Modèle linéaire et extension

Sélection de variables et régularisation

M1 Math et Interactions – UEVE/ENSIIE

semestre d'automne 2016

`http://julien.cremeriefamily.info/teachings_M1MINT_Reg.html`

Laboratoire de Mathématiques et Modélisation d'Évry
LaMME

université evry val-d'essonne

Plan

Plan

Plan

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 a

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

$\rightsquigarrow$ Un bon modèle doit prédire correctement de nouvelles réponses.

Notations

Soient

- Y la variable aléatoire de réponse,
- $X = (X_1, \ldots, X_p)$ un vecteur aléatoire tel que X_j est le j^{e} prédicteur.

Les données

Pour un échantillon $\{(y_i, x_i), i = 1, \ldots, n\}$ i.i.d. de (Y, X), on note

- $\mathcal{D} = \{i : (y_i, x_i) \in$ l'ensemble d'entraînement$\}$,
- $\mathcal{T} = \{i : (y_i, x_i) \in$ 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 $n \times p$ de données (ou design) de l'ensemble d'entraînement dont la j^{e} ligne est $\mathbf{x}_j$,
- $(\mathbf{y}_{\mathcal{T}}, \mathbf{X}_{\mathcal{T}})$ les données de test.

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

Stratégie

1. On se donne une famille $\mathcal{F}$ de modèles
   Pour la régression linéaire, $\mathcal{F} = \left\{X^T\beta, \beta \in \mathbb{R}^p\right\}$.

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

Stratégie d'apprentissage

Problème

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

Stratégie

1. On se donne une famille $\mathcal{F}$ de modèles
   Pour la régression linéaire, $\mathcal{F} = \left\{X^T\boldsymbol{\beta}, \boldsymbol{\beta} \in \mathbb{R}^p\right\}$.

2. On ajuste $\hat{f} \in \mathcal{F}$ sur des données d'entraînement $\mathcal{D}$
   On calcule l'estimateur des moindre carré $\hat{\boldsymbol{\beta}}^{\mathrm{ols}}$ *et* $\hat{f} = \hat{Y} = \mathbf{X}\hat{\boldsymbol{\beta}}^{\mathrm{ols}}$

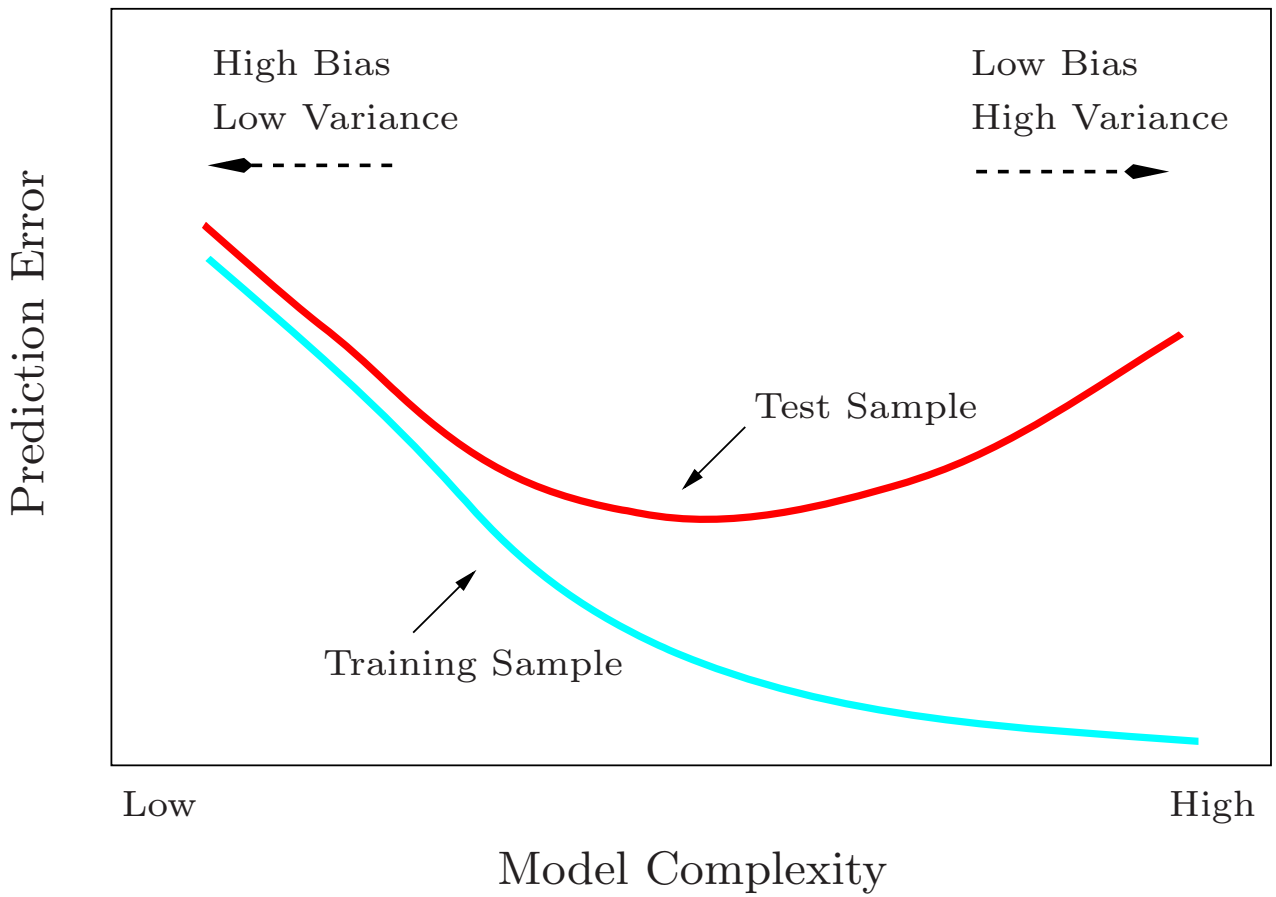
3. On estime l'erreur de prédiction err à l'aide des données test $\mathcal{T}$.

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

Compromis Biais/Variance

À un nouveau point $X = x$,

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



Cas de la régression linéaire

Erreur de prédiction

On peut montrer pour $\mathbf{X}$ fixé que

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

Thérorème de Gauss-Markov

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

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

Plan

OLS et colinéarité: Gram-Schmidt (I)

Régression par orthogonalisations successives

Algorithme de Gram-Schmidt

S0 Initialisation

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

S2 Régression sur une base orthonormale de $\text{vect}(\mathbf{X})$

for $j = 1, \ldots, p$ **do**

  for $k = 1, \ldots, j-1$ **do**

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

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

  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 $\hat{\beta}_p$

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

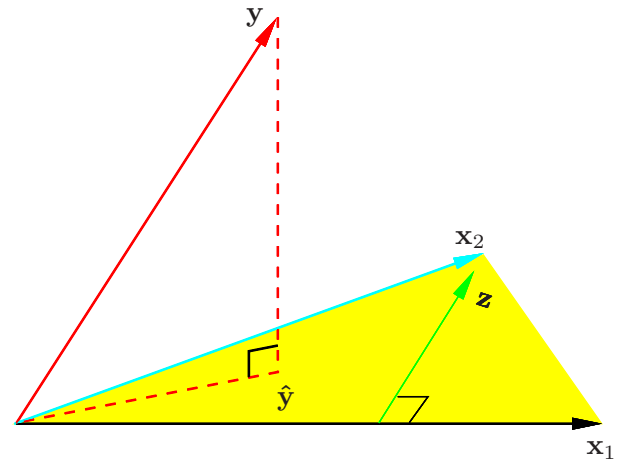


Figure: Exemple avec deux prédicteurs

L'étape 2 peut s'écrire (avec $\mathbf{D}$ diagonale telle que $\mathbf{D}_{jj} = \mathbf{z}_j^\intercal \mathbf{z}_j$)

$$\mathbf{X} = \mathbf{Z\Gamma} = \mathbf{ZD}^{-1}\mathbf{D\Gamma} = \mathbf{QR},$$

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

OLS et colinéarité: Gram-Schmidt (II)

Apportée par la factorisation QR

Estimateur et prédiction en fonction de la factorisation QR

$$\hat{\boldsymbol{\beta}}^{\text{ols}} = \mathbf{R}^{-1}\mathbf{Q}^\intercal\mathbf{y}, \qquad \hat{\mathbf{y}} = \mathbf{Q}\mathbf{Q}^\intercal\mathbf{y}.$$

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

- $\hat{\beta}_j$ est la contribution additionelle 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

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

$\rightsquigarrow$ prédicteurs colinéaires $\Rightarrow$ **mauvaise estimation de β**.

OLS et colinéarité: limite de l'interprétabilité (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.

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

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

Cas gaussien

Si X, Y, Z sont jointement gaussiens, alors

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

OLS et colinéarité: limite de l'interprétabilité (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.

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

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

Cas gaussien

Si X, Y, Z sont jointement gaussiens, alors

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

OLS et colinéarité: limite de l'interprétabilité (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.

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

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

Cas gaussien

Si X, Y, Z sont jointement gaussiens, alors

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

OLS et colinéarité: limite de l'interprétabilité (II)

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

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

Alors on peut montrer que

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

$\rightsquigarrow$ 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.

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

$\rightsquigarrow$ Les prédicteurs fortement liés impliquent des covariances négatives entre les coefficients de régression!

OLS et colinéarité: limite de l'interprétabilité (II)

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

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

Alors on peut montrer que

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

$\rightsquigarrow$ 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.

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

$\rightsquigarrow$ Les prédicteurs fortement liés impliquent des covariances négatives entre les coefficients de régression!

Plan

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 ($\mathbf{y}$) 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
```

Corrélations entre prédicteurs I

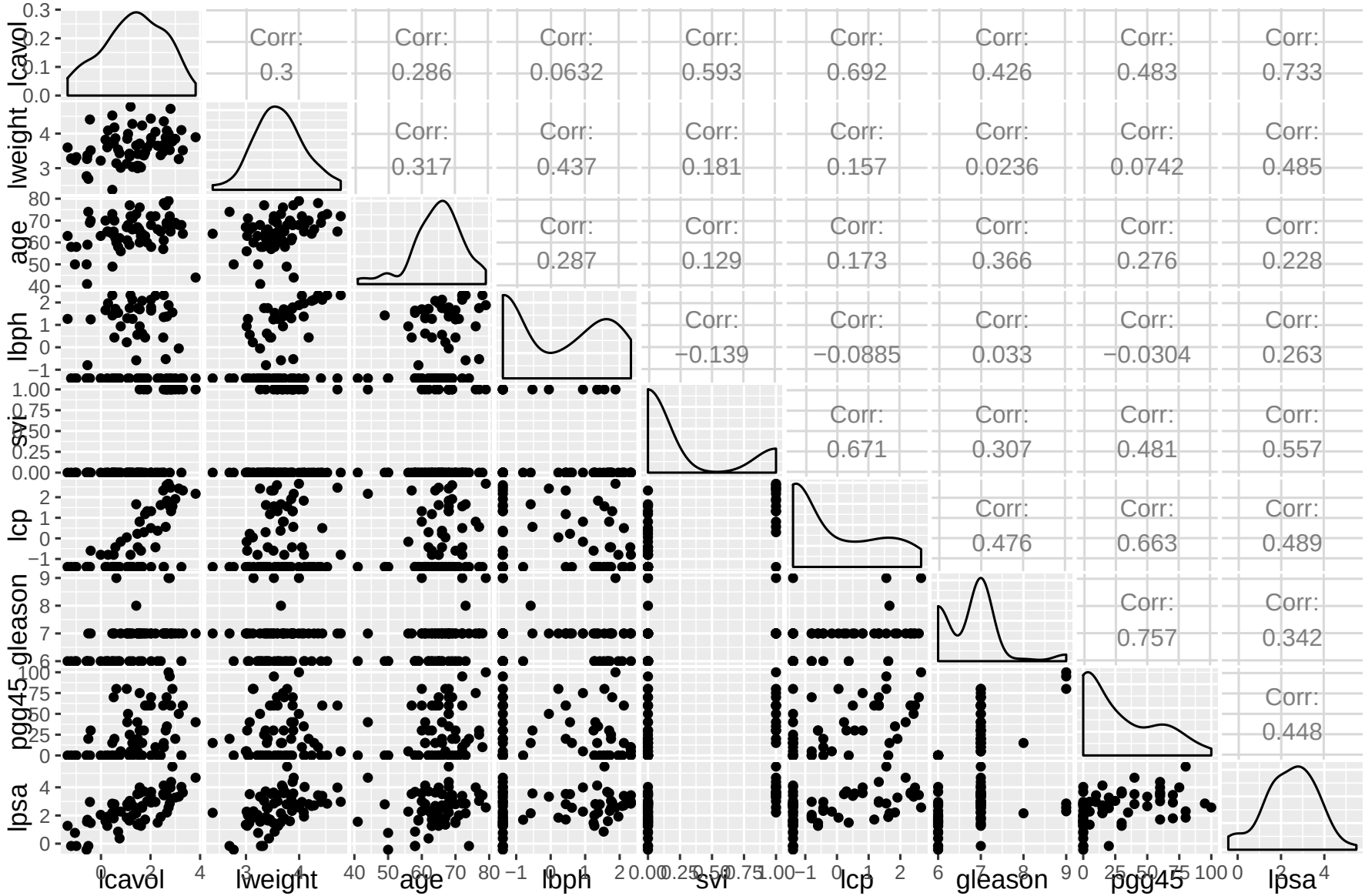
```
print(as.dist(var(prostate[prostate$train,1:8])),digits=1)

