Université de Montréal

Le Lasso Linéaire : une méthode pour des données de petites et grandes dimensions en régression linéaire

par

Yan Watts

Département de mathématiques et de statistique
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en statistique

14 avril 2023

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

Le Lasso Linéaire : une méthode pour des données de petites et grandes dimensions en régression linéaire

présenté par

Yan Watts

a été évalué par un jury composé des personnes suivantes :

Florian Maire
(président-rapporteur)

Mylène Bédard
(directrice de recherche)

Janie Coulombe
(membre du jury)

Résumé

Dans ce mémoire, nous nous intéressons à une façon géométrique de voir la méthode du Lasso en régression linéaire. Le Lasso est une méthode qui, de façon simultanée, estime les coefficients associés aux prédicteurs et sélectionne les prédicteurs importants pour expliquer la variable réponse. Les coefficients sont calculés à l'aide d'algorithmes computationnels. Malgré ses vertus, la méthode du Lasso est forcée de sélectionner au maximum n variables lorsque nous nous situons en grande dimension ($p > n$). De plus, dans un groupe de variables corrélées, le Lasso sélectionne une variable "au hasard", sans se soucier du choix de la variable.

Pour adresser ces deux problèmes, nous allons nous tourner vers le Lasso Linéaire. Le vecteur réponse est alors vu comme le point focal de l'espace et tous les autres vecteurs de variables explicatives gravitent autour du vecteur réponse. Les angles formés entre le vecteur réponse et les variables explicatives sont supposés fixes et nous serviront de base pour construire la méthode. L'information contenue dans les variables explicatives est projetée sur le vecteur réponse. La théorie sur les modèles linéaires normaux nous permet d'utiliser les moindres carrés ordinaires (MCO) pour les coefficients du Lasso Linéaire.

Le Lasso Linéaire (LL) s'effectue en deux étapes. Dans un premier temps, des variables sont écartées du modèle basé sur leur corrélation avec la variable réponse; le nombre de variables écartées (ou ordonnées) lors de cette étape dépend d'un paramètre d'ajustement γ. Par la suite, un critère d'exclusion basé sur la variance de la distribution de la variable réponse est introduit pour retirer (ou ordonner) les variables restantes. Une validation croisée répétée nous guide dans le choix du modèle final.

Des simulations sont présentées pour étudier l'algorithme en fonction de différentes valeurs du paramètre d'ajustement γ. Des comparaisons sont effectuées entre le Lasso Linéaire et des méthodes compétitrices en petites dimensions (Ridge, Lasso, SCAD, etc.). Des améliorations dans l'implémentation de la méthode sont suggérées, par exemple l'utilisation de la règle du 1se nous permettant d'obtenir des modèles plus parcimonieux. Une implémentation de l'algorithme LL est fournie dans la fonction **R** intitulée `linlasso`, disponible au https://github.com/yanwatts/linlasso.

Mots clés : régression linéaire, Lasso, moindres carrés ordinaires, sélection de variables, inférence, grande dimension

Abstract

In this thesis, we are interested in a geometric way of looking at the Lasso method in the context of linear regression. The Lasso is a method that simultaneously estimates the coefficients associated with the predictors and selects the important predictors to explain the response variable. The coefficients are calculated using computational algorithms. Despite its virtues, the Lasso method is forced to select at most n variables when we are in high-dimensional contexts ($p > n$). Moreover, in a group of correlated variables, the Lasso selects a variable "at random", without caring about the choice of the variable.

To address these two problems, we turn to the Linear Lasso. The response vector is then seen as the focal point of the space and all other explanatory variables vectors orbit around the response vector. The angles formed between the response vector and the explanatory variables are assumed to be fixed, and will be used as a basis for constructing the method. The information contained in the explanatory variables is projected onto the response vector. The theory of normal linear models allows us to use ordinary least squares (OLS) for the coefficients of the Linear Lasso.

The Linear Lasso (LL) is performed in two steps. First, variables are dropped from the model based on their correlation with the response variable; the number of variables dropped (or ordered) in this step depends on a tuning parameter γ. Then, an exclusion criterion based on the variance of the distribution of the response variable is introduced to remove (or order) the remaining variables. A repeated cross-validation guides us in the choice of the final model.

Simulations are presented to study the algorithm for different values of the tuning parameter γ. Comparisons are made between the Linear Lasso and competing methods in small dimensions (Ridge, Lasso, SCAD, etc.). Improvements in the implementation of the method are suggested, for example the use of the 1se rule allowing us to obtain more parsimonious models. An implementation of the LL algorithm is provided in the function **R** entitled `linlasso` available at https://github.com/yanwatts/linlasso.

Keywords: linear regression, Lasso, ordinary least squares, variable selection, inference, high dimensionality

Table des matières

Liste des tableaux

Liste des figures

Liste des sigles et des abréviations

BLUE	de l'anglais *Best Linear Unbiased Estimator*
EQM	Erreur quadratique moyenne
MCO	Moindres carrés ordinaires
MCP	de l'anglais *Minimax Concave Penalty*
MSE	de l'anglais *Mean Squarred Error*
SCAD	de l'anglais *Smoothly Clipped Absolute Deviations*
SIS	de l'anglais *Sure Independence Screening*

Remerciements

Je tiens à exprimer ma profonde gratitude envers ma superviseure, Mylène Bédard, pour son soutien tout au long de ma recherche. Son encadrement, ses précieux conseils ainsi que ses encouragements m'ont permis d'avancer de façon fluide dans mon mémoire. Elle a toujours su me guider dans les choix méthodologiques les plus pertinents et répondre rapidement à mes nombreuses questions ou sollicitations. Je suis très reconnaissant de ton support dans mon parcours en maîtrise et je te remercie.

Je remercie les professeurs que j'ai côtoyés durant mon parcours universitaire qui m'ont fourni les outils nécessaires à la réussite de mes études universitaires. Je suis également reconnaissant d'avoir eu de merveilleux collègues durant mon parcours, Yuxi, Émilyne et Étienne, qui m'ont aidé tout au long de mon cheminement.

Je suis également reconnaissant envers des amitiés qui se sont formées durant mon parcours au baccalauréat : Stefan, Joseph, Antoine et Béni. Merci d'avoir été une source constante d'encouragement et de soutien tout au long de mon parcours universitaire.

Merci à mes tous mes amis et amies : Félix, Corey, Éloi et tous les autres. Vous avez toujours su m'encourager et me distraire avec les nombreux matchs de basketball. Merci à Katia qui m'a toujours encouragé durant mon parcours universitaire.

Je remercie mes très chers parents, Elena et Nikita, qui ont toujours été là pour moi. Votre confiance en moi et vos encouragements ont été une source constante de motivation et d'inspiration, et je ne pourrais jamais assez vous remercier pour cela. Je remercie également ma belle famille qui m'a également supporté tout au long de ce mémoire.

Merci à mes collègues chez Numea qui m'ont supporté durant la fin de mon parcours.

Finalement, la meilleure pour la fin, merci à Stella pour tes encouragements, ton aide précieuse et pour avoir été là quand j'en avais besoin. Je suis très chanceux et reconnaissant de t'avoir dans ma vie.

Introduction

La régression linéaire est une technique statistique utilisée dans plusieurs domaines tels que les sciences de la santé, les sciences humaines, la finance, l'ingénierie et l'apprentissage automatique. Elle nous permet d'étudier la relation entre la variable dépendante continue, ou expliquée, et une ou plusieurs variables indépendantes, ou explicatives. La régression linéaire repose sur l'idée qu'il existe un lien linéaire entre les variables dépendantes et indépendantes. Il suffit d'associer un certain poids, ou coefficient, à chacune des variables explicatives afin d'expliquer la variable dépendante. Sous certaines conditions, le vecteur des coefficients, ou estimateur des moindres carrés ordinaires (MCO), constitue la base de la régression linéaire. L'intérêt de l'estimateur MCO est que parmi tous les estimateurs linéaires et sans-biais, il est celui ayant la plus petite variance et il est également simple à calculer. Pour ces raisons, l'estimateur MCO est populaire en régression linéaire. Ainsi, lorsque nous avons de nouvelles données, nous pouvons prédire la variable dépendante à l'aide de l'estimateur MCO. La prédiction est une partie importante en régression linéaire, puisqu'elle nous permet d'utiliser l'information des variables explicatives pour prendre des décisions éclairées concernant la variable dépendante.

Une des conditions nous permettant de calculer l'estimateur MCO est que le rang de la matrice de design, c'est-à-dire la matrice comportant toutes les variables explicatives, soit égal à la dimension de la matrice de design p. Si le rang devient inférieur à p, l'estimateur MCO n'est plus calculable. Ce phénomène se produit en présence de multicolinéarité entre les variables explicatives ou lorsque nous nous trouvons en grande dimension; dans ce deuxième cas, il y a alors plus de variables explicatives p que d'observations n dans la matrice de design. Afin de régler ces deux problèmes, il suffit de trouver une façon de réduire la dimension p pour obtenir un modèle contenant seulement les variables explicatives les plus importantes. C'est pourquoi nous utilisons les méthodes de sélection de variables, qui nous permettent d'améliorer l'interprétabilité du modèle et, par extension, d'améliorer la prédiction de la variable dépendante.

