tapply(bss$rss , bss.size, min))
```

```
plot(0:8, bss.best.rss, ylim=c(30, 135), type="b",
     xlab="subset size", ylab="RSS", col="red2")
points(bss.size, bss$rss, pch=20, col="gray", cex=0.7)
```

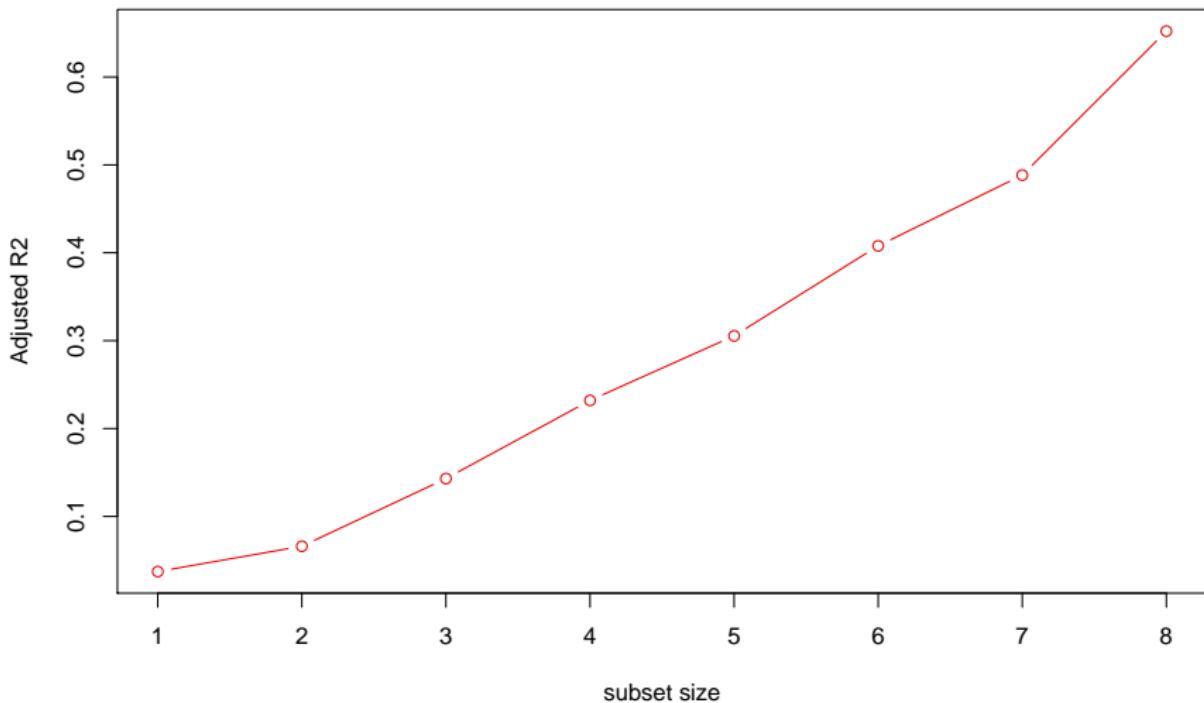
## Recherche exhaustive II



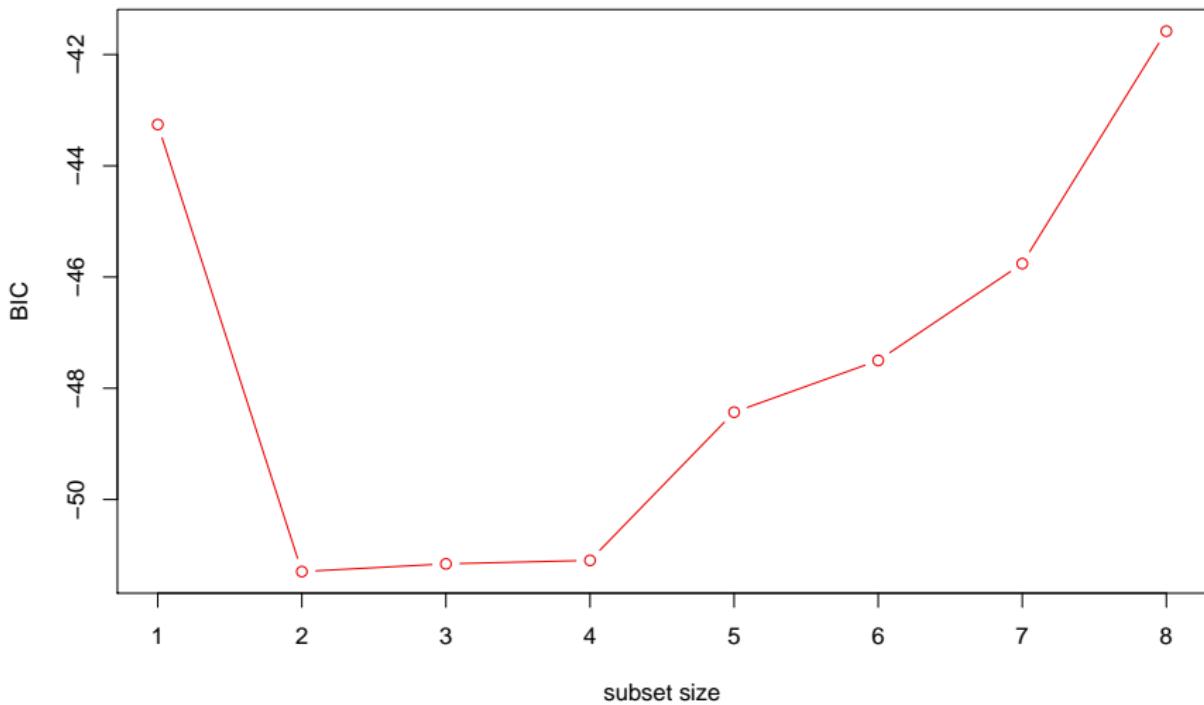
## Recherche exhaustive III



## Recherche exhaustive VI



## Recherche exhaustive V



# Forward-Stepwise dans R (I)

Création du modèle nul et du modèle plein

```
null <- lm(lpsa ~ 1, data=prostate.train)
full <- lm(lpsa ~ ., data=prostate.train)
```

Création de l'ensemble des modèles à parcourir ("scope")

```
lower <- ~1
upper <- ~lcavol+lweight+age+lbph+svi+lcp+gleason+pgg45
scope <- list(lower=lower,upper=upper)
```

Stepwise avec AIC: forward, backward, both

```
fwd <- step(null, scope, direction="forward" , trace=FALSE)
bwd <- step(full, scope, direction="backward", trace=FALSE)
both <- step(null, scope, direction="both"     , trace=FALSE)
```

↔ 3 modèles équivalents

# Forward regression

```
fwd

##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi + lbph, data = prostate.train)
##
## Coefficients:
## (Intercept)      lcavol       lweight        svi        lbph
## -0.3259       0.9177       1.9853       0.3203       0.2052

fwd$anova

##          Step Df Deviance Resid. Df Resid. Dev      AIC
## 1             NA      NA           66   96.28145  26.29306
## 2 + lcavol -1  51.752862           65   44.52858 -23.37361
## 3 + lweight -1   7.436737           64   37.09185 -33.61680
## 4     + svi -1   2.184097           63   34.90775 -35.68291
## 5     + lbph -1   2.092754           62   32.81499 -37.82507
```

# Backward regression

```
bwd

##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + age + lbph + svi + lcp +
##     pgg45, data = prostate.train)
##
## Coefficients:
## (Intercept)      lcavol       lweight        age        lbph
##      0.2591      1.0419      2.2814     -1.2791      0.2116
##      svi          lcp       pgg45
##      0.3536     -0.2911      0.3532

bwd$anova

##      Step Df Deviance Resid. Df Resid. Dev      AIC
## 1           NA      NA      58   29.42638 -37.12766
## 2 - gleason  1 0.01091586      59   29.43730 -39.10281
```

# Stepwise regression

```
both

##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi + lbph, data = prostate.train)
##
## Coefficients:
## (Intercept)      lcavol       lweight        svi        lbph
## -0.3259       0.9177       1.9853       0.3203       0.2052

both$anova

##          Step Df Deviance Resid. Df Resid. Dev      AIC
## 1             NA      NA      66   96.28145  26.29306
## 2 + lcavol -1  51.752862      65   44.52858 -23.37361
## 3 + lweight -1   7.436737      64   37.09185 -33.61680
## 4     + svi -1   2.184097      63   34.90775 -35.68291
## 5     + lbph -1   2.092754      62   32.81499 -37.82507
```

# Évaluation sur données test

```
print(err.ols)

## [1] 0.5221043

print(err.AIC.fwd <- mean((y.test-predict(fwd ,prostate.test))^2))

## [1] 0.4520967

print(err.AIC.bwd <- mean((y.test-predict(bwd ,prostate.test))^2))

## [1] 0.517824

print(err.AIC <- mean((y.test-predict(both,prostate.test))^2))

## [1] 0.4520967
```

# Stepwise en R: modification pour le BIC

Modèle plus parcimonieux

```
BIC <- step(null, scope, k=log(n <- nrow(prostate)), trace=FALSE)
BIC

##
## Call:
## lm(formula = lpsa ~ lcavol + lweight, data = prostate.train)
##
## Coefficients:
## (Intercept)      lcavol      lweight
##       -1.049       1.139       2.720

print(err.BIC  <- mean((y.test-predict(BIC ,prostate.test))^2))

## [1] 0.4908699
```

# Sélection variable : quelques remarques

## Interprétabilité

1. Si le vrai  $\mathcal{S}$  ne contient que **quelques variables liées à la réponse**,  
~~ les algorithmes de sélection peuvent retrouver les prédicteurs pertinents.
2. Si le vrai  $\mathcal{S}$  contient **beaucoup de variables très corrélées**  
~~ les variables sélectionnées seront difficiles à interpréter.

## Limites liées à la stabilité

En présence de prédicteurs très corrélés ou lorsque  $n < p$ , **de petites perturbations** des données peuvent provoquer de **grandes différences** entre les ensembles de variables sélectionnées.

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

## Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

**Motivations et principe**

La régression Ridge

Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Objectifs

Contrôler le vecteur  $\hat{\beta}$  pour

## 1. Régulariser le problème

- ▶ Pour des questions numériques, (conditionnement de  $\mathbf{X}^T \mathbf{X}$ ),
- ▶ Pour des questions de stabilité, (corrélations entre les  $(X_1, \dots, X_p)$ ).

## 2. Améliorer la prédiction

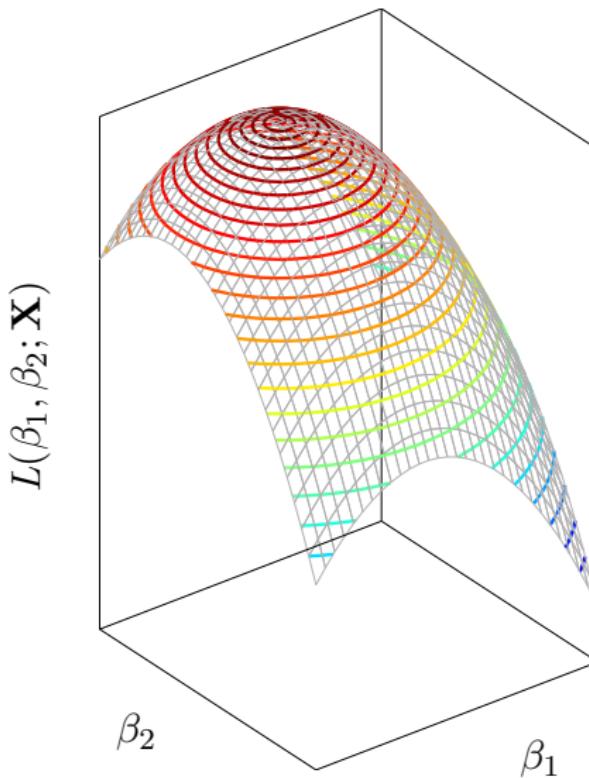
- ▶ En augmentant le biais pour diminuer la variance
- ▶ En contrôlant les variables non pertinentes

## 3. Favoriser l'interprétabilité

- ▶ En contrôlant la complexité du modèle,
- ▶ En intégrant la sélection de variable (Lasso).

# Une vue géométrique de la régularisation

## Optimisation sous contraintes



On veut résoudre un problème de la forme

$$\underset{\beta_1, \beta_2}{\text{maximize}} \quad L(\beta_1, \beta_2; \mathbf{X})$$

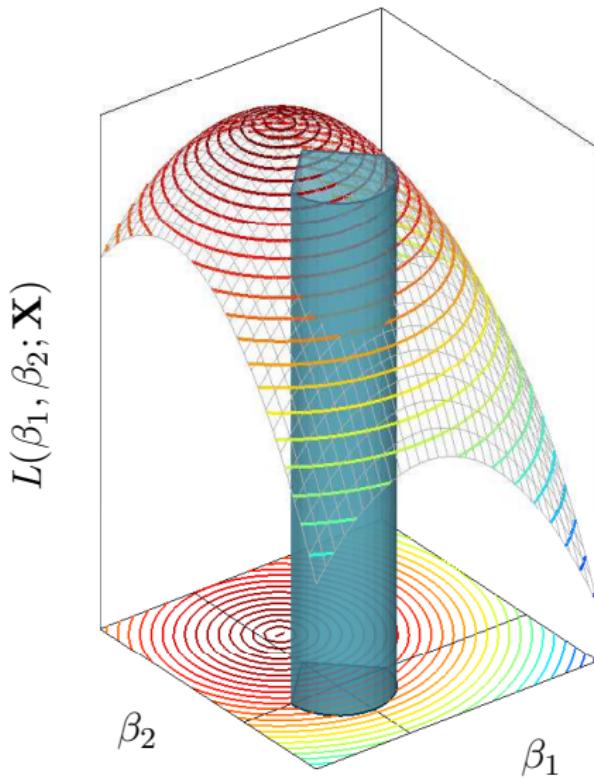
où  $L$  est typiquement une vraisemblance concave, ce qui est équivalent à

$$\underset{\beta_1, \beta_2}{\text{minimize}} \quad L'(\beta_1, \beta_2; \mathbf{X})$$

où  $L' = -L$  est convexe (par exemple, la perte de l'OLS).

# Une vue géométrique de la régularisation

## Optimisation sous contraintes



On restreint l'espace des  $\beta$ , ainsi

$$\begin{cases} \underset{\beta_1, \beta_2}{\text{maximize}} & L(\beta_1, \beta_2; \mathbf{X}) \\ \text{s.t.} & \Omega(\beta_1, \beta_2) \leq c \end{cases},$$

où  $\Omega$  définit un ensemble de  
*contraintes* sur  $\beta$ .

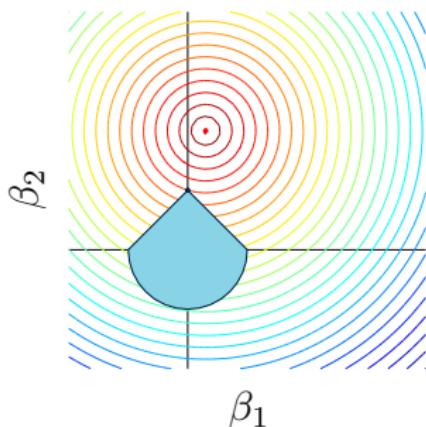
# Une vue géométrique de la régularisation

Optimisation sous contraintes

On restreint l'espace des  $\beta$ , ainsi

$$\begin{cases} \underset{\beta_1, \beta_2}{\text{maximize}} & L(\beta_1, \beta_2; \mathbf{X}) \\ \text{s.t.} & \Omega(\beta_1, \beta_2) \leq c \end{cases},$$

où  $\Omega$  définit un ensemble de *contraintes* sur  $\beta$ .



$\Updownarrow$

$$\underset{\beta_1, \beta_2}{\text{minimize}} J(\beta),$$

où  $J$  est la fonction objectif (convexe) définie par

$$J(\beta) = -L(\beta_1, \beta_2; \mathbf{X}) + \lambda \Omega(\beta_1, \beta_2)$$

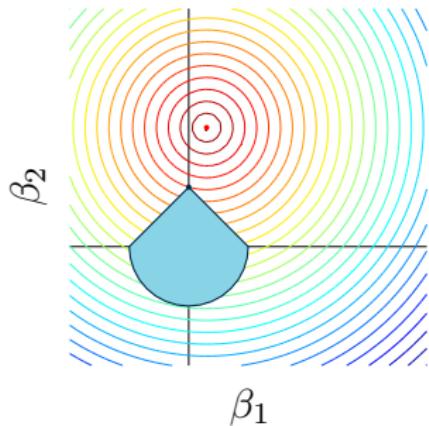
# Une vue géométrique de la régularisation

Optimisation sous contraintes

On restreint l'espace des  $\beta$ , ainsi

$$\begin{cases} \underset{\beta_1, \beta_2}{\text{maximize}} & L(\beta_1, \beta_2; \mathbf{X}) \\ \text{s.t.} & \Omega(\beta_1, \beta_2) \leq c \end{cases},$$

où  $\Omega$  définit un ensemble de *contraintes* sur  $\beta$ .



$\Updownarrow$

$$\underset{\beta_1, \beta_2}{\text{minimize}} J(\beta),$$

où  $J$  est la fonction objectif (convexe) définie par

$$J(\beta) = -L(\beta_1, \beta_2; \mathbf{X}) + \lambda \Omega(\beta_1, \beta_2)$$

Comment choisir  $\Omega$  ?

# Plan

## Prérequis

## Motivations: les limites du modèle linéaire

## Sélection de variables

## Régularisation

### Motivations et principe

### La régression Ridge

Définition de l'estimateur

Propriétés et résolution pratique

Choix du paramètre de régularisation

### Régression Lasso

### Variations autour du Lasso

### Modèles graphiques gaussiens parcimonieux

# Plan

## Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

## Régularisation

Motivations et principe

### La régression Ridge

Définition de l'estimateur

Propriétés et résolution pratique

Choix du paramètre de régularisation

### Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

## Définition

### Remarque

Si les  $\beta_j$  ne sont pas contraints, ils peuvent prendre de très grandes valeurs et donc avoir une grande variance.

### Idée

Pour contrôler la variance, il faut contrôler la taille des coefficients de  $\beta$ .  
Cette approche pourrait réduire sensiblement l'erreur de prédiction.

### La Ridge comme problème de régularisation

$$\hat{\beta}^{\text{ridge}} = \arg \min_{\beta \in \mathbb{R}^{p+1}} \text{RSS}(\beta), \quad \text{s.c. } \sum_{j=1}^p \beta_j^2 \leq s,$$

où  $s$  est un facteur de rétrécissement.

# Définition

## Remarque

Si les  $\beta_j$  ne sont pas contraints, ils peuvent prendre de très grandes valeurs et donc avoir une grande variance.

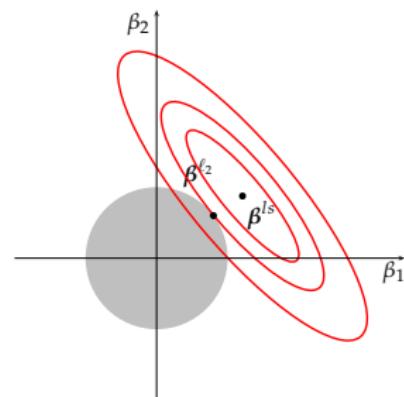
## Idée

Pour contrôler la variance, il faut contrôler la taille des coefficients de  $\beta$ .  
Cette approche pourrait réduire sensiblement l'erreur de prédiction.

## La Ridge comme problème de régularisation

$$\hat{\beta}^{\text{ridge}} = \arg \min_{\beta \in \mathbb{R}^{p+1}} \text{RSS}(\beta), \quad \text{s.c. } \sum_{j=1}^p \beta_j^2 \leq s,$$

où  $s$  est un facteur de rétrécissement.



## Un exemple en deux dimensions

Considérons que le vrai modèle est  $Y = X_1\beta_1 + X_2\beta_2 + \varepsilon$ . Si  $X_1$  et  $X_2$  sont très corrélés, alors  $X_1 \approx X_2$ . De plus pour tout  $\gamma \geq 0$ ,

$$\begin{aligned} Y &= X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \gamma(X_1 - X_2) + \varepsilon \\ &\approx X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \varepsilon. \end{aligned}$$

On prédit une réponse similaire pour un large panel d'estimateur de  $\beta$  indexés sur  $\gamma$ .

Pour de petits  $s$ , la régression Ridge contrôle

$$(\beta_1 + \gamma)^2 + (\beta_2 - \gamma)^2$$

qui est minimale pour  $\gamma = (\beta_2 - \beta_1)/2$ , et dans ce cas  $\beta_j = (\beta_1 + \beta_2)/2$ .

~ La Ridge “moyenne” les coefficients associés aux prédicteurs corrélés.

## Un exemple en deux dimensions

Considérons que le vrai modèle est  $Y = X_1\beta_1 + X_2\beta_2 + \varepsilon$ . Si  $X_1$  et  $X_2$  sont très corrélés, alors  $X_1 \approx X_2$ . De plus pour tout  $\gamma \geq 0$ ,

$$\begin{aligned} Y &= X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \gamma(X_1 - X_2) + \varepsilon \\ &\approx X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \varepsilon. \end{aligned}$$

On prédit une réponse similaire pour un large panel d'estimateur de  $\beta$  indexés sur  $\gamma$ .

Pour de petits  $s$ , la régression Ridge contrôle

$$(\beta_1 + \gamma)^2 + (\beta_2 - \gamma)^2$$

qui est minimale pour  $\gamma = (\beta_2 - \beta_1)/2$ , et dans ce cas  $\beta_j = (\beta_1 + \beta_2)/2$ .

~ La Ridge “moyenne” les coefficients associés aux prédicteurs corrélés.

## Un exemple en deux dimensions

Considérons que le vrai modèle est  $Y = X_1\beta_1 + X_2\beta_2 + \varepsilon$ . Si  $X_1$  et  $X_2$  sont très corrélés, alors  $X_1 \approx X_2$ . De plus pour tout  $\gamma \geq 0$ ,

$$\begin{aligned} Y &= X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \gamma(X_1 - X_2) + \varepsilon \\ &\approx X_1(\beta_1 + \gamma) + X_2(\beta_2 - \gamma) + \varepsilon. \end{aligned}$$

On prédit une réponse similaire pour un large panel d'estimateur de  $\beta$  indexés sur  $\gamma$ .

Pour de petits  $s$ , la régression Ridge contrôle

$$(\beta_1 + \gamma)^2 + (\beta_2 - \gamma)^2$$

qui est minimale pour  $\gamma = (\beta_2 - \beta_1)/2$ , et dans ce cas  $\beta_j = (\beta_1 + \beta_2)/2$ .

~ La Ridge “moyenne” les coefficients associés aux prédicteurs corrélés.

# Un exemple en deux dimensions (en R) I

On génère deux prédicteurs corrélés