##         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[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



Corrélations entre prédicteurs III

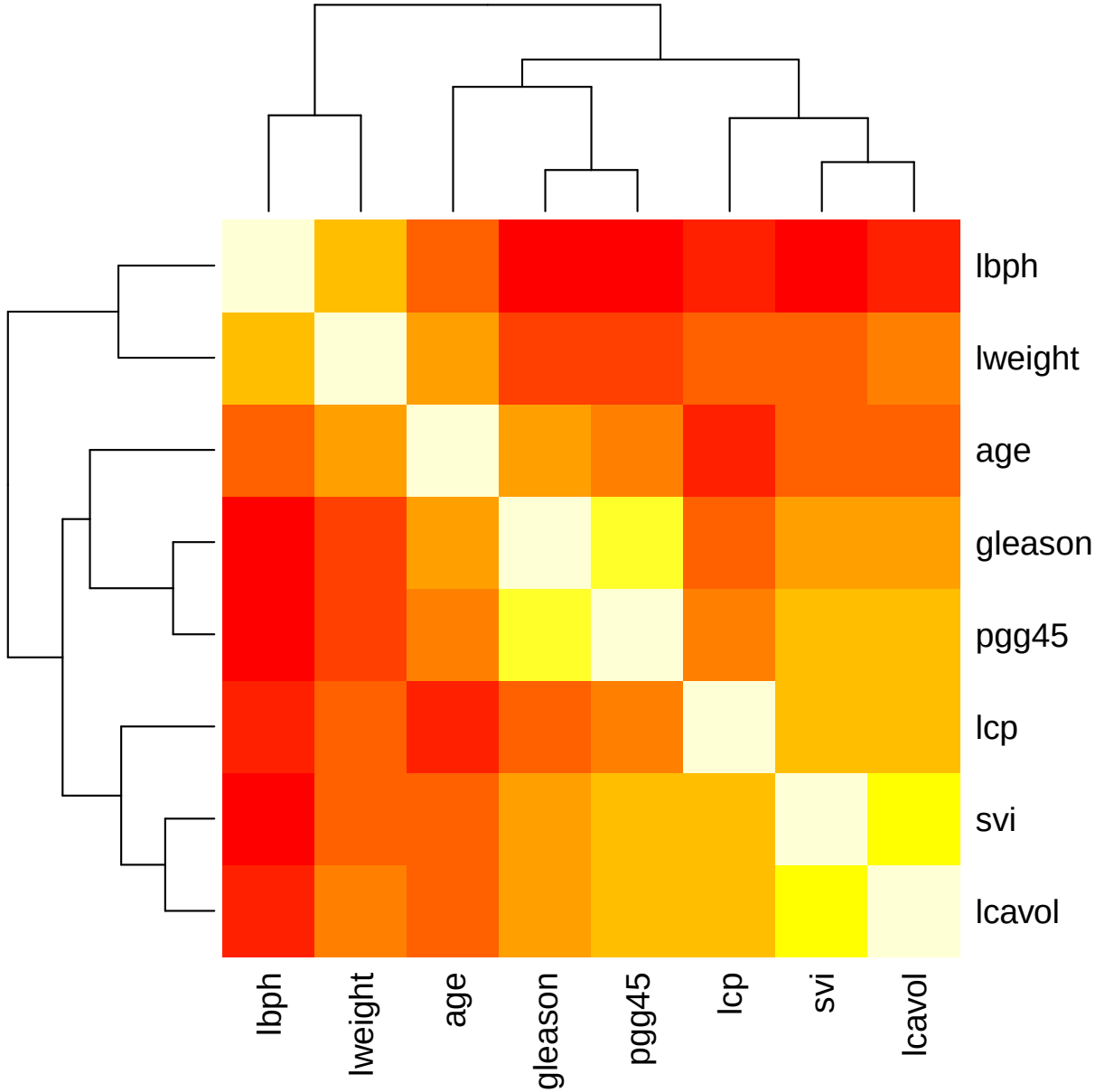


Illustration des limites de l'estimateur OLS I

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: $\mathrm{cov}(\hat{\boldsymbol{\beta}}_i, \hat{\boldsymbol{\beta}}_j) \propto -\mathrm{cor}(X_i, X_j | X_k, k \neq i, j)$.

Considération d'ordre numérique

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

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

$\rightsquigarrow$ L'interprétation devient problématique

Solutions

Sélection de variable

Si le vrai modèle contient peu de prédicteurs liés à la réponse, on peut vouloir sélectionner ceux ayant un grand pouvoir prédictif. On vise

- 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 difficile.
Une solution est de **régulariser** le problème en **contraignant** les paramètres β à vivre dans un espace approprié, de sorte à

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

Solutions

Sélection de variable

Si le vrai modèle contient peu de prédicteurs liés à la réponse, on peut vouloir sélectionner ceux ayant un grand pouvoir prédictif. On vise

- 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 difficile.
Une solution est de **régulariser** le problème en **contraignant** les paramètres β à vivre dans un espace approprié, de sorte à

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

Plan

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

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

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

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

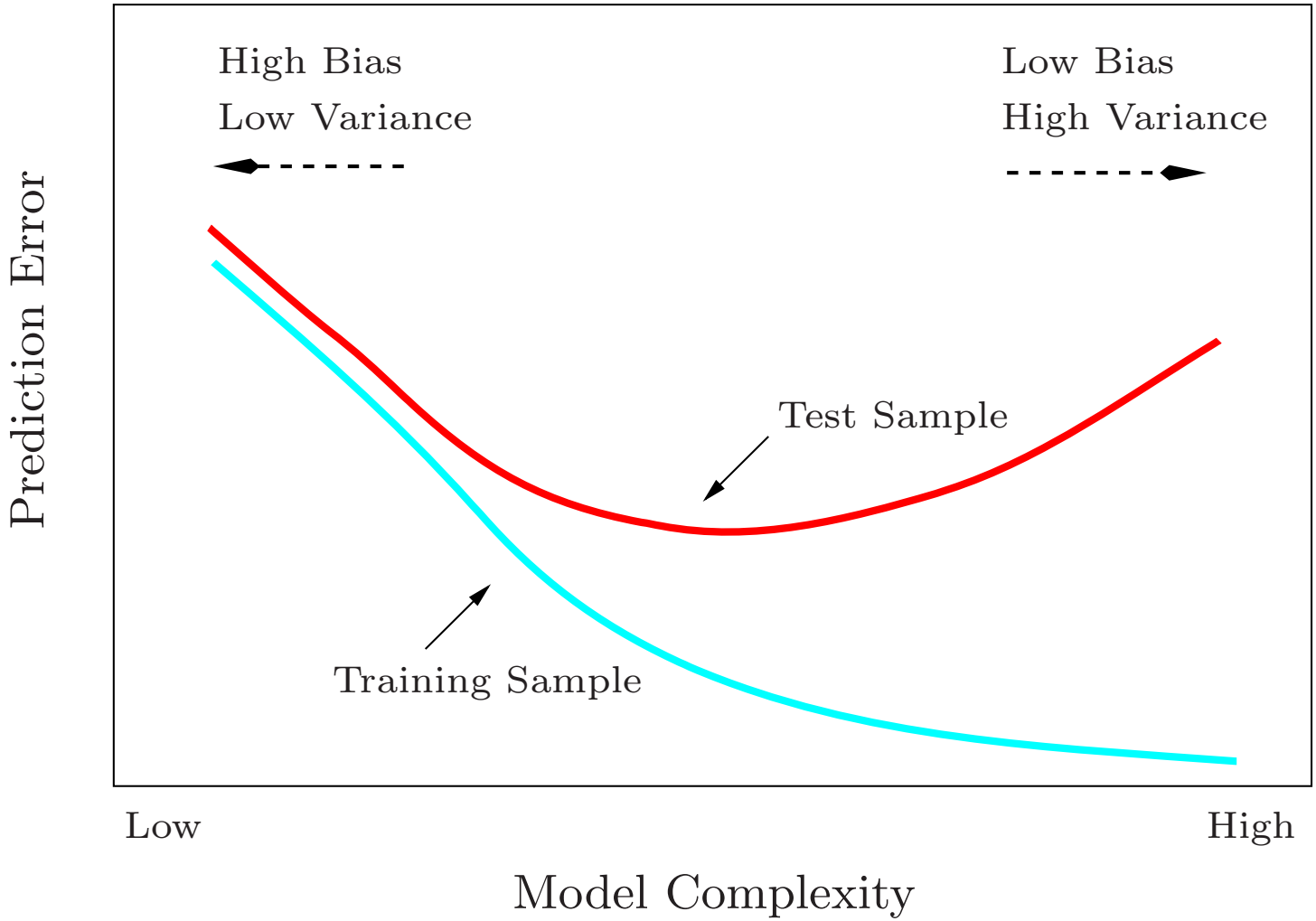
Ingrédients

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

Rappel


High Bias
Low Variance
Low Bias
High Variance
Prediction Error
Test Sample
Training Sample
Low
High
Model Complexity

Estiamtion de l'erreur 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 $\kappa : \{1, \ldots, n\} \to \{1, \ldots, K\}$ une fonction indicatrice de la partition de la $i$ème observation. On note $\hat{\boldsymbol{\beta}}^{-k}$ les paramètres estimés sur les données privées du $k$ième sous-ensemble. Alors

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

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

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

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

Critères pour le modèle de régression linéaire σ connue

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

- **C_p de Mallows**

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

- **Akaïke Information Criteria** équivalent au C_p quand σ est connue

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

- **Bayesian Information Criterion**

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

Critères pénalisés

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

Critères pour le modèle de régression linéaire σ inconnue

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

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

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

- **Akaïke Information Criteria** σ^2 estimée par $\mathrm{err}_{\mathcal{D}}/n$

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

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

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

C_p/AIC: preuve

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

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

En espérance, on a

- $\mathbb{E}[\|\varepsilon\|^2] = n\sigma^2$
- $\mathbb{E}[\varepsilon^\intercal(\mathbf{I} - \mathbf{P}_{\mathbf{X}})\boldsymbol{\mu}] = 0$
- $\mathbb{E}[2\varepsilon^\intercal\mathbf{P}_{\mathbf{X}}\varepsilon] = 2\mathbb{E}[\text{trace}(\varepsilon^\intercal\mathbf{P}_{\mathbf{X}}\varepsilon)] = 2\text{trace}(\mathbf{P}_{\mathbf{X}})\sigma^2$

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

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

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

Plan

Recherche exhaustive (best-subset)

Algorithme

Pour $k = 0, \ldots, p$, 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 $\mathcal{S} = \{1, \ldots, p\}$
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

Recherche exhaustive I

```
library(leaps)
```

On calcule tous les modèles possibles

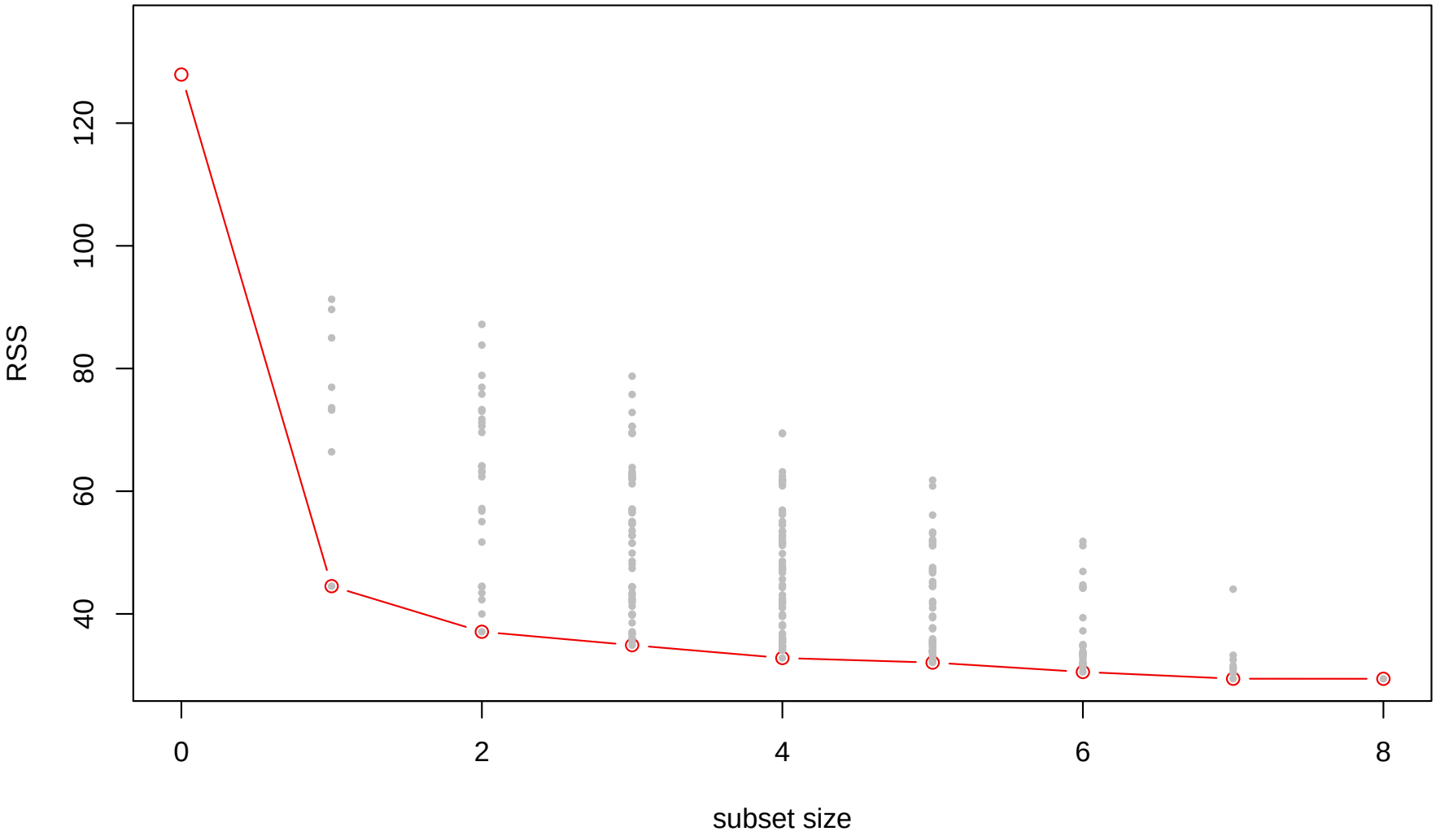
```
out <- regsubsets(lpsa ~ . , data=prostate.train,
                  nbest=100, really.big=TRUE)
bss <- summary(out)
```