Les méthodes traditionnelles de sélection de variables (régression ascendante ou descendante, AIC, BIC, C_p de Mallows) sont utiles en présence de multicolinéarité, afin d'éliminer les variables explicatives peu pertinentes dans la modélisation de la variable dépendante.

Toutefois, ces méthodes sont extrêmement coûteuses quand la dimensionnalité p est très grande, ce qui rend ces méthodes infaisables computationnellement. De plus, un petit changement dans les données pourrait mener à un modèle final complètement différent, ce qui rend ces méthodes instables. Par conséquent, plusieurs nouvelles méthodes inspirées de la régression pénalisée ont pris de l'ampleur dans les dernières décennies; une approche particulièrement populaire est le Lasso de Tibshirani (1996). Cette méthode nous permet à la fois de rétrécir la taille des coefficients et d'effectuer une sélection de variables. C'est une méthode très utilisée, mais qui souffre également de deux problèmes : la méthode du Lasso est forcée de sélectionner au maximum n variables lorsque nous nous situons en grande dimension ($p > n$). De plus, dans un groupe de variables explicatives corrélées, le Lasso sélectionne une variable "au hasard", sans réellement se soucier du choix de la variable.

Pour pallier aux problèmes du Lasso, nous nous tournons vers une méthode développée par Fraser et Bédard (2022), qui s'intitule le Lasso Linéaire. Cette méthode est une façon géométrique de voir le Lasso; le vecteur contenant les réponses est vu comme le point focal de l'espace, alors que tous les autres vecteurs de variables explicatives gravitent autour du vecteur contenant les réponses. Sous l'hypothèse que les directions des différents vecteurs sont fixes, nous utilisons les angles formés par le vecteur réponse et les vecteurs de variables explicatives pour nous guider vers la théorie du modèle de position utilisé. En projetant l'information contenue dans les variables explicatives sur le vecteur réponse $\mathbf{y}$, nous pouvons alors proposer un modèle linéaire pour la variable réponse y et fournir une certaine mesure de la variabilité qui est expliquée par ce modèle. Cette technique nous permet de passer d'un problème de sélection de variables se situant dans un espace multidimensionnel à un problème en une seule dimension; nous qualifions donc la méthode de *dimension free*. La sélection de variables s'effectue par la suite en deux étapes. À l'aide du paramètre d'ajustement $\gamma \in [0,1]$, nous pouvons préalablement ordonner certaines variables explicatives en nous basant sur leur corrélation avec le vecteur réponse et, par la suite, ordonner les variables restantes en nous référant à la variance de la distribution de y. La validation croisée répétée nous guide dans le choix du modèle final. Contrairement au Lasso, la méthode du Lasso Linéaire ne nécessite pas de computation puisque l'estimateur final est celui des moindres carrés ordinaires.

Dans le chapitre 1, nous introduisons la théorie derrière la régression linéaire. Nous présentons également les grandes lignes des méthodes traditionnelles de sélection de variables et, par la suite, nous introduisons les méthodes du Ridge et du Lasso. Dans le chapitre 2, nous décrivons en détail la théorie derrière le Lasso linéaire et nous présentons l'algorithme. Dans le chapitre 3, des ajouts sont faits à la fonction de Fraser et Bédard (2022) pour l'optimiser; nous introduisons la règle du 1se et nous validons les différents paramètres pour la méthode, dont le paramètre d'ajustement γ. Le paramètre $\gamma = 0{,}1$ semble un candidat à privilégier pour le Lasso Linéaire. Dans le chapitre 4, le Lasso Linéaire est comparé à

plusieurs méthodes inspirées de la régression pénalisée, soit le Ridge de Hoerl et Kennard (1970), le Lasso de Tibshirani (1996), le SCAD de Fan et Li (2001a), le Filet élastique de Zou et Hastie (2005) et le Lasso adaptatif de Zou (2006). Nous observons une bonne capacité prédictive du Lasso Linéaire dans divers exemples basés sur des données artificielles, expressément choisis pour étudier des contextes d'une difficulté croissante. Nous présentons également des exemples basés sur des données réelles, en petites et grandes dimensions, pour illustrer l'implémentation du Lasso Linéaire dans ces deux contextes.

Chapitre 1

Théorie de la régression linéaire

La régression linéaire est une méthode de modélisation permettant d'établir un lien linéaire entre une variable expliquée, ou dépendante, et un ensemble de variables explicatives, ou indépendantes. La variable réponse est toujours continue, mais les variables explicatives peuvent être continues, discrètes ou catégorielles. La régression linéaire nous permet de prédire la variable dépendante en associant un certain poids, ou coefficient, à chacune des variables indépendantes. On peut penser à un modèle qui prédit la concentration d'ozone (en µg/ml) en fonction de la température ou encore le taux de criminalité en fonction de la scolarité. Afin d'effectuer de telles prédictions, il faut d'abord dériver le modèle linéaire et comprendre comment estimer les coefficients associés aux différentes variables explicatives.

Soit la variable aléatoire y, qui représente la variable d'intérêt, et les p variables aléatoires explicatives $x_1, \ldots, x_p$. Nous aimerions estimer les coefficients associés à chacune des variables explicatives afin de pouvoir prédire la variable d'intérêt. Pour ce faire, supposons le modèle de régression linéaire suivant,

$$\mathbf{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

où $\mathbf{y}$ est le vecteur réponse n-dimensionnel et $\boldsymbol{X} = (1_n, \mathbf{x}_1, \ldots, \mathbf{x}_p)$ est la matrice de design $n \times (p+1)$. Le vecteur 1_n est un vecteur n-dimensionnel formé exclusivement de 1, alors que $\mathbf{x}_j$ $(j = 1, \ldots, p)$ est un vecteur n-dimensionnel contenant les observations associées à la j-ième variable explicative (ou prédicteur). Le vecteur $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^\top$ contient les $p+1$ coefficients de régression associés aux $p+1$ colonnes de la matrice de design et $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^\top$ est un vecteur d'erreurs de longueur n. Pour faciliter la lecture, les lettres en caractère gras désignent les vecteurs d'observations, soit $\mathbf{y}$ et $\mathbf{x}_j$ $(j = 1, \ldots, p)$. Cette notation nous permettra éventuellement de faire la distinction avec les variables aléatoires y et x_j $(j = 1, \ldots, p)$, qui nous permettront d'établir le cadre théorique de la régression linéaire. Dans ce chapitre cependant, nous ferons le même abus de notation que la majorité des ouvrages de régression: nous utiliserons la notation $\mathbf{y}$ pour désigner à la fois le vecteur

d'observations et le vecteur de variables aléatoires. Pour le moment, nous supposons également que les variables explicatives observées sont exactement mesurées (pas d'erreur associée à ces observations et donc la notation $x_1, \ldots, x_p$ ne sera pas nécessaire). Cependant, dans les chapitres ultérieurs, cette notation prendra tout son sens.

À partir d'ici et dans tout le reste du mémoire, nous supposerons que le vecteur réponse et les vecteurs de variables explicatives ont été standardisés. Ceci signifie que les variables initiales sont d'abord centrées, de sorte à obtenir

$$n^{-1}\sum_{i=1}^{n} y_i = 0 \quad \text{et} \quad n^{-1}\sum_{i=1}^{n} x_{ij} = 0,\ j = 1, \ldots, p; \tag{1.0.1}$$

les variables centrées sont ensuite rééchelonnées, de sorte à ce que l'écart-type de chaque vecteur soit égal à 1 :

$$n^{-1}\sum_{i=1}^{n} y_i^2 = 1 \quad \text{et} \quad n^{-1}\sum_{i=1}^{n} x_{ij}^2 = 1,\ j = 1, \ldots, p. \tag{1.0.2}$$

En d'autres mots, le vecteur réponse et les $p + 1$ colonnes de la matrice de design ont chacun une moyenne de 0 et une variance de 1. Cette manipulation nous permet de retirer l'ordonnée à l'origine du modèle. De plus, elle fait en sorte que les prédicteurs ne dépendent plus de l'unité par rapport à laquelle ils ont été mesurés (livres ou kilogrammes, par exemple); les coefficients sont alors plus faciles à comparer, ce qui facilitera l'application de certaines méthodes de sélection de variables. Pour alléger la notation, nous supposons maintenant que le jeu de données a été standardisé et utilisons la notation habituelle $(\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p)$.

