INSEA

27

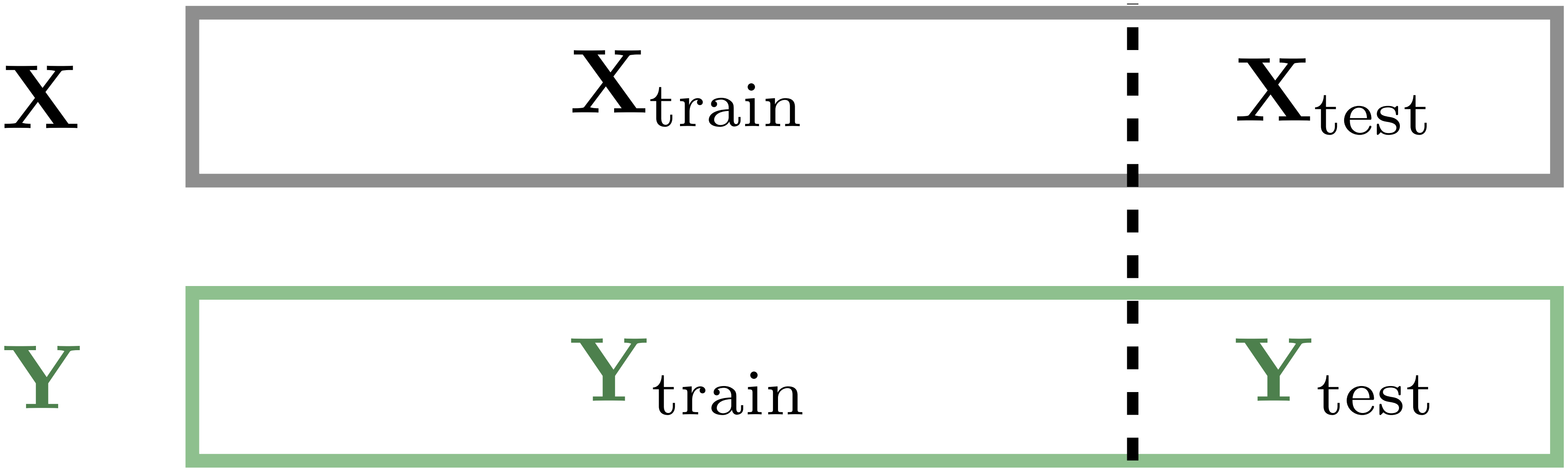
Regularization

Machine learning classique: zero-to-hero

Comment choisir C ?

On veut le C qui donne la meilleure performance sur le test

Pour cela on peut:


$\mathbf{X}$
$\mathbf{X}_{\text{train}}$
$\mathbf{X}_{\text{test}}$
$\mathbf{Y}$
$\mathbf{Y}_{\text{train}}$
$\mathbf{Y}_{\text{test}}$