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

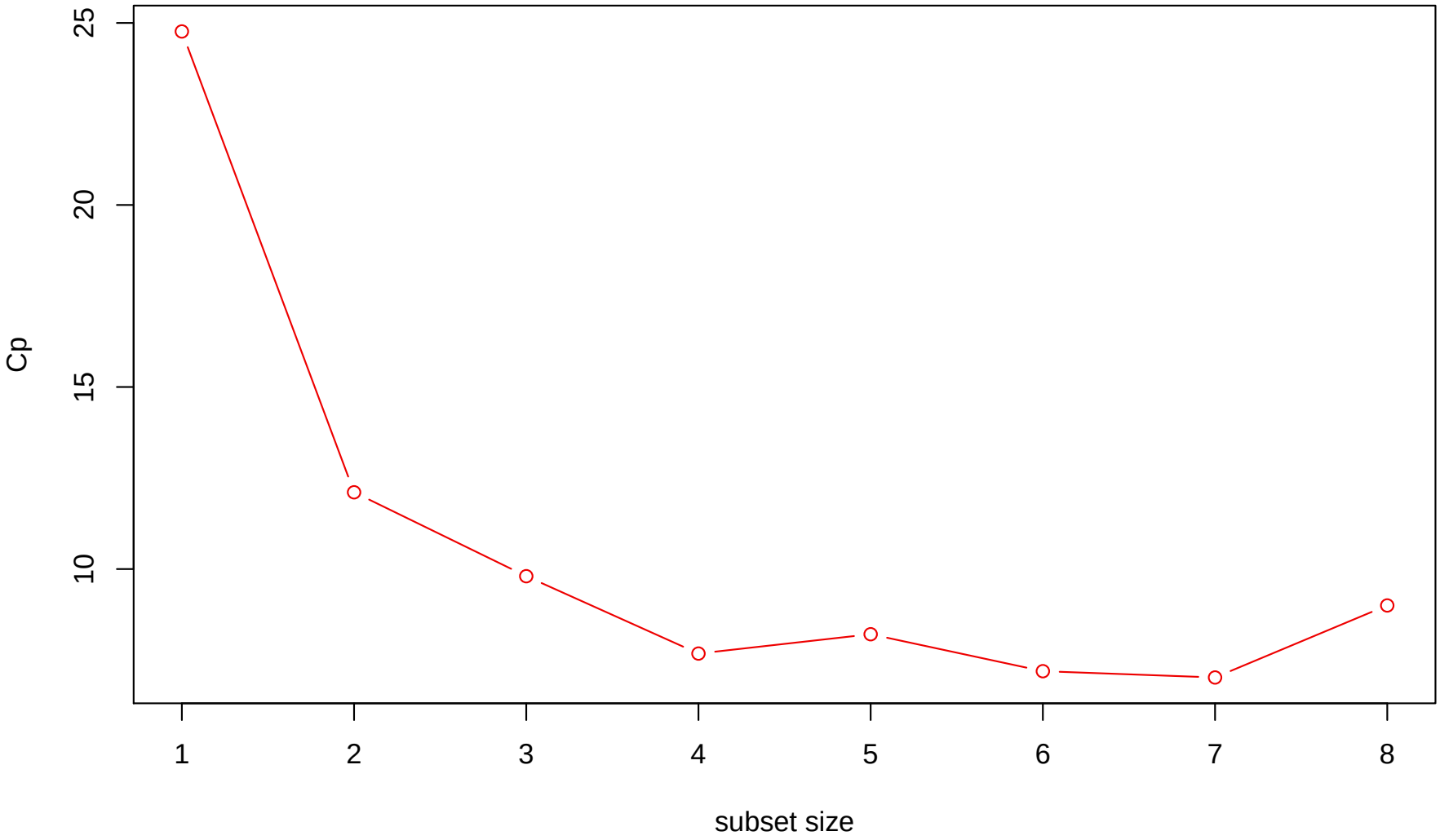
```
bss.size  <- as.numeric(rownames(bss$which))
intercept <- lm(lpsa ~ 1, data=prostate)
bss.best.rss  <- c(sum(resid(intercept)^2), tapply(bss$rss  , bss.size, min))
```

```
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)
```

