

MS BGD

MDI 720 : Ridge / Tikhonov

Joseph Salmon
<http://josephsalmon.eu>
Télécom ParisTech

Plan

Définitions de l'estimateur Ridge

- Point de vue par SVD

- Point de vue par pénalisation

- Analyse du biais par la SVD

- Analyse de la variance par la SVD

Choix du paramètre de régularisation

- Notion de chemin de régularisation

- Validation Croisée (CV)

Algorithmes et aspects computationnels

Sommaire

Définitions de l'estimateur Ridge

- Point de vue par SVD

- Point de vue par pénalisation

- Analyse du biais par la SVD

- Analyse de la variance par la SVD

Choix du paramètre de régularisation

- Notion de chemin de régularisation

- Validation Croisée (CV)

Algorithmes et aspects computationnels

Rappel

$$\mathbf{y} = X\boldsymbol{\theta}^* + \boldsymbol{\varepsilon}$$

- ▶ $\mathbf{y} \in \mathbb{R}^n$ est le vecteur des observations
- ▶ $X \in \mathbb{R}^{n \times p}$ est la matrice des variables explicatives
- ▶ $\boldsymbol{\theta}^* \in \mathbb{R}^p$ est le **vrai** paramètre du modèle que l'on veut retrouver.
- ▶ $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ est le bruit

Rem: possiblement une variable supplémentaire pour la constante

La décomposition en valeur singulières

Théorème : Golub et Van Loan (2013)

Pour toute matrice $X \in \mathbb{R}^{n \times p}$, il existe deux matrices orthogonales $U = [\mathbf{u}_1, \dots, \mathbf{u}_n] \in \mathbb{R}^{n \times n}$ et $V = [\mathbf{v}_1, \dots, \mathbf{v}_p] \in \mathbb{R}^{p \times p}$, telles que

$$U^\top X V = \text{diag}(s_1, \dots, s_{\text{rg}(X)}) = \Sigma \in \mathbb{R}^{n \times p}$$

avec $s_1 \geq s_2 \geq \dots \geq s_{\text{rg}(X)} > 0$, avec $\text{rg}(X) = \text{rang}(X)$.

$$X = U \Sigma V^\top \Leftrightarrow X = \sum_{i=1}^{\text{rg}(X)} s_i \mathbf{u}_i \mathbf{v}_i^\top$$

Une solution des moindres carrés est alors :

$$\hat{\boldsymbol{\theta}}^{\text{MCO}} = X^+ \mathbf{y} = \sum_{i=1}^{\text{rg}(X)} \frac{1}{s_i} \mathbf{v}_i \mathbf{u}_i^\top \mathbf{y}$$

Retour sur les problèmes numériques

$$\hat{\boldsymbol{\theta}}^{\text{MCO}} = X^+ \mathbf{y} = \sum_{i=1}^{\text{rg}(X)} \frac{1}{s_i} \mathbf{v}_i \mathbf{u}_i^\top \mathbf{y}$$

Si les plus petites valeurs singulières s_i s'approchent de zéro alors la solution numérique de la SVD n'est pas stable !

Rem: le défaut n'est pas propre aux moindres carrés, mais est inhérent aux problème difficiles (on dit aussi “mal posé” en analyse numérique et en traitement du signal)

Les équations normales

Une solution $\boldsymbol{\theta}$ des moindres carrés doit vérifier :

$$X^{\top} X \boldsymbol{\theta} = X^{\top} \mathbf{y} \Leftrightarrow V \Sigma^{\top} \Sigma V^{\top} \boldsymbol{\theta} = V \Sigma^{\top} U^{\top} \mathbf{y}$$

et si l'on cherche $\boldsymbol{\theta}$ sous la forme $\boldsymbol{\theta} = V \boldsymbol{\beta}$, c'est équivalent à

$$\Sigma^{\top} \Sigma \boldsymbol{\beta} = \Sigma^{\top} U^{\top} \mathbf{y}$$

$\Sigma^{\top} \Sigma$ diagonale avec $r = \text{rang}(X)$ éléments non nuls qui sont les s_i^2

$$\Sigma^{\top} \Sigma = \left[\begin{array}{cc|c} s_1^2 & & 0 \\ & \ddots & \\ 0 & & s_r^2 \\ \hline & & 0 \\ & 0 & 0 \end{array} \right] \in \mathbb{R}^{p \times p}$$

Les équations normales (suite)