```
suppressMessages(library(quadrupen))
x1 <- rnorm(5)
x2 <- x1 + rnorm(5,0, 0.5)
cor(x1,x2)

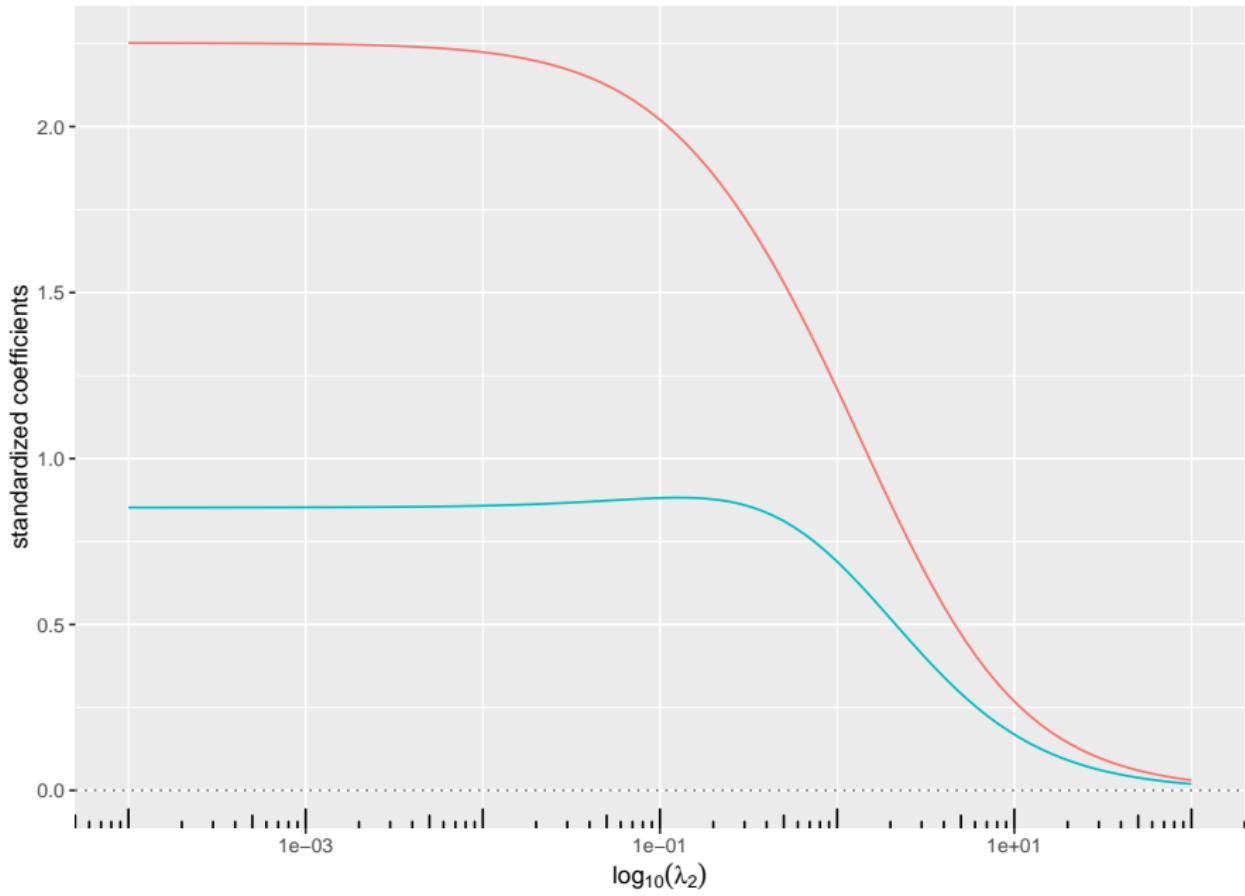
## [1] 0.7188671
```

On génère  $Y$  et on trace le **chemin de régularisation**

```
y <- x1 + x2 +rnorm(5)
plot(ridge(cbind(x1,x2),y))
```

## Un exemple en deux dimensions (en R) II

ridge path



# La ridge comme un problème de régression pénalisée

On ne pénalise pas la constante et on considère donc  $\beta = (\beta_1, \dots, \beta_p)$  et on pose

- ▶  $\hat{\beta}_0 = \bar{y} - \bar{x}\hat{\beta}$
- ▶ on centre  $y$  et  $x_j$ ,  $j = 1, \dots, p$ .

On normalise  $x_j$  pour ajuster et on renvoie les estimations  $\hat{\beta}^{\text{ridge}}$  dans l'échelle d'origine.

## Forme Lagrangienne (convexe)

$$\begin{aligned}\hat{\beta}^{\text{ridge}} &= \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|y - X\beta\|^2 + \lambda \|\beta\|^2 \\ &= (X^\top X + \lambda I_p)^{-1} X^\top y = H_\lambda y.\end{aligned}$$

## Forte convexité

Contrairement à l'OLS, une solution unique existe toujours quand  $\lambda > 0$  quelque soit le conditionnement de la matrice  $X^\top X$ .

# Plan

## Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

## Régularisation

Motivations et principe

### La régression Ridge

Définition de l'estimateur

Propriétés et résolution pratique

Choix du paramètre de régularisation

### Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

## Connexion à l'OLS

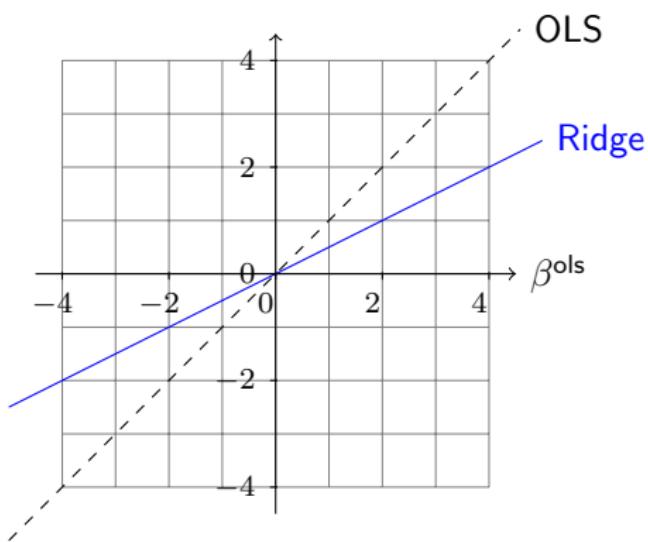
Soit  $\mathbf{S}_\lambda = \mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}_p$ . Alors

$$\hat{\boldsymbol{\beta}}^{\text{ridge}} = \mathbf{S}_\lambda^{-1} \mathbf{X}^\top \mathbf{X} \hat{\boldsymbol{\beta}}^{\text{ols}} = (\mathbf{I}_p - \lambda \mathbf{S}_\lambda^{-1}) \hat{\boldsymbol{\beta}}^{\text{ols}}.$$

Lorsque  $\lambda = 0$ ,  $\hat{\boldsymbol{\beta}}^{\text{ridge}}$  et  $\hat{\boldsymbol{\beta}}^{\text{ols}}$  coïncident.

Dans le cas d'un design orthonormal,  $\mathbf{X}^\top \mathbf{X} = \mathbf{I}$  et

$$\hat{\boldsymbol{\beta}}^{\text{ridge}} = \frac{1}{1 + \lambda} \hat{\boldsymbol{\beta}}^{\text{ols}}.$$



# Biais et variance de l'estimateur Ridge

## Proposition

$$\mathbb{E} \left( \hat{\boldsymbol{\beta}}^{\text{ridge}} - \hat{\boldsymbol{\beta}}^{\text{ols}} \right) = -\lambda \mathbf{S}_\lambda^{-1} \boldsymbol{\beta}, \quad \mathbb{V} \left( \hat{\boldsymbol{\beta}}^{\text{ridge}} \right) = \sigma^2 \mathbf{S}_\lambda^{-1} \mathbf{X}^\top \mathbf{X} \mathbf{S}_\lambda^{-1}.$$

et

$$\mathbb{V}(\hat{\boldsymbol{\beta}}^{\text{ols}}) - \mathbb{V}(\hat{\boldsymbol{\beta}}^{\text{ridge}}) \succeq 0$$

donc  $\mathbb{V}(x^T \hat{\boldsymbol{\beta}}^{\text{ols}}) \geq \mathbb{V}(x^T \hat{\boldsymbol{\beta}}^{\text{ridge}})$  un  $x$  fixé.

- ▶ quand  $\lambda \rightarrow 0$ , sans biais, grande variance (OLS)
  - ▶ quand  $\lambda \rightarrow \infty$ , grand biais, variance nulle.
- ~~> Un compromis est nécessaire (*i.e.*, un bon choix pour  $\lambda$ ).

# Calcul pratique du chemin de solution

Ridge et SVD

Décomposition en valeur singulière – lien vers une illustration

$$\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^T,$$

- ▶  $\mathbf{U}$  est une matrice  $n \times \min(n, p)$  orthogonale génératrice de l'espace ligne,
- ▶  $\mathbf{V}$  est une matrice  $p \times \min(n, p)$  orthogonale génératrice de l'espace colonne,
- ▶  $\mathbf{D} = \text{diag}(d_1, \dots, d_i, \dots, d_m \in \mathbb{R}^{n, p})$  contient les valeurs singulières de  $\mathbf{X}$ .

$$\hat{\boldsymbol{\beta}}^{\text{ridge}} = \mathbf{V}\boldsymbol{\Delta}_\lambda \mathbf{U}^T \mathbf{y},$$

où  $\boldsymbol{\Delta}_\lambda$  est une matrice diagonale telle que  $\Delta_i = d_i / (d_i^2 + \lambda)$ .

Coût algorithmique

Le calcul du chemin de solution complet pour  $K$  valeurs de  $\lambda$  nécessite

# Interprétation liée à la SVD

Pour les moindres carrés

$$\hat{\mathbf{X}}\hat{\boldsymbol{\beta}}^{\text{ols}} = \mathbf{X}(\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y} = \mathbf{U}\mathbf{U}^\top \mathbf{y}.$$

où  $\mathbf{U}^\top \mathbf{y}$  sont les coordonnées de  $\mathbf{y}$  dans la base orthonormale  $\mathbf{U}$ .

Pour la ridge

$$\hat{\mathbf{X}}\hat{\boldsymbol{\beta}}^{\text{ridge}} = \mathbf{X}(\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{I}_p)^{-1}\mathbf{X}^\top \mathbf{y} = \sum_{j=1}^p \mathbf{u}_j \frac{d_j^2}{d_j^2 + \lambda} \mathbf{u}_j^\top \mathbf{y}.$$

~~ La régularisation ridge régularise plus les coefficients associés aux axes de faibles variance  $d_j$ .

# Une implémentation R possible pour la régression Ridge

```
ridge.regression <- function(x,y,lambda=0){  
  ## x is assumed to be centered/scaled by the user  
  
  n.lambda <- length(lambda)  
  variables <- colnames(x)  
  p <- length(variables)  
  
  SVD <- svd(x)  
  d <- rep(SVD$d,n.lambda)  
  d2 <- rep(SVD$d^2,n.lambda)  
  lambdas <- rep(lambda,each=p)  
  V <- SVD$v  
  U <- SVD$u  
  
  Delta <- d/(d^2+lambdas)  
  beta <- t(V %*% matrix((rep(t(U) %*% y, n.lambda) * Delta),nrow=p))  
  df <- colSums(matrix(d2/rep(d2+lambdas,nrow=p)))  
  colnames(beta) <- variables  
  
  return(list(beta=beta,beta0=mean(y),df=df))  
}
```

# Ridge et données du cancer de la prostate

## Calcul du chemin de solution

```
ridge.path <- ridge(x.train,y.train)
```

Calcul de l'erreur de prédiction sur l'ensemble test pour tout  $\lambda$

```
err <- colMeans((y.test-predict(ridge.path,x.test))^2)
```

Ainsi, le  $\lambda^*$  qui minimise cette erreur est

```
ridge.path@lambda2[which.min(err)]
```

```
## [1] 0.5722368
```

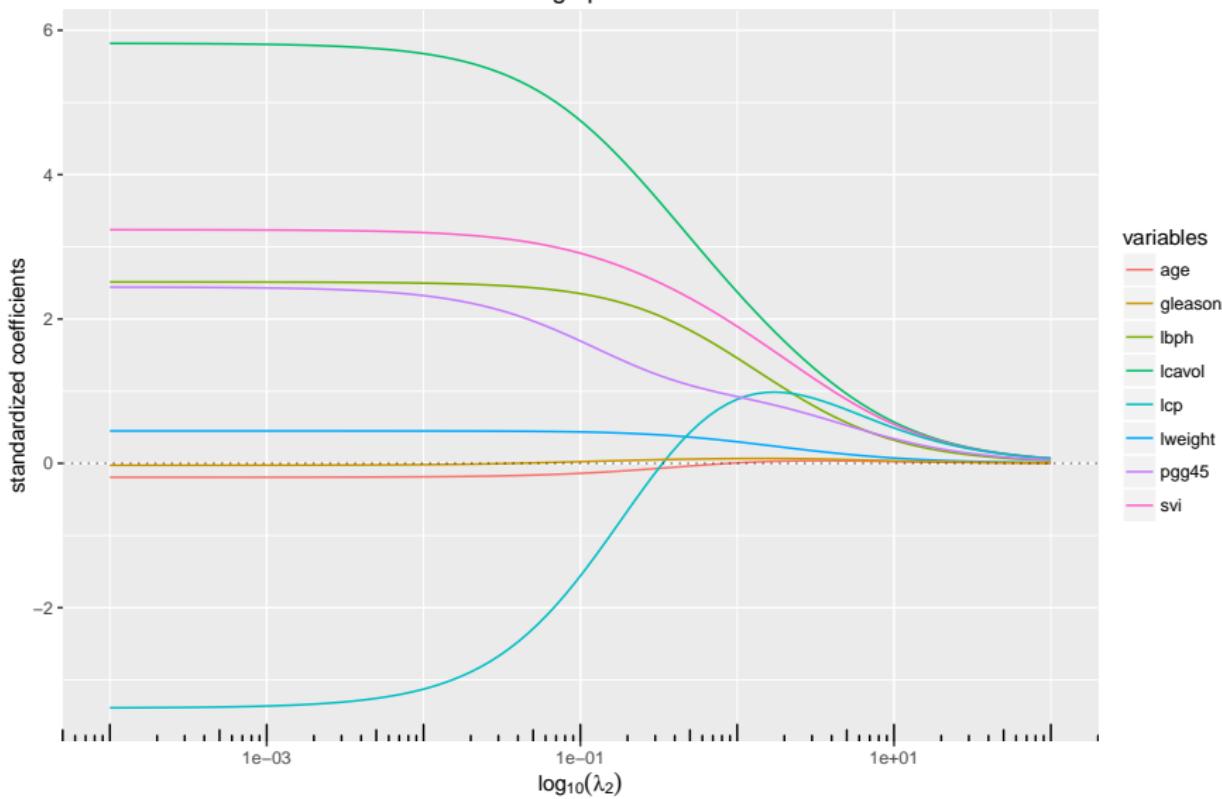
L'erreur de prédiction est légèrement meilleure que celle de l'OLS

```
err.ridge <- err[which.min(err)]; err.ridge; err.ols
```

```
##      0.572
## 0.5148989
## [1] 0.5221043
```

# Ridge et données du cancer de la prostate

ridge path



# Ridge généralisée (I)

## Critère généralisé

$$\begin{aligned}\hat{\beta}^{\text{ridge}} &= \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \beta^\top \mathbf{L} \beta \\ &= (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{L})^{-1} \mathbf{X}^\top \mathbf{y} = \mathbf{H}_\lambda^{\mathbf{L}} \mathbf{y}.\end{aligned}$$

La matrice  $\mathbf{L}$  doit être semi-définie positive pour préserver la convexité.

## Résolution

- ▶ Considérons les données transformées  $\mathbf{X}\mathbf{C}^{-1}$  où  $\mathbf{C} = \mathbf{L}^{1/2}$  e.g. par factorisation de Cholesky  $\mathbf{C}^T \mathbf{C} = \mathbf{L}$ .
- ▶ Considérons la SVD  $\mathbf{U}\mathbf{D}\mathbf{V}^T = \mathbf{X}\mathbf{C}^{-1}$  des données transformées

Alors

$$\hat{\beta}_{\mathbf{L}}^{\text{ridge}} = \mathbf{C}^{-1} \mathbf{V} (\mathbf{D}^2 + \lambda \mathbf{I})^{-1} \mathbf{D} \mathbf{U}^T \mathbf{y}.$$

# Ridge généralisée (I)

## Critère généralisé

$$\begin{aligned}\hat{\beta}^{\text{ridge}} &= \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|^2 + \lambda \beta^\top \mathbf{L} \beta \\ &= (\mathbf{X}^\top \mathbf{X} + \lambda \mathbf{L})^{-1} \mathbf{X}^\top \mathbf{y} = \mathbf{H}_\lambda^{\mathbf{L}} \mathbf{y}.\end{aligned}$$

La matrice  $\mathbf{L}$  doit être semi-définie positive pour préserver la convexité.

## Résolution

- ▶ Considérons les données transformées  $\mathbf{X}\mathbf{C}^{-1}$  où  $\mathbf{C} = \mathbf{L}^{1/2}$  e.g. par factorisation de Cholesky  $\mathbf{C}^T \mathbf{C} = \mathbf{L}$ .
- ▶ Considérons la SVD  $\mathbf{UDV}^T = \mathbf{X}\mathbf{C}^{-1}$  des données transformées

Alors

$$\hat{\beta}_{\mathbf{L}}^{\text{ridge}} = \mathbf{C}^{-1} \mathbf{V} (\mathbf{D}^2 + \lambda \mathbf{I})^{-1} \mathbf{D} \mathbf{U}^T \mathbf{y}.$$

## Ridge généralisée (II)

### Interpretation graphique

Soit  $\mathcal{G} = (\mathcal{E}, \mathcal{V}, \mathcal{W})$  un graphe pondéré. Alors

$$\sum_{(i,j) \in \mathcal{E}} w_{ij}(\beta_j - \beta_i)^2 = \boldsymbol{\beta}^T \mathbf{L} \boldsymbol{\beta},$$

où  $\mathbf{L}$  est la matrice *Laplacienne* associé au graphe (semi-définie positive).

### Interpretation bayésienne

Supposons un prior Gaussien  $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{0}, \tilde{\mathbf{X}} = \mathbf{U} \mathbf{D} \mathbf{L}^{-1})$ , i.e.,

$$\log \mathbb{P}(\boldsymbol{\beta} | \mathbf{L}) = \boldsymbol{\beta}^T L \boldsymbol{\beta} + cst.,$$

alors  $\hat{\boldsymbol{\beta}}_{\mathbf{L}}^{\text{ridge}}$  est la moyenne a posteriori (le mode).

# Plan

## Prérequis

## Motivations: les limites du modèle linéaire

## Sélection de variables

## Régularisation

Motivations et principe

### La régression Ridge

Définition de l'estimateur

Propriétés et résolution pratique

### Choix du paramètre de régularisation

## Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Les options classiques

En estimant l'erreur de prédiction

On choisit le  $\lambda$  minimisant l'erreur estimée

- ▶ par l'estimateur hold-out ou
- ▶ par validation croisée.

Par critère pénalisé

On choisit le  $\lambda$  minimisant le critère de forme générale

$$\text{crit}(\lambda) = \text{err}_{\mathcal{D}}(\lambda) + \text{pen}(\text{df}_\lambda)$$

~~ Quel sens donner aux degrés de liberté pour la Ridge?

# En estimant l'erreur de prédiction par validation croisée

Pour la régression: PRESS (*predicted residual sum of squares*)

## Principe

1. Partitionner les données en  $K$  sous-ensembles,
2. Utiliser successivement chaque sous-ensemble comme test,
3. Calculer l'erreur de test pour les  $K$  sous-ensembles,
4. Moyenner les  $K$  erreurs pour obtenir l'estimation finale.

## Formalisme

Soit  $\kappa : \{1, \dots, n\} \rightarrow \{1, \dots, K\}$  une fonction indicatrice de la partition de la  $i$ ème observation. On note  $\hat{\beta}^{-k}$  les paramètres estimés sur les données privées du  $k$ ième sous-ensemble. Alors

$$\text{CV}(\hat{\beta}) = \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \hat{\beta}^{-\kappa(i)})^2$$

donne l'estimation de l'erreur de prédiction par validation croisée.

# En estimant l'erreur de prédiction par validation croisée

Pour la régression: PRESS (*predicted residual sum of squares*)

## Principe

1. Partitionner les données en  $K$  sous-ensembles,
2. Utiliser successivement chaque sous-ensemble comme test,
3. Calculer l'erreur de test pour les  $K$  sous-ensembles,
4. Moyenner les  $K$  erreurs pour obtenir l'estimation finale.

## Formalisme

Soit  $\kappa : \{1, \dots, n\} \rightarrow \{1, \dots, K\}$  une fonction indicatrice de la partition de la  $i$ ème observation. On note  $\hat{\beta}^{-k}$  les paramètres estimés sur les données privées du  $k$ ième sous-ensemble. Alors

$$\text{CV}(\hat{\beta}) = \frac{1}{n} \sum_{i=1}^n (y_i - x_i^T \hat{\beta}^{-\kappa(i)})^2$$

donne l'estimation de l'erreur de prédiction par validation croisée.

# Validation croisée: remarques et compléments

## Remarques

1. Si  $K = n$ , LOOCV,
2. Si  $K = 2$ , estimation "hold-out",
3. En grande dimension,  $K$  doit être choisi avec "attention",
4.  $CV(\lambda)$  calcule l'erreur de CV le long du chemin de régularisation.

## Proposition

Soit  $\mathbf{H}$  une matrice de lissage (*smoothing*), i.e., un opérateur linéaire fini tel que  $\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}$ . Alors l'erreur CV LOO vérifie

$$CV = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{f}(x_i)^{-i})^2 = \frac{1}{n} \sum_{i=1}^n \left( \frac{y_i - \hat{f}(x_i)}{1 - H_{ii}} \right)^2.$$

# Degrés de liberté effectifs

## Idées

- ▶ Les degrés de liberté décrivent le niveau de complexité d'un modèle .
- ▶ Pour l'OLS,  $df = p$  (plus 1 pour la constante).

## Définition

*Considérons une prédiction  $\hat{\mathbf{y}}$  ajustée depuis une observation  $\mathbf{y}$ . Les degrés de liberté généralisés de la prédiction sont définis par*

$$df(\hat{\mathbf{y}}) = \frac{1}{\sigma^2} \sum_{i=1}^n \text{cov}(\hat{y}_i, y_i).$$

$\rightsquigarrow$  *Plus l'ajustement est proche des données, plus le modèle est complexe, plus grande est la covariance.*

## Degrés de liberté effectifs: cas de la Ridge

### Proposition

Considérons une méthode linéaire qui s'écrit:

$$\hat{\mathbf{y}} = \mathbf{H}\mathbf{y}.$$

Les degrés de liberté effectifs du modèle  $\hat{\mathbf{y}}$  vérifient

$$df(\hat{\mathbf{y}}) = \text{Tr}(\mathbf{H}).$$

### Ridge

Pour la régression Ridge, df est une fonction décroissante de  $\lambda$  qui tend vers 0 (ou 1 en considérant la constante):

$$df(\hat{\mathbf{y}}_\lambda) = \sum_{i=1}^p \frac{d_i^2}{d_i^2 + \lambda},$$

# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$