2. Choisir une liste de valeurs de C, par ex: [0.01, 0.05, 0.1, 1., 10]

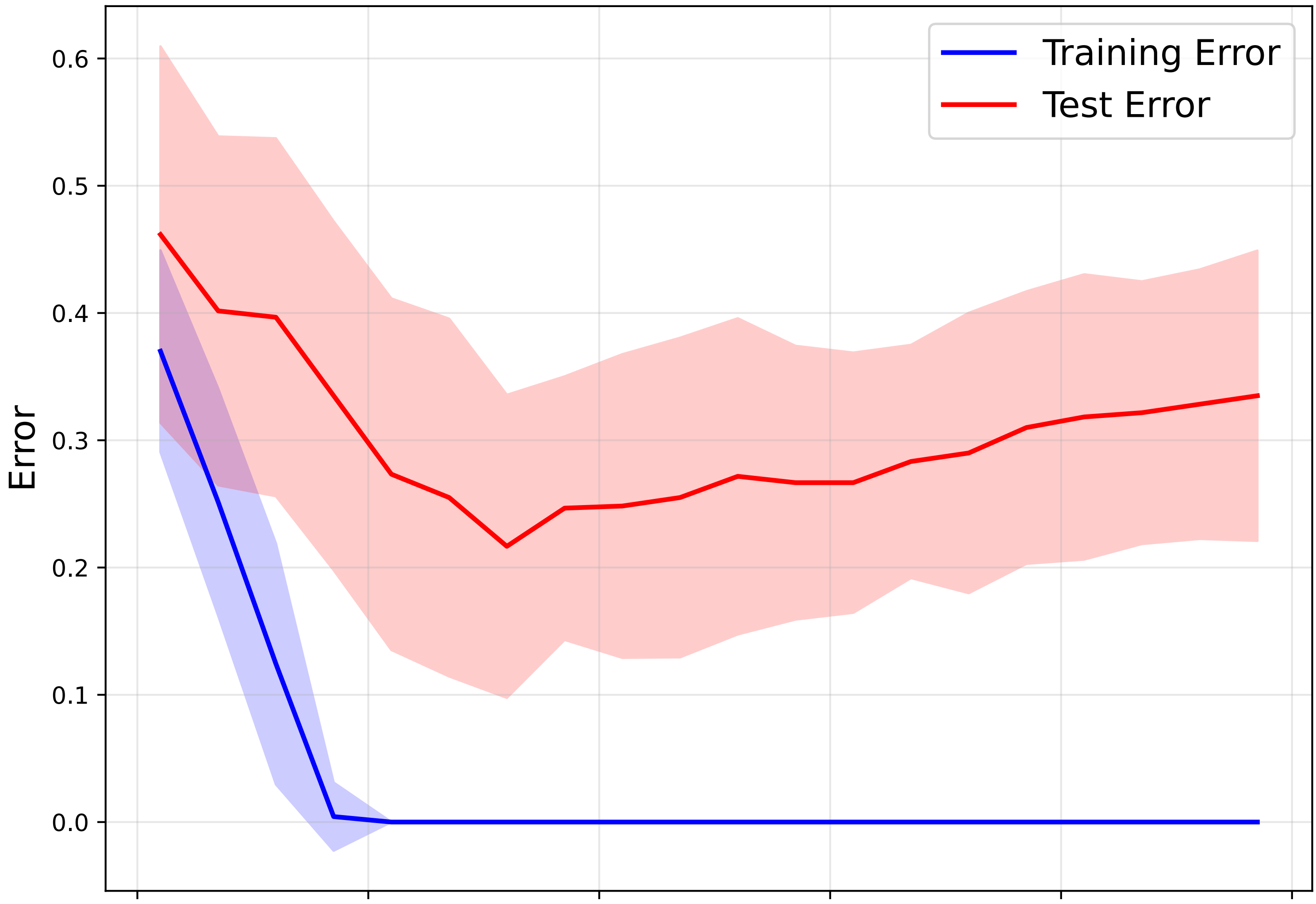
1. Couper le dataset en deux train et test:

Pour chaque C:

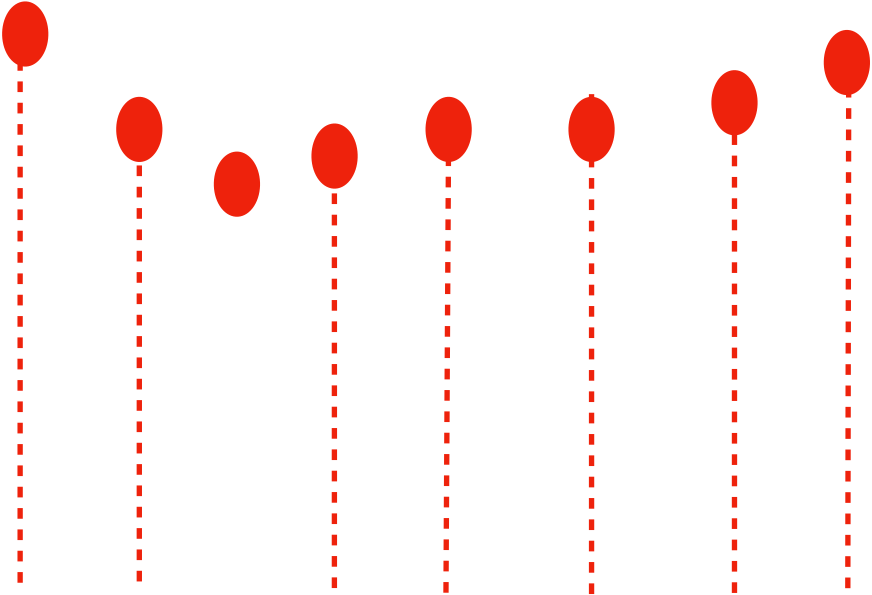
1. Optimiser sur $\mathbf{X}_{\text{train}}$ $\mathbf{Y}_{\text{train}}$ $$\min_{\theta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \text{loss}(f_\theta(\mathbf{x}_i), y_i) + \frac{1}{C} \text{pénalité}(\theta)$$

2. Évaluer l'erreur de prédiction sur $\mathbf{X}_{\text{test}}$ $\mathbf{Y}_{\text{test}}$