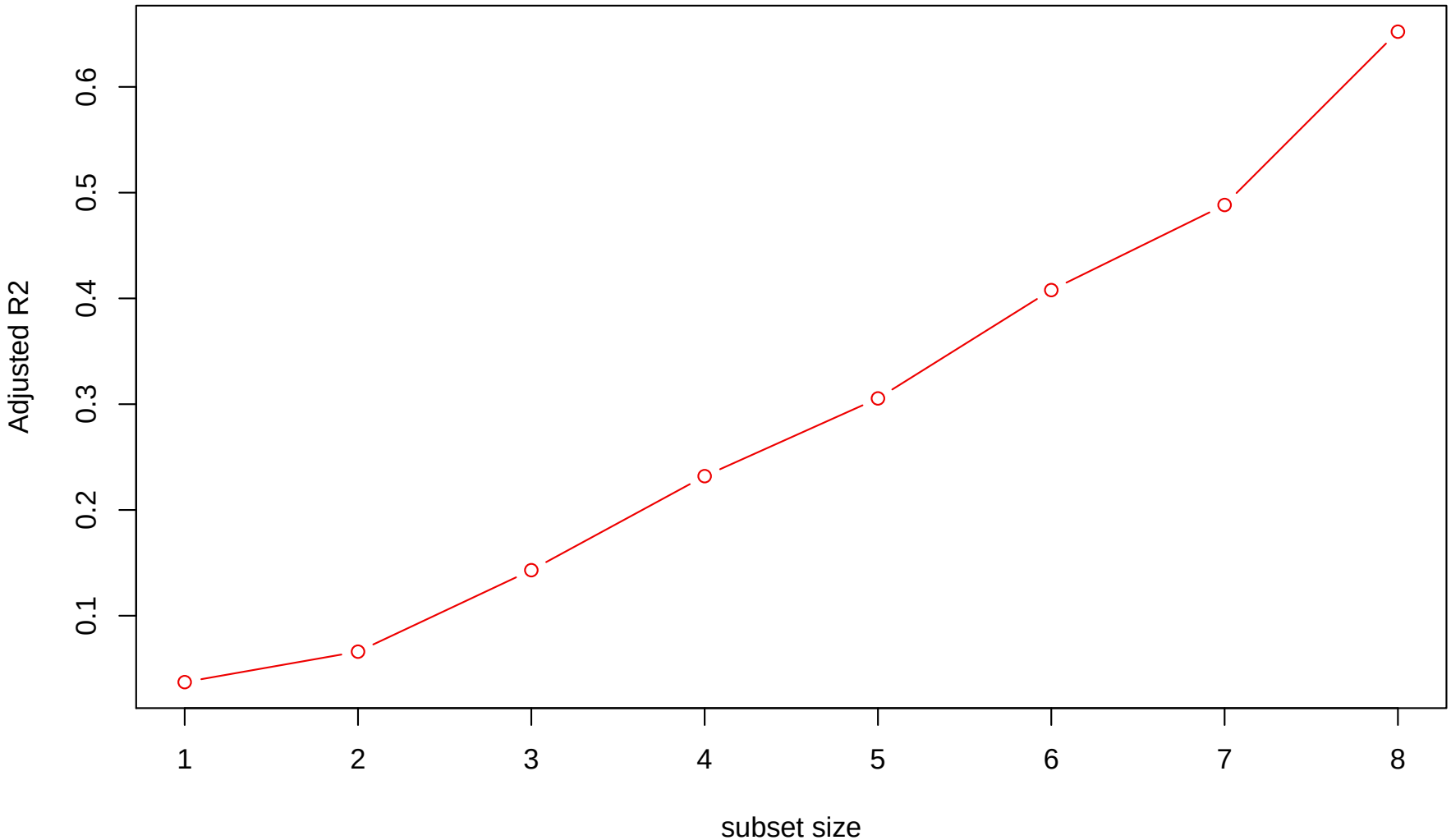
Recherche exhaustive II



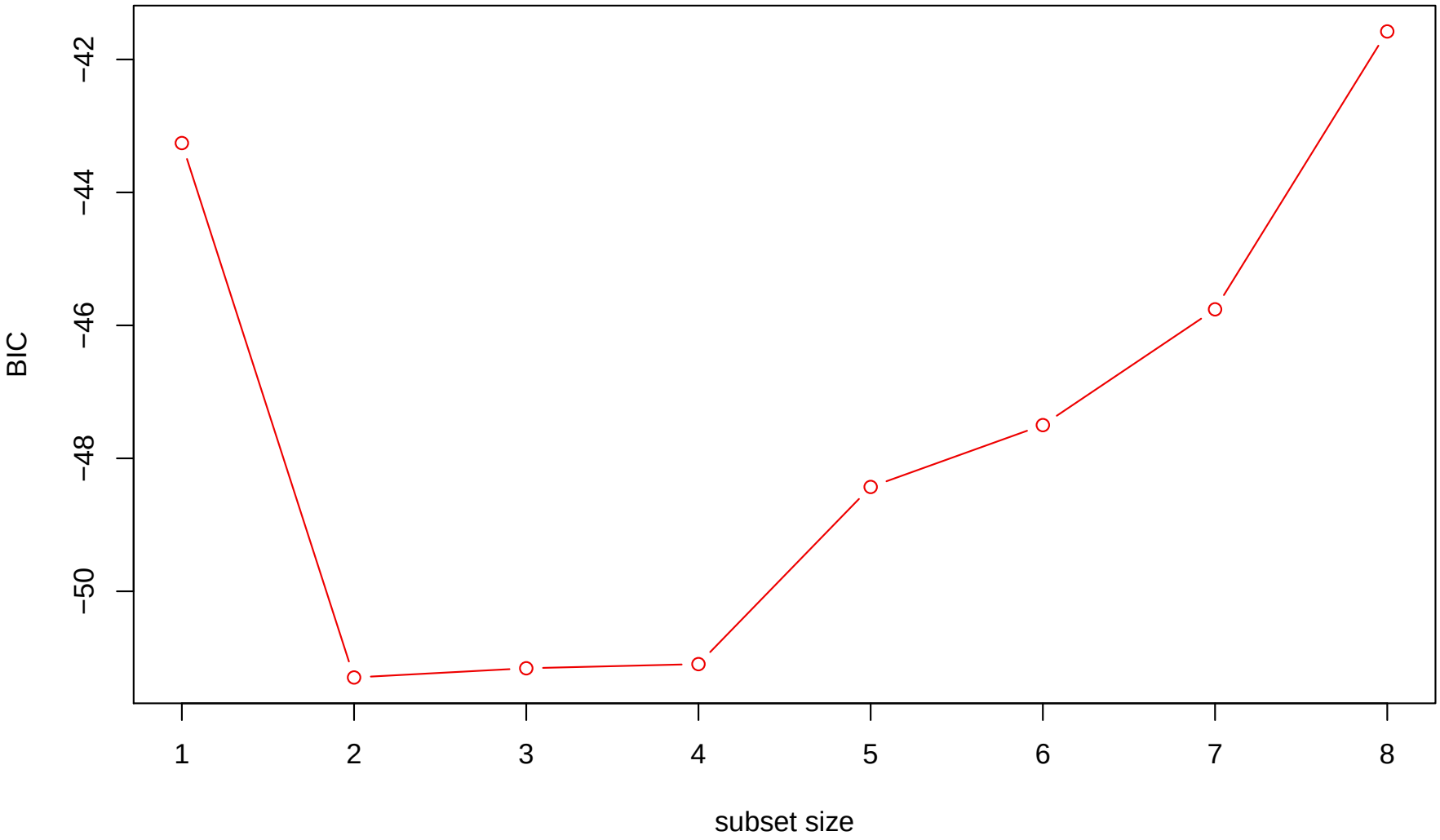
Recherche exhaustive III



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)
```

$\rightsquigarrow$ 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 response**,
   $\rightsquigarrow$ les algorithmes de sélection peuvent retrouver les prédicteurs pertinents.
2. Si le vrai $\mathcal{S}$ contient **beaucoup de variables très corrélées**
   $\rightsquigarrow$ 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

Plan

Objectifs

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

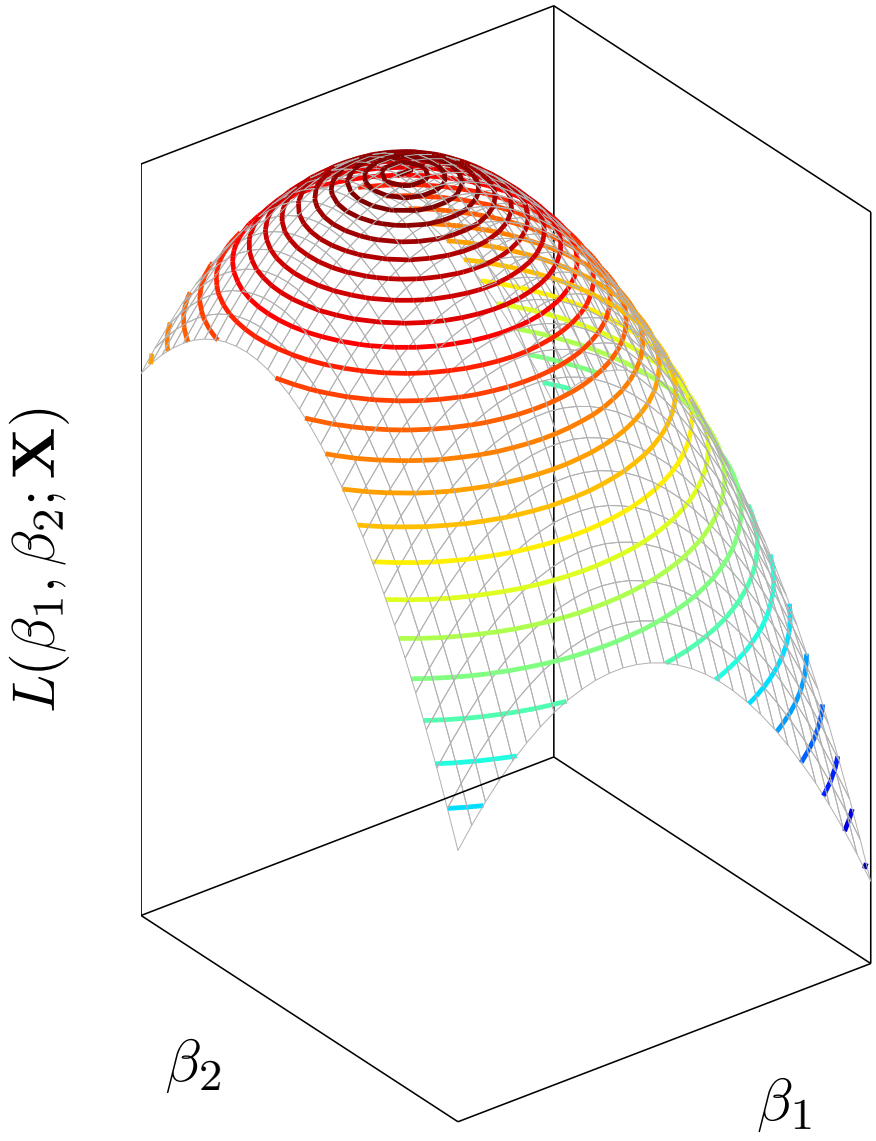
1. Régulariser le problème
   - Pour des questions numériques, (conditionnement de $\mathbf{X}^T\mathbf{X}$),
   - Pour des questions de stabilité, (corrélations entre les $(X_1, \ldots, X_p)$).

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

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

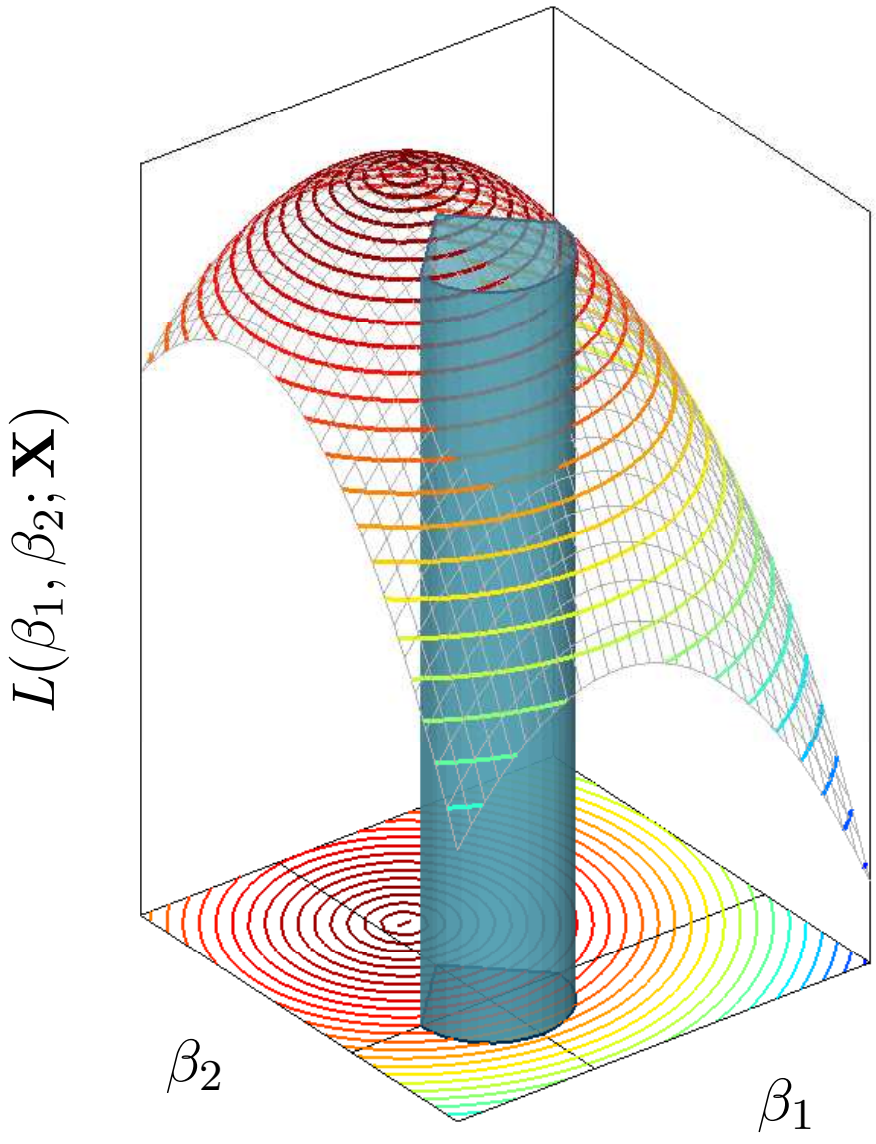
où L est typiquement une vraisemblance concave, ce qui est équivalent à

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

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

Une vue géométrique de la régularisation

Optimisation sous contraintes



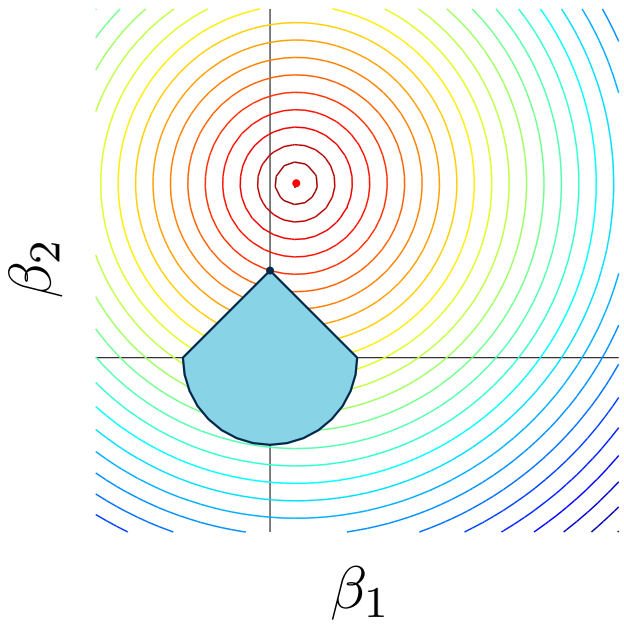
On restreint l'espace des $\boldsymbol{\beta}$, ainsi

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

où Ω définit un ensemble de *constraintes* sur $\boldsymbol{\beta}$.

Une vue géométrique de la régularisation

Optimisation sous contraintes



On restreint l'espace des $\boldsymbol{\beta}$, 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 Ω définit un ensemble de *constraintes* sur $\boldsymbol{\beta}$.

$$\Updownarrow$$

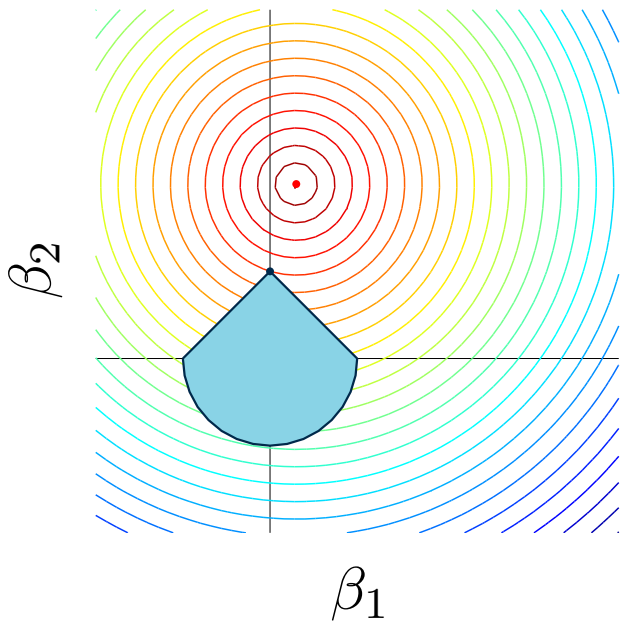
$$\underset{\beta_1,\beta_2}{\text{minimize}}\, J(\boldsymbol{\beta}),$$

où J est la fonction objectif (convexe) définie par

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

Une vue géométrique de la régularisation

Optimisation sous contraintes



On restreint l'espace des $\boldsymbol{\beta}$, 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 Ω définit un ensemble de *constraintes* sur $\boldsymbol{\beta}$.

$$\Updownarrow$$

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

où J est la fonction objectif (convexe) définie par

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

Comment choisir Ω ?

Plan

Plan

Définition

Remarque

Si les β_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 $\boldsymbol{\beta}$. Cette approche pourrait réduire sensiblement l'erreur de prédiction.

La Ridge comme problème de régularisation

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

où s est un facteur de rétrécissement.

Définition

Remarque

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

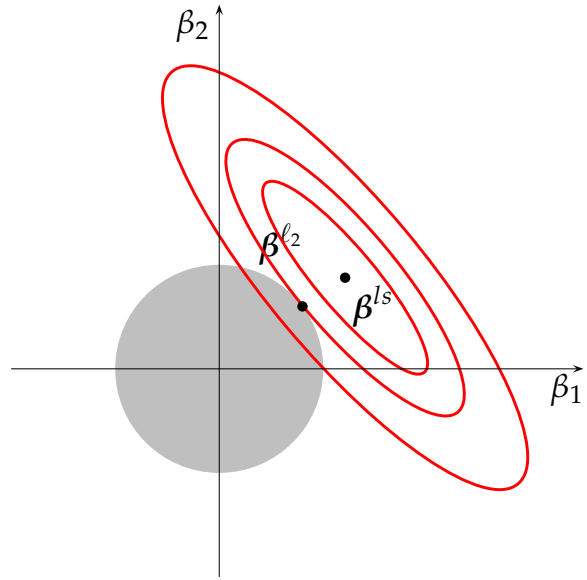
Idée

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

La Ridge comme problème de régularisation

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

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 $\boldsymbol{\beta}$ indexés sur γ.

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 $\beta_j = (\beta_1 + \beta_2)/2$.

$\rightsquigarrow$ 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 $\boldsymbol{\beta}$ indexés sur γ.

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 $\beta_j = (\beta_1 + \beta_2)/2$.

$\rightsquigarrow$ 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 $\boldsymbol{\beta}$ indexés sur γ.

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 $\beta_j = (\beta_1 + \beta_2)/2$.

$\rightsquigarrow$ La Ridge "moyenne" les coefficients associés aux prédicteurs corrélés.

Un exemple en deux dimensions (en R) I

On génère deux prédicteurs corrélés

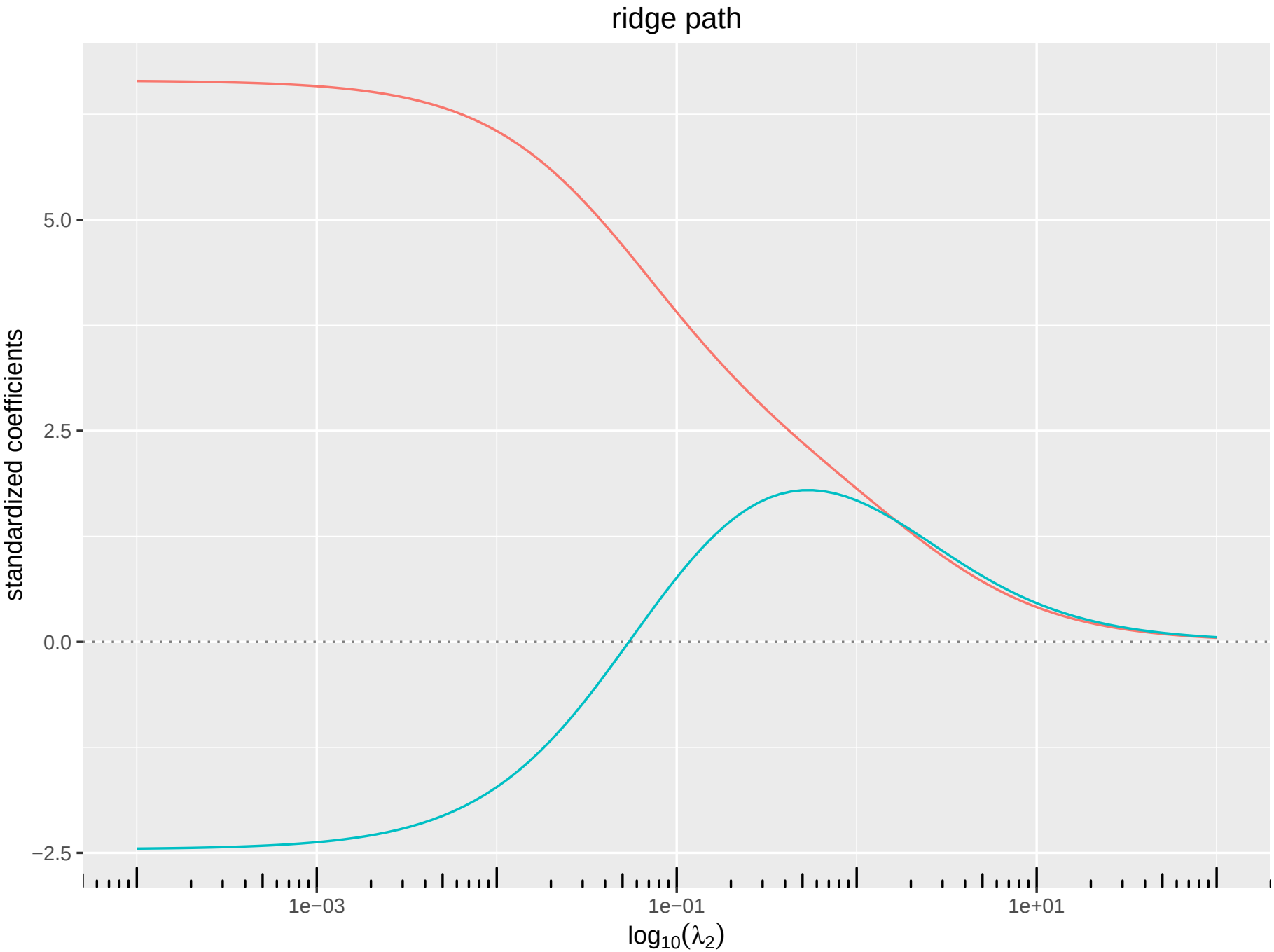
```
suppressMessages(library(quadrupen))
x1 <- rnorm(5)
x2 <- x1 + rnorm(5,0, 0.5)
cor(x1,x2)

## [1] 0.936967
```

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



La ridge comme un problème de régression pénalisée

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

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

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

Forme Lagrangienne (convexe)

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

Forte convexité

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

Ridge et données du cancer de la prostate

Calcul du chemin de solution