```
crit <- criteria(ridge.path, plot=FALSE)
print(head(crit$criterion), digits=3)

##      AIC    BIC    GCV df lambda fraction
## 1 3.77 3.78 0.00575 9 0.000100      1
## 2 3.77 3.78 0.00575 9 0.000115      1
## 3 3.77 3.78 0.00575 9 0.000132      1
## 4 3.77 3.78 0.00575 9 0.000152      1
## 5 3.77 3.78 0.00575 9 0.000175      1
## 6 3.77 3.78 0.00575 9 0.000201      1

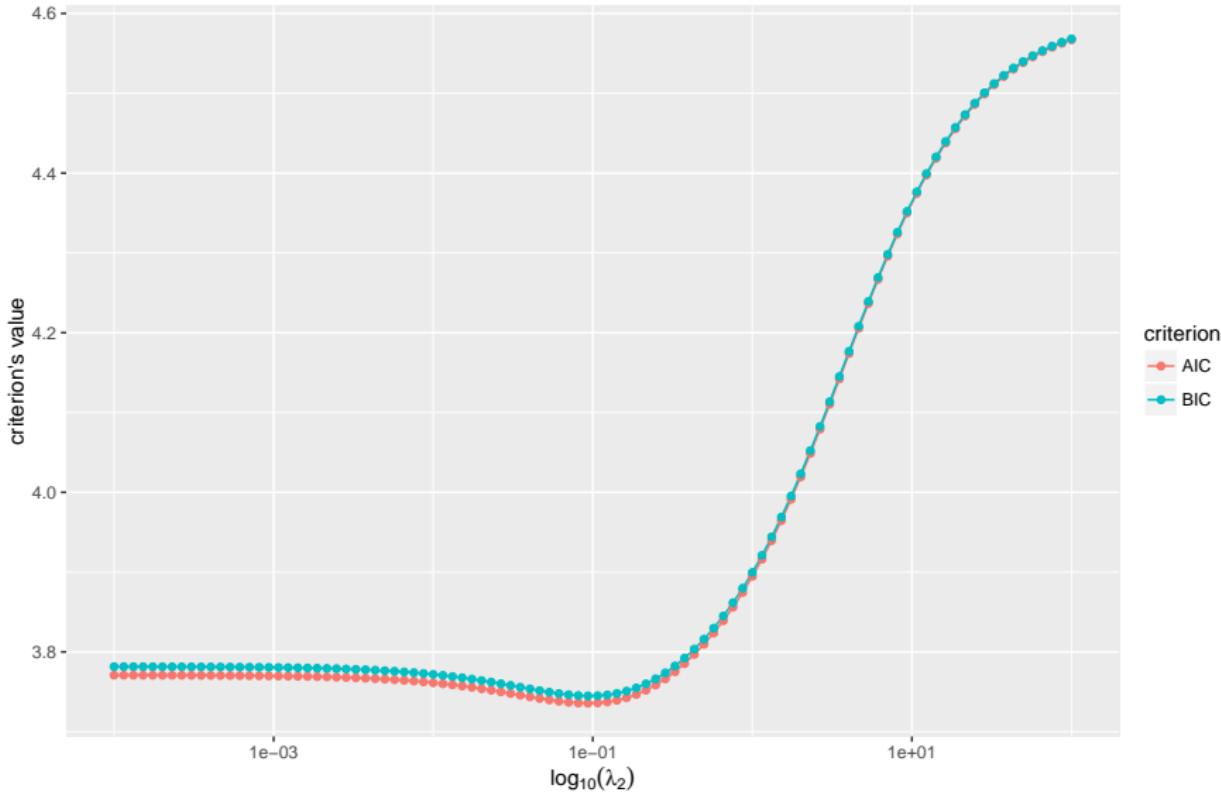
print(head(crit$beta.min))

## 6 x 2 Matrix of class "dgeMatrix"
##          AIC        BIC
## lcavol 0.8635928 0.8635928
## lweight 0.4152738 0.4152738
## age    -0.1520864 -0.1520864
## lbph   0.2911599 0.2911599
## svi    0.4092488 0.4092488
## lcp   -0.2047139 -0.2047139
```

# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$  (plot)

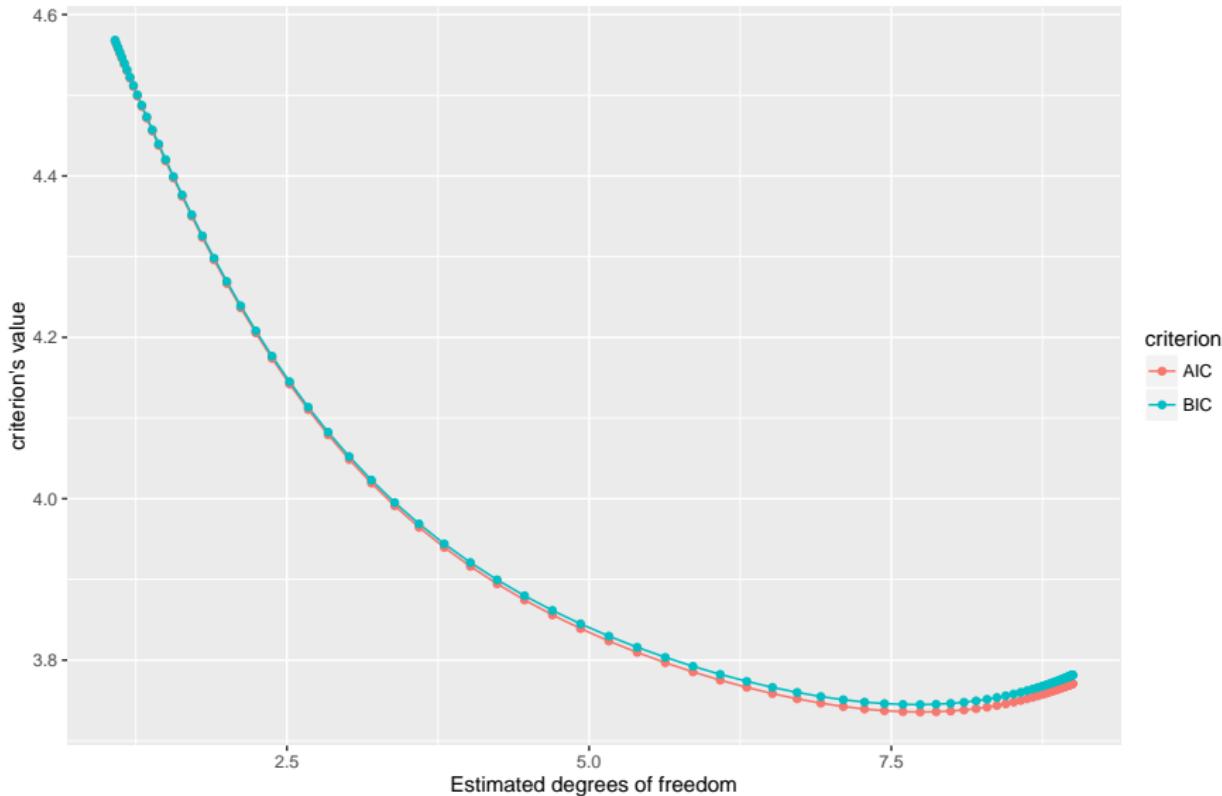
Information Criteria for a ridge fit



# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$  (plot 2)

Information Criteria for a ridge fit



## Validation croisée

La validation croisée se parallélise facilement et ne prend que peu de temps sur un petit jeu de données

```
system.time(loo <- quadrupen::crossval(x.train,y.train,"ridge",K=n,normalize=FALSE)

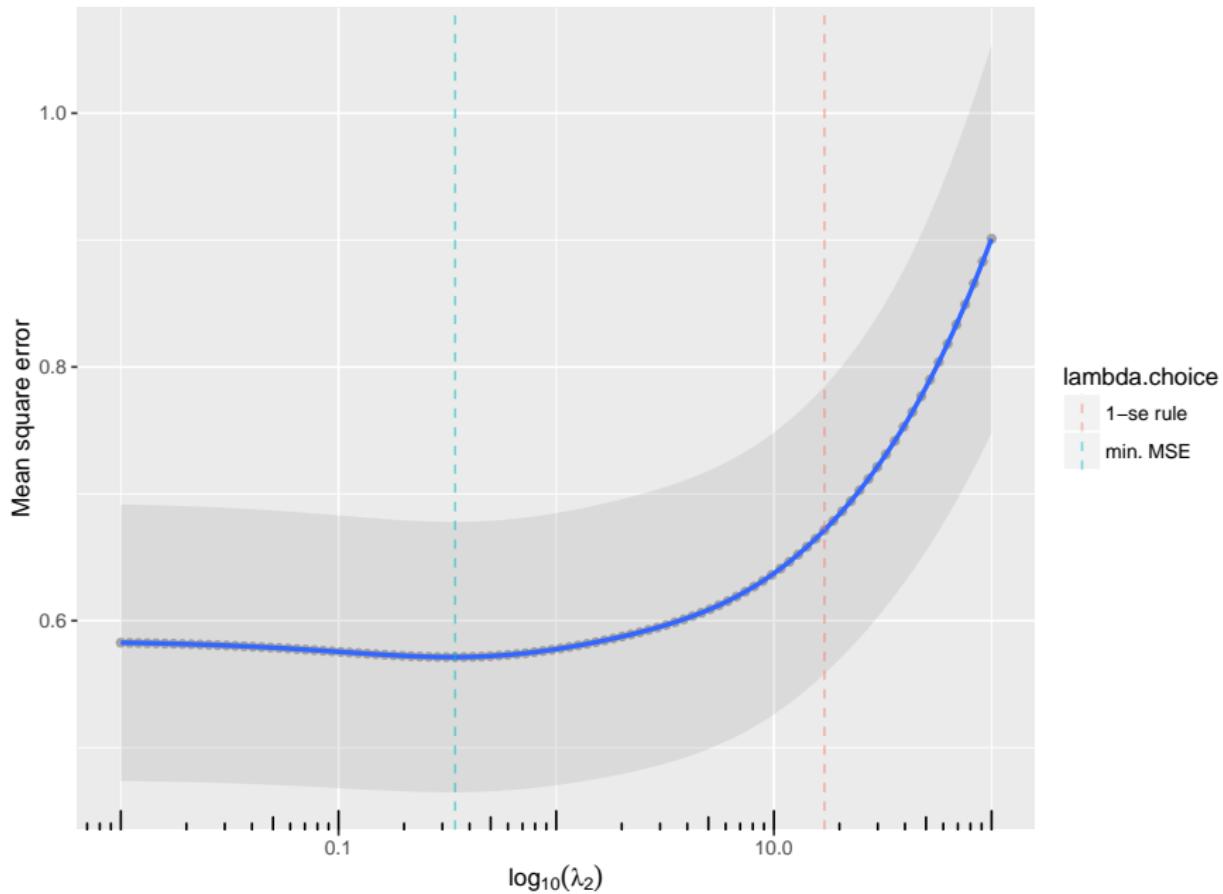
## 
## CROSS-VALIDATION FOR ridge REGULARIZER
##
## 67-fold CV on the lambda2 grid.
##     user   system elapsed
## 0.708   0.256   0.330

system.time(CV10 <- quadrupen::crossval(x.train,y.train,"ridge",K=10,normalize=FALSE)