Alternative régularisée : résoudre les équations normales

$$\left[\begin{array}{ccc|c} s_1^2 & & 0 & 0 \\ & \ddots & & \\ 0 & & s_r^2 & \\ \hline & 0 & & 0 \end{array} \right] \text{ remplacé par } \left[\begin{array}{ccc|c} s_1^2 & & 0 & 0 \\ & \ddots & & \\ 0 & & s_r^2 & \\ \hline & 0 & & 0 \end{array} \right] + \lambda \text{Id}_p$$

De manière synthétique cela s'écrit :

$$(\lambda \text{Id}_p + \Sigma^\top \Sigma) \boldsymbol{\beta} = \Sigma^\top U^\top \mathbf{y}$$

i.e., on ajoute à toutes les valeurs propres de $X^\top X$ un terme $\lambda > 0$ "petit", λ est nommé **paramètre de régularisation**

$$\boldsymbol{\beta} = (\lambda \text{Id}_p + \Sigma^\top \Sigma)^{-1} \Sigma^\top U^\top \mathbf{y}$$

et donc

$$\boldsymbol{\theta} = V(\lambda \text{Id}_p + \Sigma^\top \Sigma)^{-1} \Sigma^\top U^\top \mathbf{y}$$

Ridge forme explicite

Avec la SVD, l'équation suivante se simplifie :

$$\boldsymbol{\theta} = V(\lambda \text{Id}_p + \Sigma^\top \Sigma)^{-1} \Sigma^\top U^\top \mathbf{y}$$

Cela donne une première forme de l'estimateur *Ridge*

$$\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} = (\lambda \text{Id}_p + X^\top X)^{-1} X^\top \mathbf{y}$$

Rappel : sous l'hypothèse de plein rang $\hat{\boldsymbol{\theta}}^{\text{MCO}} = (X^\top X)^{-1} X^\top \mathbf{y}$

Rem:

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

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

Astuce du noyau (Kernel trick)

Astuce du noyau : Selon si $n > p$ ou $n \leq p$, une méthode qui cherche à trouver une solution des moindres carrés par inversion peut préférer l'une des deux formulations suivantes :

$$X^{\top}(XX^{\top} + \lambda \text{Id}_n)^{-1}\mathbf{y} = (X^{\top}X + \lambda \text{Id}_p)^{-1}X^{\top}\mathbf{y}$$

- ▶ membre de gauche : on inverse une matrice $n \times n$
- ▶ membre de droite : on inverse une matrice $p \times p$

Rem: cette propriété est aussi très utile pour les méthodes à noyaux de type SVM (*cf.* cours de *machine learning*)

Exo: Démontrer la propriété précédente avec la SVD

Sommaire

Définitions de l'estimateur Ridge

Point de vue par SVD

Point de vue par pénalisation

Analyse du biais par la SVD

Analyse de la variance par la SVD

Choix du paramètre de régularisation

Notion de chemin de régularisation

Validation Croisée (CV)

Algorithmes et aspects computationnels

Ridge / Tikhonov : la définition pénalisée

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

- ▶ Noter que l'estimateur *Ridge* est **unique** pour un λ fixé

- ▶ On retrouve de nouveau les cas limites :

$$\lim_{\lambda \rightarrow 0} \hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \hat{\boldsymbol{\theta}}^{\text{MCO}} \text{ (solution de norme } \|\cdot\|_2 \text{ minimale)}$$

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

- ▶ Lien des deux formulations par les conditions du premier

ordre : pour $f(\boldsymbol{\theta}) = \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2/2 + \lambda \|\boldsymbol{\theta}\|_2^2/2$

$$\nabla f(\boldsymbol{\theta}) = X^{\top}(X\boldsymbol{\theta} - \mathbf{y}) + \lambda\boldsymbol{\theta} = 0 \Leftrightarrow (X^{\top}X + \lambda \text{Id}_p)\boldsymbol{\theta} = X^{\top}\mathbf{y}$$

Rem: version codée dans `sklearn`

Interprétation contrainte

Un problème de la forme “Lagrangienne” suivante :

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

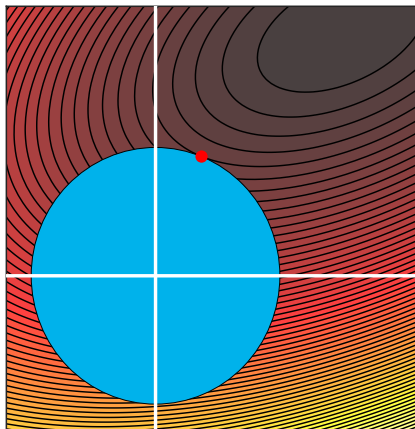
admet pour un certain $T > 0$ la même solution que :