1.1. Régression des moindres carrés ordinaires

Nous souhaitons maintenant étudier la relation linéaire entre la variable réponse et les p variables explicatives par le biais des coefficients de régression $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)^\top$. Le but de la régression est de pouvoir expliquer le mieux possible la variable réponse en estimant les coefficients $\boldsymbol{\beta}$ par $\hat{\boldsymbol{\beta}}$. Une approche populaire pour estimer ces coefficients est celle des moindres carrés ordinaires (MCO). Celle-ci nous permet d'obtenir des estimateurs intuitifs et aussi d'avoir accès aux lois de ces estimateurs. Les hypothèses nécessaires liées à cette approche seront introduites sous peu dans le but, par la suite, de faire de la prédiction.

Définition 1.1.1 (Fonction objective MCO)**.** *L'estimateur des moindres carrés ordinaires $\hat{\boldsymbol{\beta}}$ est défini comme étant l'estimateur qui minimise la fonction objective*

$$\boldsymbol{S}(\boldsymbol{\beta}) = (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}).$$

L'estimateur peut également s'écrire sous la forme suivante :

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}).$$

Proposition 1.1.2 (Estimateur des MCO). *Soit $n > p$, avec une matrice de design $\boldsymbol{X}$ de rang plein p; alors l'unique estimateur des moindres carrés de $\boldsymbol{\beta}$ s'écrit*

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \mathbf{y}. \tag{1.1.1}$$

Démonstration. Il suffit de développer la fonction objective et ensuite trouver le minimum de la fonction,

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{\beta}) &= (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) \\ &= \mathbf{y}^\top \mathbf{y} - \boldsymbol{\beta}^\top \boldsymbol{X}^\top \mathbf{y} - \mathbf{y}^\top \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta} \\ &= \mathbf{y}^\top \mathbf{y} - 2\boldsymbol{\beta}^\top \boldsymbol{X}^\top \mathbf{y} + \boldsymbol{\beta}^\top \boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta}. \end{aligned}$$

La dérivée s'écrit comme suit :

$$\frac{\partial \boldsymbol{S}(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}} = -2\boldsymbol{X}^\top \mathbf{y} + 2\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta}.$$

Ainsi, le minimum, s'il existe, est

$$-2\boldsymbol{X}^\top \mathbf{y} + 2\boldsymbol{X}^\top \boldsymbol{X}\hat{\boldsymbol{\beta}} = 0 \implies \hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \mathbf{y}.$$

□

En minimisant la fonction objective $S(\boldsymbol{\beta})$, il est possible d'obtenir un unique minimum pour $\hat{\boldsymbol{\beta}}$. Ceci est sujet à la satisfaction de l'hypothèse clé stipulant que le $rang(\boldsymbol{X}) = p$. Si le $rang(\boldsymbol{X}) < p$, il n'est alors pas possible d'inverser la matrice $\boldsymbol{X}^\top \boldsymbol{X}$; ceci survient, entre autres, lorsqu'il y a de la multicolinéarité entre les colonnes de la matrice de design $\boldsymbol{X}$ ou lorsque le nombre de variables explicatives p est plus grand que le nombre d'observations n. Ces problèmes seront abordés dans la section 1.2.2.

Nous étudions maintenant l'espérance et la variance de l'estimateur $\hat{\boldsymbol{\beta}}$. Pour ce faire, il sera pratique de poser certaines conditions connues sous le nom des conditions de Gauss-Markov.

Définition 1.1.3 (Conditions de Gauss-Markov). *Supposons que les erreurs $(\varepsilon_1, \ldots, \varepsilon_n)$ sont centrées, de même variance (homoscédasticité) et non corrélées entre elles. Alors,*

(1) $\mathbb{E}(\varepsilon_i) = 0$ pour $i \in \{1, \ldots, n\}$,

(2) $Cov(\varepsilon_i, \varepsilon_j) = \partial_{i,j}\sigma^2$ pour $(i, j) \in \{1, \ldots, n\}^2$,

où $\sigma^2 > 0$, $\partial_{i,j} = 1$ lorsque $i = j$ et $\partial_{i,j} = 0$ sinon.

Les hypothèses peuvent être réécrites sous forme matricielle pour $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^\top$:

(1) $\mathbb{E}(\boldsymbol{\varepsilon}) = 0_n$,

(2) $\mathbb{E}(\boldsymbol{\varepsilon}\boldsymbol{\varepsilon}^\top) = \sigma^2 I_n$,

où $0_n = (0, \ldots, 0)^\top$ est un vecteur n-dimensionnel dont toutes les composantes sont des zéros et I_n est la matrice identité $n \times n$.

Basé sur les conditions de Gauss-Markov, il est évident que $\mathbb{E}(\mathbf{y}) = \boldsymbol{X}\boldsymbol{\beta}$ et $\mathbb{V}(\mathbf{y}) = \sigma^2 I_n$. Nous étudions maintenant les propriétés de l'estimateur $\hat{\boldsymbol{\beta}}$. De façon générale, le biais d'un estimateur mesure à quel point l'espérance de cet estimateur s'écarte de la valeur qu'il est censé estimer.

Définition 1.1.4 (Biais d'un estimateur)**.** *Soit $\hat{\boldsymbol{\beta}}$, un estimateur de $\boldsymbol{\beta}$; alors le biais d'un estimateur est défini comme suit*

$$\boldsymbol{B}(\hat{\boldsymbol{\beta}}) = \mathbb{E}(\hat{\boldsymbol{\beta}}) - \boldsymbol{\beta}.$$

Proposition 1.1.5 (Biais et variance de l'estimateur MCO)**.** *Sous les conditions de Gauss-Markov, l'estimateur $\hat{\boldsymbol{\beta}}$ est un estimateur sans biais et sa variance vaut $\mathbb{V}(\hat{\boldsymbol{\beta}}) = \sigma^2(\boldsymbol{X}^\top\boldsymbol{X})^{-1}$.*

Démonstration. La seule variable aléatoire dans l'expression $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\mathbf{y}$ étant $\mathbf{y}$, alors l'espérance et la variance de $\hat{\boldsymbol{\beta}}$ sont calculées par rapport à la distribution de $\mathbf{y}$:

$$\mathbb{E}(\hat{\boldsymbol{\beta}}) = \mathbb{E}((\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\mathbf{y}) = (\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\mathbb{E}(\mathbf{y}) = (\boldsymbol{X}^\top\boldsymbol{X})^{-1}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{\beta}.$$

L'estimateur $\hat{\boldsymbol{\beta}}$ est donc sans biais. La variance est

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

□

En statistique, l'erreur quadratique moyenne (*EQM*) est une mesure de précision qui nous permet de comparer les différents estimateurs. Son expression sera constituée d'un terme de variance et de biais; le meilleur estimateur est celui qui a la plus petite *EQM*.

Définition 1.1.6 (Erreur quadratique moyenne)**.** *L'erreur quadratique moyenne (EQM) d'un estimateur $\hat{\boldsymbol{\beta}}$ de $\boldsymbol{\beta}$ est définie comme suit*

$$EQM(\hat{\boldsymbol{\beta}}) = \mathbb{E}((\hat{\boldsymbol{\beta}} - \boldsymbol{\beta})^2) = \mathbb{V}(\hat{\boldsymbol{\beta}}) + \boldsymbol{B}(\hat{\boldsymbol{\beta}})^2 = \mathbb{V}(\hat{\boldsymbol{\beta}}).$$

Cette quantité est essentiellement une mesure de performance et de qualité pour l'estimateur $\hat{\boldsymbol{\beta}}$.

On voit que l'*EQM* de $\hat{\boldsymbol{\beta}}$ ne dépend que de la variance, car le biais est nul. Alors, on aimerait maintenant trouver le meilleur estimateur possible parmi la classe des estimateurs sans biais, ce qui revient à trouver l'estimateur ayant la plus petite variance.

Théorème 1.1.7 (Estimateurs BLUE)**.** *Sous les hypothèses de Gauss-Markov, pour toute combinaison linéaire des paramètres, $a^\top\boldsymbol{\beta}$ avec $a = (a_1, \ldots, a_p)^\top$, l'estimateur $a^\top\hat{\boldsymbol{\beta}}$ possède la plus petite variance parmi tous les estimateurs linéaires et sans biais. C'est donc un estimateur BLUE ou* Best Linear Unbiased Estimator *(estimateur sans biais à variance minimale).*

DÉMONSTRATION. Soit $\tilde{\boldsymbol{\beta}} = C\boldsymbol{y}$, un estimateur linéaire et sans biais; alors

$$\boldsymbol{\beta} = \mathbb{E}(C\mathbf{y}) = C\boldsymbol{X}\boldsymbol{\beta},\ \forall \boldsymbol{\beta}.$$

Étant sans biais, alors $C\boldsymbol{X} = I_p$. Sous les hypothèses de Gauss-Markov,

$$\begin{aligned}
\mathbb{V}(a^\top \hat{\boldsymbol{\beta}}) &= \sigma^2 a^\top (\boldsymbol{X}^\top \boldsymbol{X})^{-1} a \\
&= \sigma^2 a^\top C\boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top C^\top a \\
&\leq \sigma^2 a^\top C I_n C^\top a \\
&= \mathbb{V}(a^\top \tilde{\boldsymbol{\beta}}).
\end{aligned}$$

L'inégalité provient du fait que la quantité $I_n - \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top$ est positive semi définie, car la matrice de design est de rang plein. Ainsi, l'inégalité est valide et donc l'estimateur $\hat{\boldsymbol{\beta}}$ est BLUE.