## 
## CROSS-VALIDATION FOR ridge REGULARIZER
##
## 10-fold CV on the lambda2 grid.
##     user   system elapsed
## 0.472   0.636   0.260
```

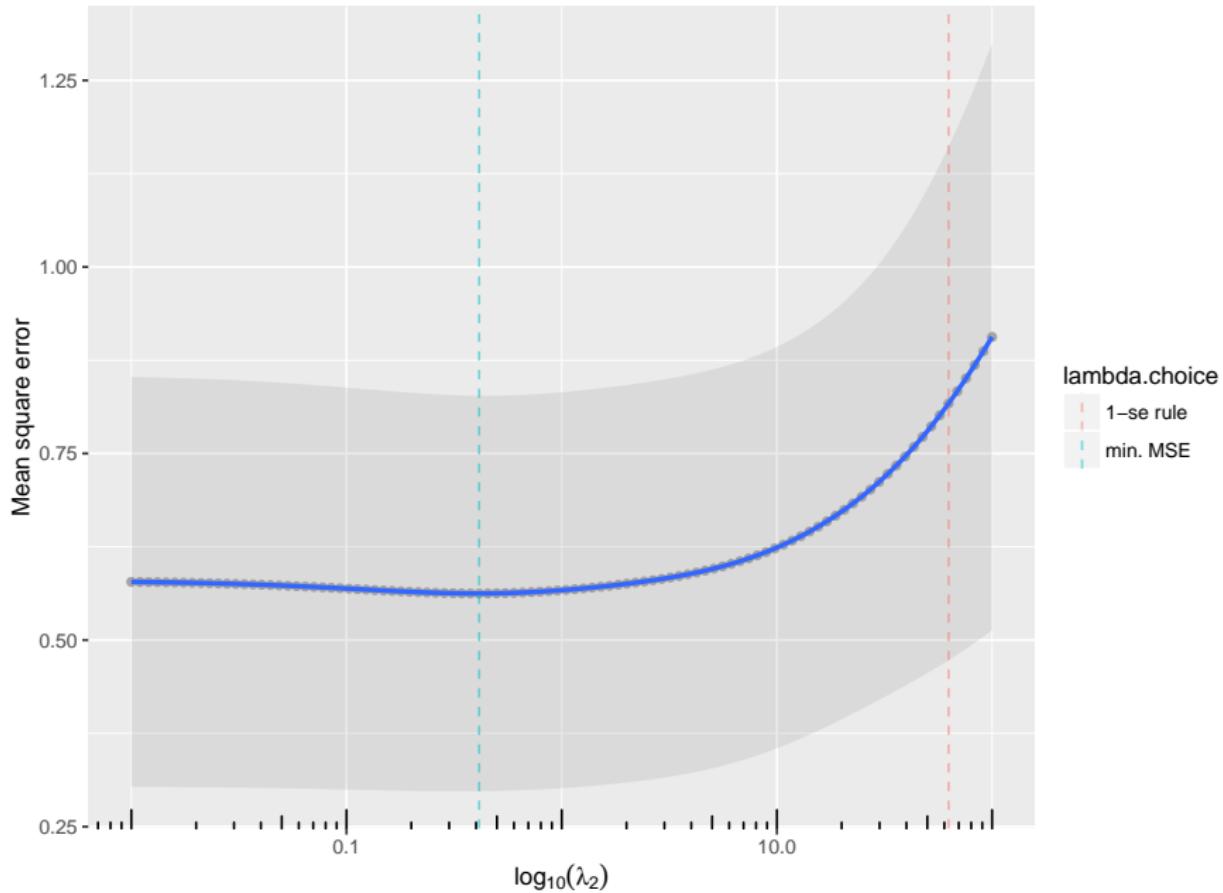
# Validation croisée ("leave one out")

LOO CV error



# Validation croisée ("ten fold")

10-fold CV error



# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Définition de l'estimateur

Propriétés et résolution pratique

Choix du paramètre de régularisation

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Définition de l'estimateur

Propriétés et résolution pratique

Choix du paramètre de régularisation

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Le Lasso

Least Absolute Shrinkage and Selection Operator

Limite de la ridge

La Ridge régularise... mais on aimerait également sélectionner les prédicteurs significatifs.

Idée

Proposer une contrainte qui force la **parcimonie** (en forçant des entrées de  $\hat{\beta}$  à zéro).

Le Lasso comme problème d'optimisation

Le Lasso estime  $\hat{\beta}^{\text{lasso}}$  via

$$\underset{\beta \in \mathbb{R}^{p+1}}{\text{minimize}} \text{RSS}(\beta), \quad \text{s.t. } \sum_{j=1}^p |\beta_j| \leq s,$$

où  $s$  est un niveau de régularisation.

# Le Lasso

Least Absolute Shrinkage and Selection Operator

Limite de la ridge

La Ridge régularise... mais on aimerait également sélectionner les prédicteurs significatifs.

Idée

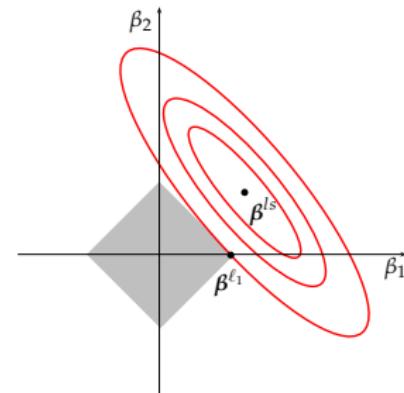
Proposer une contrainte qui force la **parcimonie** (en forçant des entrées de  $\hat{\beta}$  à zéro).

Le Lasso comme problème d'optimisation

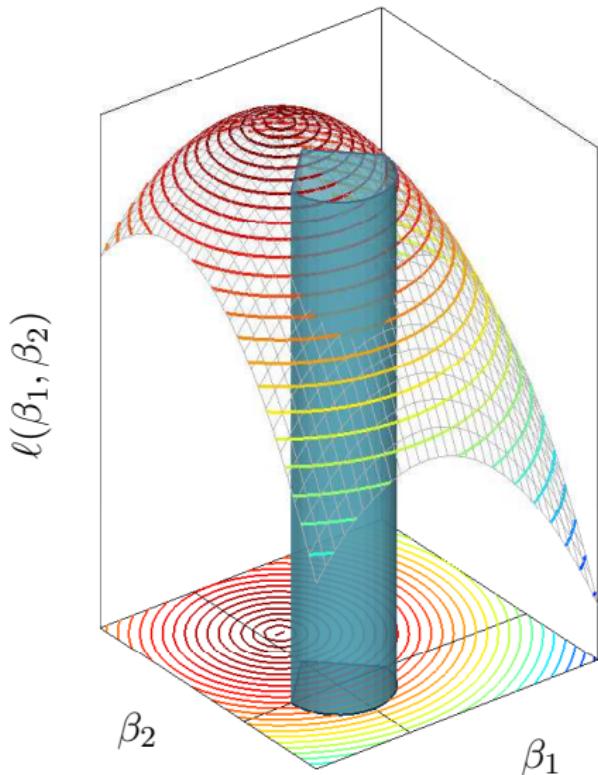
Le Lasso estime  $\hat{\beta}^{\text{lasso}}$  via

$$\underset{\beta \in \mathbb{R}^{p+1}}{\text{minimize}} \text{RSS}(\beta), \quad \text{s.t. } \sum_{j=1}^p |\beta_j| \leq s,$$

où  $s$  est un niveau de régularisation.



# Interprétation géométrique de la parcimonie



$$\begin{aligned} & \underset{\beta_1, \beta_2}{\text{maximize}} \quad \ell(\beta_1, \beta_2) \\ \text{s.c.} \quad & \Omega(\beta_1, \beta_2) \leq c \end{aligned}$$

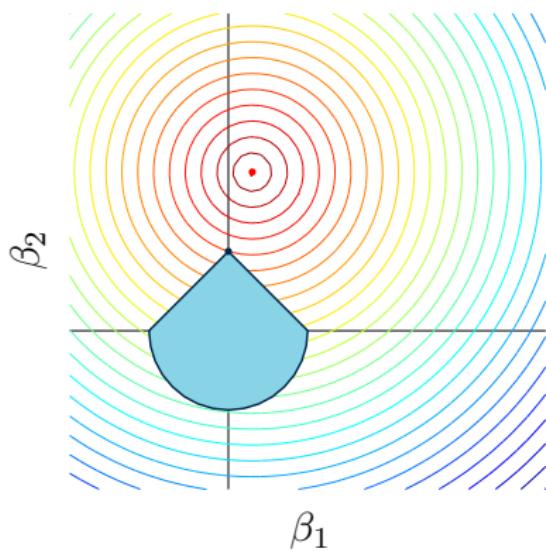
$\Updownarrow$

$$\begin{aligned} & \underset{\beta_1, \beta_2}{\text{minimize}} \quad -\ell(\beta_1, \beta_2) \\ \text{s.c.} \quad & \Omega(\beta_1, \beta_2) \leq c \end{aligned}$$

$\Updownarrow$

$$\underset{\beta_1, \beta_2}{\text{minimize}} -\ell(\beta_1, \beta_2) + \lambda \Omega(\beta_1, \beta_2)$$

# Interprétation géométrique de la parcimonie



$$\begin{aligned} & \underset{\beta_1, \beta_2}{\text{maximize}} \quad \ell(\beta_1, \beta_2) \\ \text{s.c.} \quad & \Omega(\beta_1, \beta_2) \leq c \end{aligned}$$

$\Updownarrow$

$$\begin{aligned} & \underset{\beta_1, \beta_2}{\text{minimize}} \quad -\ell(\beta_1, \beta_2) \\ \text{s.c.} \quad & \Omega(\beta_1, \beta_2) \leq c \end{aligned}$$

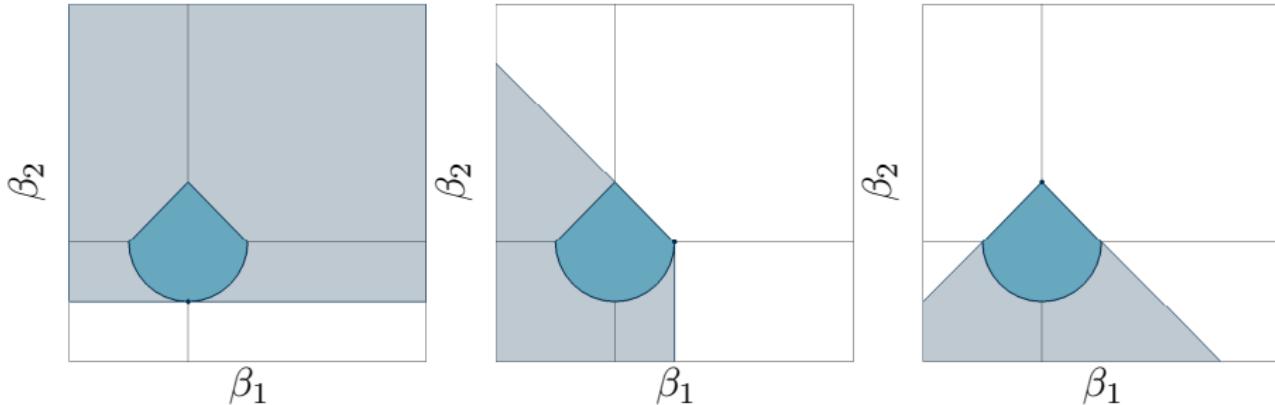
$\Updownarrow$

$$\underset{\beta_1, \beta_2}{\text{minimize}} -\ell(\beta_1, \beta_2) + \lambda \Omega(\beta_1, \beta_2)$$

# Interprétation géométrique de la parcimonie

Cône de dualité

généralise la notion de vecteur normal



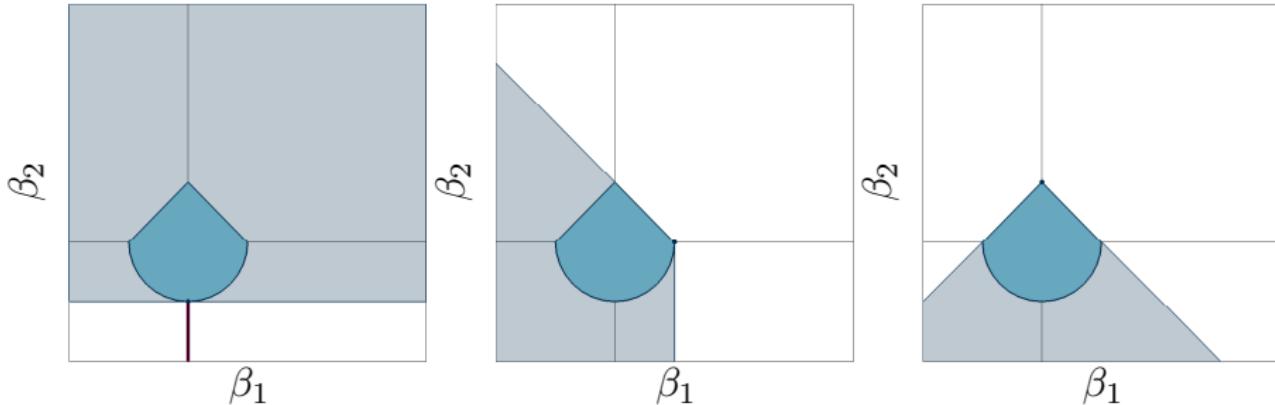
Soit  $C$  un ensemble convexe,

- $C^*(x_0) = \{y | y^T(x - x_0) \geq 0, x \in C\}$  est le cône dual en  $x_0$ ,
- $N_C(x_0) = \{y | y^T(x - x_0) \leq 0, x \in C\}$  est le cône normal,

# Interprétation géométrique de la parcimonie

Cône de dualité

généralise la notion de vecteur normal



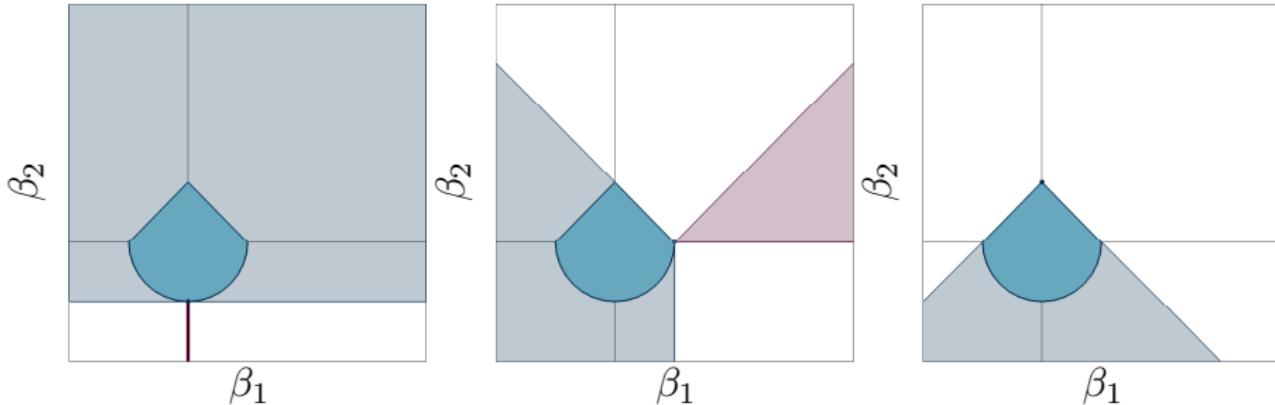
Soit  $C$  un ensemble convexe,

- $C^*(x_0) = \{y | y^T(x - x_0) \geq 0, x \in C\}$  est le cône dual en  $x_0$ ,
- $N_C(x_0) = \{y | y^T(x - x_0) \leq 0, x \in C\}$  est le cône normal,

# Interprétation géométrique de la parcimonie

Cône de dualité

généralise la notion de vecteur normal



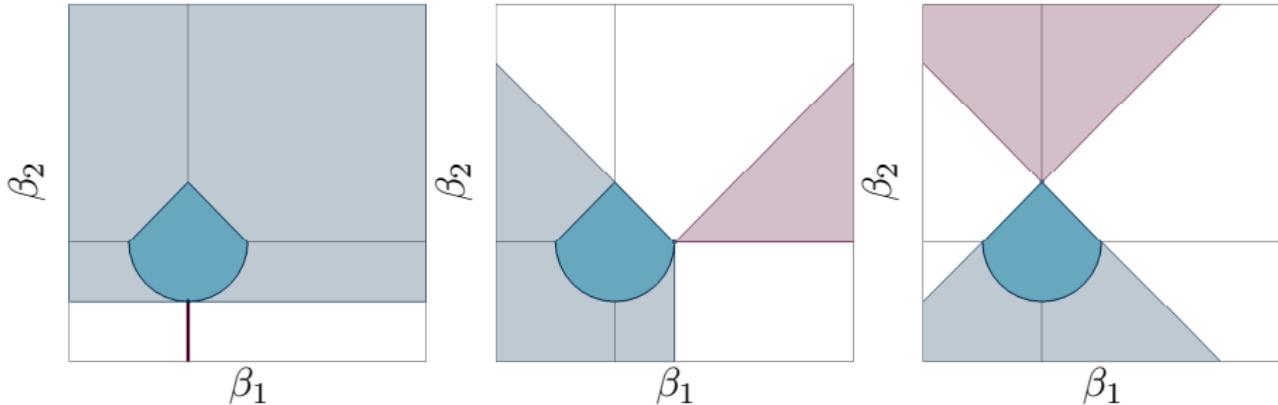
Soit  $C$  un ensemble convexe,

- $C^*(x_0) = \{y | y^T(x - x_0) \geq 0, x \in C\}$  est le cône dual en  $x_0$ ,
- $N_C(x_0) = \{y | y^T(x - x_0) \leq 0, x \in C\}$  est le cône normal,

# Interprétation géométrique de la parcimonie

Cône de dualité

généralise la notion de vecteur normal



Soit  $C$  un ensemble convexe,

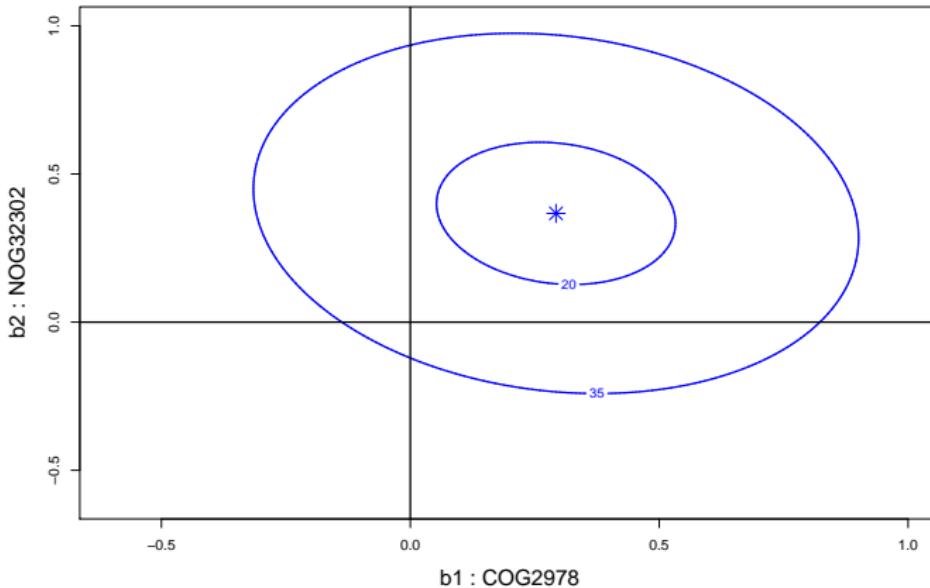
- $C^*(x_0) = \{y | y^T(x - x_0) \geq 0, x \in C\}$  est le cône dual en  $x_0$ ,
- $N_C(x_0) = \{y | y^T(x - x_0) \leq 0, x \in C\}$  est le cône normal,

Forme des cônes  $\Rightarrow$  pattern de parcimonie

# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

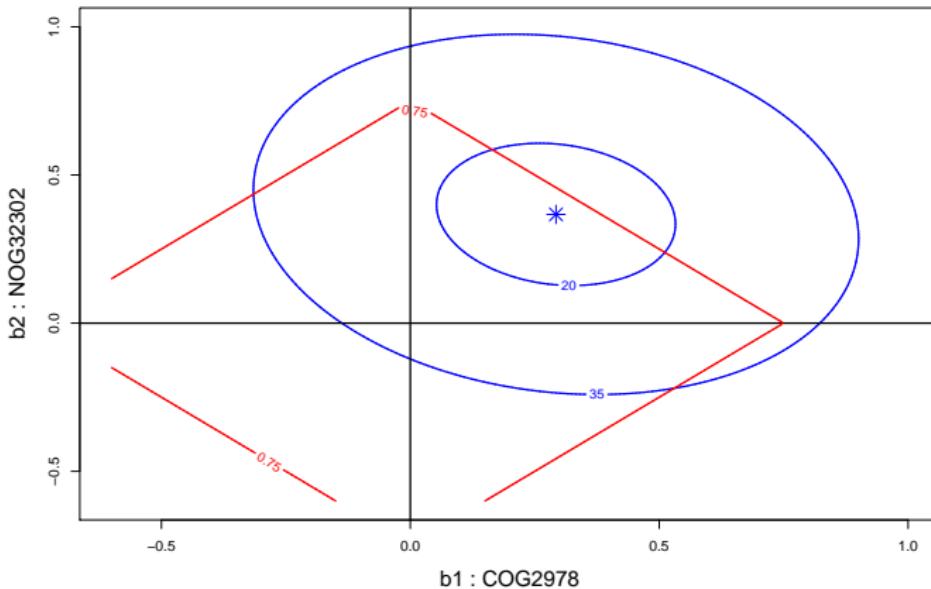
$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{pas de contrainte}$$



# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

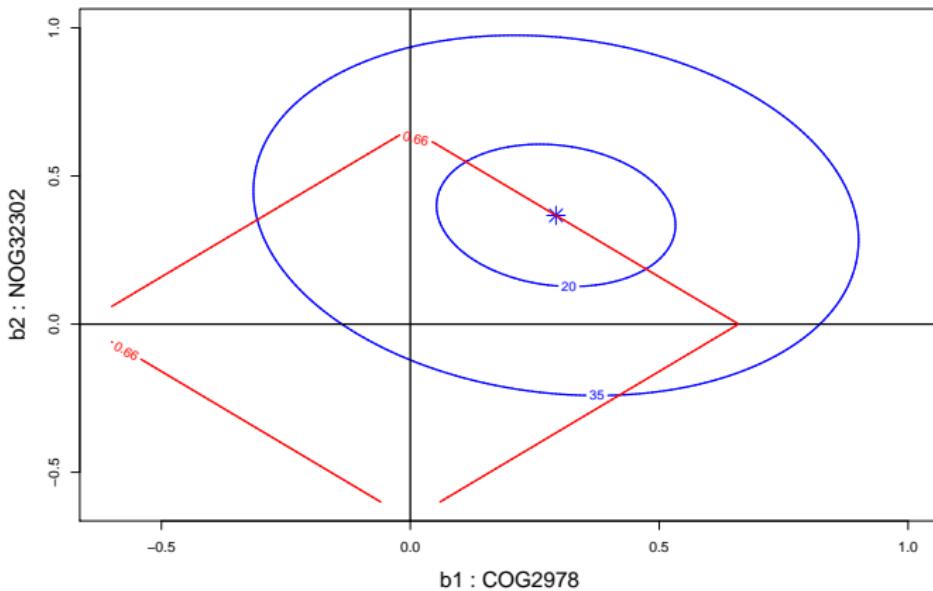
$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{s.c. } |\beta_1| + |\beta_2| < 0.75$$



# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

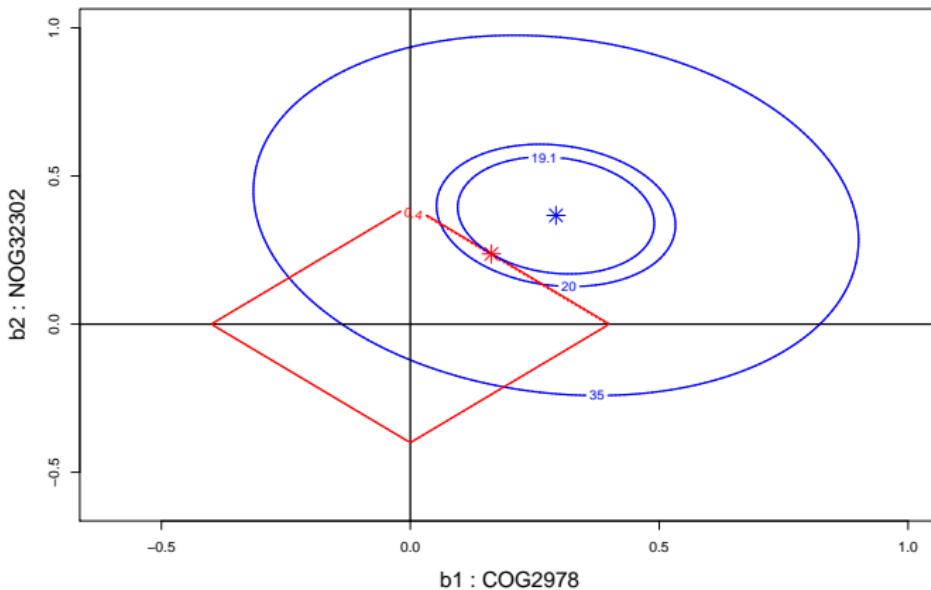
$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{s.c. } |\beta_1| + |\beta_2| < 0.66$$



# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

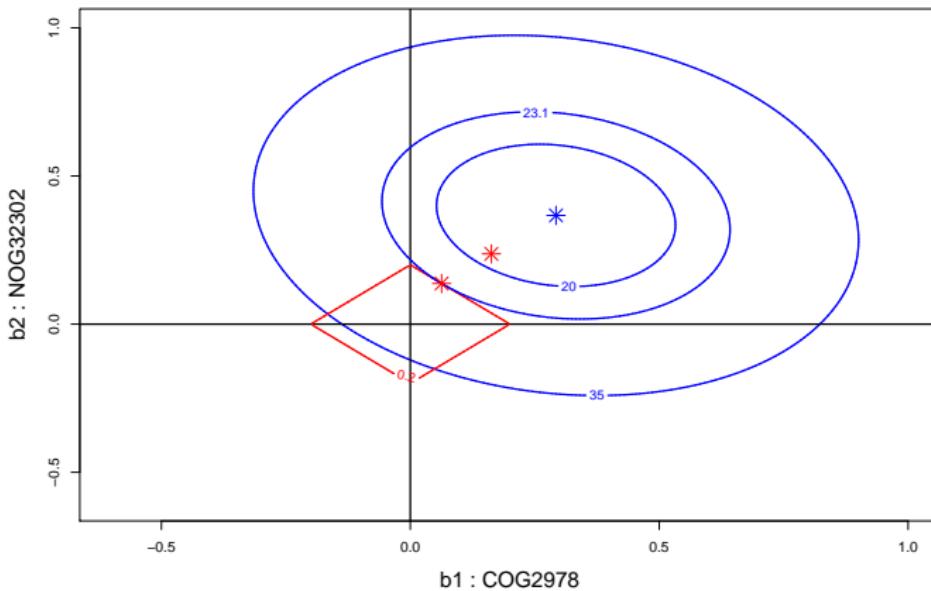
$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{s.c. } |\beta_1| + |\beta_2| < 0.4$$



# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

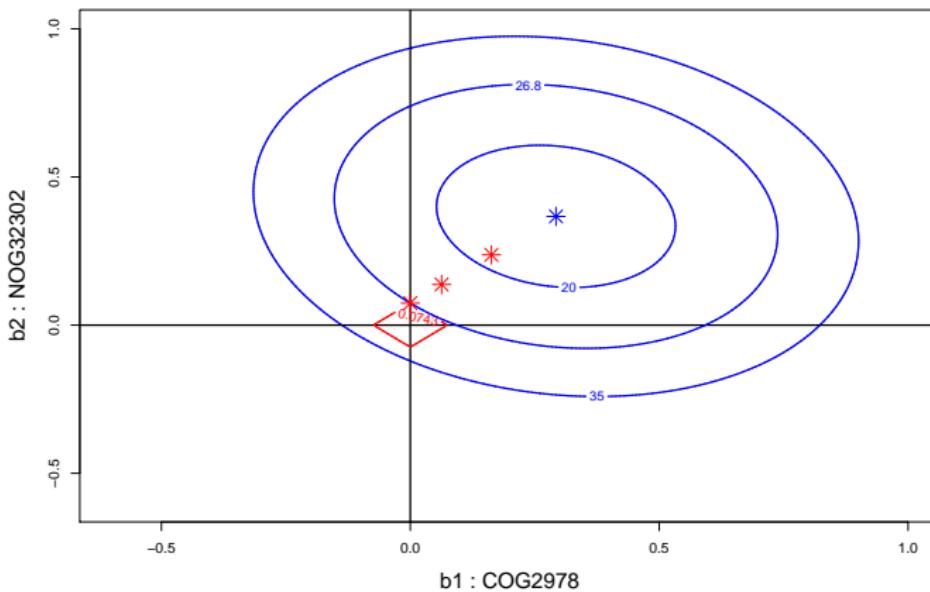
$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{s.c. } |\beta_1| + |\beta_2| < 0.2$$



# Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

$$\sum_{i=1}^n (y_i - x_i^1 \beta_1 - x_i^2 \beta_2)^2, \quad \text{s.c. } |\beta_1| + |\beta_2| < 0.0743$$



# Le Lasso comme méthode de régression pénalisée

On ne pénalise pas la constante, donc

- ▶  $\hat{\beta}_0 = \bar{y}$ ,
- ▶ on centre  $y$  et  $x_j$ ,  $j = 1, \dots, p$ ,
- ▶ on normalise les prédicteurs avant d'ajuster,
- ▶ on renvoie  $\hat{\beta}$  dans l'échelle d'origine.

Résolution d'un problème d'optimisation convexe

$$\hat{\beta}^{\text{lasso}} = \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|y - X\beta\|^2 + \lambda \|\beta\|_1,$$

Pas de forme close, mais existe toujours et est unique lorsque  $X^T X$  est de plein rang.

~~ Le Lasso régularise et sélectionne les prédicteurs, mais n'a pas de solution explicite.

# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Définition de l'estimateur

**Propriétés et résolution pratique**

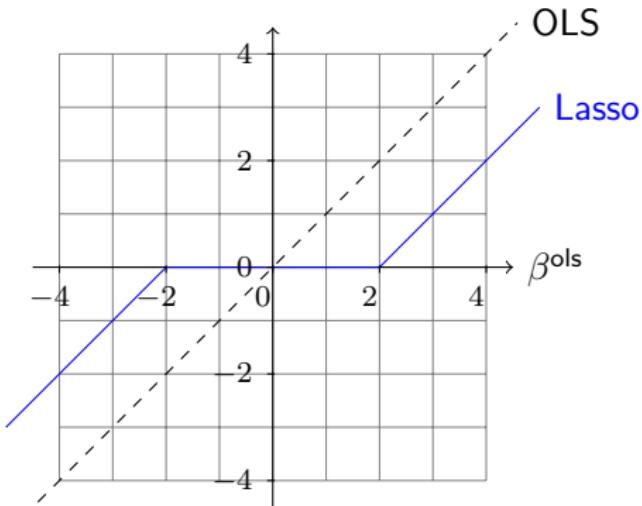
Choix du paramètre de régularisation

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

## Cas orthogonal et connexion avec l'OLS

À supposer que  $\mathbf{X}^\top \mathbf{X} = \mathbf{I}$ , alors



~~ correspond au "seuillage-doux"  $S_{\text{thres}}$  de Donoho et al.

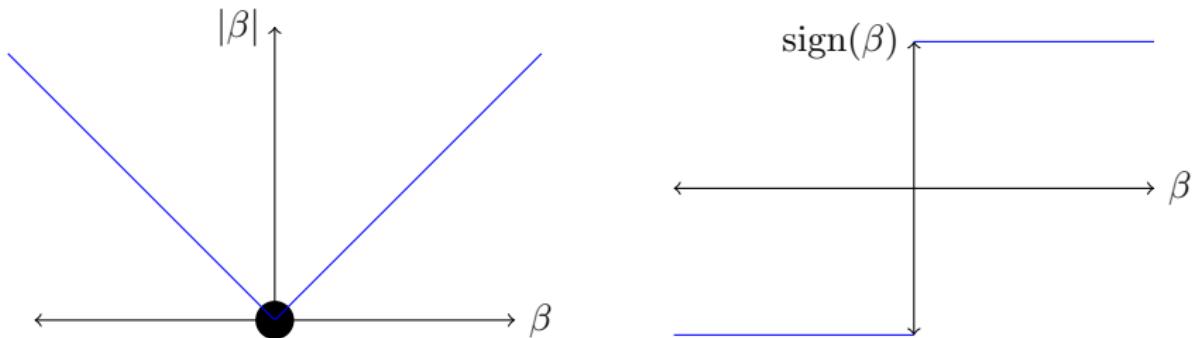
$$\hat{\beta}_j^{\text{lasso}} = \left( 1 - \frac{\lambda}{|\hat{\beta}_j^{\text{ols}}|} \right)^+ \\ = S_{\text{thres}}(\hat{\beta}_j^{\text{ols}}, \lambda), \hat{\beta}_j^{\text{ols}},$$

$$|\hat{\beta}_j^{\text{lasso}}| = (|\hat{\beta}_j^{\text{ols}}| - \lambda)^+ .$$

# Notion de sous-gradient et résolution

Un vecteur  $\beta$  est solution du Lasso si et seulement si

$$-\mathbf{X}^\top(\mathbf{X}\beta - \mathbf{y}) + \lambda\theta = \mathbf{0}, \quad \text{with} \quad \begin{cases} \theta_j = \text{sign}(\beta_j) & \text{if } \beta_j \neq 0, \\ \theta_j \in [-1, 1] & \text{if } \beta_j = 0. \end{cases}$$



 Boyd, S. and Vandenberghe, L. 2006. Convex optimization.

# L'algorithme de shooting



Fu, W., 1998.

Penalized regressions: the bridge vs. the lasso.

Soit

$$S(x, \lambda) = \text{sign}(x) \max(0, |x| - \lambda).$$

l'opérateur de seuillage doux.

1. Start with  $\hat{\beta} = \beta^{\text{ols}}$
2. For each  $j = 1, \dots, p$ , set

$$\hat{\beta}_j = S \left( \sum_{i=1}^n x_{ij} (y_i - \tilde{y}_i^{(j)}), \lambda \right) / \mathbf{x}_j^\top \mathbf{x}_j,$$

with  $\tilde{y}_i^{(j)} = \sum_{k \neq j} x_{ik} \hat{\beta}_k$ .

3. Repeat 2 until convergence

# LARs: Least angle regression

Une méthode populaire pour ajuster le Lasso

- 
- B. Efron, T. Hastie, I. Johnstone, R. Tibshirani, 2004.  
Least Angle Regression.

Algorithme efficace de calcul du chemin de solution

La solution du LARS consiste en une fonction décrivant  $\hat{\beta}$  pour chaque valeur de  $\lambda$ .

- ▶ construit un chemin *linéaire par morceau* de la solution en partant du vecteur nul,
- ▶ Coût proche de celui de l'OLS,
- ▶ bien adapté à la validation croisée.

# Lasso sur données cancer de la prostate I

## Calcul du chemin de solution du Lasso