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

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

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

Lignes de niveau et ensemble de contraintes



Optimisation sous contraintes ℓ_2

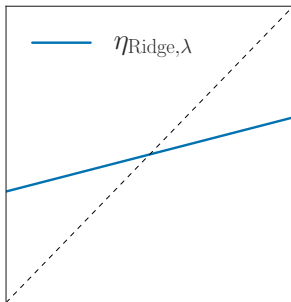
Le cas orthogonal

Retour sur un cas simple $X^\top X = \text{Id}_p$

$$\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} = (\lambda \text{Id}_p + X^\top X)^{-1} X^\top \mathbf{y}$$

$$\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}} = (\lambda \text{Id}_p + \text{Id}_p)^{-1} X^\top \mathbf{y} = \frac{1}{\lambda + 1} X^\top \mathbf{y}$$

$$\hat{\mathbf{y}} = \frac{1}{\lambda + 1} \mathbf{y} = (\eta_{\text{rdg},\lambda}(\mathbf{y}_i))_{i=1,\dots,n}$$



Rem: la fonction réelle $\eta_{\text{rdg},\lambda}$ est une contraction linéaire (shrinkage)

Prédiction associée


Partant du coefficient *Ridge* :

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = (\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \mathbf{y}$$

la prédiction associée s'obtient ainsi :

$$\hat{\mathbf{y}} = X \hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = X(\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} \mathbf{y}$$

Rem: l'estimateur $\hat{\mathbf{y}}$ est toujours linéaire en \mathbf{y}


Rem: l'équivalent de la matrice chapeau ( : *hat matrix*) est

$$H_{\lambda} = X(\lambda \text{Id}_p + X^{\top} X)^{-1} X^{\top} = \sum_{j=1}^{\text{rg}(X)} \frac{s_j^2}{s_j^2 + \lambda} \mathbf{u}_j \mathbf{u}_j^{\top}. \text{ Cependant, si}$$

$\lambda \neq 0$, on n'a plus $H_{\lambda}^2 = H_{\lambda} = \sum_{j=1}^{\text{rg}(X)} \mathbf{u}_j \mathbf{u}_j^{\top}$, H_{λ} n'est donc pas un projecteur en général.

Point normalisation

Rappel : normaliser les p variables de la même manière pour que la pénalisation contraigne de manière similaire toutes les variables

- ▶ centrer l'observation et les variables explicatives \Rightarrow pas de coefficient pour la variable constante (donc pas de contrainte)
- ▶ ne pas centrer les variables explicatives \Rightarrow ne pas mettre de contrainte sur la variable constante ( : *bias*)

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta} - \theta_0 \mathbf{1}_n\|_2^2 + \lambda \sum_{j=1}^p \theta_j^2$$

Alternative (si l'on n'a pas normalisé) : changer la pénalité en

$$\arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p} \|\mathbf{y} - X\boldsymbol{\theta}\|_2^2 + \lambda \sum_{j=1}^p \alpha_j \theta_j^2 \quad (\text{e.g., } \alpha_j = \|\mathbf{x}_j\|_2^2)$$

Rem: pour la validation croisée on utilisera plus naturellement $\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2n}$ que $\frac{\|\mathbf{y} - X\boldsymbol{\theta}\|_2^2}{2}$ pour l'attache aux données

Normalisation (bis)

Ridge avec variables $X' = [\mathbf{x}'_1, \dots, \mathbf{x}'_K]$ dont il existe une relation de linéarité $\sum_{k=1}^K \mu_k \mathbf{x}'_k = \mathbf{1}$ (e.g., des variables qualitatives normalisées ou non) :

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^p, \boldsymbol{\theta}' \in \mathbb{R}^K} \|\mathbf{y} - X\boldsymbol{\theta} - X'\boldsymbol{\theta}' - \theta_0 \mathbf{1}_n\|_2^2 + \lambda \sum_{j=1}^p \theta_j^2 + \lambda \sum_{k=1}^K \theta_k'^2$$

est équivalent à résoudre le programme :

$$\hat{\boldsymbol{\theta}}_{\lambda}^{\text{rdg}} = \arg \min_{\substack{\boldsymbol{\theta} \in \mathbb{R}^p, \boldsymbol{\theta}' \in \mathbb{R}^K \\ \sum_{k=1}^K \theta_k' = 0}} \|\mathbf{y} - X\boldsymbol{\theta} - X'\boldsymbol{\theta}' - \theta_0 \mathbf{1}_n\|_2^2 + \lambda \sum_{j=1}^p \theta_j^2 + \lambda \sum_{k=1}^K \theta_k'^2$$