□

L'intérêt de l'estimateur MCO est que parmi tous les estimateurs linéaires et sans-biais, il est celui qui a la plus petite variance. De plus, cet estimateur est simple à calculer. Pour ces raisons, c'est l'estimateur le plus utilisé en régression linéaire multiple. Par contre, bien qu'on puisse calculer $\hat{\boldsymbol{\beta}}$, on n'a pas directement accès à sa variance $\mathbb{V}(\hat{\boldsymbol{\beta}})$ puisqu'elle dépend de σ^2. Il est alors important de comprendre comment estimer σ^2 afin de pouvoir estimer la variance $\mathbb{V}(\hat{\boldsymbol{\beta}})$.

1.1.1. Les résidus et la variance de l'estimateur MCO

Pour pouvoir estimer la variance de $\hat{\boldsymbol{\beta}}$, il faut définir la notion de résidu. Le résidu d'une observation donnée est la différence entre sa valeur observée et sa valeur prédite. Les résidus nous permettront de développer un estimateur sans biais pour σ^2.

Définition 1.1.8 (Résidus)**.** *Le vecteur des résidus est défini comme*

$$\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \hat{\mathbf{y}},$$

où $\hat{\boldsymbol{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}}$ est le vecteur des valeurs prédites étant donné les variables explicatives observées. En développant l'équation, on obtient

$$\hat{\boldsymbol{\varepsilon}} = \mathbf{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}} = \mathbf{y} - \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \mathbf{y} = \mathbf{y} - H\mathbf{y} = (I_n - H)\mathbf{y} = H_\perp \mathbf{y},$$

où $H = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top$ est la matrice de projection et $H_\perp = I_n - H$.

En régression linéaire, les résidus permettent de quantifier la distance entre la réponse observée y_i et la réponse prédite $\hat{y}_i$ pour $i \in \{1, \ldots, n\}$. Ainsi, il est important que les valeurs $\hat{y}_i$ se rapprochent le plus possible des vraies valeurs y_i, ce qui témoigne d'un modèle

de régression linéaire bien ajusté pour les données. Nous étudions maintenant l'espérance et la variance des résidus.

Proposition 1.1.9 (Espérance et variance des résidus)**.** *Le vecteur de résidus est tel que* $\mathbb{E}(\hat{\boldsymbol{\varepsilon}}) = 0$ *et la variance des résidus satisfait* $\mathbb{V}(\hat{\boldsymbol{\varepsilon}}) = \sigma^2 H_\perp$.

Démonstration. L'espérance et la variance sont calculées par rapport à la distribution de la variable aléatoire y :

$$\mathbb{E}(\hat{\boldsymbol{\varepsilon}}) = \mathbb{E}(H_\perp \mathbf{y}) = H_\perp \mathbb{E}(\mathbf{y}) = H_\perp \boldsymbol{X}\boldsymbol{\beta} = \boldsymbol{X}\boldsymbol{\beta} - \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{\beta} = 0.$$

Sachant que $H_\perp$ est une matrice symétrique et idempotente, la variance satisfait

$$\mathbb{V}(\hat{\boldsymbol{\varepsilon}}) = \mathbb{V}(H_\perp \mathbf{y}) = H_\perp \mathbb{V}(\mathbf{y}) H_\perp^\top = \sigma^2 H_\perp.$$

□

On définit les sommes de carrés suivantes qui nous aident à évaluer la pertinence globale d'un modèle de prédiction en régression linéaire. Les quantités suivantes seront étudiées dans la section 1.2 pour faire de la sélection de modèle.

Définition 1.1.10 (Somme des carrés)**.** *La somme des carrés totaux (SCT) est définie comme suit*

$$SCT = \|\mathbf{y} - \bar{y}\mathbf{1}_n\|^2,$$

où $\bar{y} = n^{-1}\sum_{i=1}^n y_i$ *est la moyenne de la variable réponse et* $\|\cdot\|$ *est la norme euclidienne. La somme résiduelle des carrés (SRC) est définie comme suit*

$$SRC = \|\hat{\boldsymbol{\varepsilon}}\|^2 = \|\mathbf{y} - \hat{\mathbf{y}}\|^2,$$

où $\hat{\mathbf{y}} = \boldsymbol{X}\hat{\boldsymbol{\beta}}$ *est le vecteur de valeurs prédites. La somme des carrés expliqués (par la régression) est*

$$SCE = \|\hat{\mathbf{y}} - \bar{y}\mathbf{1}_n\|^2.$$

Il est possible de vérifier que $SCT = SCE + SRC$.

Naturellement, une petite somme résiduelle des carrés indique un très bon ajustement du modèle de régression linéaire, car ceci signifie que les résidus sont minimes et donc que les valeurs prédites $\hat{y}_i$ sont près des valeurs observées y_i. De façon générale, il est possible d'utiliser $\|\hat{\boldsymbol{\varepsilon}}\|^2$ comme estimateur pour σ^2. Calculons l'espérance de cet estimateur afin de voir s'il est biaisé. Puisque $\|\hat{\boldsymbol{\varepsilon}}\|^2$ est un scalaire, qui peut être vu comme une matrice de dimension 1×1, alors il est possible par les propriétés de la trace d'écrire

$$\mathbb{E}(\|\hat{\boldsymbol{\varepsilon}}\|^2) = \mathbb{E}(\hat{\boldsymbol{\varepsilon}}^\top \hat{\boldsymbol{\varepsilon}}) = \mathbb{E}(\mathrm{Tr}(\hat{\boldsymbol{\varepsilon}}^\top \hat{\boldsymbol{\varepsilon}})) = \mathbb{E}(\mathrm{Tr}(\hat{\boldsymbol{\varepsilon}}\hat{\boldsymbol{\varepsilon}}^\top)) = \mathrm{Tr}\left(\mathbb{E}(\hat{\boldsymbol{\varepsilon}}\hat{\boldsymbol{\varepsilon}}^\top)\right) = \mathrm{Tr}\left(\mathbb{V}(\hat{\boldsymbol{\varepsilon}}) + \mathbb{E}(\hat{\boldsymbol{\varepsilon}})\mathbb{E}(\hat{\boldsymbol{\varepsilon}})^\top\right).$$

Selon la proposition 1.1.9, l'espérance de $\hat{\boldsymbol{\varepsilon}}$ est nulle et sa variance est $\sigma^2(I_n - H)$. Puisque la trace de la matrice identité est n et que celle de la matrice de projection est p, alors selon

les propriétés de la trace,

$$\mathrm{Tr}\left(\mathbb{V}(\hat{\boldsymbol{\varepsilon}})\right) = \mathrm{Tr}\left(\sigma^2(I_n - H)\right) = \sigma^2(n-p).$$

Il est clair que l'estimateur $\|\hat{\boldsymbol{\varepsilon}}\|^2$ est biaisé; il suffit alors de le diviser par $n-p$ pour le rendre sans biais.

Proposition 1.1.11. *Soit S^2, défini comme suit*

$$S^2 = \frac{\|\hat{\boldsymbol{\varepsilon}}\|^2}{n-p};$$

alors, S^2 est sans biais pour σ^2.

Démonstration. Sachant que $\mathbb{E}(\|\hat{\boldsymbol{\varepsilon}}\|^2) = \sigma^2(n-p)$, alors

$$\mathbb{E}(S^2) = \frac{\mathbb{E}(\|\hat{\boldsymbol{\varepsilon}}\|^2)}{n-p} = \frac{\sigma^2(n-p)}{n-p} = \sigma^2.$$

□

Ayant identifié un estimateur pour σ^2, il est maintenant possible d'estimer la variance de $\hat{\boldsymbol{\beta}}$.

Proposition 1.1.12. *Soit la variance estimée de l'estimateur MCO $\hat{\boldsymbol{\beta}}$, définie comme*

$$\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}}) = S^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1}; \tag{1.1.2}$$

alors, $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})$ est sans biais pour $\mathbb{V}(\hat{\boldsymbol{\beta}})$.

Démonstration. Sachant que $\mathbb{E}(S^2) = \sigma^2$, alors

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

□

En pratique, on peut maintenant calculer $\hat{\boldsymbol{\beta}}$ à l'aide de l'équation (1.1.1), ainsi qu'estimer la matrice de variance-covariance à l'aide de l'équation (1.1.2). La variance estimée de chacun des termes $\hat{\beta}_j$ pour $j \in \{1, \ldots, p\}$ se trouve sur la diagonale de $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})$. Néanmoins, il serait intéressant de développer des lois pour $\hat{\boldsymbol{\beta}}$ afin d'effectuer des tests d'hypothèse ou de construire des intervalles de confiance pour chacun des $\hat{\beta}_j$, $j \in \{1, \ldots, p\}$. Pour ce faire, il sera nécessaire d'introduire une nouvelle hypothèse concernant le vecteur des termes d'erreur.