```
ridge.path <- glmnet(x.train,y.train, alpha=0)
```

Calcul de l'erreur de prédiction sur l'ensemble test pour tout λ

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

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

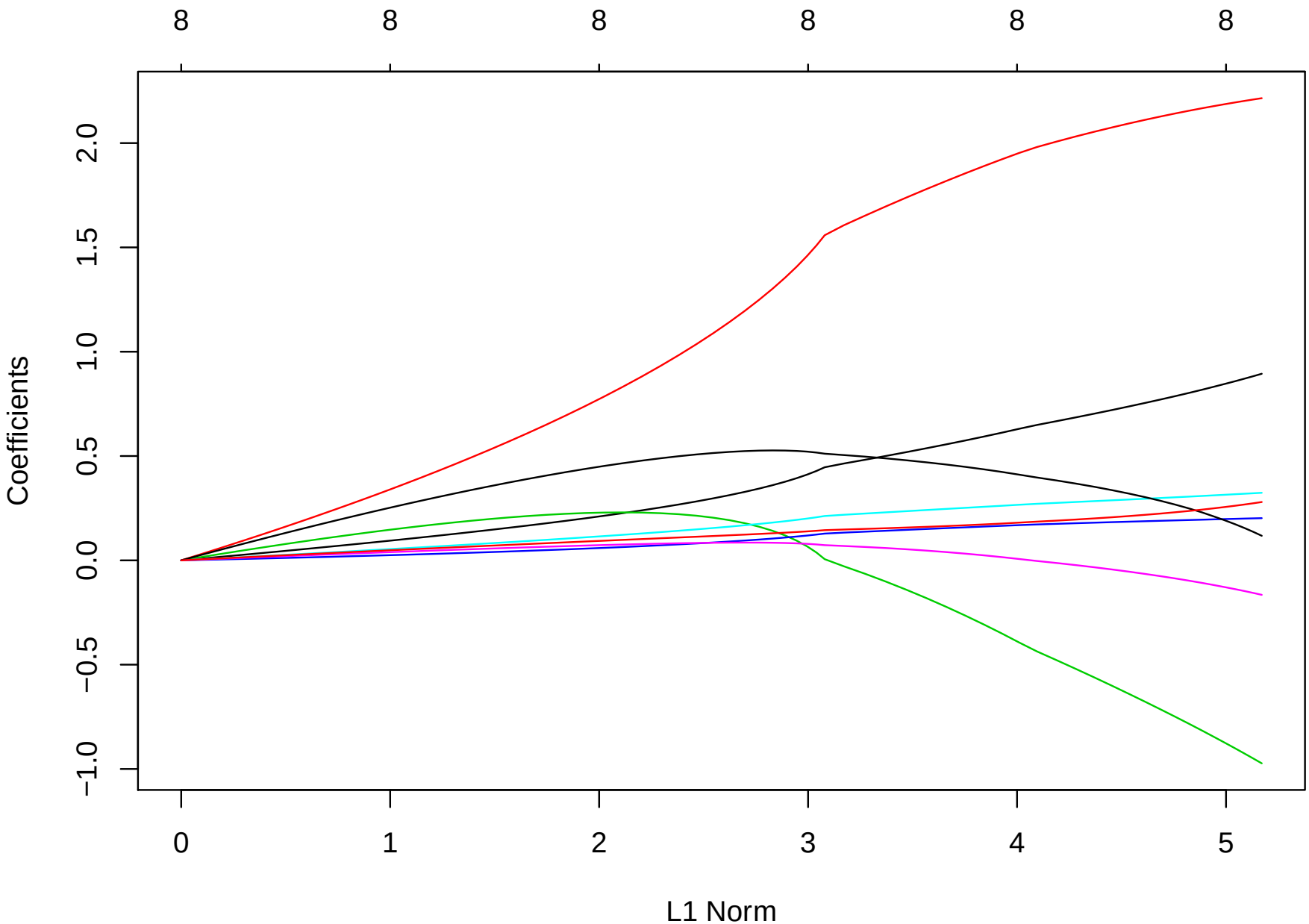
```
ridge.path$lambda[which.min(err)]

## [1] 0.2228282
```

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

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

##       s89
## 0.4866302
## [1] 0.5221043
```


8
8
8
8
8
8
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
Coefficients
0
1
2
3
4
5
L1 Norm

Plan

Les options classiques

En estimant l'erreur de prédiction

On choisit le λ minimisant l'erreur estimée

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

Par critère pénalisé

On choisit le λ minimisant le critère de forme générale

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

$\rightsquigarrow$ Quel sens donner aux degrés de liberté pour la Ridge?

Degrés de liberté effectifs

Idées

- Les degrés de liberté décrivent le niveau de complexité d'un modèle .
- Pour l'OLS, $\mathrm{df} = p$ (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

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

$\rightsquigarrow$ *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'écrive:

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

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

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

Ridge

Pour la régression Ridge, df est une fonction décroissant de λ qui tend vers 0 (ou 1 en considérant la constante):

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

Validation croisée

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

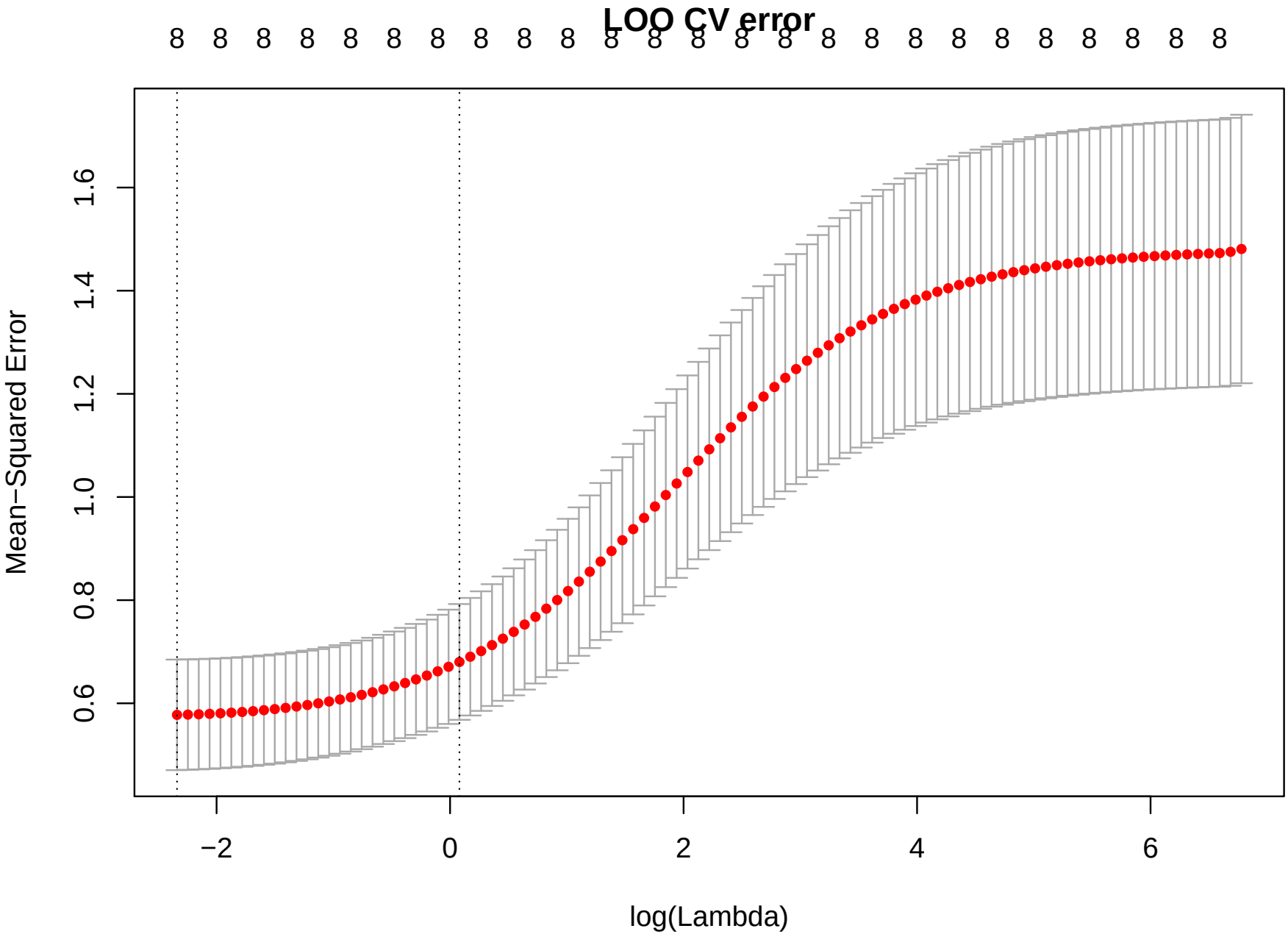
```
system.time(loo <- cv.glmnet(x.train,y.train,alpha=0,nfolds=n))
```

```
##    user  system elapsed
##   0.356   0.000   0.356
```

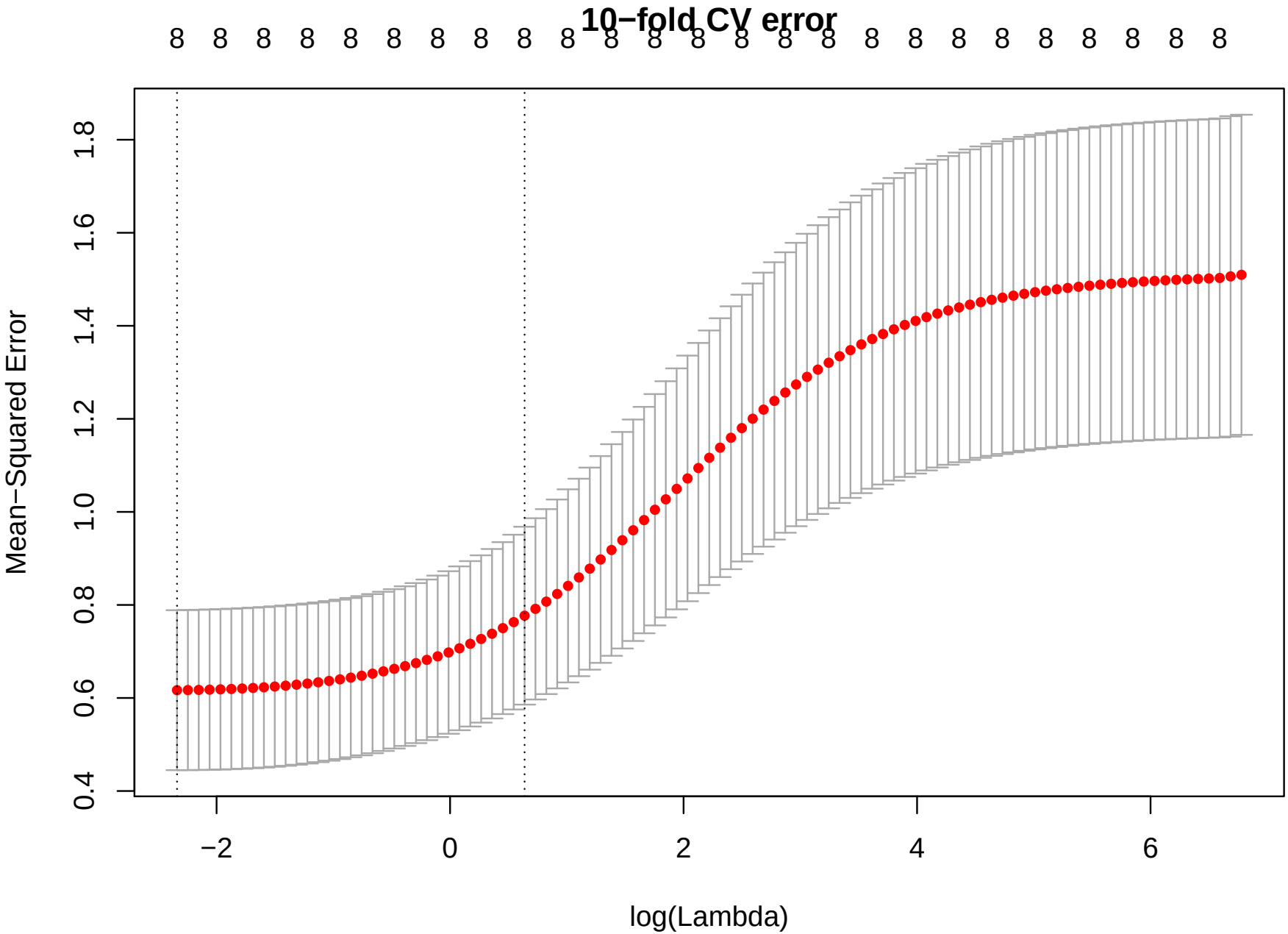
```
system.time(CV10 <- cv.glmnet(x.train,y.train,alpha=0,nfolds=10))
```

```
##    user  system elapsed
##    0.06    0.00    0.06
```

Validation croisée ("leave one out")



Validation croisée (”ten fold”)



Plan

Plan

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 $\hat{\boldsymbol{\beta}}$ à zéro).

Le Lasso comme problème d'optimisation

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

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

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

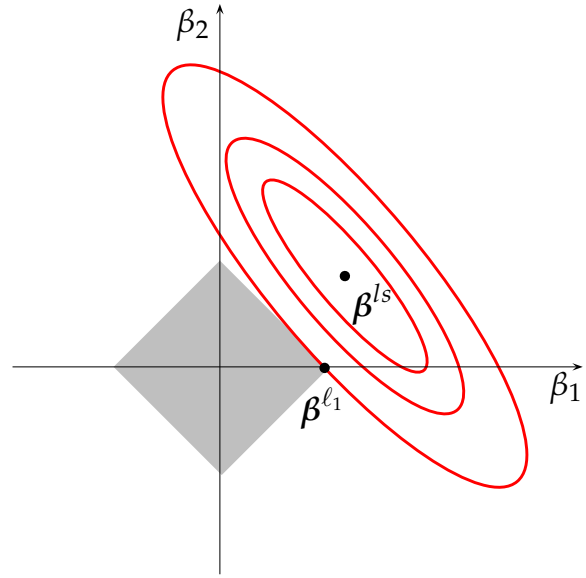
Proposer une contrainte qui force la parcimonie (en forçant des entrées de $\hat{\boldsymbol{\beta}}$ à zéro).