Exo: Preuve du résultat ; aide : cf. [Park \(2006\) p.35](#)

Sommaire

Définitions de l'estimateur Ridge

Point de vue par SVD

Point de vue par pénalisation

Analyse du biais par la SVD

Analyse de la variance par la SVD

Choix du paramètre de régularisation

Notion de chemin de régularisation

Validation Croisée (CV)

Algorithmes et aspects computationnels

Le biais dans le cas général

Sous l'hypothèse de bruit "blanc" $\mathbf{y} = X\boldsymbol{\theta}^\star + \boldsymbol{\varepsilon}$ avec $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$:

$$\begin{aligned}\mathbb{E}(\hat{\boldsymbol{\theta}}_\lambda^{\text{rdg}}) &= \mathbb{E}((\lambda \text{Id}_p + X^\top X)^{-1} X^\top \mathbf{y}) \\ &= \mathbb{E}((\lambda \text{Id}_p + X^\top X)^{-1} X^\top X \boldsymbol{\theta}^\star + (\lambda \text{Id}_p + X^\top X)^{-1} X^\top \boldsymbol{\varepsilon}) \\ &= (\lambda \text{Id}_p + X^\top X)^{-1} X^\top X \boldsymbol{\theta}^\star \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^2}{s_i^2 + \lambda} \mathbf{v}_i \mathbf{v}_i^\top \boldsymbol{\theta}^\star\end{aligned}$$

Rem: on retrouve $\mathbb{E}(\hat{\boldsymbol{\theta}}^{\text{MCO}}) = \sum_{i=1}^{\text{rg}(X)} \mathbf{v}_i \mathbf{v}_i^\top \boldsymbol{\theta}^\star$ quand $\lambda \rightarrow 0$

Rem: le biais vaut $-\lambda(X^\top X + \lambda \text{Id}_p)^{-1} \boldsymbol{\theta}^\star$

Sommaire

Définitions de l'estimateur Ridge

Point de vue par SVD

Point de vue par pénalisation

Analyse du biais par la SVD

Analyse de la variance par la SVD

Choix du paramètre de régularisation

Notion de chemin de régularisation

Validation Croisée (CV)

Algorithmes et aspects computationnels

Variance dans le cas général

Sous l'hypothèse de bruit "blanc" (*i.e.*, $\mathbb{E}(\varepsilon) = 0$) et de modèle homoscédastique : $\mathbb{E}(\varepsilon\varepsilon^\top) = \sigma^2 \text{Id}_n$

Variance / Covariance

$$V_\lambda^{\text{rdg}} = \mathbb{E} \left((\hat{\theta}_\lambda^{\text{rdg}} - \mathbb{E}(\hat{\theta}_\lambda^{\text{rdg}})) (\hat{\theta}_\lambda^{\text{rdg}} - \mathbb{E}(\hat{\theta}_\lambda^{\text{rdg}}))^\top \right)$$

Calcul explicite :

$$\begin{aligned} V_\lambda^{\text{rdg}} &= \mathbb{E}((\lambda \text{Id}_p + X^\top X)^{-1} X^\top \varepsilon \varepsilon^\top X (\lambda \text{Id}_p + X^\top X)^{-1}) \\ &= (\lambda \text{Id}_p + X^\top X)^{-1} X^\top \mathbb{E}(\varepsilon \varepsilon^\top) X (\lambda \text{Id}_p + X^\top X)^{-1} \\ &= \sigma^2 (\lambda \text{Id}_p + X^\top X)^{-2} X^\top X \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^2 \sigma^2}{(s_i^2 + \lambda)^2} \mathbf{v}_i \mathbf{v}_i^\top \end{aligned}$$

Rem: on retrouve $V^{\text{MCO}} = \sum_{i=1}^{\text{rg}(X)} \frac{\sigma^2}{s_i^2} \mathbf{v}_i \mathbf{v}_i^\top$ quand $\lambda \rightarrow 0$

Rem: on retrouve une variance nulle quand $\lambda \rightarrow \infty$

Risque de prédiction

Hypothèse de modèle homoscedastique : $\mathbb{E}(\epsilon\epsilon^\top) = \sigma^2 \text{Id}_n$

Risque (quadratique) de prédiction $\mathbb{E}\|X\theta^\star - X\hat{\theta}_\lambda^{\text{rdg}}\|^2$

Sous l'hypothèse de modèle homoscedastique :