1.1.2. Inférence pour l'estimateur MCO

L'inférence statistique nous permet de tirer des conclusions sur la vraie valeur de $\boldsymbol{\beta}$ à l'aide de statistiques de test ou encore d'intervalles de confiance. Nous avons déjà introduit deux hypothèses simplificatrices spécifiant que la matrice de design doit être de rang plein et que les conditions de Gauss-Markov doivent être respectées. Afin de pouvoir construire la

théorie pour l'inférence du paramètre $\boldsymbol{\beta}$, il faut cependant ajouter une troisième hypothèse concernant le vecteur des termes d'erreur.

Définition 1.1.13. *Soit le vecteur de bruit* $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)^\top$. *On suppose que les termes* ε_i *sont indépendants et que* $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ *pour* $i \in \{1, \ldots, n\}$. *De façon spécifique,*

$$\varepsilon_1, \ldots, \varepsilon_n \overset{iid}{\sim} \mathcal{N}(0, \sigma^2).$$

Ceci peut également être exprimé comme

$$\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n).$$

Cette hypothèse introduit la normalité des erreurs, ce qui nous permet de construire des lois pour $\mathbf{y}$ et $\hat{\boldsymbol{\beta}}$. On sait déjà que $\mathbb{E}(\mathbf{y}) = \boldsymbol{X}\boldsymbol{\beta}$ et $\mathbb{V}(\mathbf{y}) = \sigma^2 I_n$; avec la nouvelle hypothèse, nous avons alors

$$\mathbf{y} \sim \mathcal{N}(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 I_n). \tag{1.1.3}$$

Le théorème de Cochran nous aide à déterminer la loi de l'estimateur $\hat{\boldsymbol{\beta}}$ et à construire, par la suite, les tests statistiques et leurs intervalles de confiances correspondants.

Théorème 1.1.14 (Théorème de Cochran)**.** *Soit* $\mu \in \mathbb{R}^n$, $\mathbf{y} \sim \mathcal{N}(\mu, \sigma^2 I_n)$ *et* M, *un sous-espace de* $\mathbb{R}^n$ *de dimension* p. *Soit* $H = \boldsymbol{X}(\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top$, *la matrice de projection de* $\mathbb{R}^n$ *sur* M *et son orthogonal* $H_\perp = I_n - H$ *de dimension* $n - p$. *On a*

(i) $H\mathbf{y} \sim \mathcal{N}(H\mu, \sigma^2 H)$.

(ii) Les vecteurs $\mathbf{y}$ *et* $\mathbf{y} - H\mathbf{y}$ *sont indépendants.*

(iii) $\|H_\perp \mathbf{y} - H_\perp \mu\|^2/\sigma^2 \sim \chi^2_{n-p}$.

Démonstration. La preuve complète du théorème de Cochran peut être retrouvée dans la section 3.3 de Bapat (2020).

□

Ayant introduit le théorème de Cochran, il est maintenant possible de déterminer les lois de $\hat{\boldsymbol{\beta}}$ et de S^2.

Proposition 1.1.15. *Sachant que* $rang(\boldsymbol{X}) = p$ *et* $\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n)$, *alors*

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

(2) $(n-p)S^2/\sigma^2 \sim \chi^2_{n-p}$,

(3) $\hat{\boldsymbol{\beta}}$ *et* S^2 *sont indépendants.*

Démonstration. En utilisant le théorème de Cochran, chacun des énoncés de la proposition se démontre facilement.

(1) Sachant que $\hat{\boldsymbol{\beta}}$ est une fonction linéaire de $\mathbf{y}$ et que $\mathbf{y}$ suit la loi (1.1.3), alors $\hat{\boldsymbol{\beta}}$ suit également une normale d'espérance $\boldsymbol{\beta}$ et de variance $\sigma^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1}$, selon la proposition (1.1.5).

(2) Sachant que $H_\perp = I_n - H$ et que la dimension de $H_\perp$ est $n-p$, ainsi que $H_\perp \boldsymbol{X}\boldsymbol{\beta} = 0_n$, alors

$$S^2 = \frac{\|\hat{\boldsymbol{\varepsilon}}\|^2}{n-p} = \frac{\|\mathbf{y} - \hat{\mathbf{y}}\|^2}{n-p} = \frac{\|\mathbf{y} - H\mathbf{y}\|^2}{n-p} = \frac{\|H_\perp \mathbf{y}\|^2}{n-p} = \frac{\|H_\perp \mathbf{y} - H_\perp \boldsymbol{X}\boldsymbol{\beta}\|^2}{n-p}.$$

En multipliant par $(n-p)/\sigma^2$ de chaque côté, il est évident selon l'énoncé (iii) du théorème de Cochran que

$$\frac{(n-p)S^2}{\sigma^2} \sim \chi^2_{n-p}.$$

(3) On sait que $\hat{\boldsymbol{\beta}}$ est une fonction de $\mathbf{y}$ et que S^2 est une fonction de $\mathbf{y} - H\mathbf{y}$; par conséquent, selon l'énoncé (ii) du théorème de Cochran, on sait que $\hat{\boldsymbol{\beta}}$ et S^2 sont indépendants.

□

Le théorème de Cochran et la proposition 1.1.15 nous permettent de développer la théorie nécessaire à l'inférence de l'estimateur MCO. Ainsi, il est maintenant possible de construire des tests statistiques et des intervalles de confiance pour les coefficients de régression du modèle linéaire.

Proposition 1.1.16. *Sachant que* $rang(\boldsymbol{X}) = p$ *et* $\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n)$*, alors*

- *Un test statistique pour* β_j *est*

$$T_j = \frac{\hat{\beta}_j - \beta_j}{\sqrt{\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj}}} \sim t_{n-p}, \tag{1.1.4}$$

  où $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj} = S^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1}_{jj}$*, pour* $j \in \{1, \ldots, p\}$ *et* t_{n-p} *est une distribution Student à* $n-p$ *degrés de liberté.*
- *Un intervalle de confiance de niveau* $1-\alpha$ *pour* β_j *est donné par*

$$\hat{\beta}_j \ \pm \ t_{1-\alpha/2, n-p} \times \sqrt{\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj}},$$

  pour $j \in \{1, \ldots, p\}$*, où* $t_{1-\alpha/2,n-p}$ *est le quantile* $1-\alpha/2$ *de la distribution Student à* $n-p$ *degrés de liberté.*

Démonstration. Sachant que $\hat{\boldsymbol{\beta}}$ suit une normale d'espérance $\boldsymbol{\beta}$ et de variance $\mathbb{V}(\hat{\boldsymbol{\beta}}) = \sigma^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1}$, alors pour chaque β_j, on a

$$Z_j = \frac{\hat{\beta}_j - \beta_j}{\sigma\sqrt{(\boldsymbol{X}^\top \boldsymbol{X})^{-1}_{jj}}} \sim \mathcal{N}(0,1).$$

En général, si $Z \sim \mathcal{N}(0,1)$, $U \sim \chi^2_d$ et Z est indépendant de U, alors on peut construire la statistique $T = Z/\sqrt{U/d}$ de loi t_d. Dans notre contexte de régression, en utilisant les énoncés

(ii) et (iii) de la proposition 1.1.15, on a

$$T_j = \frac{\hat{\beta}_j - \beta_j}{\sqrt{\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj}}} \sim t_{n-p},$$

où $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj} = S^2(\boldsymbol{X}^\top \boldsymbol{X})^{-1}_{jj}$, pour $j \in \{1, \ldots, p\}$.

□

L'inférence de l'estimateur $\hat{\boldsymbol{\beta}}$ joue un rôle primordial dans la régression linéaire, car il est important de pouvoir construire des intervalles de confiance et de pouvoir faire des tests d'hypothèse sur les β_j. En pratique, il est commun de confronter les hypothèses $H_0 : \{\beta_j = 0\}$ et $H_a : \{\beta_j \neq 0\}$. Ceci nous permet de déterminer si la j-ème variable explicative est pertinente ou non pour le modèle de régression linéaire. De ce fait, il sera important de voir quelles sont les variables qui sont pertinentes dans notre modèle. Ceci nous amène à aborder la sélection de modèle dans un contexte de régression linéaire.

1.2. Sélection de modèle

Dans la section précédente, nous avons posé trois hypothèses clés :

(1) Le $rang(\boldsymbol{X}) = p$.

(2) Les erreurs sont centrées, de même variance $\sigma^2 > 0$ et non corrélés entre elles.

(3) Les erreurs sont $\varepsilon_1, \ldots, \varepsilon_n \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$.