```
library(quadrupen)
lasso.path <- quadrupen::lasso(x.train,y.train)
```

## Calcul de l'erreur de prédiction sur l'ensemble test

```
err <- colMeans((y.test-predict(lasso.path,x.test))^2)
```

On choisit  $\lambda^*$  minimisant cette erreur

```
lasso.path@lambda[which.min(err)]
## [1] 0.9291339
```

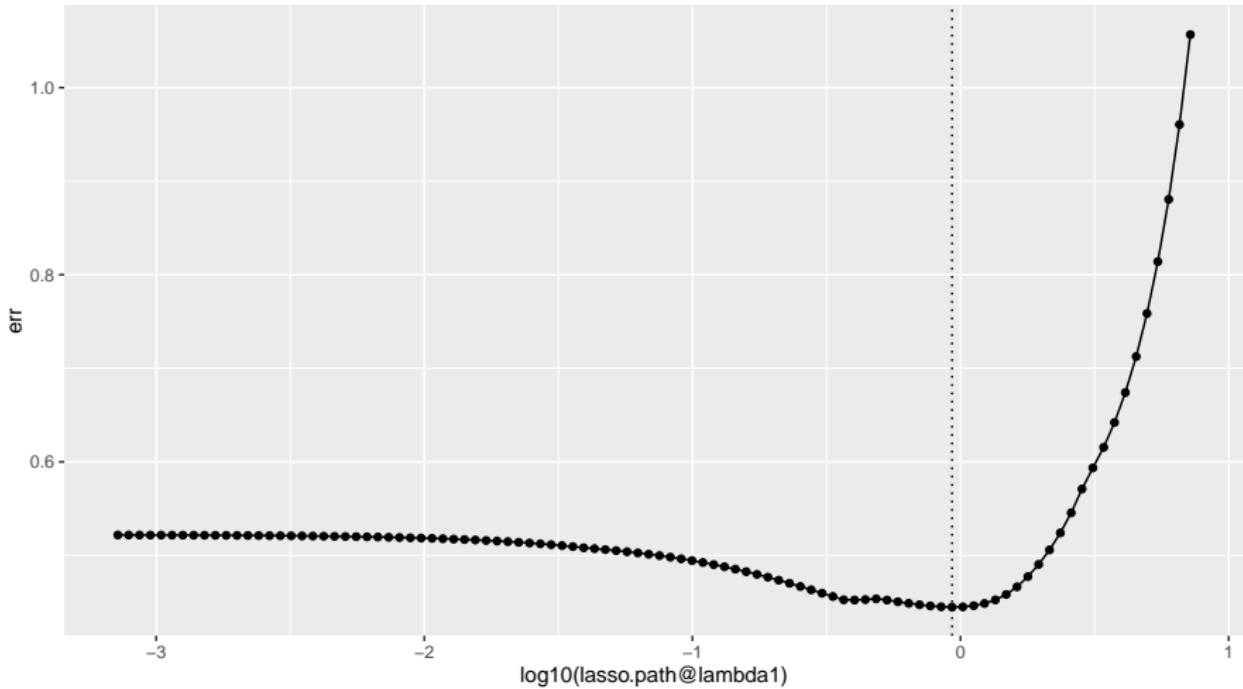
## Lasso sur données cancer de la prostate II

L'erreur de prédiction est significativement plus petite que pour l'OLS et la ridge, avec seulement 5 coefficients + la constante

```
err[which.min(err)]  
  
##      0.929  
## 0.4447334  
  
lasso.path@coefficients[which.min(err), ]  
  
##      lcavol     lweight         age       lbph        svi       lcp  
## 0.83764934 1.74070516 0.00000000 0.09308986 0.18471854 0.00000000  
##      gleason      pgg45  
## 0.00000000 0.07755607
```

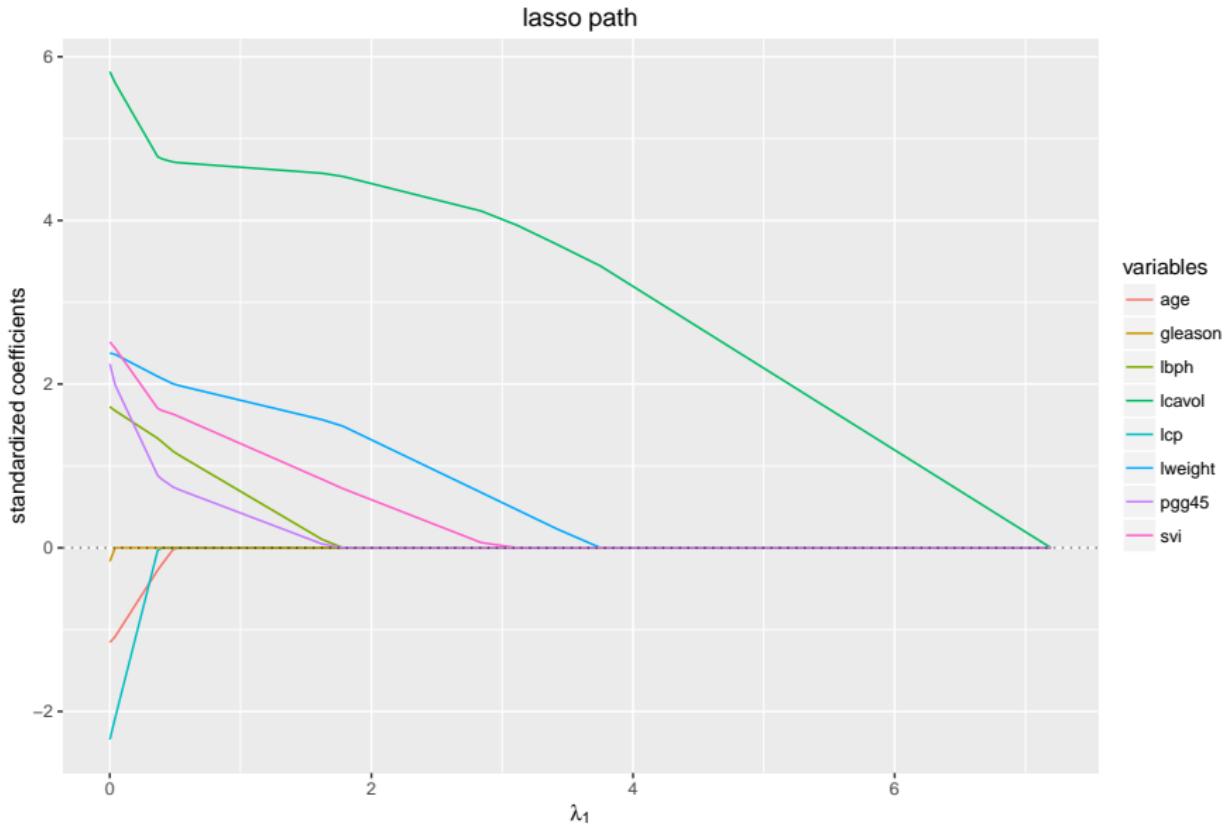
# Erreurs de prédition sur les données test

```
qplot(log10(lasso.path@lambda1), err) + geom_line() +  
geom_vline(xintercept=log10(lasso.path@lambda1[which.min(err)]), lty=3)
```



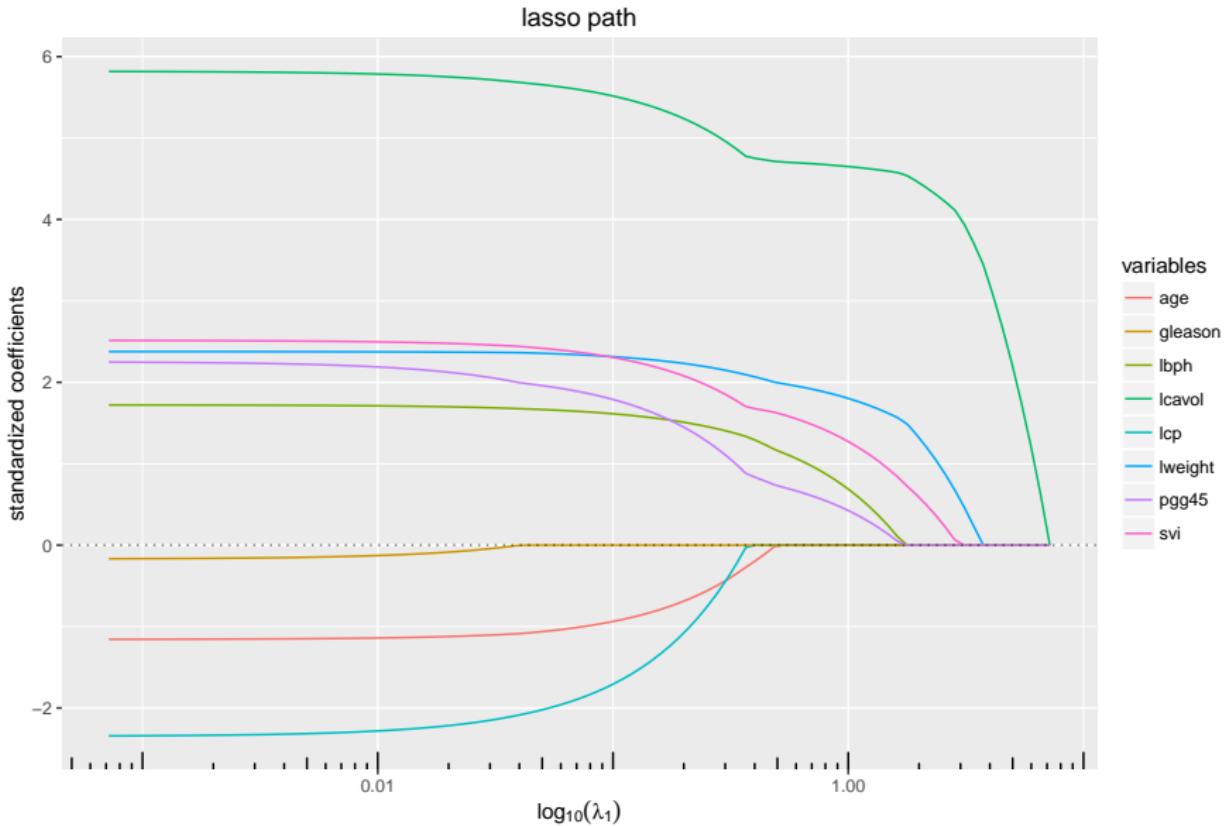
# Chemin de solution (en fonction de $\lambda$ )

```
plot(lasso.path, labels=colnames(x.train), log.scale=FALSE)
```



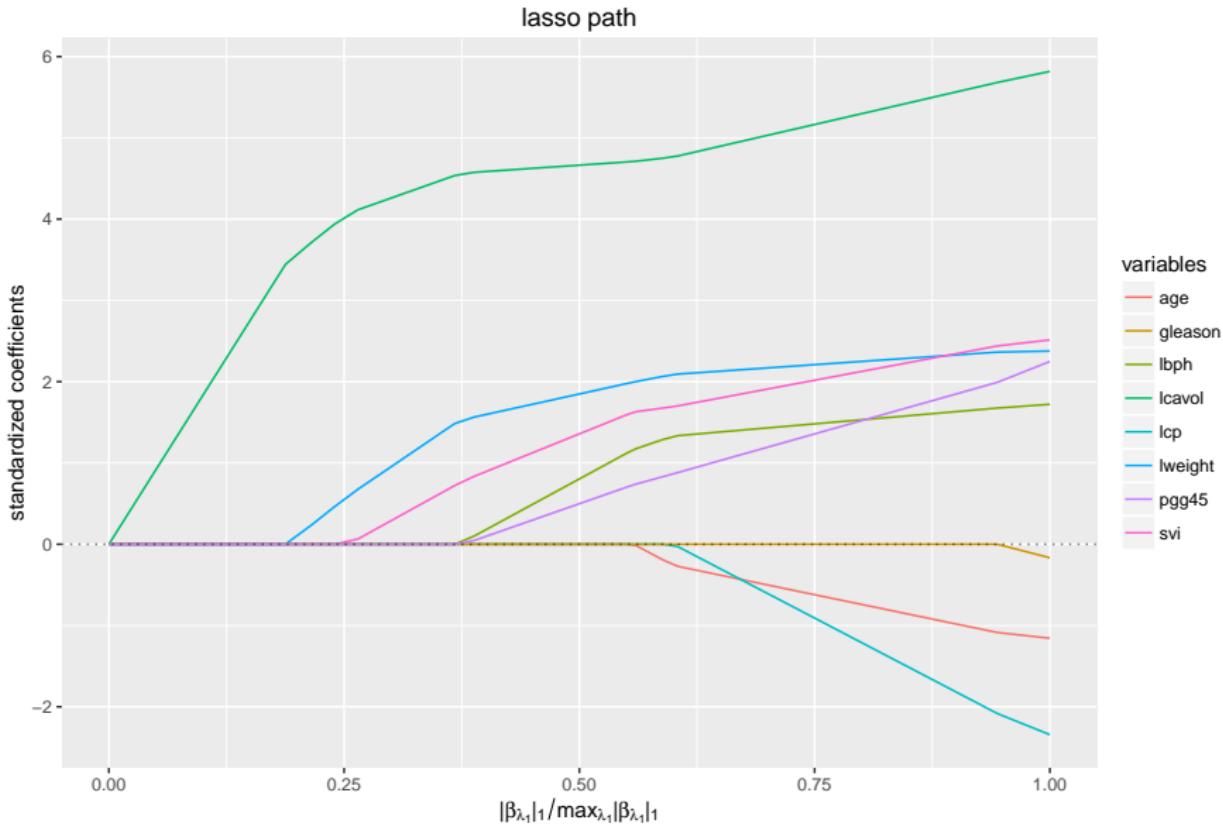
# Chemin de solution (en fonction de $\log_{10}(\lambda)$ )