Le Lasso comme problème d'optimisation

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

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

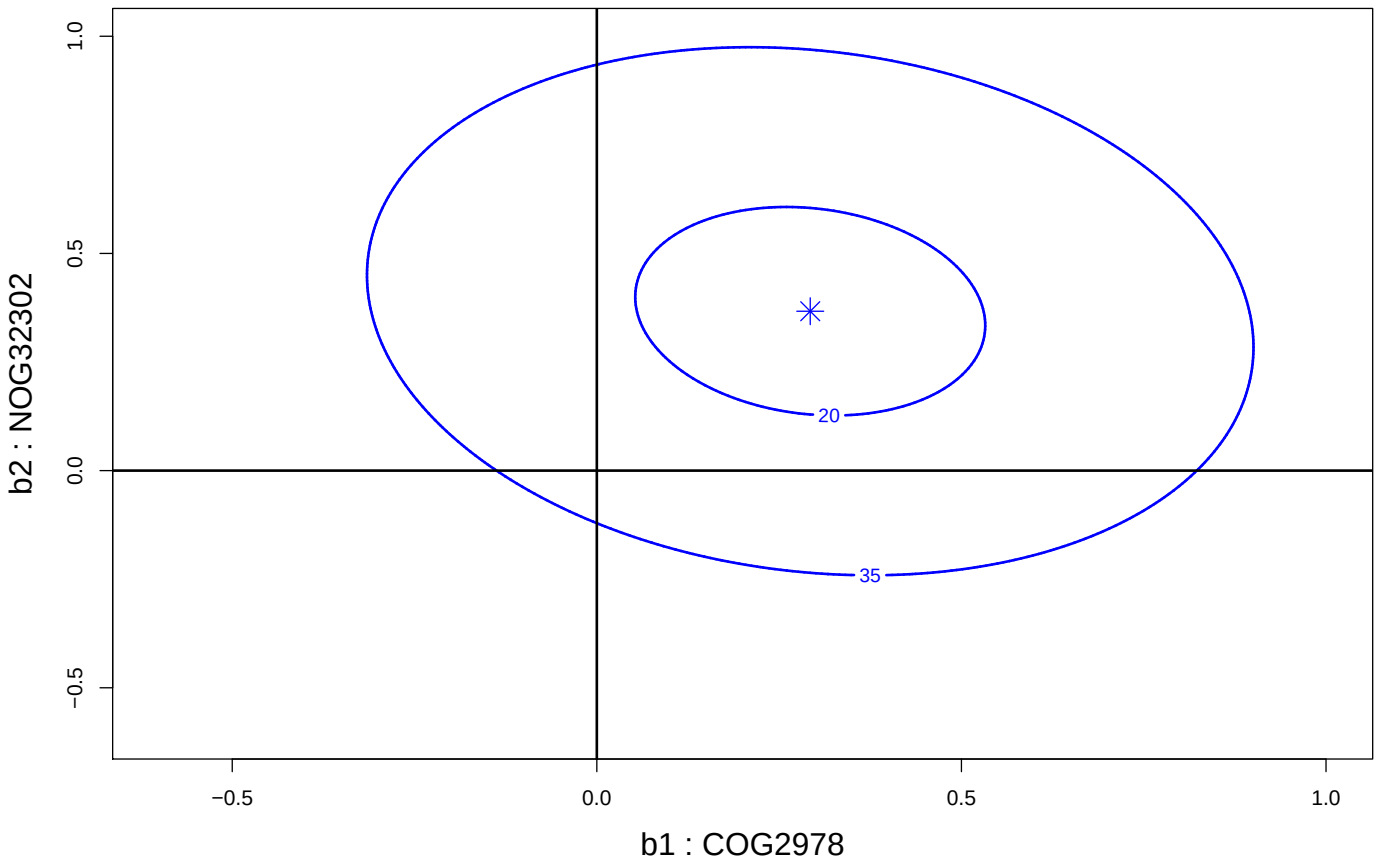
où s est un niveau de régularisation.



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

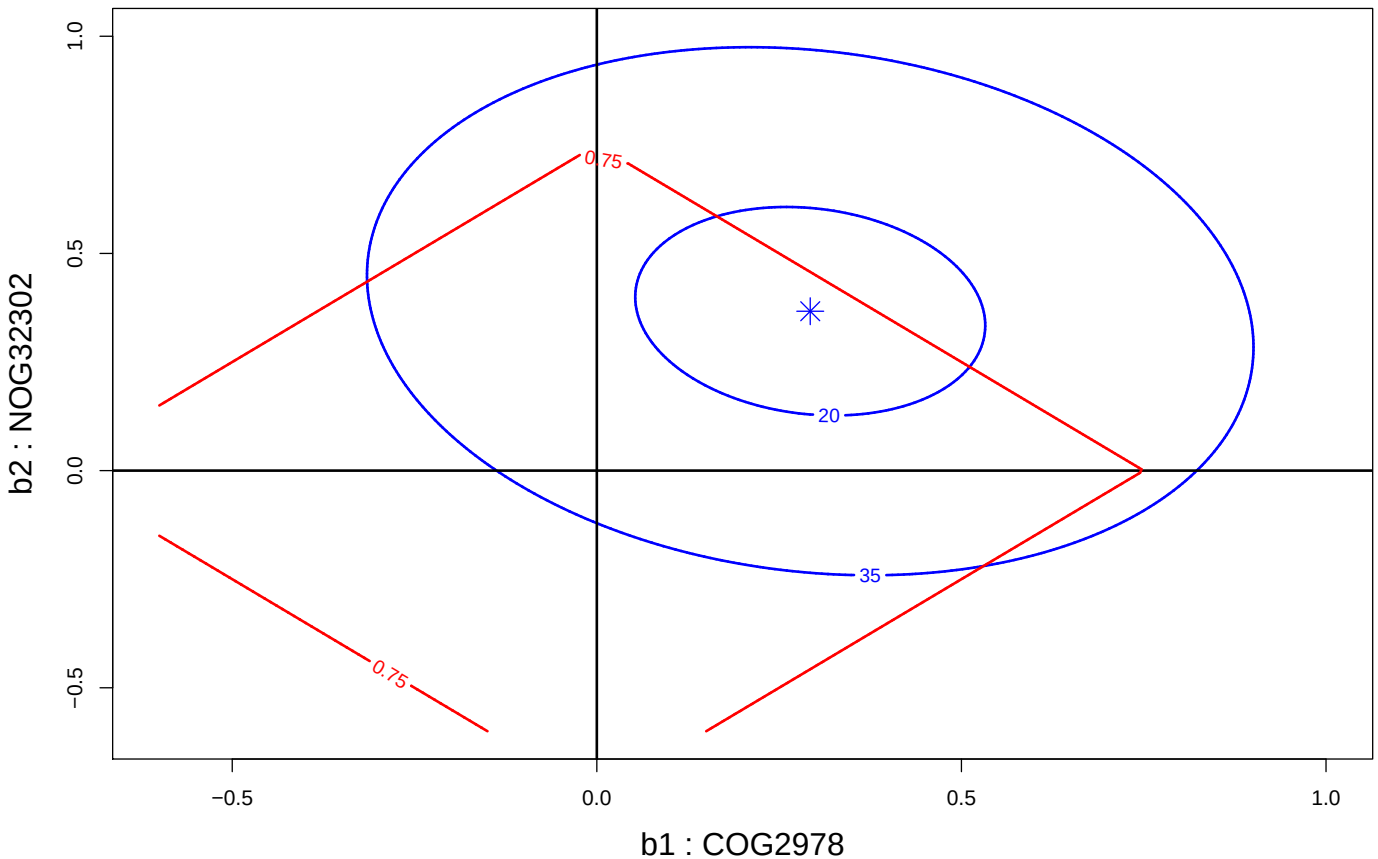
$$\sum_{i=1}^{n}(y_i - x_i^1\beta_1 - x_i^2\beta_2)^2, \qquad \text{pas de constrainte}$$



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

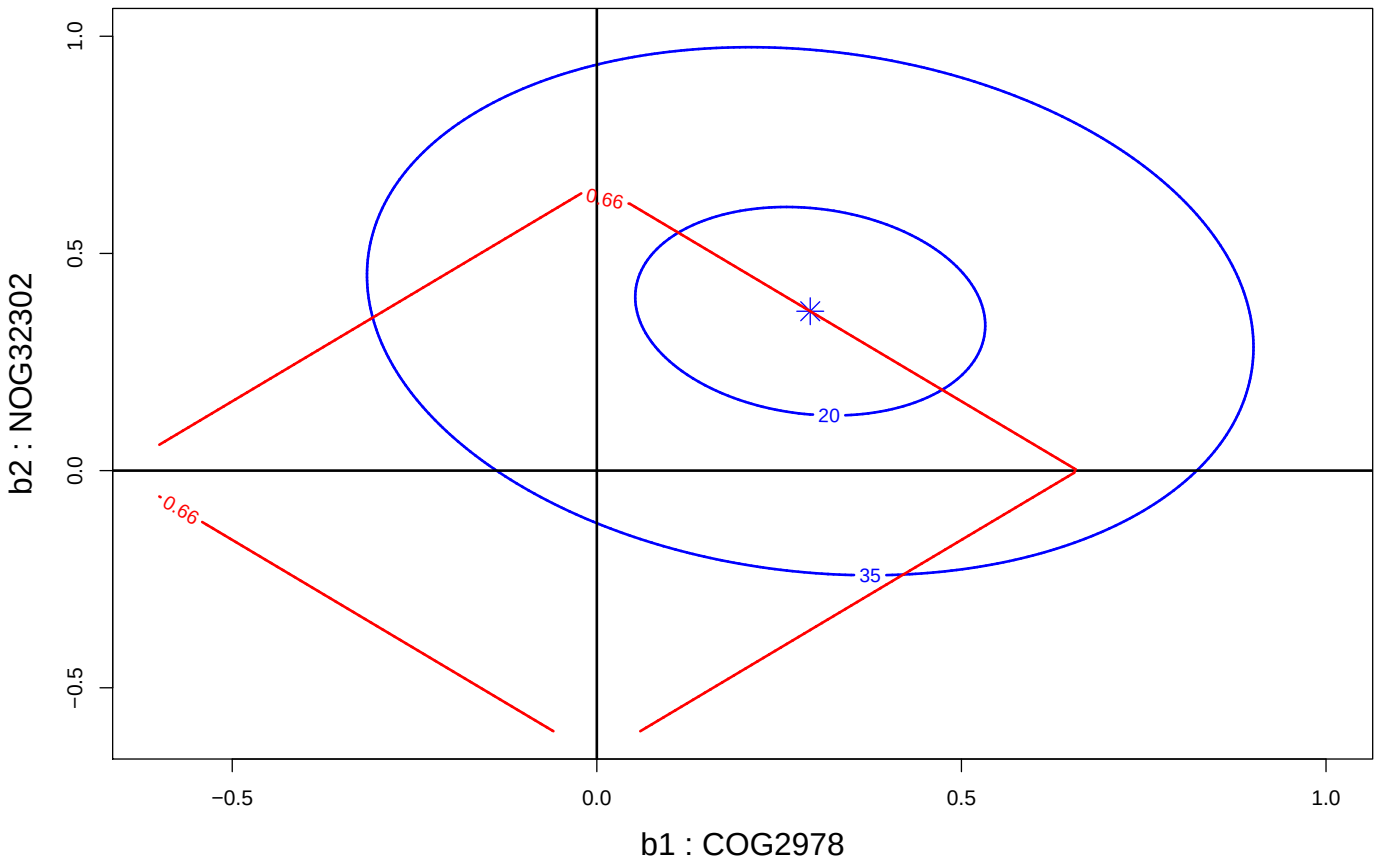
$$\sum_{i=1}^{n}(y_i - x_i^1\beta_1 - x_i^2\beta_2)^2, \qquad \text{s.c. } |\beta_1| + |\beta_2| < 0.75$$



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

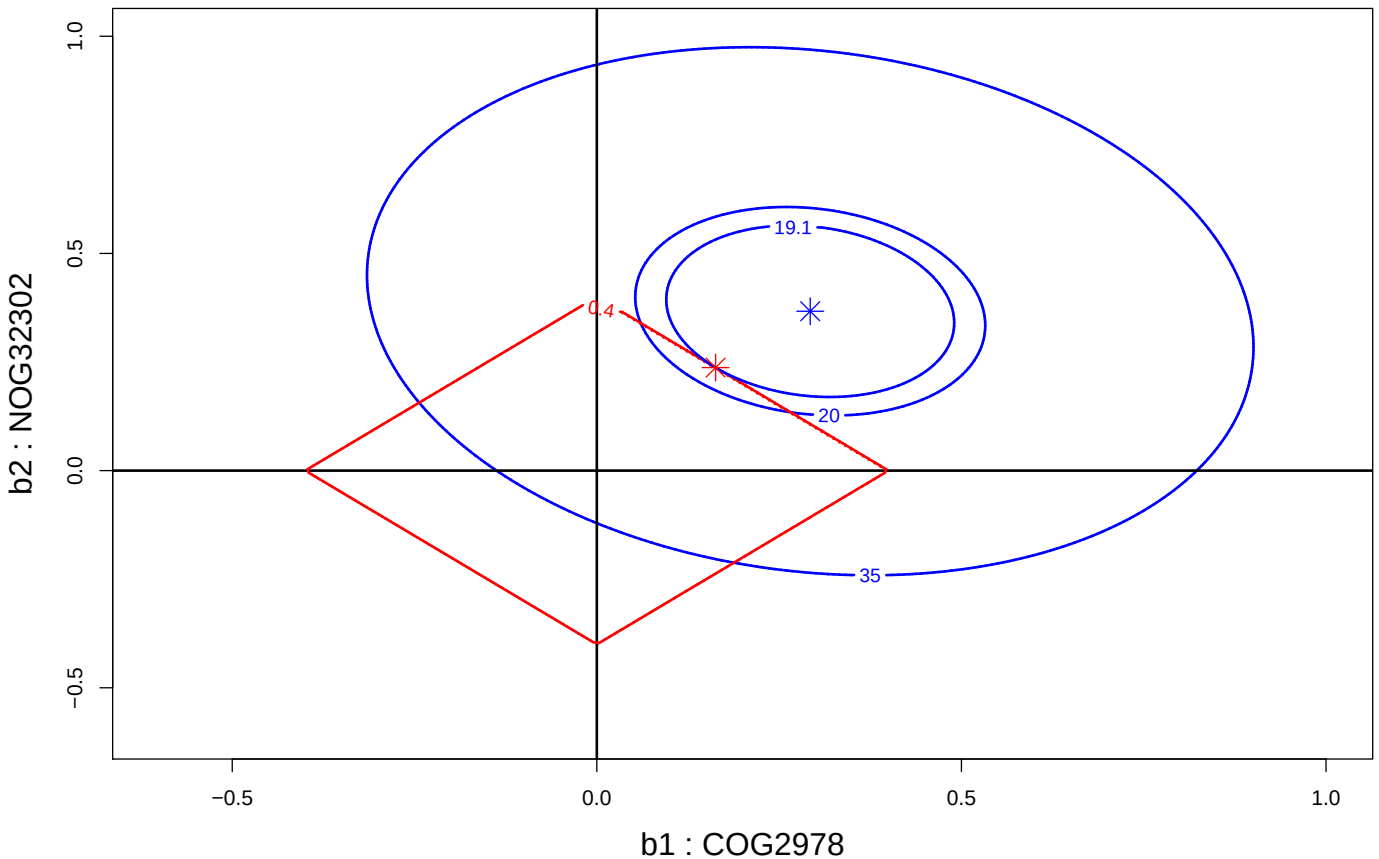
$$\sum_{i=1}^{n}(y_i - x_i^1\beta_1 - x_i^2\beta_2)^2, \qquad \text{s.c. } |\beta_1| + |\beta_2| < 0.66$$