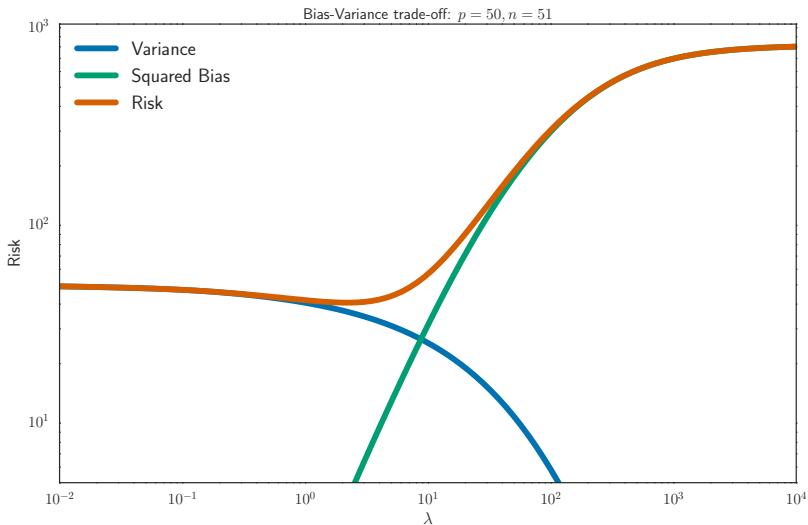
$$R_{\text{pred}}(\theta^\star, \hat{\theta}_\lambda^{\text{rdg}}) = \mathbb{E} \left[(\hat{\theta}_\lambda^{\text{rdg}} - \theta^\star)^\top (X^\top X) (\hat{\theta}_\lambda^{\text{rdg}} - \theta^\star) \right]$$

Calcul explicite (début identique) :

$$\begin{aligned} R_{\text{pred}}(\theta^\star, \hat{\theta}_\lambda^{\text{rdg}}) &= \mathbb{E} \left[(\hat{\theta}_\lambda^{\text{rdg}} - \theta^\star)^\top (X^\top X) (\hat{\theta}_\lambda^{\text{rdg}} - \theta^\star) \right] \\ &= \mathbb{E} \left[(X(X^\top X + \lambda \text{Id}_p)^{-1} X^\top \epsilon)^\top (X(X^\top X + \lambda \text{Id}_p)^{-1} X^\top \epsilon) \right. \\ &\quad \left. + \theta^{\star\top} \lambda^2 (X^\top X) (X^\top X + \lambda \text{Id}_p)^{-2} \theta^\star \right] \\ &= \sum_{i=1}^{\text{rg}(X)} \frac{s_i^4 \sigma^2}{(s_i^2 + \lambda)^2} + \theta^{\star\top} \lambda^2 (X^\top X) (X^\top X + \lambda \text{Id}_p)^{-2} \theta^\star \end{aligned}$$

Rem: $\lim_{\lambda \rightarrow 0} R_{\text{pred}}(\theta^\star, \hat{\theta}_\lambda^{\text{rdg}}) = \text{rg}(X) \sigma^2$, $\lim_{\lambda \rightarrow \infty} R_{\text{pred}}(\theta^\star, \hat{\theta}_\lambda^{\text{rdg}}) = \|X\theta^\star\|_2^2$

Biais / Variance : exemple de simulation



$$X \in \mathbb{R}^{50 \times 50}, \boldsymbol{\theta}^* = (2, 2, 2, 2, 2, 0, \dots, 0)^\top$$

Sommaire

Définitions de l'estimateur Ridge

Point de vue par SVD

Point de vue par pénalisation

Analyse du biais par la SVD

Analyse de la variance par la SVD

Choix du paramètre de régularisation

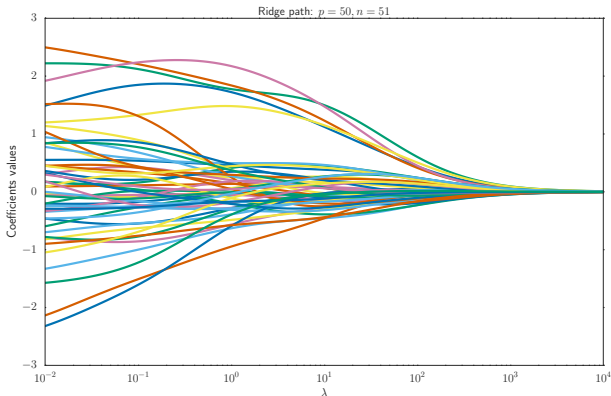
Notion de chemin de régularisation

Validation Croisée (CV)

Algorithmes et aspects computationnels

Choix de λ

```
n_features = 50; n_samples = 51
X = np.random.randn(n_samples, n_features)
theta_true = np.zeros([n_features, ])
theta_true[0:5] = 2.
y_true = np.dot(X, theta_true)
y = y_true + 1. * np.random.rand(n_samples,)
```