```
plot(lasso.path, labels=colnames(x.train))
```



# Chemin de solution (proportion de régularisation $s$ )

```
plot(lasso.path, labels=colnames(x.train), xvar="fraction")
```



# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Définition de l'estimateur

Propriétés et résolution pratique

**Choix du paramètre de régularisation**

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

## Critères pénalisés

### Degrés de liberté du LASSO

On peut montrer que c'est simplement le nombre de prédicteurs actifs

$$\text{df}(\hat{\mathbf{y}}_{\lambda}^{\text{lasso}}) = \text{card}(\{j : \beta_j(\lambda) \neq 0\}) = |\mathcal{A}|.$$

- Akaike Information Criterion équivalent au  $C_p$  en régression

$$\text{AIC} = -2\text{loglik} + 2\frac{|\mathcal{A}|}{n},$$

- Bayesian Information Criterion

$$\text{BIC} = -2\text{loglik} + |\mathcal{A}| \log(n),$$

- modified BIC (lorsque  $n < p$ )

$$\text{mBIC} = -2\text{loglik} + |\mathcal{A}| \log(p),$$

- Extended BIC ajoute un prior sur le nombre de modèles de taille  $|\mathcal{A}|$

$$\text{eBIC} = -2\text{loglik} + |\mathcal{A}|(\log(n) + 2 \log(p)).$$

# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$

```
crit <- criteria(lasso.path, plot=FALSE)
print(head(crit$criterion), digits=3)

##      AIC    BIC    GCV df lambda fraction
## 1 4.60 4.60 0.0208  1    7.19  0.0000
## 2 4.53 4.53 0.0184  2    6.55  0.0192
## 3 4.44 4.45 0.0168  2    5.97  0.0367
## 4 4.37 4.37 0.0156  2    5.44  0.0527
## 5 4.30 4.30 0.0145  2    4.96  0.0672
## 6 4.23 4.24 0.0136  2    4.52  0.0805

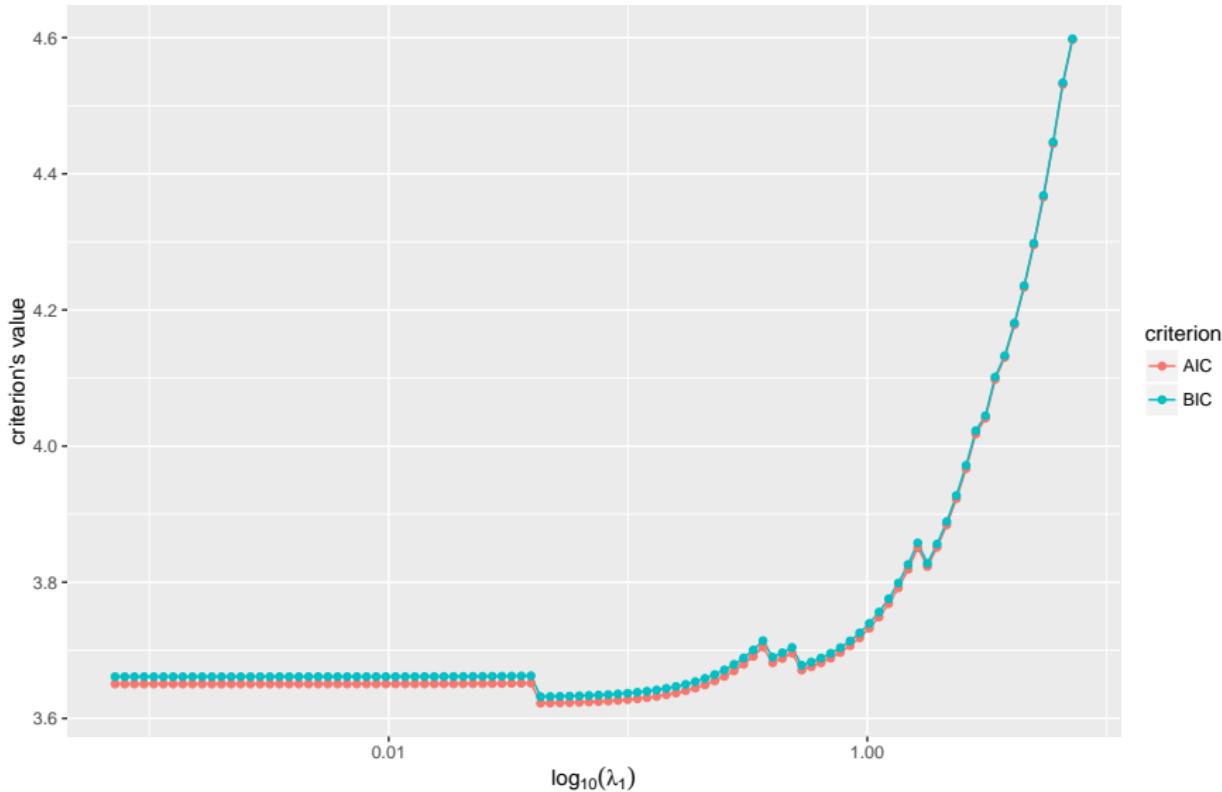
print(head(crit$beta.min))

## 6 x 2 sparse Matrix of class "dgCMatrix"
##          AIC        BIC
## lcavol 1.0203631 1.0203631
## lweight 2.2474028 2.2474028
## age     -1.1631499 -1.1631499
## lbph    0.2061171 0.2061171
## svi     0.3400045 0.3400045
## lcp    -0.2573030 -0.2573030
```

# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$  (plot)

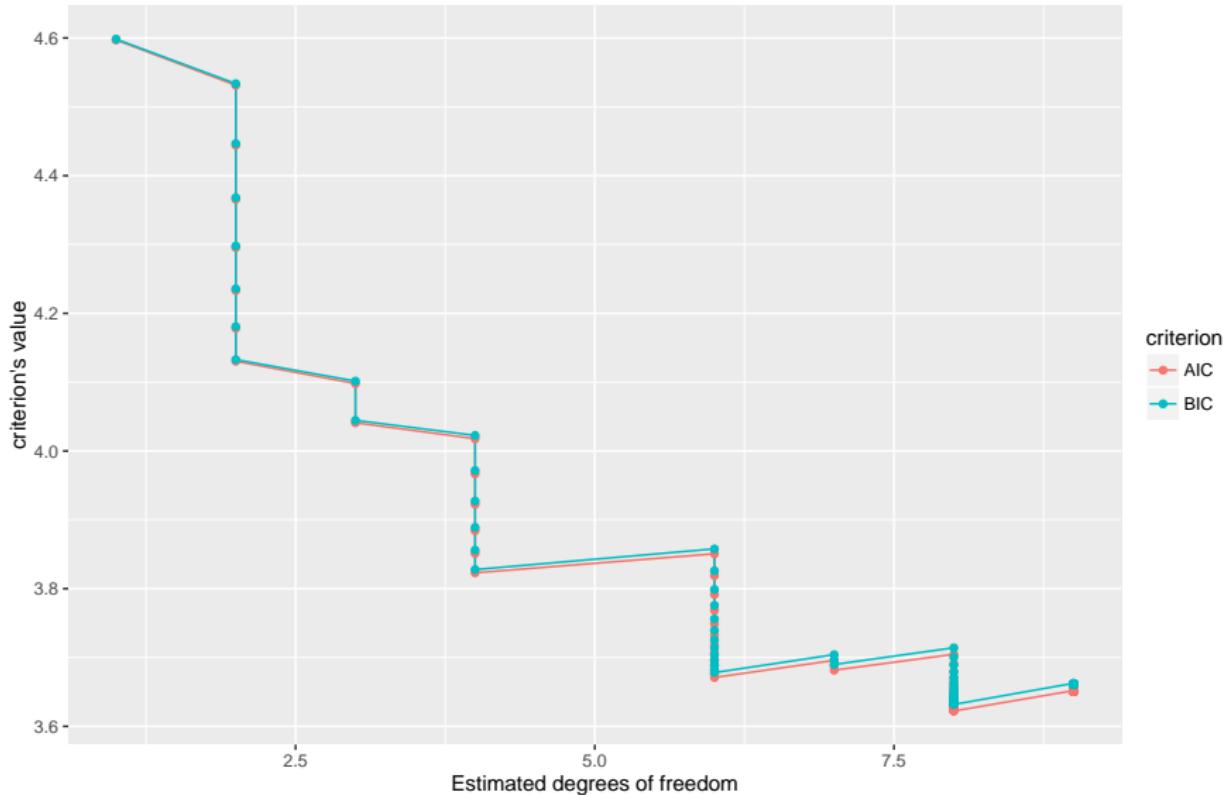
Information Criteria for a lasso fit



# Cancer de la prostate

Calcul de l' AIC/BIC en estimant  $\sigma$  (plot 2)

Information Criteria for a lasso fit



## Validation croisée

La validation croisée se parallélise facilement et ne prend que peu de temps sur un petit jeu de données

```
system.time(loo <- crossval(x.train,y.train,"lasso",K=n,normalize=FALSE))

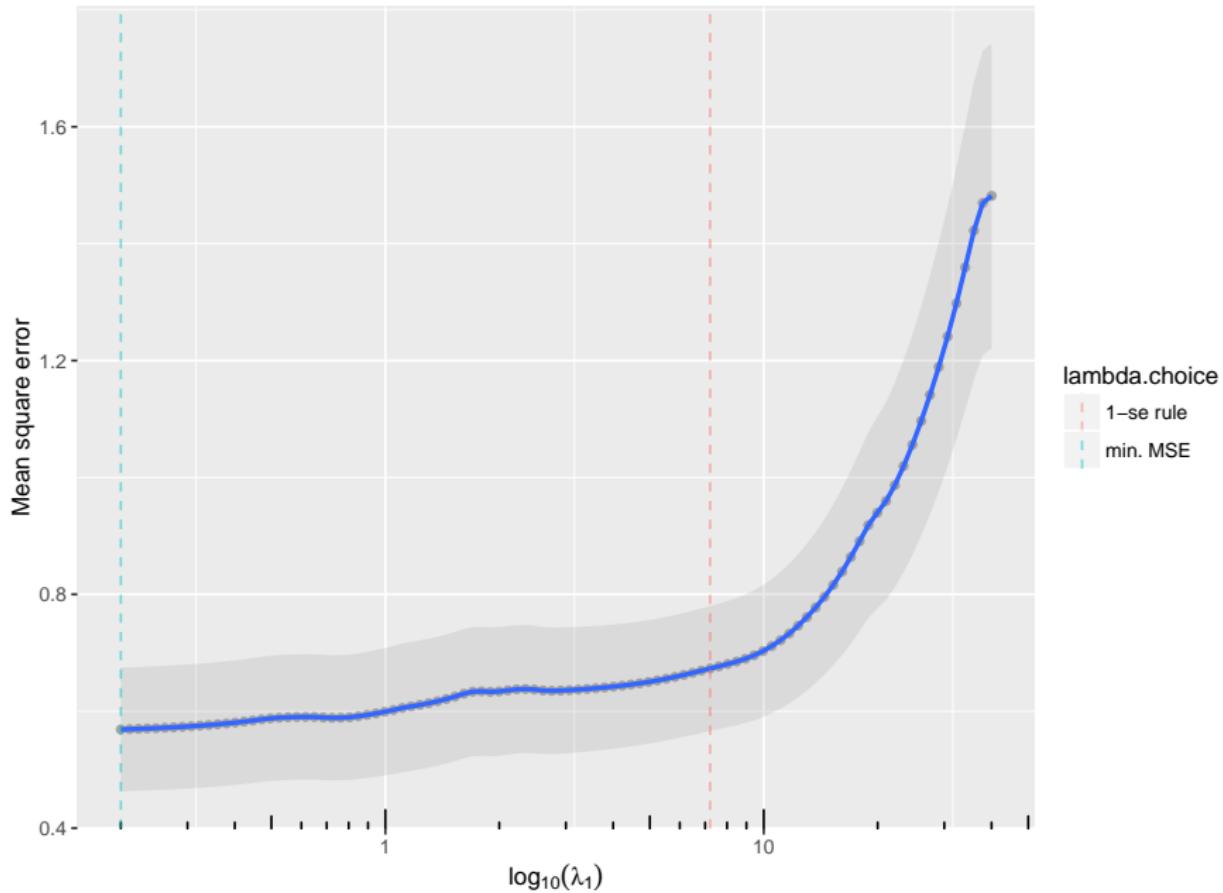
##
## CROSS-VALIDATION FOR lasso REGULARIZER
##
## 67-fold CV on the lambda1 grid, lambda2 is fixed.
##     user   system elapsed
## 0.724   0.236   0.343
```

```
system.time(CV10 <- crossval(x.train,y.train,"lasso",K=10,normalize=FALSE))