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

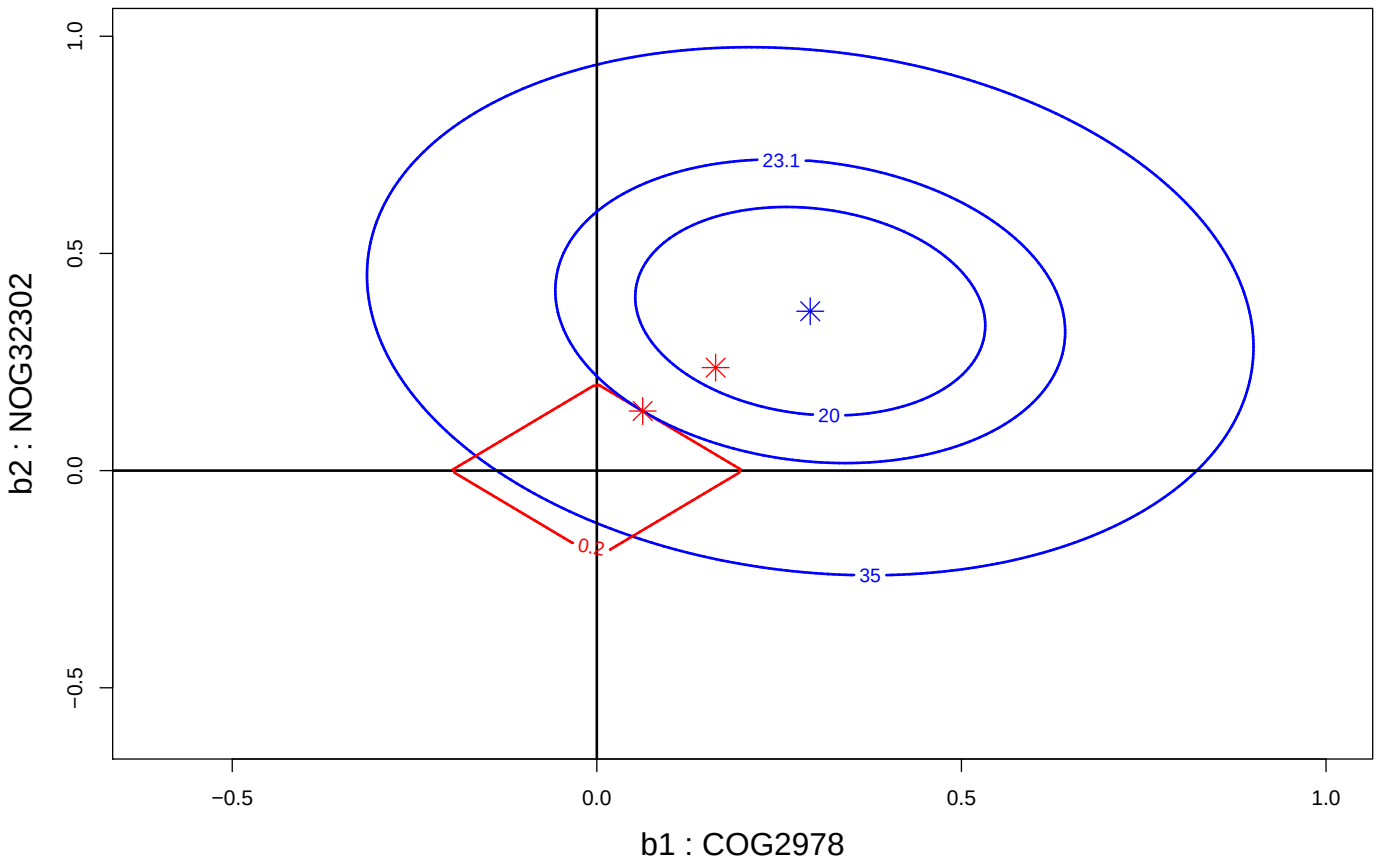
$$\sum_{i=1}^{n}(y_i - x_i^1\beta_1 - x_i^2\beta_2)^2, \qquad \text{s.c. } |\beta_1| + |\beta_2| < 0.4$$



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

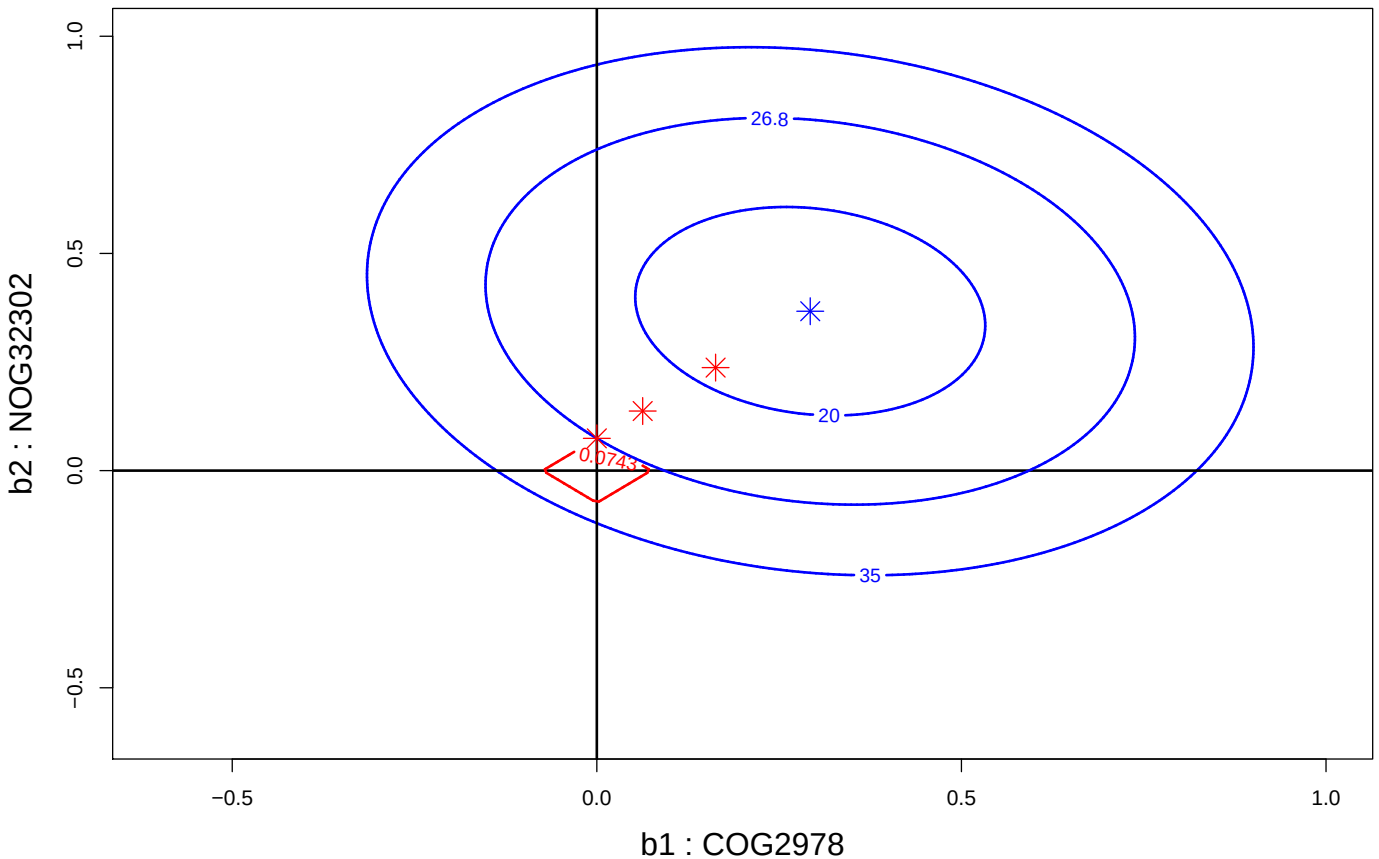
$$\sum_{i=1}^{n}(y_i - x_i^1\beta_1 - x_i^2\beta_2)^2, \qquad \text{s.c. } |\beta_1| + |\beta_2| < 0.2$$



Les singularités induisent de la "sparsité"

Figures de Sylvie Huet

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



Le Lasso comme méthode de régression pénalisée

On ne pénalise pas la constante, donc

- $\hat{\beta}_0 = \bar{\mathbf{y}}$,
- on centre $\mathbf{y}$ et $\mathbf{x}_j$, $j = 1, \ldots, p$,
- on normalise les prédicteurs avant d'ajuster,
- on renvoie $\hat{\boldsymbol{\beta}}$ dans l'échelle d'origine.

Résolution d'un problème d'optimisaiton convexe

$$\hat{\boldsymbol{\beta}}^{\text{lasso}} = \underset{\boldsymbol{\beta} \in \mathrm{R}^p}{\arg\min} \frac{1}{2} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|_1,$$

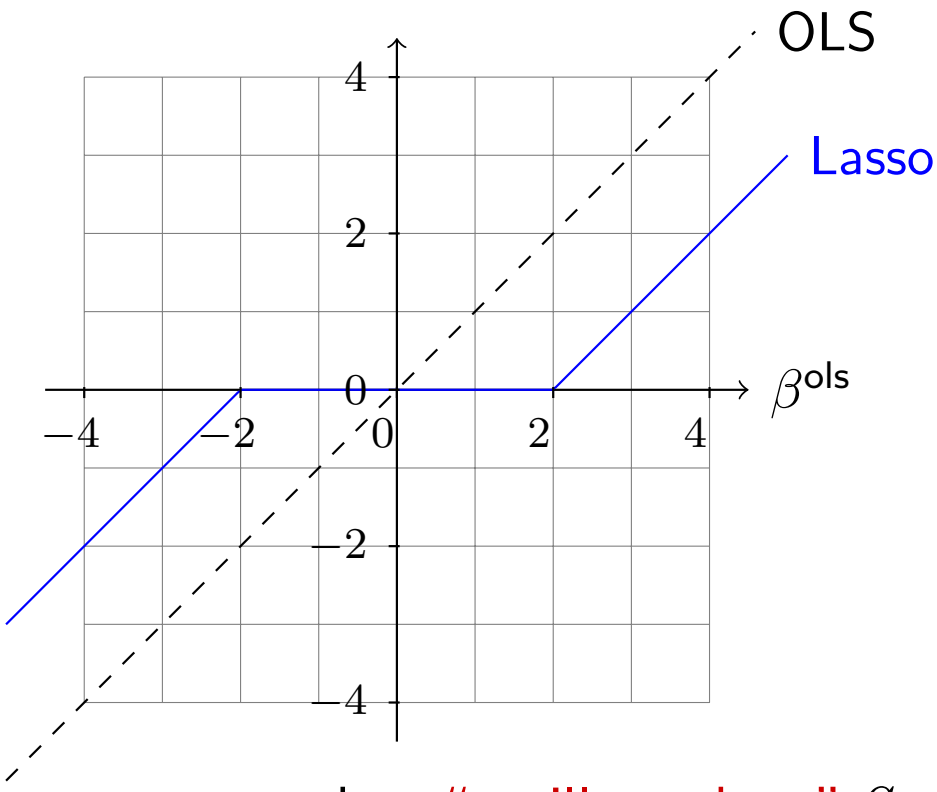
Pas de forme close, mais existe toujours et est unique lorsque $\mathbf{X}^\intercal \mathbf{X}$ est de plein rang.

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

Plan

Connexion avec l'OLS

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



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

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

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

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 calcuyl du chemin de solution

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

- construit un chemin *linéaire par morceau* de la solution en aprtant 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(glmnet)
lasso.path <- glmnet(x.train,y.train)
```

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

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

On choisit $\lambda^\star$ minimisant cette erreur

```
lasso.path$lambda[which.min(err)]

