Projet 1 SY19

Contents

1 Régression **2**
- 1.1 Modèle avec méthode BIC . . . 2
- 1.2 Modèle linéaire avec cross-validation . . . 2
- 1.3 Modèle linéaire avec nested cross-validation . . . 2
- 1.4 Modèle avec méthode ridge . . . 3
- 1.5 Modèle avec méthode lasso . . . 3
- 1.6 Comparaison des modèles . . . 3

2 Classification **5**

Introduction

Dans le cadre de l'UV SY19, nous avons réalisé un projet consistant en la sélection des modèles optimaux pour un problème de régression et un problème de classification. Ce rapport est constitué d'une première partie résumant nos différentes approches pour la régression, puis une deuxième partie concernant la classification.

1 Régression

Avant toute chose, il nous faut séparer notre jeu de données en 2 groupes : un groupe d'entraînement et un groupe de test. Pour cela, on utilise la fonction sample en tant que masque sur notre ensemble de données. Ici, nous avont choisi un paramètre 4/5, donc 400 variables d'entraînement pour 100 variables de test.

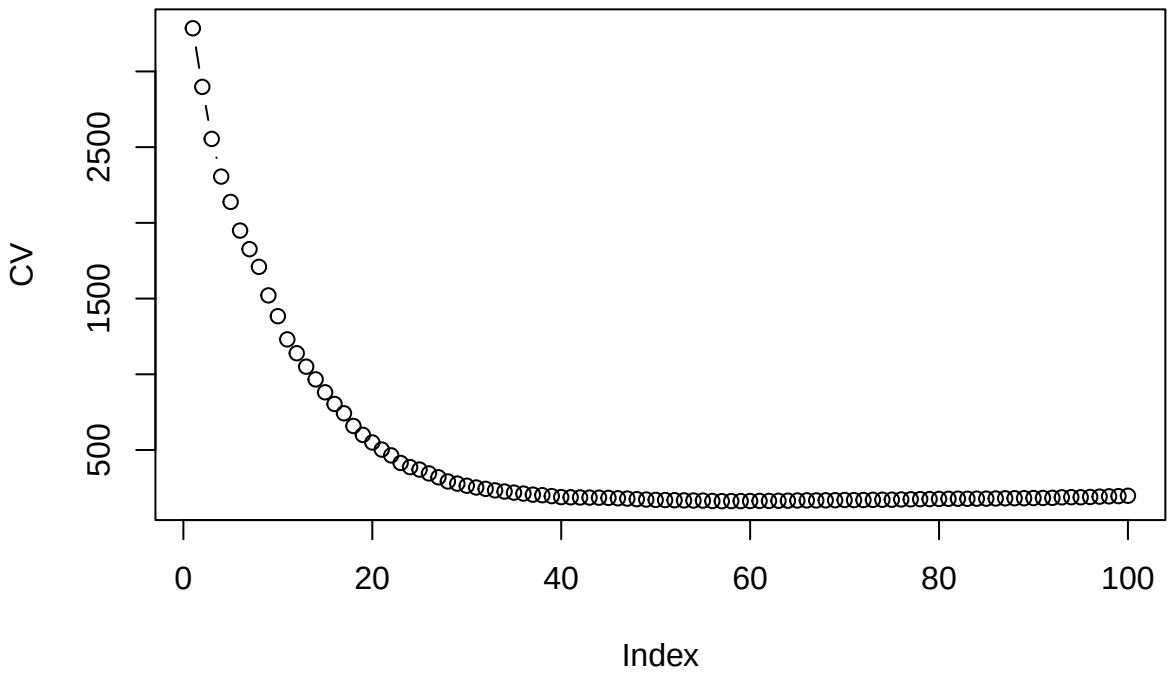
1.1 Modèle avec méthode BIC

Avec la méthode BIC, on trouve un modèle plutôt cohérent avec peu de prédicteurs (seulement 45).

1.2 Modèle linéaire avec cross-validation

La validation croisée permet de tester différents sous-ensembles de données comme modèles d'apprentissage et de test. Ici, on fait de coupes de 1/10e et on se rend compte que 50 prédicteurs semblent être un bon compromis pour notre modèle.