##
## CROSS-VALIDATION FOR lasso REGULARIZER
##
## 10-fold CV on the lambda1 grid, lambda2 is fixed.
##     user   system elapsed
## 0.472   0.632   0.258
```

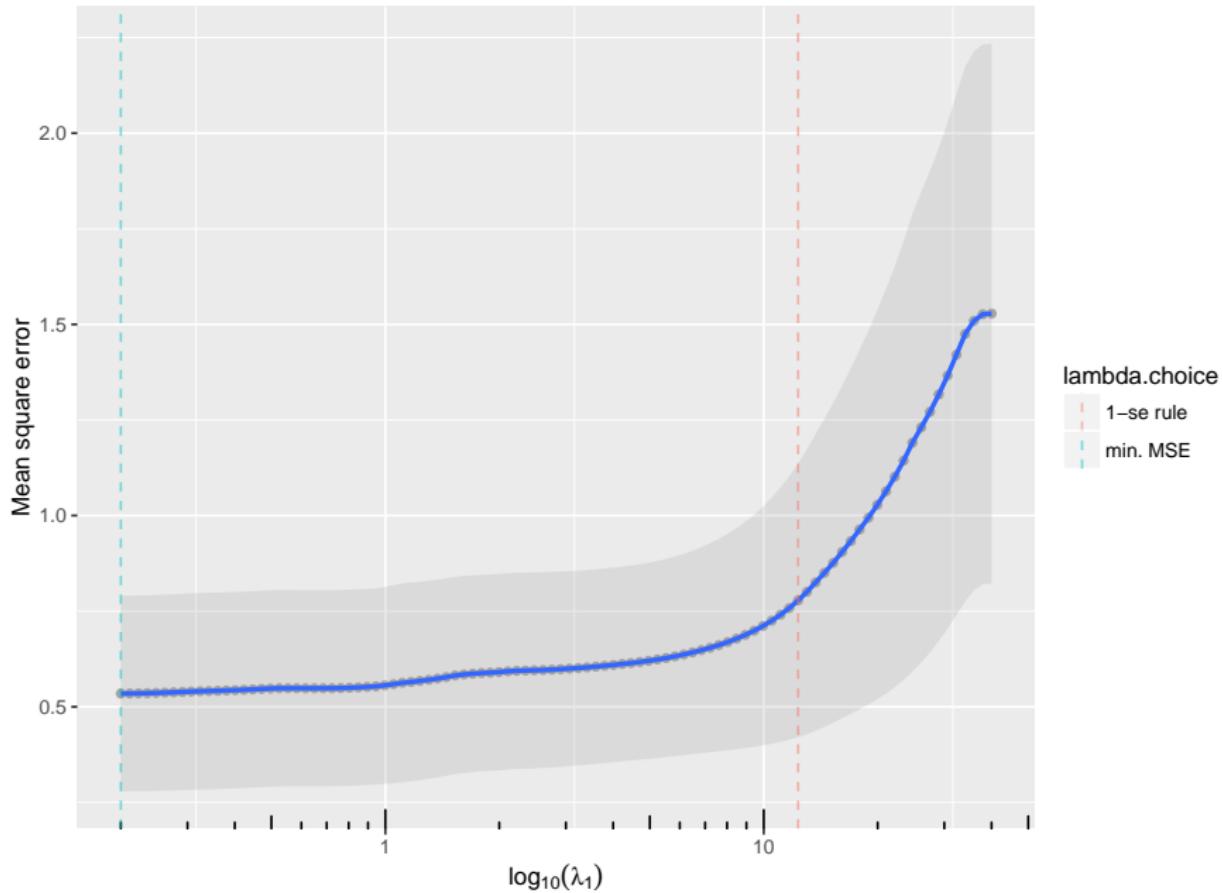
# Validation croisée ("leave one out")

LOO CV error



# Validation croisée ("ten fold")

10-fold CV error



# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

## Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

**Variations autour du Lasso**

Modèles graphiques gaussiens parcimonieux

# Bridge regression

A simple interpretable model: Gaussian linear regression

Assume a linear relationship between the features  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_p)$  and the outcome  $\mathbf{y} = (y_1, \dots, y_n)$  plus an iid Gaussian noise:

$$\mathbf{y} = \mu \mathbf{1}_n + \mathbf{X}\beta^* + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_n, \sigma^2 \mathbf{I}_n).$$

The bridge estimator

Force  $\hat{\beta}$  to live in balls associated with the  $\ell_\gamma$ -norms:

$$\hat{\beta}_{\lambda, \gamma} = \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{such that} \quad \|\beta\|_\gamma^\gamma \leq c$$

where  $c$  is a constant.

For  $\gamma = 1$ , we get Lasso.

For  $\gamma = 2$ , we get Ridge.

For  $\gamma < 1$ , we get SCAD.

For  $\gamma > 2$ , we get Bridge.

# Bridge regression

A simple interpretable model: Gaussian linear regression

Assume a linear relationship between the features  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_p)$  and the outcome  $\mathbf{y} = (y_1, \dots, y_n)$  plus an iid Gaussian noise:

$$\mathbf{y} = \mu \mathbf{1}_n + \mathbf{X}\beta^* + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_n, \sigma^2 \mathbf{I}_n).$$

- ▶ OLS would fail in the  $n < p$  setup or in presence of highly correlated features
- ▶ Bridge regression regularizes by controlling the size of the coefficients

The bridge estimator

Force  $\hat{\beta}$  to live in balls associated with the  $\ell_\gamma$ -norms:

$$\hat{\beta}_{\lambda, \gamma} = \arg \min_{\beta \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 \quad \text{such that} \quad \|\beta\|_\gamma^\gamma \leq c$$

## Bridge regression

A simple interpretable model: Gaussian linear regression

Assume a linear relationship between the features  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_p)$  and the outcome  $\mathbf{y} = (y_1, \dots, y_n)$  plus an iid Gaussian noise:

$$\mathbf{y} = \mu \mathbf{1}_n + \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_n, \sigma^2 \mathbf{I}_n).$$

The bridge estimator

Force  $\hat{\boldsymbol{\beta}}$  to live in balls associated with the  $\ell_\gamma$ -norms:

$$\hat{\boldsymbol{\beta}}_{\lambda, \gamma} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad \text{such that} \quad \|\boldsymbol{\beta}\|_\gamma^\gamma \leq c$$

The Lagrangian form is

$$\hat{\boldsymbol{\beta}}_{\lambda, \gamma} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_\gamma^\gamma.$$

## Bridge regression

A simple interpretable model: Gaussian linear regression

Assume a linear relationship between the features  $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_p)$  and the outcome  $\mathbf{y} = (y_1, \dots, y_n)$  plus an iid Gaussian noise:

$$\mathbf{y} = \mu \mathbf{1}_n + \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}_n, \sigma^2 \mathbf{I}_n).$$

The bridge estimator

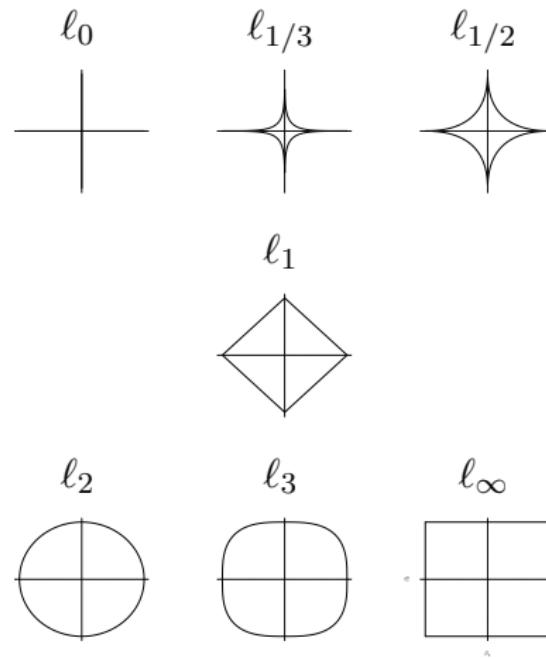
Force  $\hat{\boldsymbol{\beta}}$  to live in balls associated with the  $\ell_\gamma$ -norms:

$$\hat{\boldsymbol{\beta}}_{\lambda, \gamma} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 \quad \text{such that} \quad \|\boldsymbol{\beta}\|_\gamma^\gamma \leq c$$

The Lagrangian form is

$$\hat{\boldsymbol{\beta}}_{\lambda, \gamma} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_\gamma^\gamma.$$

# Bridge regression



**Figure:** Contours of the feasible sets for various  $\gamma$  when  $\beta \in \mathbb{R}^2$ .

# Numerical illustration

True model mimicking the block-wise structure between SNP for the predictors

Dispatched the true parameters in 5 groups

$$\boldsymbol{\beta}^* = \left( \underbrace{0.25, \dots, 0.25}_{p/4 \text{ times}}, \underbrace{1, \dots, 1}_{p/8 \text{ times}}, \underbrace{-0.25, \dots, -0.25}_{p/4 \text{ times}}, \underbrace{-1, \dots, -1}_{p/8 \text{ times}}, \underbrace{0.25, \dots, 0.25}_{p/4 \text{ times}} \right).$$

Faithful block-wise pattern between the predictors

We let  $\mathbf{x}_i \sim \mathcal{N}(\mathbf{0}_p, \Sigma)$  with

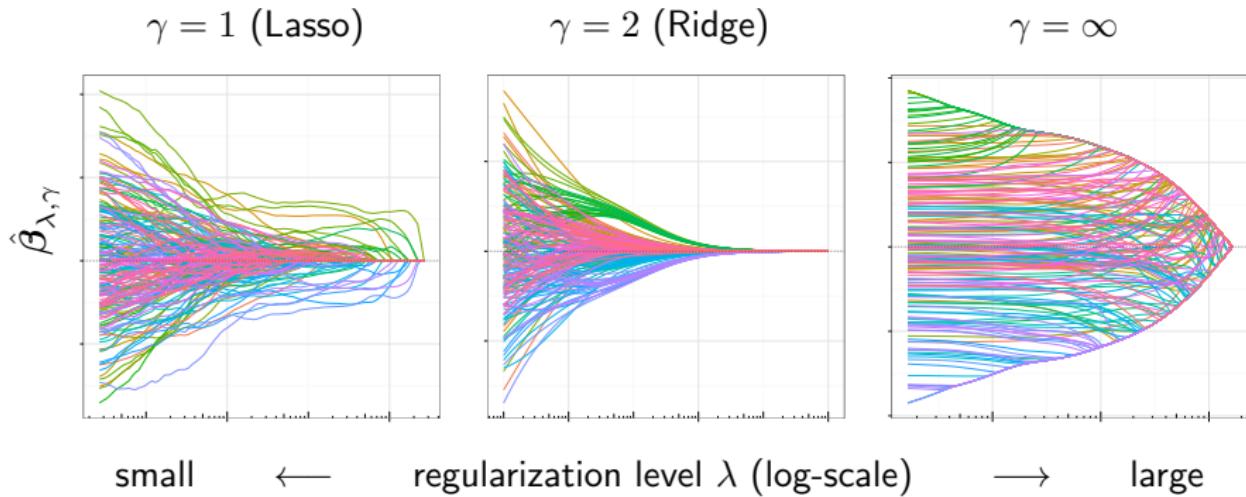
$$\Sigma = \begin{matrix} & \text{---} \\ \text{---} & \begin{matrix} .25 & & & \\ & .75 & & \\ & & .25 & \\ & & & .75 \end{matrix} \end{matrix}$$

$$\text{with } \Sigma_{ij} = \begin{cases} 1 & i = j, \\ .25 & i, j \in \text{blocks } \{1, 3, 5\}, i \neq j, \\ .75 & i, j \in \text{blocks } \{2, 4\}, i \neq j, \\ 0 & \text{otherwise.} \end{cases}$$

+ Variance  $\sigma^2$  of the noise chosen to met  $R^2 \approx 0.8$  on the training set.

# Numerical illustration

Bridge regularization paths ( $p = 192, n = 200$ )



**Figure:** Regularization paths for the bridge estimators

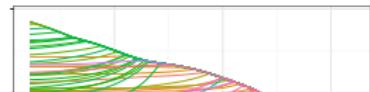
# Numerical illustration

Bridge regularization paths ( $p = 192, n = 200$ )

$\gamma = 1$  (Lasso)

$\gamma = 2$  (Ridge)

$\gamma = \infty$



More structure?

How inducing broader types of structures?

Idea: “blend”  $\Omega$  to introduce a broad variety of structures



small

←

regularization level  $\lambda$  (log-scale)

→

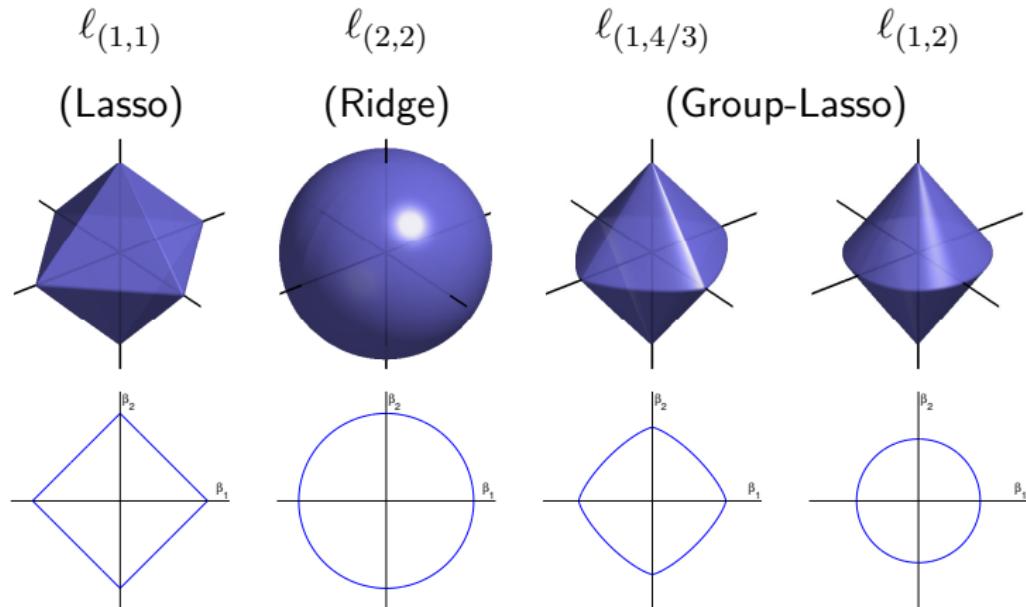
large

Figure: Regularization paths for the bridge estimators

# Accounting for group structures

## Group-Lasso norms

$$\|\boldsymbol{\beta}\|_{1,\eta} = \sum_{k=1}^K \omega_k \left( \sum_{j \in \mathcal{G}_k} \beta_j^\eta \right)^{1/\eta} = \sum_{k=1}^K \omega_k \|\boldsymbol{\beta}_{\mathcal{G}_k}\|_\eta.$$



# Account for direct relationships between the features (I)

By means of fusion and graph penalties

Prior knowledge: a proximity graph  $\mathcal{G}$  between the variables.

A weighted graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$  with vertices  $\mathcal{V} = \{1, \dots, p\}$  and edges  $\mathcal{E}$  weighted by values  $\mathcal{W} = \{\omega_{ij}, (i, j) \in \mathcal{E}\}$ .

Generalized “fusion”/total-variation penalty

$$\sum_{(i,j) \in \mathcal{E}} \omega_{ij} |\beta_i - \beta_j| = \|\mathbf{D}\boldsymbol{\beta}\|_1,$$

Generalized ridge/Laplacian penalty

$$\sum_{(i,j) \in \mathcal{E}} \omega_{ij} (\beta_i - \beta_j)^2 = \boldsymbol{\beta}^\top \mathbf{D}^T \mathbf{D} \boldsymbol{\beta} = \boldsymbol{\beta}^\top \mathbf{L} \boldsymbol{\beta} = \|\boldsymbol{\beta}\|_{\mathbf{L}}^2.$$

## Account for direct relationships between the features (II)

By means of fusion and graph penalties

Example: connected neighbors

$\mathcal{G}$  is a chain graph with edges  $\mathcal{E} = \{(1, 2), (2, 3), \dots, (p-1, p)\}$ , and

$$\mathbf{D} = \begin{bmatrix} 1 & & & & \\ \vdots & \begin{bmatrix} 1 & \dots & \dots & p \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \\ p-1 & \end{bmatrix}, \quad \mathbf{L} = \mathbf{D}^T \mathbf{D} = \begin{bmatrix} 1 & & & & & & & & p \\ \vdots & \begin{bmatrix} 1 & -1 & & & & & & \\ -1 & 2 & -1 & & & & & \\ \ddots & \ddots & \ddots & \ddots & & & & \\ -1 & 2 & 1 & & & & & \\ & -1 & 1 & & & & & \end{bmatrix} \\ p & \end{bmatrix}.$$

“Classical” TV/fusion penalty

$$\|\mathbf{D}\boldsymbol{\beta}\|_1 = \sum_{i=1}^{p-1} |\beta_{i+1} - \beta_i|, \quad \boldsymbol{\beta}^\top \mathbf{L} \boldsymbol{\beta} = \sum_{i=1}^{p-1} (\beta_{i+1} - \beta_i)^2$$

# Combining several effects

Typically structure + sparsity

Mixture of penalties

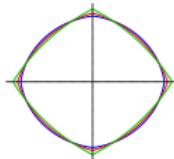
For  $\alpha \in [0, 1]$ , consider for instance

$$\alpha \|\beta\|_\gamma + (1 - \alpha) \|\beta\|_L^2$$

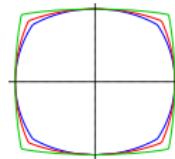
$$\alpha \|\beta\|_\gamma + (1 - \alpha) \|\mathbf{D}\beta\|_1.$$

$$\ell_1 + \ell_2^2$$

elastic-net

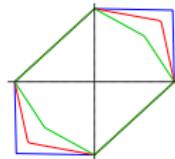


$$\ell_\infty + \ell_2^2$$



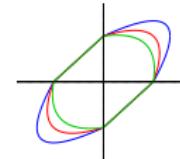
$$\ell_1 + \text{TV}$$

fused-Lasso

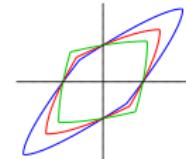


$$\ell_1 + \ell_2 - \text{TV}$$

structured enet



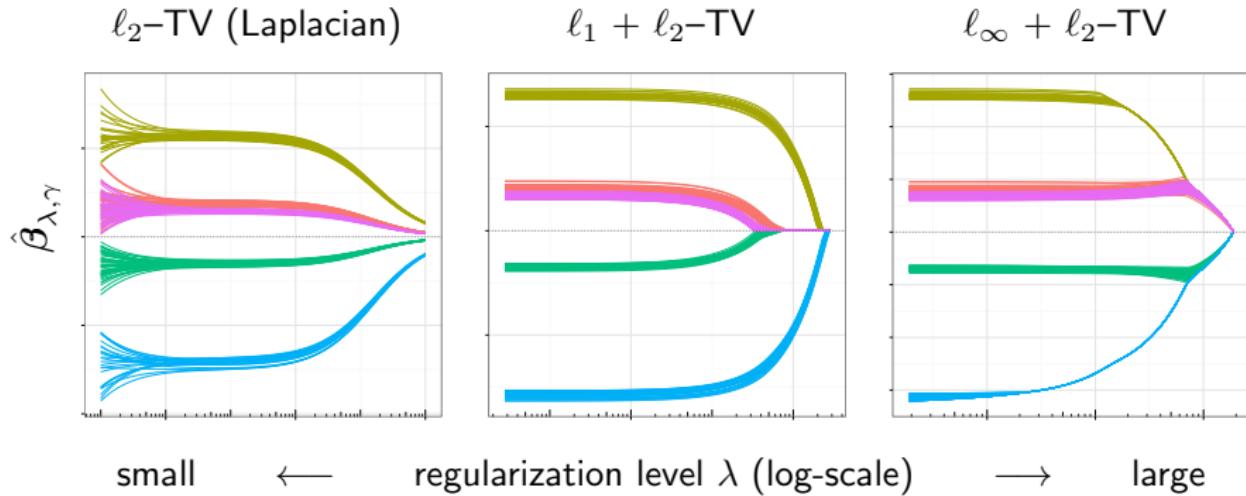
$$\ell_\infty + \ell_2 - \text{TV}$$



**Figure:** A couple of examples for various  $\alpha$

# Numerical illustration now accounting for structure

Basic structure integration pays for **interpretability**



**Figure:** Regularization paths for the structured estimators

## Variants: the Lasso zoo I

The elastic-net (Zou and Hastie, 2005)

Tends to activate correlated variables simultaneously:

$$\hat{\boldsymbol{\beta}}^{\text{e-net}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \frac{1}{2} \text{RSS}(\boldsymbol{\beta}) + \lambda (\alpha \|\boldsymbol{\beta}\|_1 + (1 - \alpha) \|\boldsymbol{\beta}\|_2^2) \right\},$$

Adaptive/Weighted-Lasso

Weights each entry according to a previous estimate e.g. the ols:

$$\hat{\boldsymbol{\beta}}^{\text{lasso}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \frac{1}{2} \text{RSS}(\boldsymbol{\beta}) + \lambda \|\mathbf{w} \circ \boldsymbol{\beta}\|_1 \right\}, \quad \text{with } \mathbf{w} = \max \left( 1, 1/\boldsymbol{\beta}^{\text{ols}} \right).$$

Group-Lasso (Yuan and Lin, 2006)

$$\hat{\boldsymbol{\beta}}^{\text{group}} = \arg \min_{\boldsymbol{\beta} \in \mathbb{R}^p} \left\{ \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \sum_{k=1}^K w_k \|\boldsymbol{\beta}_{G_k}\| \right\},$$

# Example: prostate cancer I

Elastic-net, Adaptive Lasso, Group-Lasso

```
out.elas <- elastic.net(x,y,lambda2=1)
```

```
beta.ols <- lm(y~x)$coefficients[-1]
p.fact <- pmax(1,1/abs(beta.ols))
out.alasso <- elastic.net(x,y,penscale=p.fact)
```

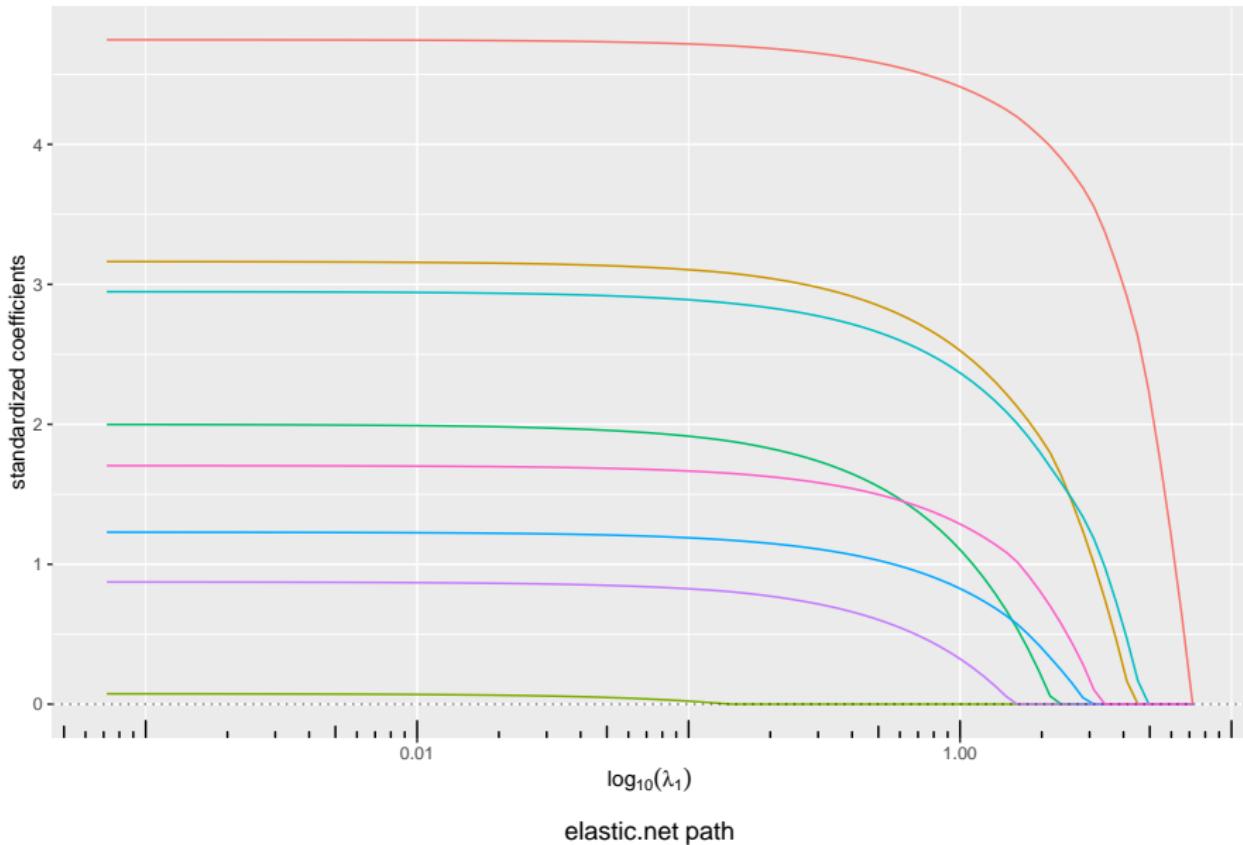
According to the Lasso solution path, we fell like grouping 2 (lweight) and 5 (svi), as well as 3 (age) and 6 (scp); the same for 4 (lbph) and 8 (pgg45).