Sommaire

Définitions de l'estimateur Ridge

Point de vue par SVD

Point de vue par pénalisation

Analyse du biais par la SVD

Analyse de la variance par la SVD

Choix du paramètre de régularisation

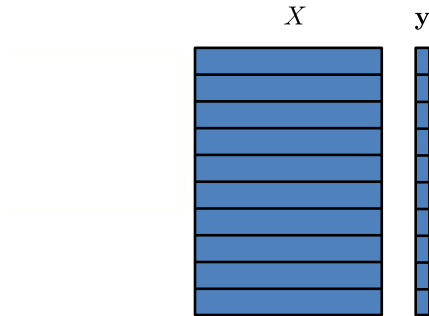
Notion de chemin de régularisation

Validation Croisée (CV)

Algorithmes et aspects computationnels

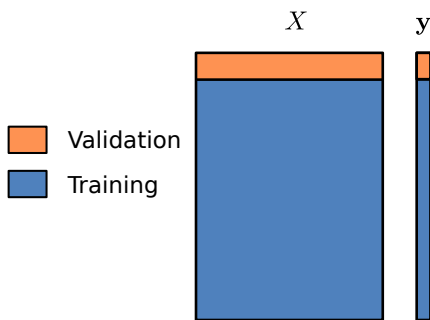
Validation croisée K -fold ($K = 10$)

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



Validation croisée K -fold ($K = 10$)

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

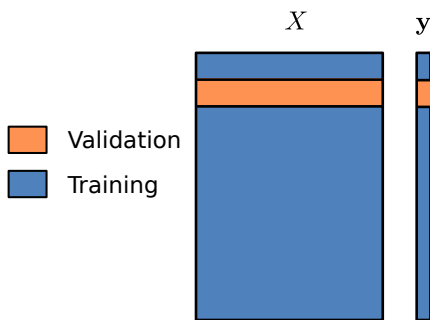


$k = 1$

1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

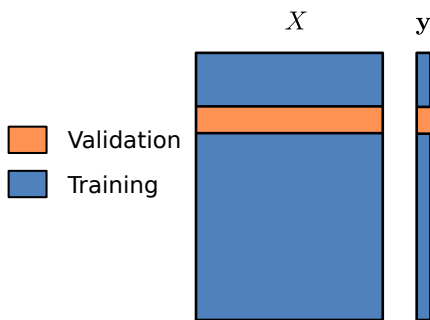
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

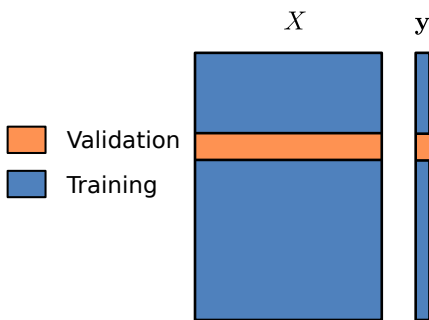
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



- $k = 3$
1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
 2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

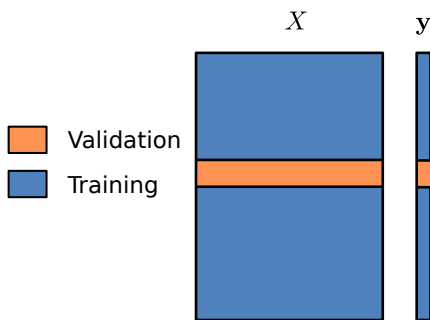
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

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

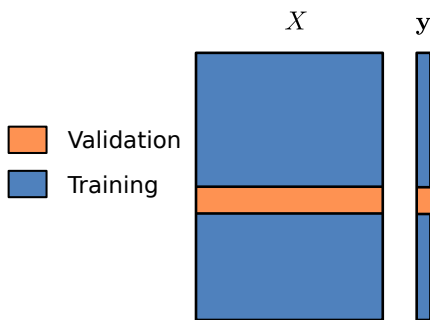


$$k = 5$$

1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

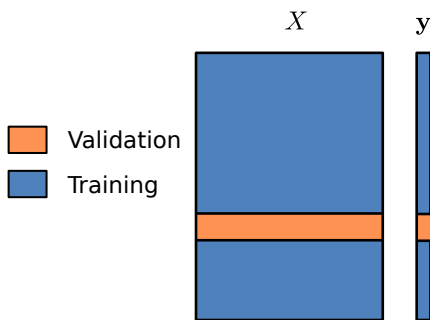
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