Les trois hypothèses nous permettent respectivement d'expliciter la forme de $\hat{\boldsymbol{\beta}}$ à l'aide de l'équation (1.1.1), de trouver l'espérance et la variance estimée de $\hat{\boldsymbol{\beta}}$ à l'aide de l'équation (1.1.2) et de mener une inférence sur $\hat{\boldsymbol{\beta}}$ tel que décrit dans la section 1.1.2. Tel que vu précédemment, l'intérêt principal de cet estimateur est sa simplicité; il est, de surcroît, un estimateur sans biais et à variance minimale parmi les estimateurs linéaires de $\hat{\boldsymbol{\beta}}$.

Puisque nous travaillons souvent avec un grand nombre de variables explicatives, il est important de comprendre quelles sont les variables qui expliquent le mieux la variable réponse. Celles-ci ne sont généralement pas toutes importantes pour le modèle. Nous verrons que le fait d'utiliser un nombre de variables explicatives relativement restreint dans le modèle de régression mène souvent à une plus petite variance pour $\hat{\boldsymbol{\beta}}$, des estimateurs $\hat{\boldsymbol{\beta}}$ plus précis et une réduction du surapprentissage du modèle.

1.2.1. Méthodes de sélection de variables traditionnelles

Les méthodes de sélection de variables traditionnelles en régression linéaire sont très intéressantes et souvent performantes. Ces méthodes nous permettent de choisir un sous-modèle offrant des prédictions de bonne qualité et une interprétation adéquate du modèle. Dans le but de bien comprendre ces méthodes, nous introduisons maintenant quelques mesures

d'ajustement de modèles telles que les coefficients R^2 et R^2 ajusté (R^2_a), ainsi que le C_p de Mallows. Ces quantités évaluent l'ajustement d'un modèle de régression linéaire et nous permettront éventuellement de faire le choix d'un modèle spécifique.

Définition 1.2.1. *Le coefficient R^2 est défini par*

$$R^2 = \frac{SCE}{SCT} = 1 - \frac{SRC}{SCT}.$$

Le coefficient R^2 mesure l'ajustement d'un modèle de régression linéaire en utilisant les sommes de carrés de la définition 1.1.10. Ayant des valeurs entre 0 et 1, plus le R^2 s'approche de 1, plus l'ajustement s'approche de la perfection. En calculant le coefficient R^2 pour tous les sous-modèles possibles, on choisit celui qui a le plus grand R^2. Par contre, il a été montré que le R^2 n'est pas fiable comme méthode de sélection de modèle lorsqu'il faut comparer des modèles avec différents nombres de variables explicatives. En effet, le coefficient R^2 ne pénalise pas en fonction du nombre de variables utilisées et donc les modèles plus complexes sont favorisés.

Pour pallier à ce problème, le R^2_a a été proposé, soit

$$R^2_a = 1 - \frac{SRC/(n-p)}{SCT/(n-1)}.$$

Le coefficient R^2_a est similaire au coefficient R^2, mais les quantités SRC et SCT sont normalisées par leurs degrés de liberté. Le R^2_a pénalise donc les modèles plus complexes l'aide de la division par $n-p$. Comme pour le coefficient R^2, il faut choisir le sous-modèle qui a le plus grand R^2_a. Ce sous-modèle nous mène généralement à des prédictions de bonne qualité et à une interprétation adéquate du modèle.

Une autre mesure d'ajustement de modèle utilisée est le C_p de Mallows. Cette mesure est définie comme suit

$$C_p = \frac{\|\tilde{\boldsymbol{\varepsilon}}\|^2}{S^2} - n + 2|\tilde{\boldsymbol{X}}|,$$

où $\tilde{\boldsymbol{X}}$ est un modèle réduit ayant un certain nombre $|\tilde{\boldsymbol{X}}|$ de variables explicatives. De plus, le vecteur des résidus pour ce modèle est $\tilde{\boldsymbol{\varepsilon}}$. En pratique, il s'agit de conserver le modèle ayant le plus petit C_p. Certains auteurs recommandent que $C_p \approx |\tilde{\boldsymbol{X}}|$; voir Hastie *et al.* (2009).

Il existe aussi des critères d'information basés sur la vraisemblance d'un modèle de régression. Sous l'hypothèse du bruit gaussien, on sait que $\mathbf{y} \sim \mathcal{N}(\boldsymbol{X}\boldsymbol{\beta}, \sigma^2 I_n)$. Connaissant la densité de la loi normale, il est alors possible de trouver la fonction de vraisemblance de ce modèle. Pour obtenir les estimateurs à vraisemblance maximale (VM) de $\boldsymbol{\beta}$ et σ^2, $\hat{\boldsymbol{\beta}}_{VM}$ et S^2_{VM}, il faut maximiser la log-vraisemblance. En régression linéaire, il est facile de vérifier que $\hat{\boldsymbol{\beta}}_{VM} = \hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1}\boldsymbol{X}^\top \mathbf{y}$ et $S^2_{VM} = \|\hat{\boldsymbol{\varepsilon}}\|^2/n \neq \|\hat{\boldsymbol{\varepsilon}}\|^2/(n-p) = S^2$. On remarque donc que l'estimateur de la moyenne est le même que sous l'approche des moindres carrés, mais

que les estimateurs de la variance sont différents. Spécifiquement, l'estimateur du maximum de vraisemblance de la variance n'est pas sans biais, contrairement à son équivalent MCO. Maintenant, la log-vraisemblance évaluée à $\hat{\boldsymbol{\beta}}_{VM}$ et S^2_{VM} satisfait

$$\begin{aligned} L(\boldsymbol{X}; \hat{\boldsymbol{\beta}}_{VM}; S^2_{VM}) &= \log\left\{(2\pi S^2_{VM})^{-n/2} \exp\left(-\frac{\|\mathbf{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{VM}\|^2}{2S^2_{VM}}\right)\right\} \\ &= -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(S^2_{VM}) - \frac{nS^2_{VM}}{2S^2_{VM}} \\ &= -\frac{n}{2}\log\left(\frac{\|\mathbf{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}_{VM}\|^2}{n}\right) - \frac{n}{2}(1 + \log(2\pi)), \end{aligned}$$

où $\hat{\boldsymbol{\beta}}_{VM} = \hat{\boldsymbol{\beta}}$ et $S^2_{VM} = \|\hat{\boldsymbol{\varepsilon}}\|^2/n$.

Cette log-vraisemblance nous permet d'énoncer deux critères d'information très utilisés en statistique, soient le critère d'information d'*Akaike* (AIC) et le critère d'information de Bayes (BIC).

Soit $\tilde{\boldsymbol{X}}$, un modèle ayant $|\tilde{\boldsymbol{X}}|$ variables explicatives; alors, le critère AIC satisfait

$$\text{AIC} = -2L(\tilde{\boldsymbol{X}}; \tilde{\boldsymbol{\beta}}_{VM}; \tilde{S}^2_{VM}) + 2|\tilde{\boldsymbol{X}}|,$$

alors que l'expression pour le BIC est

$$\text{BIC} = -2L(\tilde{\boldsymbol{X}}; \tilde{\boldsymbol{\beta}}_{VM}; \tilde{S}^2_{VM}) + |\tilde{\boldsymbol{X}}|\log(n),$$

où $\tilde{\boldsymbol{\beta}}_{VM}$ et $\tilde{S}^2_{VM}$ sont les estimateurs VM associés au modèle à $|\tilde{\boldsymbol{X}}|$ variables explicatives.

Le critère choisi (AIC ou BIC) est calculé pour tous les modèles possibles et le modèle qui minimise ce critère est sélectionné. En régression linéaire, le critère AIC est équivalent au C_p de Mallows selon Akaike (1998). Les critères AIC et BIC sont très similaires, mais se distinguent dans la pénalité qu'ils appliquent relativement à la taille du modèle. Le critère AIC pénalise chaque paramètre additionnel par un facteur de 2, alors que le BIC utilise plutôt un facteur de $\log(n)$. Ainsi, BIC a tendance à choisir des modèles plus parcimonieux que AIC.

D'autres méthodes de sélection de variables existent, par exemple la méthode de sélection ascendante, d'élimination descendante ou encore la méthode séquentielle. En fixant un seuil α et en utilisant des tests d'hypothèse de la forme $H_0 : \{\beta_j = 0\}$ vs $H_a : \{\beta_j \neq 0\}$, il est possible d'identifier (ou de sélectionner) les prédicteurs les plus importants. Les différents tests font intervenir les statistiques T_j en (1.1.4). Par exemple, pour la méthode descendante, on commence avec le modèle complet à p variables et on retire la variable avec la plus petite statistique T_j (c.-à-d. la plus grande valeur-p). On se retrouve avec $p-1$ variables explicatives et on continue à retrancher des prédicteurs jusqu'à ce que toutes les variables explicatives

du modèle soient significatives (c'est-à-dire qu'aucun prédicteur supplémentaire ne peut être écarté). Ces méthodes peuvent être utilisées en parallèle avec les critères C_p, AIC et BIC.