3. Choisir la valeur de C avec la plus petite erreur de prédiction sur le test


Training Error
Test Error
Error
0.6
0.5
0.4
0.3
0.2
0.1
0.0

C optimal

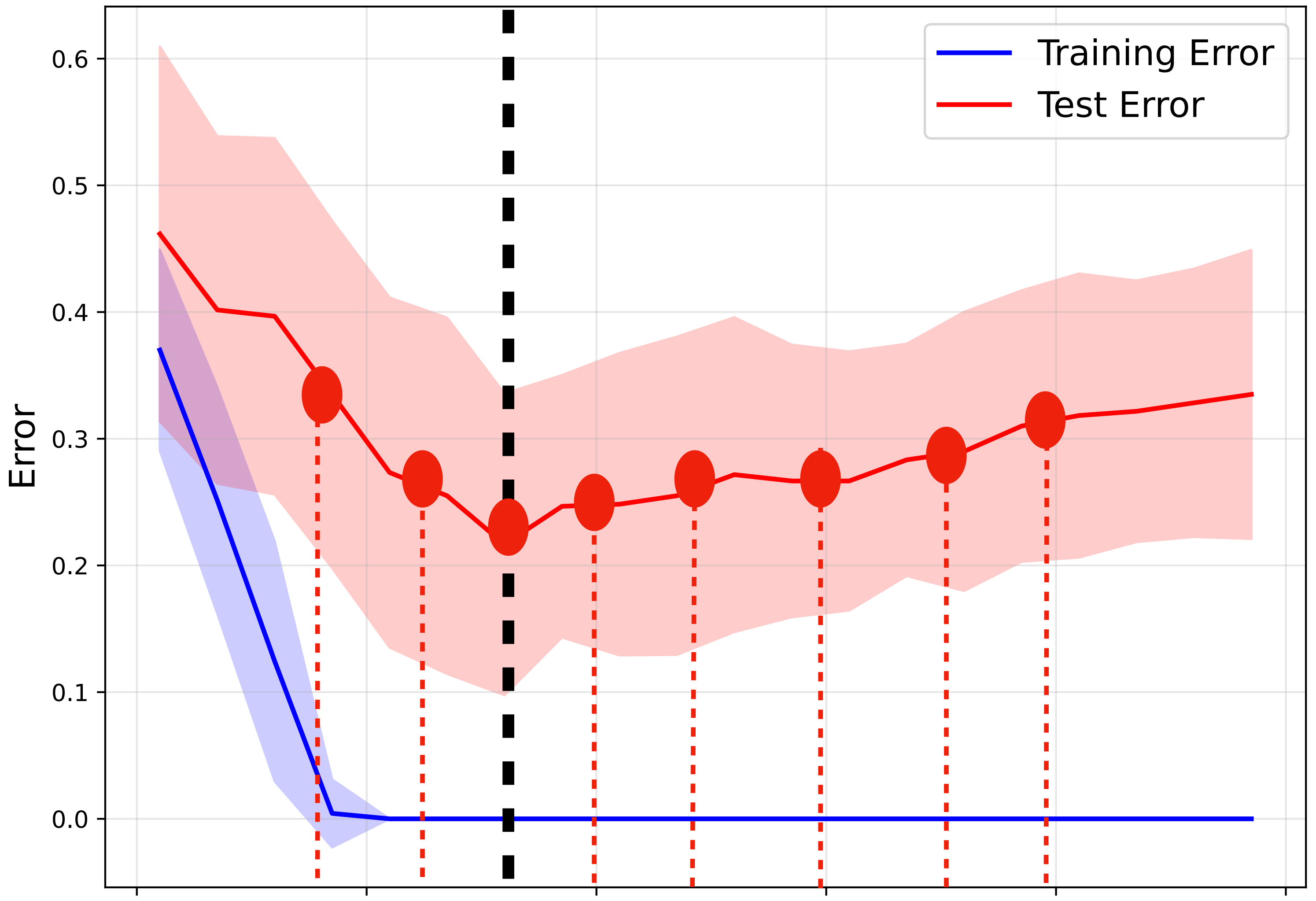


Quel est l’inconvénient principal de cette méthode ?

Le C choisi dépend du découpage aléatoire train / test

Comment choisir C ?

On veut le C qui donne la meilleure performance sur le test



C optimal

Pour cela on peut:

1. Couper le dataset en deux train et test:

$\mathbf{X}$	$\mathbf{X}_{\text{train}}$	$\mathbf{X}_{\text{test}}$
$\mathbf{Y}$	$\mathbf{Y}_{\text{train}}$	$\mathbf{Y}_{\text{test}}$

2. Choisir une liste de valeurs de C, par ex: [0.01, 0.05, 0.1, 1., 10]

Pour chaque C:

1. Optimiser sur $\mathbf{X}_{\text{train}}$ $\mathbf{Y}_{\text{train}}$ $\min_{\theta \in \mathbb{R}^p} \frac{1}{n} \sum_{i=1}^{n} \text{loss}(f_\theta(\mathbf{x}_i), y_i) + \frac{1}{C} \text{pénalité}(\theta)$

2. Évaluer l'erreur de prédiction sur $\mathbf{X}_{\text{test}}$ $\mathbf{Y}_{\text{test}}$

3. Choisir la valeur de C avec la plus petite erreur de prédiction sur le test

Quel est l'inconvénient principal de cette méthode ? Le C choisi dépend du découpage aléatoire train / test

Idée: Effectuer plusieurs découpages et moyenner l'erreur de test