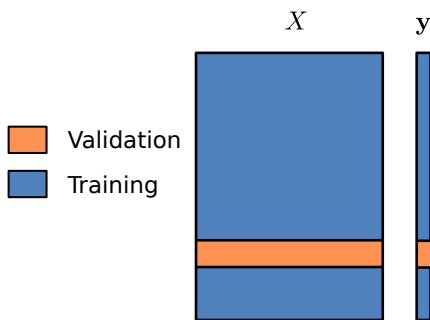
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



- $k = 7$
1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
 2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

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

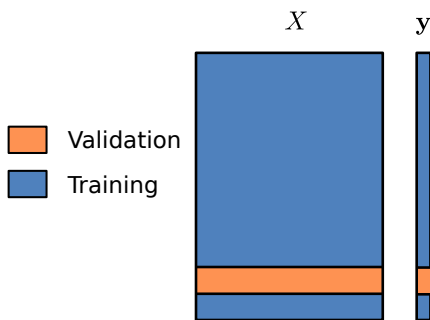


$$k = 8$$

1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

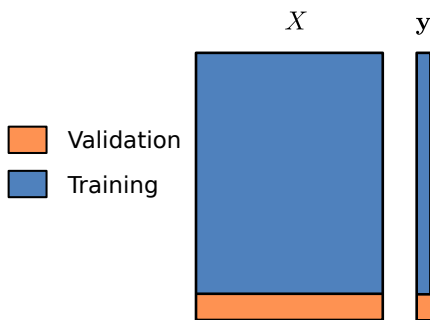
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

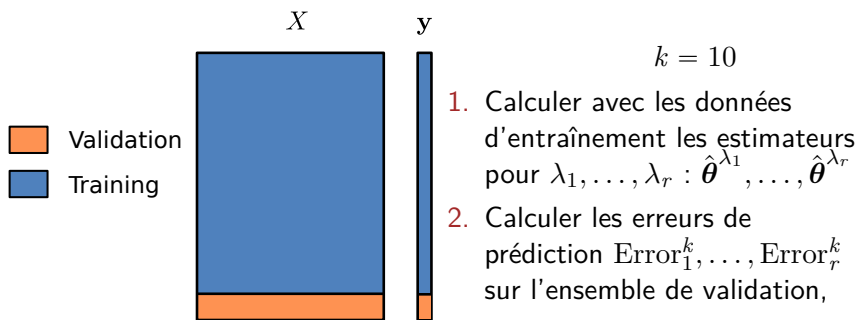
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



- $k = 10$
1. Calculer avec les données d'entraînement les estimateurs pour $\lambda_1, \dots, \lambda_r$: $\hat{\theta}^{\lambda_1}, \dots, \hat{\theta}^{\lambda_r}$
 2. Calculer les erreurs de prédiction $\text{Error}_1^k, \dots, \text{Error}_r^k$ sur l'ensemble de validation,

Validation croisée K -fold ($K = 10$)

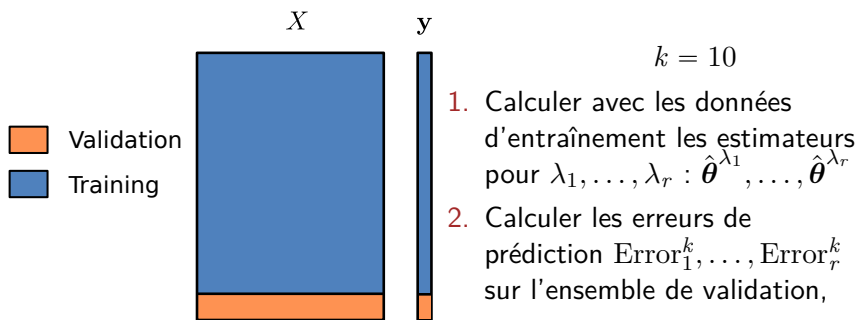
- ▶ Choisir une grille de taille r de λ à tester : $\lambda_1, \dots, \lambda_r$
- ▶ Diviser (X, y) selon les observations en K blocs :



Choix du paramètre : calculer $\widehat{\text{Error}}_1, \dots, \widehat{\text{Error}}_r$, moyennes des erreurs et choisir $\hat{i}^{\text{CV}} \in \llbracket 1, r \rrbracket$ atteignant la plus petite

Validation croisée K -fold ($K = 10$)


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



Choix du paramètre : calculer $\widehat{\text{Error}}_1, \dots, \widehat{\text{Error}}_r$, moyennes des erreurs et choisir $\hat{i}^{\text{CV}} \in \llbracket 1, r \rrbracket$ atteignant la plus petite