Précédemment, trois hypothèses ont été énoncées quant au modèle de régression linéaire. Celles-ci nous ont permis de trouver $\hat{\boldsymbol{\beta}}$, de faire de l'inférence et de présenter des méthodes de sélection de variables. Toutefois, si l'hypothèse du rang de la matrice de design n'est pas satisfaite, alors il est important de comprendre quelles sont les conséquences engendrées sur notre inférence.

1.2.2. Violation de l'hypothèse du rang

Dans la section 1.2.1, nous avons fait la supposition que $\boldsymbol{X}$ est de rang plein p. Si le $rang(\boldsymbol{X}) < p$, il est évident que l'estimateur n'est plus calculable, car la matrice $\boldsymbol{X}^\top \boldsymbol{X}$ n'est plus inversible. Ceci peut être visualisé géométriquement à l'aide des valeurs propres de la matrice $\boldsymbol{X}^\top \boldsymbol{X}$. Puisque $\boldsymbol{X}^\top \boldsymbol{X}$ est une matrice symétrique, alors

$$\boldsymbol{X}^\top \boldsymbol{X} = V \Lambda V^\top = \sum_{j=1}^{p} \kappa_j (v_j v_j^\top) \implies (\boldsymbol{X}^\top \boldsymbol{X})^{-1} = \sum_{j=1}^{n} \frac{1}{\kappa_j} (v_j v_j^\top),$$

où V est une matrice orthogonale $p \times p$ et Λ est une matrice diagonale de valeurs propres $diag(\kappa_1, \ldots, \kappa_p)$. Si $\kappa_j = 0$ pour $j \in \{1, \ldots, p\}$, alors le déterminant de la matrice symétrique $\boldsymbol{X}^\top \boldsymbol{X}$ est automatiquement égal à 0, car $det(X^T X) = V det(\Lambda) V^T = det(\Lambda) = 0$. Si le déterminant est nul, alors $\boldsymbol{X}^\top \boldsymbol{X}$ n'est pas inversible. Par conséquent, l'estimateur $\hat{\boldsymbol{\beta}}$ n'est plus unique, car il existe une infinité de solutions à la matrice $(\boldsymbol{X}^\top \boldsymbol{X})^{-1}$. Lorsque l'inverse de la matrice symétrique est théoriquement calculable, s'il existe $j \in \{1, \ldots, p\}$ tel que $\kappa_j \approx 0$, deux problèmes potentiels peuvent survenir.

(1) Les composantes de $\hat{\boldsymbol{\beta}}$ sont surévaluées. En d'autres mots, si $\kappa_j \approx 0$, alors $\hat{\boldsymbol{\beta}} \to \infty$, ce qui peut nuire à la fonction prédictive des modèles.

(2) Les variances estimées $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj}$, pour $j \in 1, \ldots, p$, peuvent également tendre vers l'infini si $\kappa_j \approx 0$, ce qui nuit à l'interprétation du modèle de régression linéaire. Ceci aura pour conséquence de potentiellement augmenter la taille des intervalles de confiance.

Ayant identifié deux conséquences majeures de la violation de l'hypothèse de rang, il est maintenant important de comprendre dans quelles situations le rang de $\boldsymbol{X}$ pourrait être inférieur à p. En pratique, ceci peut se présenter dans deux situations. D'une part, supposons que deux variables explicatives $\mathbf{x}_1$ et $\mathbf{x}_2$ sont très corrélées. Pour expliquer $\mathbf{y}$, une seule de ces variables explicatives sera alors nécessaire. Si nous sélectionnons $\mathbf{x}_1$, la variable $\mathbf{x}_2$ sera redondante, car l'information qu'elle apporte est déjà disponible via $\mathbf{x}_1$. Ainsi, le poids $\hat{\beta}_2$ n'est pas pertinent pour expliquer la variable réponse. Généralement, ceci se produit quand il y a de la multicolinéarité entre les variables explicatives X_j pour $j \in \{1, \ldots, p\}$. D'autre

part, si la dimension de $p > n$, alors il y plus de variables explicatives que d'observations, ce qui crée des dépendances linéaires entre les colonnes de la matrice de design $\boldsymbol{X}$.

Une solution possible serait d'éliminer les variables qui sont fortement corrélées lors du processus de sélection de variables. Selon Tibshirani (1996), les méthodes de sélection de variables mentionnées précédemment sont cependant instables et computationnellement coûteuses. Par exemple, lorsque $p < n$, ces méthodes sont sensibles aux données; un modèle ajusté avant le retrait d'une donnée aberrante pourrait donc être très différent d'un modèle ajusté sans cette donnée. Les méthodes sont computationnellement coûteuses dû au nombre de modèles à tester; plus p est grand, plus la matrice de design est grande et plus le nombre de modèles est élevé, ce qui peut être désavantageux en pratique. Finalement, dans le cas où $p > n$, il n'est pas possible de calculer $\hat{\boldsymbol{\beta}}$ et donc les méthodes de sélection de modèle ne nous sont d'aucune aide.

1.3. Régression pénalisée

L'hypothèse du rang de la matrice de design est violée lorsqu'il y a présence de multicolinéarité entre les variables explicatives ou lorsque nous nous retrouvons dans un contexte de grande dimension ($p > n$). L'estimateur $\hat{\boldsymbol{\beta}}$ et les variances estimées $\hat{V}(\hat{\boldsymbol{\beta}})_{jj}$ deviennent alors surévaluées. Notons également que les méthodes de sélection de modèle proposées dans la section 1.2 ne sont pas appropriées lorsque l'hypothèse du rang est violée. C'est pourquoi nous étudions maintenant la régression pénalisée, une méthode populaire qui ajoute à la fonction objective à minimiser, $(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})$, une pénalité $\mathcal{P}(\boldsymbol{\beta})$ qui est une fonction de $\boldsymbol{\beta}$. Nous verrons que la régression pénalisée nous permet principalement d'adresser le problème de multicolinéarité.

Définition 1.3.1 (Fonction objective en régression pénalisée)**.** *Le coefficient de régression pénalisée est défini comme l'estimateur $\hat{\boldsymbol{\beta}}^{(P)}$ qui minimise*

$$\hat{\boldsymbol{\beta}}^{(P)} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda\mathcal{P}(\boldsymbol{\beta}) \right\}, \tag{1.3.1}$$

où $\mathcal{P}(\boldsymbol{\beta})$ est une fonction de pénalité et $\lambda \geq 0$ est un paramètre d'ajustement.

Si $\lambda = 0$, alors il est évident qu'on obtient l'estimateur MCO, soit $\hat{\boldsymbol{\beta}}^{(P)} = \hat{\boldsymbol{\beta}}$. Il existe différents choix pour la pénalité $\mathcal{P}(\boldsymbol{\beta})$, par exemple le Lasso ($\mathcal{P}(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|_1$) ou le Ridge ($\mathcal{P}(\boldsymbol{\beta}) = \|\boldsymbol{\beta}\|^2$). L'ajout de la pénalité à la fonction objective nous permet d'adresser les problèmes soulevés dans la section 1.2.2; en effet, les composantes de $\hat{\boldsymbol{\beta}}$ et les variances estimées $\mathbb{V}(\hat{\boldsymbol{\beta}})_{jj}$ surévaluées seront rétrécies grâce à cette pénalité. Notons, de plus, que la variable d'ajustement λ permet de contrôler le degré du rétrécissement.

1.3.1. Ridge

L'estimateur Ridge introduit par Hoerl et Kennard (1970) est une forme très simple de la régression pénalisée.

Théorème 1.3.2 (Estimateur Ridge)**.** *Le coefficient de régression Ridge est défini comme l'estimateur* $\hat{\boldsymbol{\beta}}^{(R)}$ *qui minimise*

$$\hat{\boldsymbol{\beta}}^{(R)} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|^2 \right\}, \tag{1.3.2}$$

où $\|\cdot\|$ *est la norme euclidienne et* $\lambda \geq 0$ *est un paramètre d'ajustement. L'estimateur est alors*

$$\hat{\boldsymbol{\beta}}^{(R)} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)^{-1} \boldsymbol{X}^\top \mathbf{y}. \tag{1.3.3}$$

Démonstration. Voir l'annexe A.1.

□

Le Ridge est une façon simple et naïve d'attaquer les deux problèmes de la section 1.2.2. Lorsque les valeurs propres $\{\kappa_j\}_{j=1}^p$ associées à la matrice $\boldsymbol{X}^\top \boldsymbol{X}$ sont près de 0, on peut penser que si une certaine quantité était ajoutée à ces valeurs propres, alors les composantes de $\hat{\boldsymbol{\beta}}$ et les variances estimées $\hat{V}(\hat{\boldsymbol{\beta}})_{jj}$ ne serait plus surévaluées. Ainsi, la matrice $\boldsymbol{X}^\top \boldsymbol{X}$ devient $(\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)$ et par un résultat d'algèbre linéaire, les valeurs propres de cette dernière matrice sont $\{\kappa_j + \lambda\}_{j=1}^p$. De cette façon naïve, les valeurs propres se sont éloignées de 0, ce qui nous permet de régler les problèmes mentionnés dans la section 1.2.2. Il est alors possible d'obtenir l'estimateur Ridge $\hat{\boldsymbol{\beta}}^{(R)}$ en remplaçant la matrice $\boldsymbol{X}^\top \boldsymbol{X}$ de l'estimateur MCO par $(\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)$.