```
group <- c(1,2,3,4,2,3,5,4)
out.grp <- group.lasso(x, y, group)
sgrp <- selection(out.grp)
```

# Example: prostate cancer

Elastic-net: solution path (penalty)

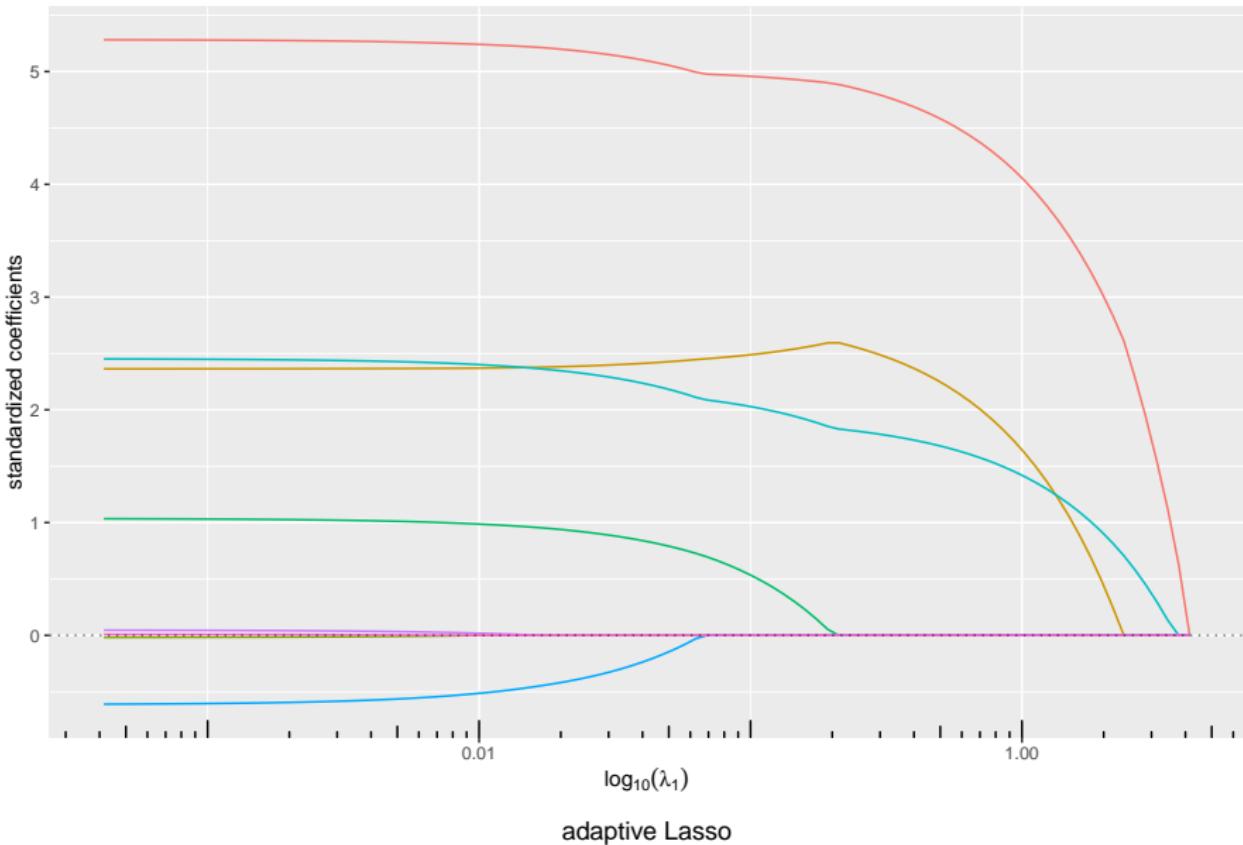
elastic.net path



# Example: prostate cancer

Adaptive-Lasso: solution path (penalty)

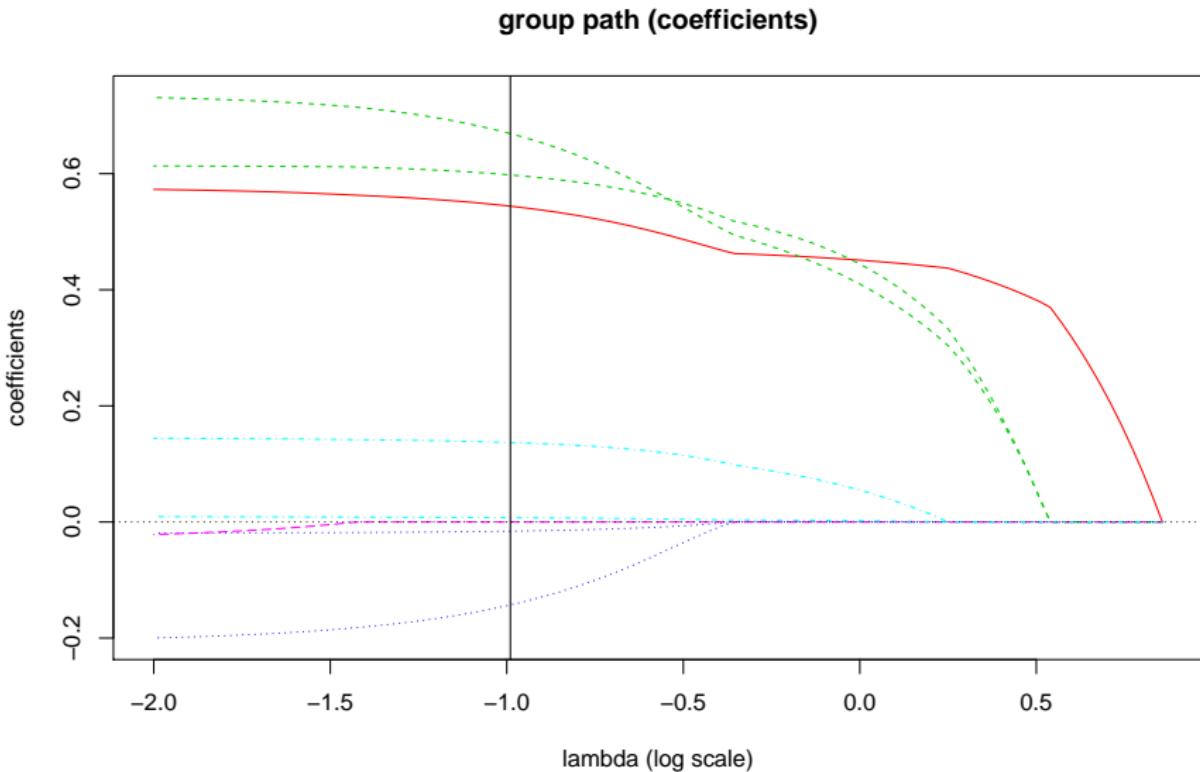
adaptive Lasso



adaptive Lasso

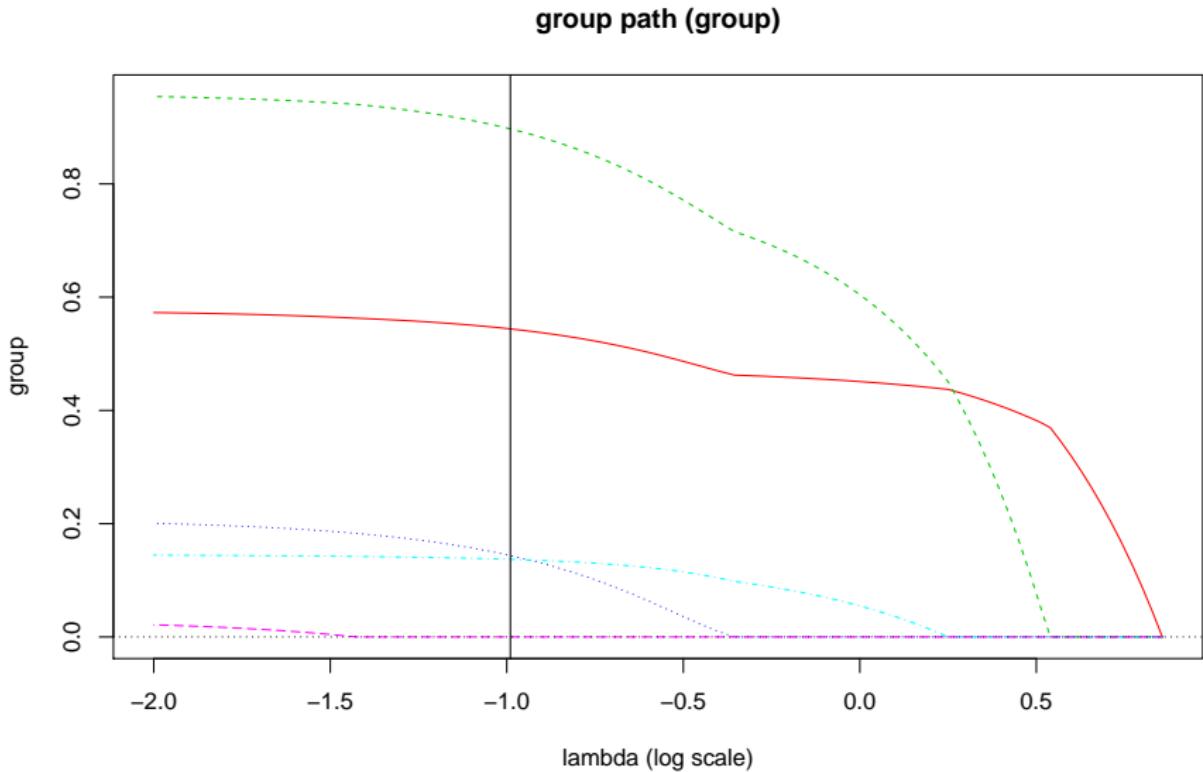
# Example: prostate cancer

Group-Lasso: solution path and AIC



# Example: prostate cancer

Group-Lasso: group-norm path and AIC



# Plan

Prérequis

Motivations: les limites du modèle linéaire

Sélection de variables

## Régularisation

Motivations et principe

La régression Ridge

Régression Lasso

Variations autour du Lasso

Modèles graphiques gaussiens parcimonieux

# Gaussian Graphical Model: canonical settings

## Assays in comparable Gaussian conditions

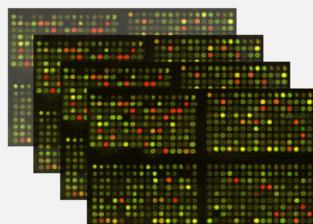
Profiles of a set  $\mathcal{P} = \{1, \dots, p\}$  of variable is described by  $X \in \mathbb{R}^p$  such as

1.  $X \sim \mathcal{N}(\mu, \Sigma)$ , with  $\Theta = \Sigma^{-1}$  the precision matrix.
2. a sample  $(X^1, \dots, X^n)$  of assays stacked in an  $n \times p$  data matrix  $\mathbf{X}$ .

## Conditional independence structure

### The data

Stacking  $(X^1, \dots, X^n)$ , we met the usual individual/variable table  $\mathbf{X}$



$$\mathbf{X} = \begin{pmatrix} x_1^1 & x_1^2 & x_1^3 & \dots & x_1^p \\ \vdots & & & & \\ x_n^1 & x_n^2 & x_n^3 & \dots & x_n^p \end{pmatrix}$$

~ "Covariance" selection

# Gaussian Graphical Model: canonical settings

Assays in comparable Gaussian conditions

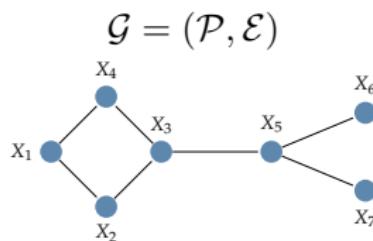
Profiles of a set  $\mathcal{P} = \{1, \dots, p\}$  of variable is described by  $X \in \mathbb{R}^p$  such as

1.  $X \sim \mathcal{N}(\mu, \Sigma)$ , with  $\Theta = \Sigma^{-1}$  the precision matrix.
2. a sample  $(X^1, \dots, X^n)$  of assays stacked in an  $n \times p$  data matrix  $\mathbf{X}$ .

Conditional independence structure

$$(i, j) \notin \mathcal{E} \Leftrightarrow X_i \perp\!\!\!\perp X_j | X_{\setminus\{i,j\}} \Leftrightarrow \Theta_{ij} = 0.$$

Graphical interpretation



~ "Covariance" selection

$\Theta$

|                |                |                |                |                |                |                |                |
|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
|                | X <sub>1</sub> | X <sub>2</sub> | X <sub>3</sub> | X <sub>4</sub> | X <sub>5</sub> | X <sub>6</sub> | X <sub>7</sub> |
| X <sub>1</sub> | ■              |                |                |                |                |                |                |
| X <sub>2</sub> |                | ■              |                |                |                |                |                |
| X <sub>3</sub> |                |                | ■              |                |                |                |                |
| X <sub>4</sub> |                |                |                | ■              |                |                |                |
| X <sub>5</sub> |                |                |                |                | ■              |                |                |
| X <sub>6</sub> |                |                |                |                |                | ■              |                |
| X <sub>7</sub> |                |                |                |                |                |                | ■              |

## Gold standard penalized approaches

Use  $\ell_1$  for both regularizing and promoting *sparsity*

Penalized likelihood (Banerjee *et al.*, Yuan and Lin, 2008)

$$\hat{\Theta}_\lambda = \arg \max_{\Theta \in \mathbb{S}_+} \ell(\Theta; \mathbf{X}) - \lambda \|\Theta\|_1$$

- + symmetric, positive-definite
- solved by the “Graphical-Lasso” ( $\mathcal{O}(p^3)$ , Friedman *et al*, 2007).

Neighborhood Selection (Meinshausen & Bühlman, 2006)

$$\hat{\beta}^{(i)} = \arg \min_{\beta \in \mathbb{R}^{p-1}} \frac{1}{n} \left\| \mathbf{X}_i - \mathbf{X}_{\setminus i} \beta \right\|_2^2 + \lambda \|\beta\|_1$$

CLIME – Pseudo-likelihood (Cai *et al.*, 2011; Yuan, 2010)

$$\hat{\Theta} = \arg \min_{\Theta} \|\Theta\|_1 \text{ subjected to } \|n^{-1} \mathbf{X}^t \mathbf{X} \Theta - \mathbf{I}\|_\infty \leq \lambda$$

## Gold standard penalized approaches

Use  $\ell_1$  for both regularizing and promoting *sparsity*

Penalized likelihood (Banerjee *et al.*, Yuan and Lin, 2008)

$$\hat{\Theta}_\lambda = \arg \max_{\Theta \in \mathbb{S}_+} \ell(\Theta; \mathbf{X}) - \lambda \|\Theta\|_1$$

Neighborhood Selection (Meinshausen & Bühlman, 2006)

$$\hat{\beta}^{(i)} = \arg \min_{\beta \in \mathbb{R}^{p-1}} \frac{1}{n} \left\| \mathbf{X}_i - \mathbf{X}_{\setminus i} \beta \right\|_2^2 + \lambda \|\beta\|_1$$

- not symmetric, not positive-definite
- +  $p$  Lasso solved with Lars-like algorithms ( $\mathcal{O}(npd)$  for  $d$  neighbors).

CLIME – Pseudo-likelihood (Cai *et al.*, 2011; Yuan, 2010)

$$\hat{\Theta} = \arg \min_{\Theta} \|\Theta\|_1 \text{ subjected to } \|n^{-1} \mathbf{X}^t \mathbf{X} \Theta - \mathbf{I}\|_\infty \leq \lambda$$

## Gold standard penalized approaches

Use  $\ell_1$  for both regularizing and promoting *sparsity*

Penalized likelihood (Banerjee *et al.*, Yuan and Lin, 2008)

$$\hat{\Theta}_\lambda = \arg \max_{\Theta \in \mathbb{S}_+} \ell(\Theta; \mathbf{X}) - \lambda \|\Theta\|_1$$

Neighborhood Selection (Meinshausen & Bühlman, 2006)

$$\hat{\beta}^{(i)} = \arg \min_{\beta \in \mathbb{R}^{p-1}} \frac{1}{n} \left\| \mathbf{X}_i - \mathbf{X}_{\setminus i} \beta \right\|_2^2 + \lambda \|\beta\|_1$$

CLIME – Pseudo-likelihood (Cai *et al.*, 2011; Yuan, 2010)

$$\hat{\Theta} = \arg \min_{\Theta} \|\Theta\|_1 \text{ subjected to } \|n^{-1} \mathbf{X}^t \mathbf{X} \Theta - \mathbf{I}\|_\infty \leq \lambda$$

- not positive-definite
- +  $p$  linear programs easily distributed ( $\mathcal{O}(p^2 d)$  for  $d$  neighbors).

# Gold standard penalized approaches

Use  $\ell_1$  for both regularizing and promoting *sparsity*

Penalized likelihood (Banerjee *et al.*, Yuan and Lin, 2008)

$$\hat{\Theta}_\lambda = \arg \max_{\Theta \in \mathbb{S}_+} \ell(\Theta; \mathbf{X}) - \lambda \|\Theta\|_1$$

## Variants and recent improvements

'13 NIPS submissions

- ▶ Use square-root Lasso in place of Lasso for tuning insensitive property package
- ▶ Solve CLIME for  $p = 10^6$  (on 400 cores).

See R package `huge`, `fastclime`, `flare`, `QUIC`.



# Estimating the covariance structure of the plasmodium data I

```
library(Matrix)
load("plasmodium_expression.Rdata")
dim(Y)

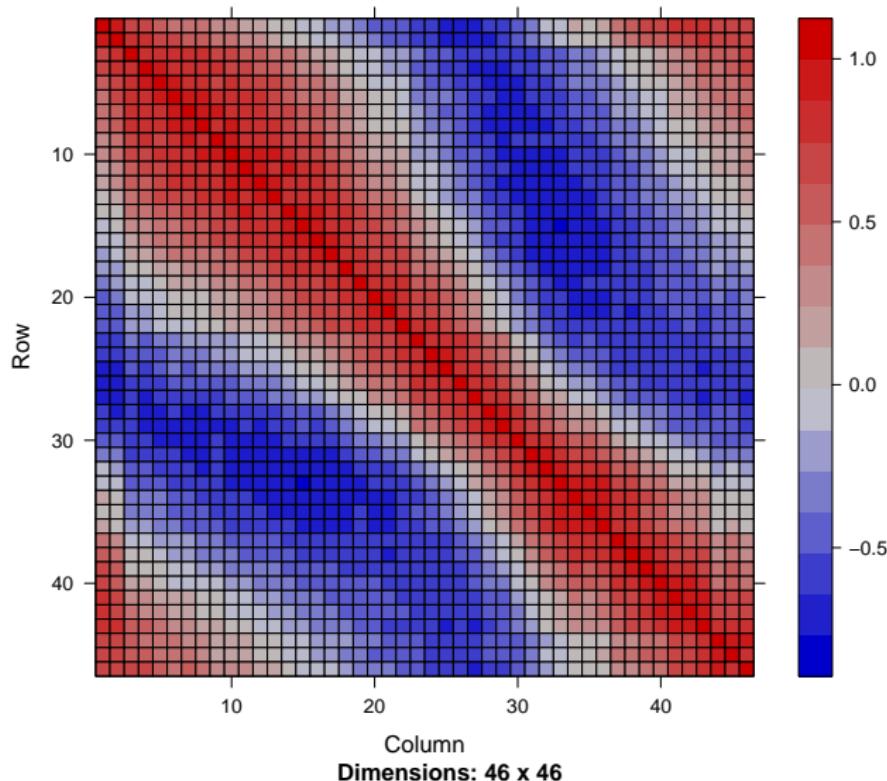
## [1] 3490    46

head(Y) [, 1:5]

##           TP1      TP2      TP3      TP4      TP5
## MAL13P1.100 0.4510 0.6532 1.0760 0.5515 0.4238
## MAL13P1.102 1.5320 1.8920 0.8803 1.0300 0.9328
## MAL13P1.103 0.5218 0.5213 0.5328 0.3719 0.3258
## MAL13P1.105 0.5515 0.5527 0.8627 0.4541 0.4299
## MAL13P1.107 0.5630 0.4463 1.0760 0.4035 0.2082
## MAL13P1.112 0.5390 0.5393 0.5642 0.5326 0.4469

image(Matrix(cor(Y)))
```

# Estimating the covariance structure of the plasmodium data II



# Estimating the covariance structure I

## Sparse Estimation

```
library(huge)
huge.out <- huge(as.matrix(Y), method="glasso", cov.output=TRUE)

## Conducting the graphical lasso (glasso) with lossless screening....in progress:0%
Conducting the graphical lasso (glasso) with lossless screening....in progress:9%
Conducting the graphical lasso (glasso) with lossless screening....in progress:19%
Conducting the graphical lasso (glasso) with lossless screening....in progress:30%
Conducting the graphical lasso (glasso) with lossless screening....in progress:40%
Conducting the graphical lasso (glasso) with lossless screening....in progress:50%
Conducting the graphical lasso (glasso) with lossless screening....in progress:60%
Conducting the graphical lasso (glasso) with lossless screening....in progress:70%
Conducting the graphical lasso (glasso) with lossless screening....in progress:80%
Conducting the graphical lasso (glasso)....done.

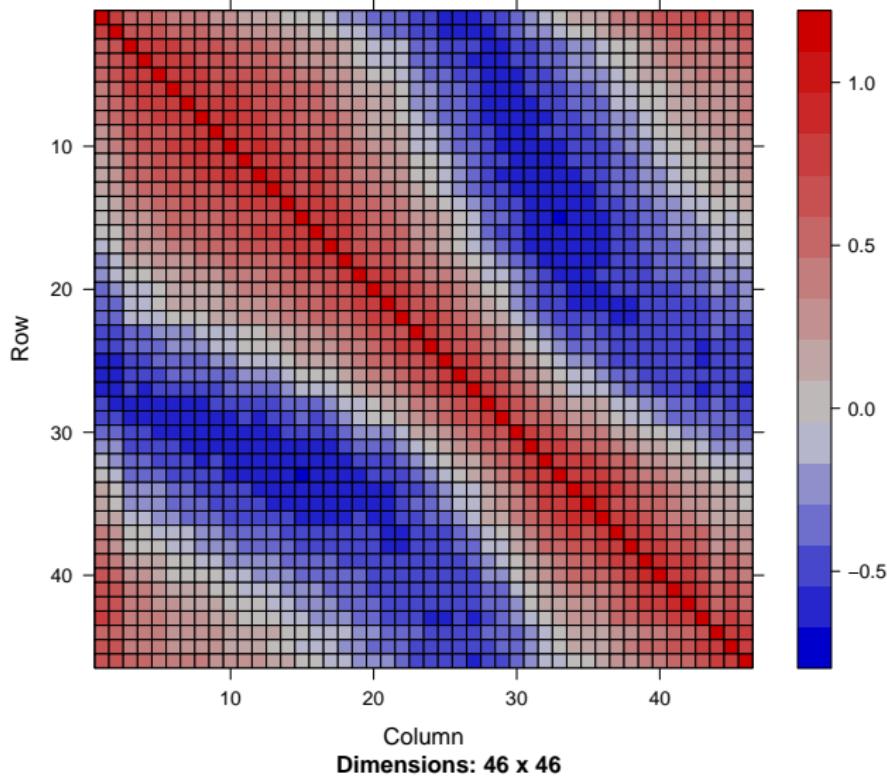
sel.out <- huge.select(huge.out)

## Conducting extended Bayesian information criterion (ebic) selection....done

image(sel.out$opt.cov)
```

# Estimating the covariance structure II

## Sparse Estimation



# Estimating the covariance structure I

## Sparse Estimation of the inverse covariance

```
sum(abs(sel.out$opt.icov) != 0)

## [1] 760

ncol(sel.out$opt.icov) ** 2

## [1] 2116

image(sel.out$opt.icov)
```

# Estimating the covariance structure II

Sparse Estimation of the inverse covariance