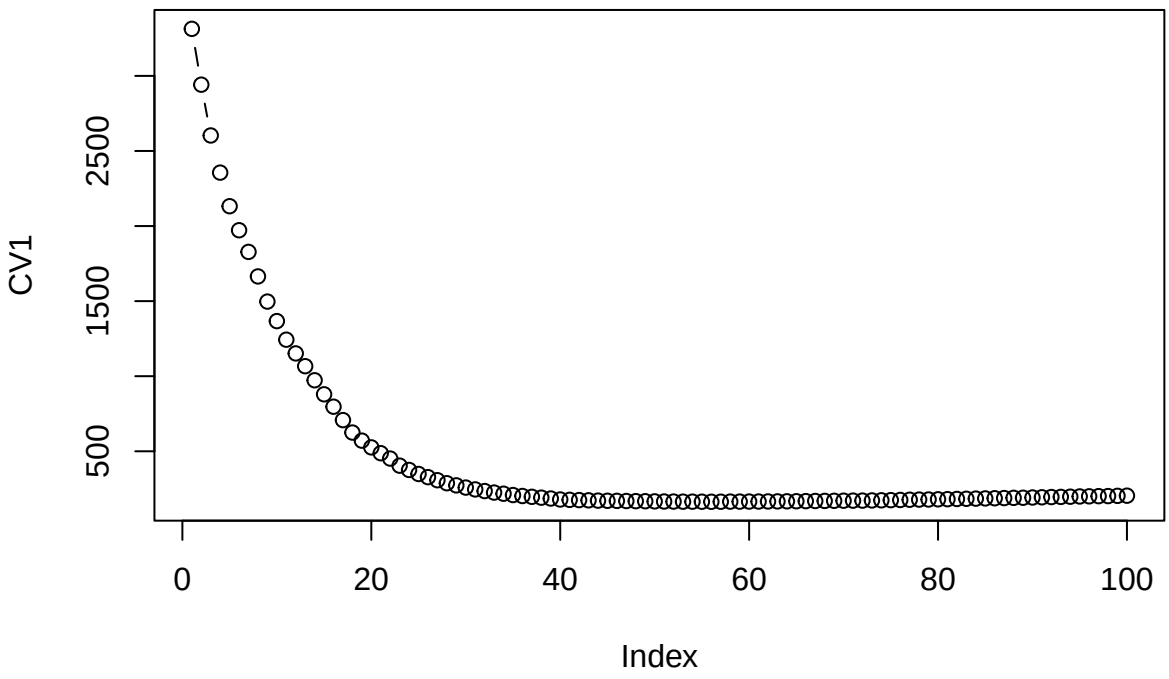
```
plot(CV, type="b")
```



1.3 Modèle linéaire avec nested cross-validation

La méthode de nested CV permet d'utiliser non pas 2 mais 3 ensembles de données. On va avoir l'ensemble d'entraînement, l'ensemble de validation, qui va permettre de sélectionner le meiller modèle et enfin l'ensemble de test, pour connaître la qualité de notre modèle. Cet ensemble de test est donc indépendant du reste ce qui permet d'éviter des biais et des fuites de données. On observe qu'avec cette méthode, le nombre optimal de prédicteurs semble être 57.

```
plot(CV1, type="b")
```



1.4 Modèle avec méthode ridge

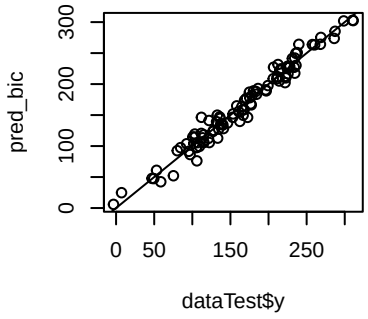
1.5 Modèle avec méthode lasso

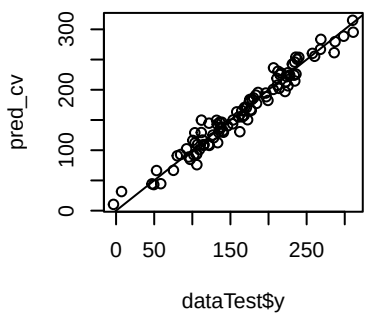
1.6 Comparaison des modèles

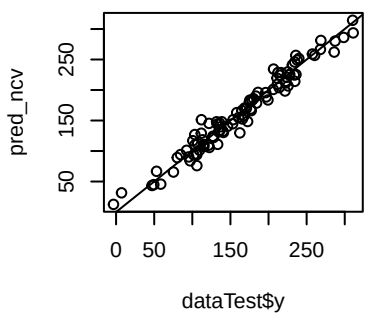
En comparant les différents modèles obtenus, on constate que celui qui admet l'erreur quadratique moyenne la plus faible est le modèle obtenu avec nested cross-validation.

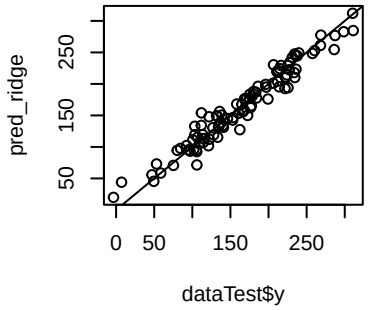
```
test_methodes(data.reg.test)
```

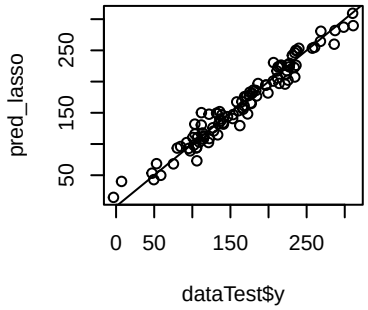
```
## La méthode BIC donne un modèle avec une erreur quadratique moyenne de : 130.2439
## Le modèle linéaire avec cross-validation a une erreur quadratique moyenne de : 167.8741
## Le modèle linéaire avec nested cross-validation a une erreur quadratique moyenne de : 165.9104
## La méthode ridge a une erreur quadratique moyenne de : 220.4128
## La méthode lasso a une erreur quadratique moyenne de : 185.2832
```


pred_bic
0
100
200
300
0
50
150
250
dataTest$y


pred_cv
0
100
200
300
0
50
150
250
dataTest$y


pred_ncv
50
150
250
0
50
150
250
dataTest$y


pred_ridge
50
150
250
0
50
150
250
dataTest$y


pred_lasso
50
150
250
0
50
150
250
dataTest$y

2 Classification