Re-calibration : calculer $\hat{\theta}^{\lambda_{\hat{i}^{\text{CV}}}}$ sur l'ensemble des observations

CV en pratique

Cas extrême de validation croisée ( : *cross-validation*)

- ▶ $K = 1$ impossible, au moins $K = 2$
- ▶ $K = n$, stratégie “*leave-one-out*” (cf. **Jackknife**) : autant de blocs que de variables

Rem: $K = n$ efficace computationnellement mais instable

Conseils pratiques :

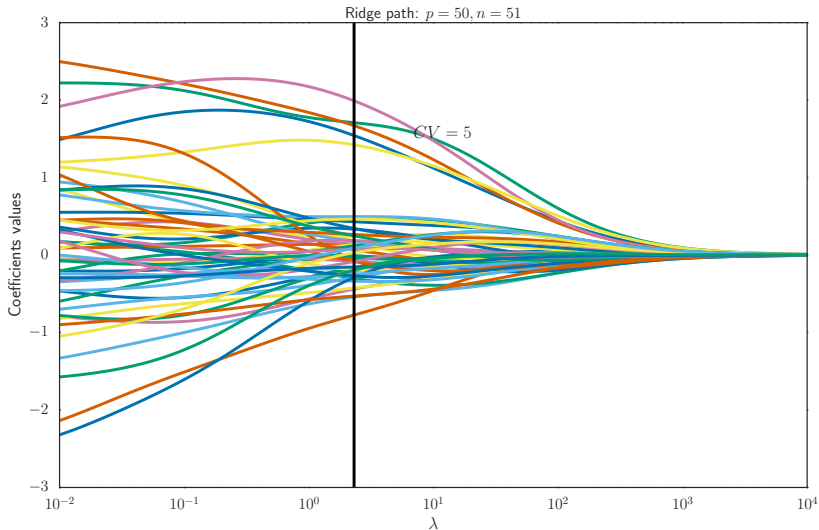
- ▶ “randomiser les observations” : observations dans un ordre aléatoire, évite des blocs de données trop similaires (chaque sous-bloc doit être représentatif de l'ensemble)
- ▶ choix habituels : $K = 5, 10$

Alternatives : partition aléatoire entre ensemble d'apprentissage et validation, version pour séries temporelles, etc.

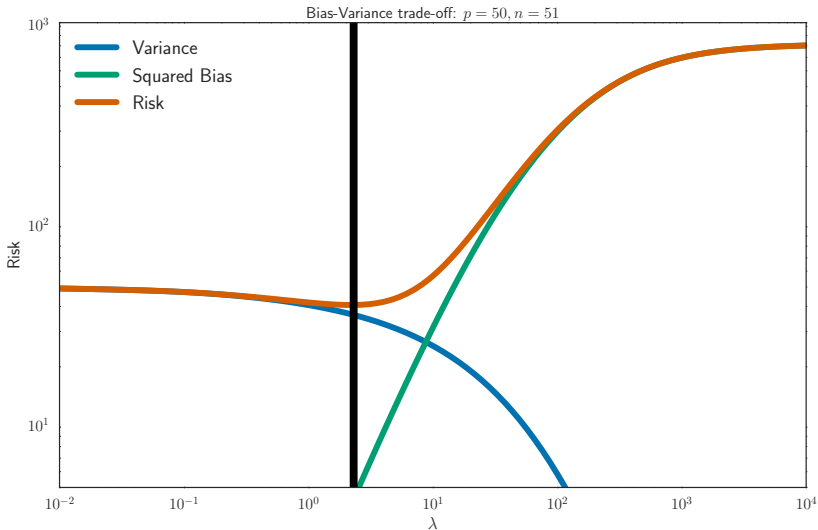
http://scikit-learn.org/stable/modules/cross_validation.html

Rem: en prédiction on peut aussi moyenner les meilleurs estimateurs obtenus plutôt que de re-calibrer sur toutes les données


Choix de λ : exemple avec $CV = 5$ (I)



Choix de λ : exemple avec $CV = 5$ (II)



Algorithmes pour la méthode *Ridge*

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

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

Rem: on calcule rarement l'estimateur *Ridge* pour un λ , en général on en calcule plusieurs (10, 100, ...) et on cherche le meilleur

Rem: enjeu crucial de calculer des SVD de grandes tailles

Références I

- ▶ G. H. Golub and C. F. van Loan.
Matrix computations.
Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.
- ▶ M. Y. Park.
Generalized linear models with regularization.
PhD thesis, Stanford University, 2006.