## [1] 0.1135118
```

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)]
```

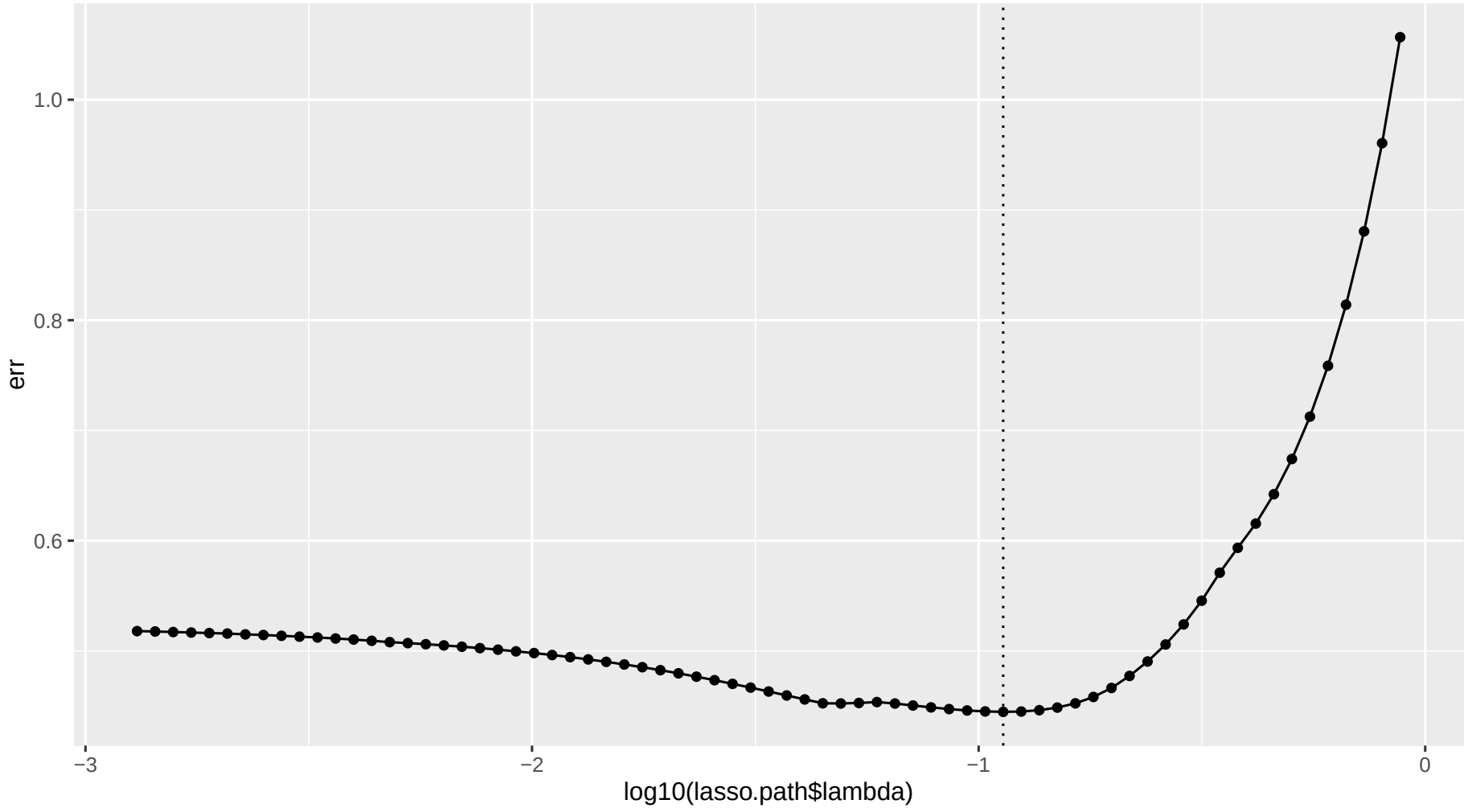
```
##       s22
## 0.4447306
```

```
lasso.path$beta[,which.min(err)]
```

```
##     lcavol    lweight        age       lbph        svi        lcp
## 0.83762993 1.74080154 0.00000000 0.09308703 0.18473598 0.00000000
##    gleason      pgg45
## 0.00000000 0.07755339
```

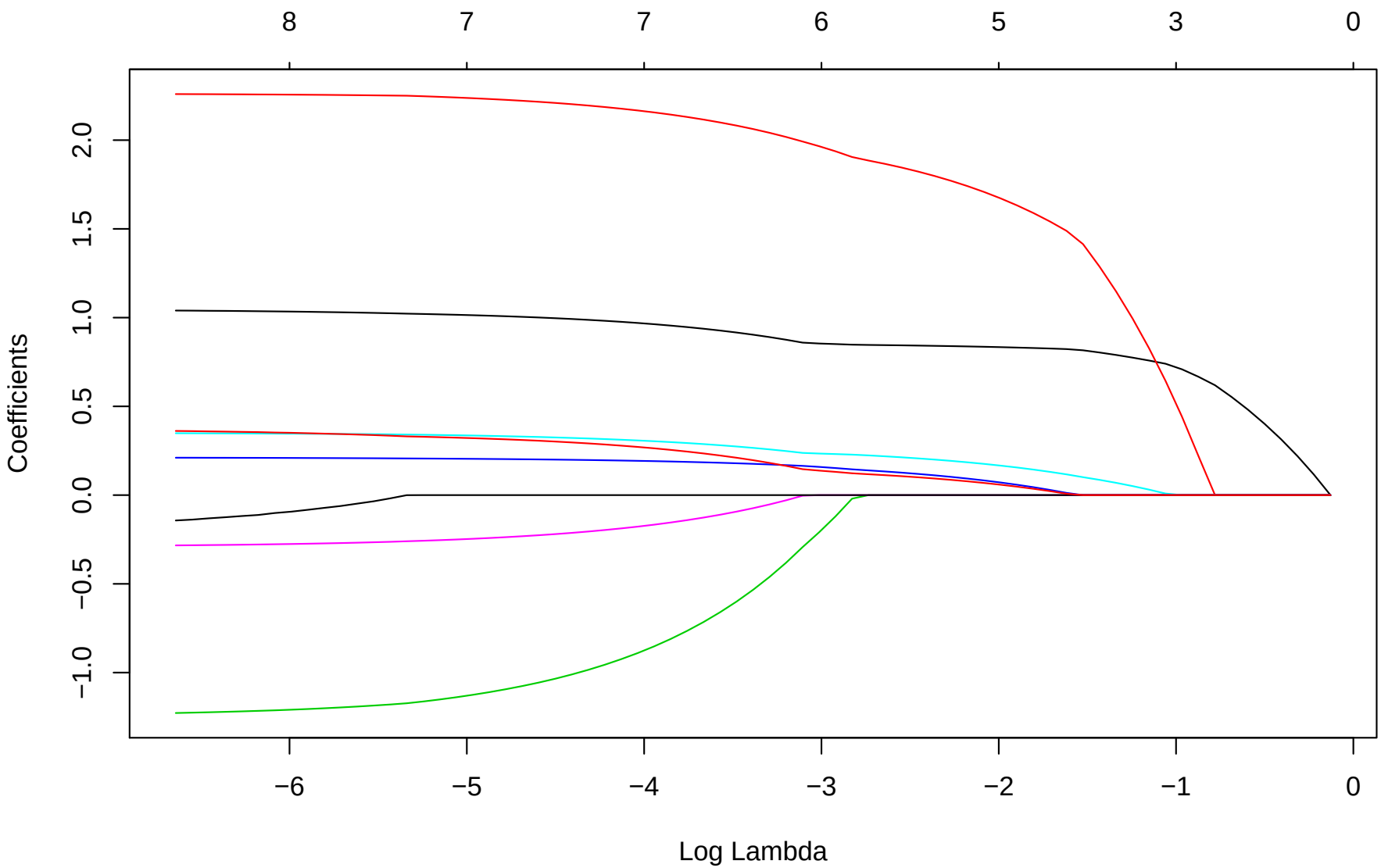
Erreur de prédiction sur les données test

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



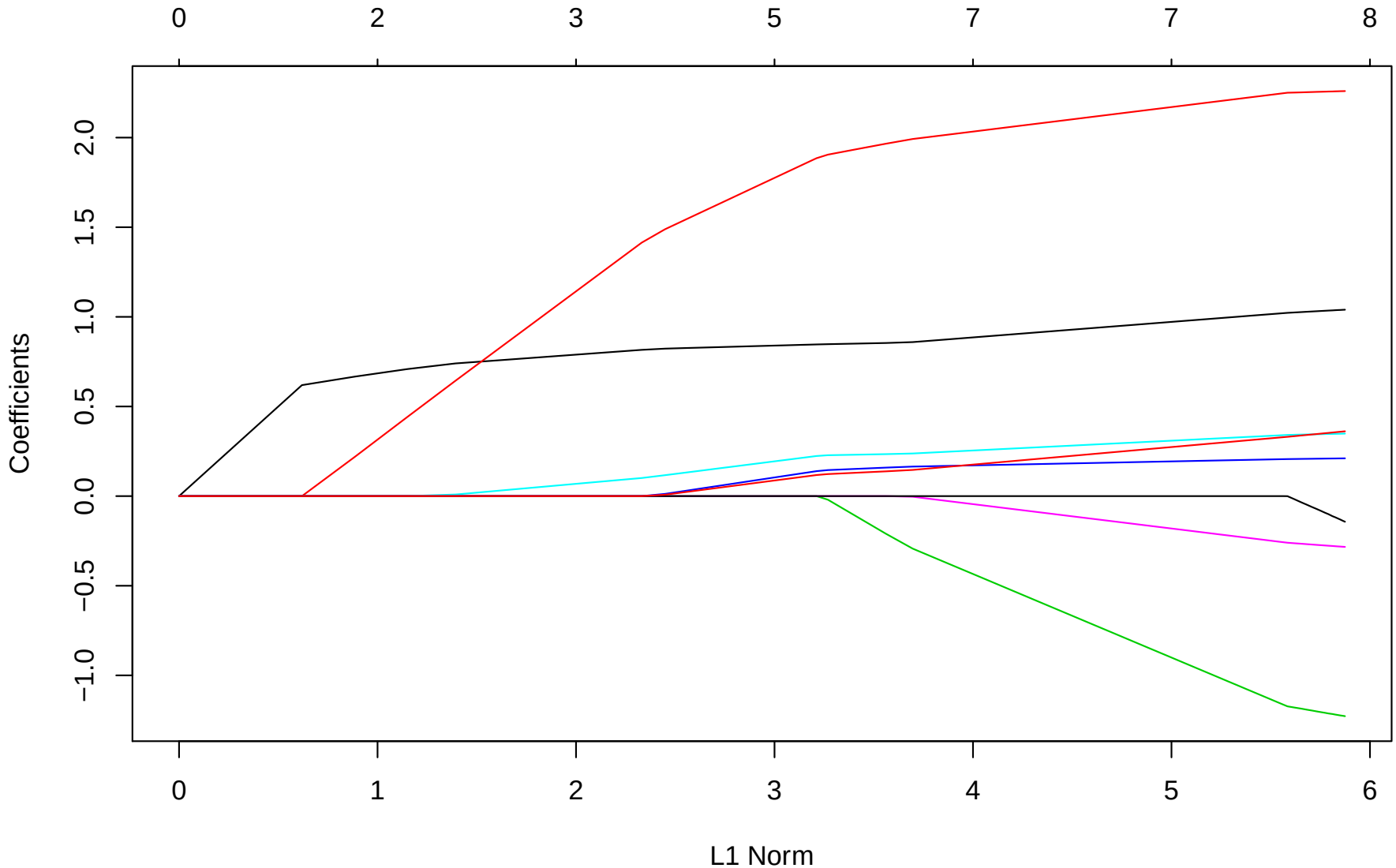
Chemin de solution (en fonction de λ)

```
plot(lasso.path, xvar="lambda")
```



Chemin de solution (proportion de régularisation s)

```
plot(lasso.path, xvar="norm")
```



Plan

Critères pénalisés

Degrés de liberté du LASSO

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

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

- Akaike Information Criterion équivalent au C_p en régression

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

- Bayesian Information Criterion

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

- modified BIC (lorsque $n < p$)

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

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

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

Validation croisée

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

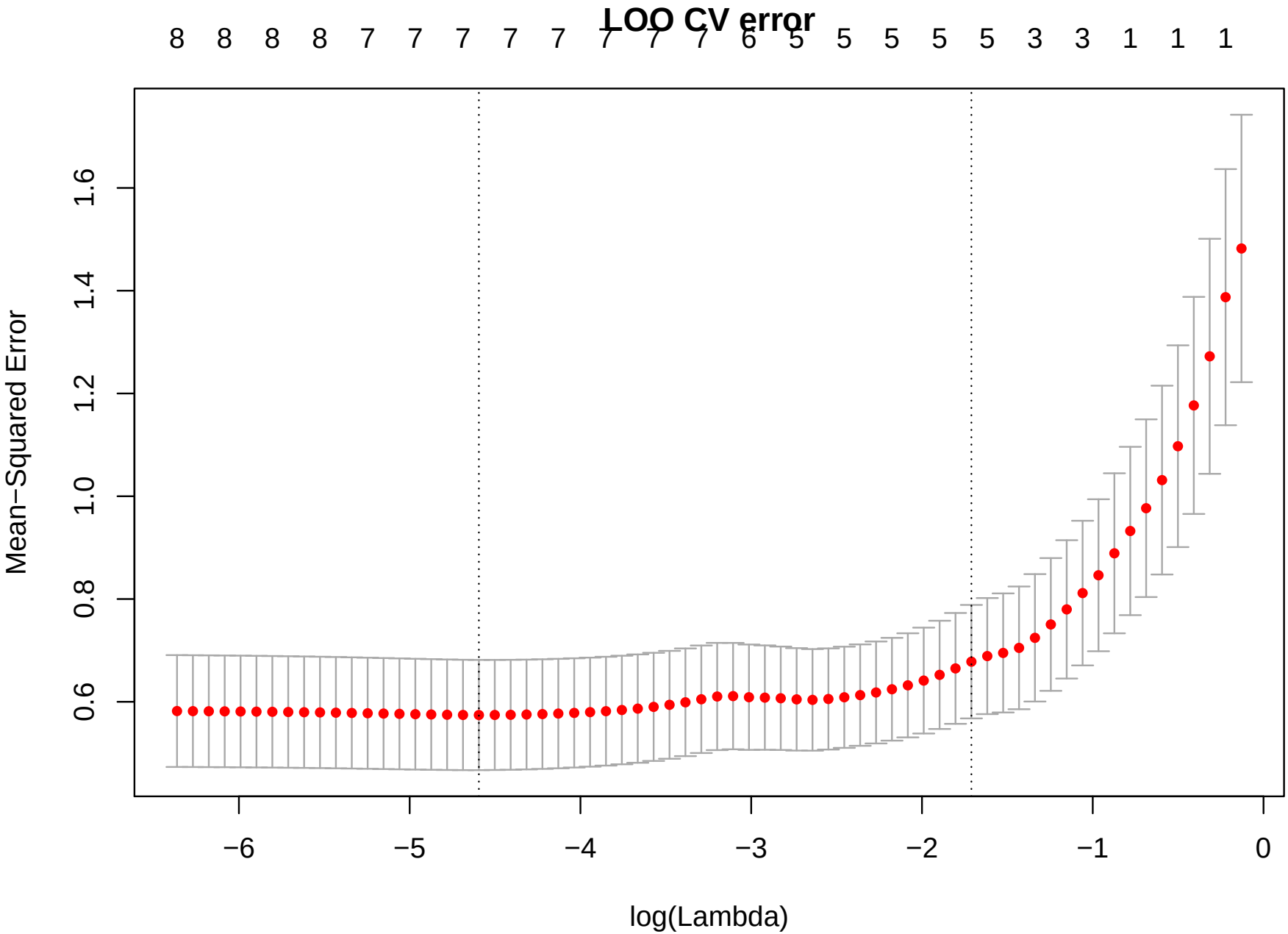
```
system.time(loo <- cv.glmnet(x.train,y.train,nfolds=n))
```

```
##    user  system elapsed
##   0.344   0.000   0.343
```

```
system.time(CV10 <- cv.glmnet(x.train,y.train,nfolds=10))
```

```
##    user  system elapsed
##   0.056   0.000   0.056
```

Validation croisée ("leave one out")



Validation croisée (”ten fold”)