Proposition 1.3.3 (Biais et variance des estimateurs Ridge)**.** *Si les conditions de Gauss-Markov sont satisfaites, alors l'estimateur* $\hat{\boldsymbol{\beta}}^{(R)}$ *est un estimateur possédant un biais égal à*

$$\boldsymbol{B}(\hat{\boldsymbol{\beta}}^{(R)}) = -\lambda (\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)^{-1} \boldsymbol{\beta},$$

et une variance satisfaisant

$$\mathbb{V}(\hat{\boldsymbol{\beta}}^{(R)}) = \sigma^2 (\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)^{-1} \boldsymbol{X}^\top \boldsymbol{X} (\boldsymbol{X}^\top \boldsymbol{X} + \lambda I_p)^{-1}.$$

Démonstration. Voir l'annexe A.2.

□

Contrairement à l'estimateur MCO, l'estimateur Ridge a un biais non nul. Dans une situation où nous aurions à choisir un estimateur adéquat pour nos données, nous pourrions alors faire usage de l'*EQM*, introduit à la définition 1.1.6.

En présence de multicolinéarité dans la matrice de design, l'estimateur Ridge serait un meilleur choix que l'estimateur MCO selon Tibshirani (1996). De ce fait, il est intéressant

de noter le comportement du biais et de la variance de l'estimateur Ridge en fonction du paramètre λ choisi.

Remarque 1.3.4. *Le paramètre λ agit sur le biais et la variance de $\hat{\boldsymbol{\beta}}^{(R)}$ comme suit :*

- *Si $\lambda \to 0$, alors $\hat{\boldsymbol{\beta}}^{(R)} \to \hat{\boldsymbol{\beta}}$, $\boldsymbol{B}(\hat{\boldsymbol{\beta}}^{(R)}) \to 0$ et $\mathbb{V}(\hat{\boldsymbol{\beta}}^{(R)}) \to \infty$.*
- *Si $\lambda \to \infty$, alors $\hat{\boldsymbol{\beta}}^{(R)} \to 0$, $\boldsymbol{B}(\hat{\boldsymbol{\beta}}^{(R)}) \to \infty$ et $\mathbb{V}(\hat{\boldsymbol{\beta}}^{(R)}) \to 0$.*

La régression Ridge est une méthode intuitive introduisant un paramètre d'ajustement λ dans l'équation de $\hat{\boldsymbol{\beta}}$. Ceci nous permet, au prix d'un biais, de rétrécir les coefficients ($\hat{\boldsymbol{\beta}}^{(R)} \to 0$) et de diminuer la variance afin de régler le problème de multicolinéarité. Malgré ses vertus, la régression Ridge ne fait pas de sélection de modèle, ce qui nous oblige à utiliser le modèle complet.

1.3.2. Lasso

Le Lasso, introduit par Tibshirani (1996), est un acronyme anglais pour *Least Absolute Shrinkage and Selection Operator*. La pénalité $\mathcal{P}(\boldsymbol{\beta})$ de cette méthode nous permet non seulement de rétrécir les coefficients $\hat{\beta}_j$ comme avec l'estimateur Ridge, mais elle nous permet également de mettre certains coefficients $\hat{\beta}_j$ à 0. De ce fait, le Lasso conserve les avantages du Ridge en rétrécissant les coefficients, tout en nous permettant de faire de la sélection de modèle par le biais d'un choix de paramètre λ.

Définition 1.3.5 (Fonction objective du Lasso)**.** *Le coefficient de régression Lasso est défini comme l'estimateur $\hat{\boldsymbol{\beta}}^{(L)}$ qui minimise*

$$\hat{\boldsymbol{\beta}}^{(L)} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{2n} (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \|\boldsymbol{\beta}\|_1 \right\}, \tag{1.3.4}$$

où $\|\cdot\|_1$ est la norme en valeur absolue (ou l_1) et $\lambda \geq 0$ est un paramètre d'ajustement.

La norme l_1 dans la pénalité Lasso nous empêche d'avoir accès à une solution analytique pour l'estimateur $\hat{\boldsymbol{\beta}}^{(L)}$. De ce fait, il faut procéder avec des algorithmes computationnels pour trouver l'estimateur Lasso. Il est alors recommandé d'ajouter la constante $1/(2n)$ devant la fonction objective, ce qui permet d'accélérer les calculs pour les différents algorithmes.

Proposition 1.3.6. *Soit des variables explicatives orthonormées, c'est à dire satisfaisant $\boldsymbol{X}^\top \boldsymbol{X} = I_p$. Il est alors possible d'obtenir une solution fermée pour l'estimateur Lasso, soit*

$$\hat{\beta}_j^{(L)} = sign\left(\hat{\beta}_j^{(MCO)}\right) \left(\left|\hat{\beta}_j^{(MCO)}\right| - n\lambda\right)^+, \tag{1.3.5}$$

pour $j = 1, \ldots, p$, où $\hat{\beta}_j^{(MCO)}$ sont les coefficients des moindres carrés ordinaires, $(x)^+ = \max(0, x)$ est l'opérateur qui prend uniquement des valeurs non-négatives et la fonction

sign$(\cdot)$ *est définie comme suit,*

$$sign\left(\hat{\beta}_j^{(MCO)}\right) = \begin{cases} -1 & si\ \hat{\beta}_j^{(MCO)} < 0, \\ 0 & si\ \hat{\beta}_j^{(MCO)} = 0, \\ 1 & si\ \hat{\beta}_j^{(MCO)} > 0. \end{cases}$$

Démonstration. Sachant que les variables explicatives sont orthonormées, la solution de l'estimateur MCO devient $\hat{\boldsymbol{\beta}}^{(MCO)} = \boldsymbol{X}^\top \mathbf{y}$. Il suffit maitenant de développer la fonction objective et de trouver son minimum. Cette fonction satisfait

$$\begin{aligned} \boldsymbol{S}(\boldsymbol{\beta}) &= \frac{1}{2n}(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda\|\boldsymbol{\beta}\|_1 \\ &= \frac{1}{2n}\mathbf{y}^\top\mathbf{y} - \frac{1}{n}\mathbf{y}^\top\boldsymbol{X}\boldsymbol{\beta} + \frac{1}{2n}\boldsymbol{\beta}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta} + \lambda\|\boldsymbol{\beta}\|_1. \end{aligned}$$

Exprimée à l'aide de sommes, celle-ci devient

$$\boldsymbol{S}(\boldsymbol{\beta}) = \frac{1}{2n}\sum_{i=1}^{n} y_i^2 - \frac{1}{n}\sum_{j=1}^{p}\hat{\beta}_j^{(MCO)}\beta_j + \frac{1}{2n}\sum_{j=1}^{p}\beta_j^2 + \lambda\sum_{j=1}^{p}|\beta_j|, \tag{1.3.6}$$

pour $j = 1, \ldots, p$. Maintenant, en examinant cette équation, il est évident que les $\hat{\beta}_j$ qui la minimiseront seront de mêmes signes que les $\hat{\beta}_j^{(MCO)}$ correspondants. En effet, pour minimiser cette équation, il est capital de faire en sorte que chacun des termes dans la somme $-\frac{1}{n}\sum_{j=1}^{p}\hat{\beta}_j^{(MCO)}\beta_j$ soit négatif.

En dérivant par rapport à β_j, nous trouvons donc

$$\begin{aligned} \frac{\partial \boldsymbol{S}(\beta_j)}{\partial \beta_j} &= -\frac{1}{n}\hat{\beta}_j^{(MCO)} + \frac{1}{n}\beta_j + \lambda sign(\beta_j) \\ &= -\frac{1}{n}\hat{\beta}_j^{(MCO)} + \frac{1}{n}\beta_j + \lambda sign\left(\hat{\beta}_j^{(MCO)}\right). \end{aligned}$$

L'estimateur $\hat{\beta}_j^{(L)}$ satisfait alors l'équation

$$-\frac{1}{n}\hat{\beta}_j^{(MCO)} + \frac{1}{n}\hat{\beta}_j^{(L)} + \lambda sign\left(\hat{\beta}_j^{(MCO)}\right) = 0.$$

En isolant $\hat{\beta}_j^{(L)}$, nous trouvons

$$\begin{aligned} \hat{\beta}_j^{(L)} &= \hat{\beta}_j^{(MCO)} - n\lambda sign\left(\hat{\beta}_j^{(MCO)}\right) \\ &= sign\left(\hat{\beta}_j^{(MCO)}\right)\left(\left|\hat{\beta}_j^{(MCO)}\right| - n\lambda\right). \end{aligned}$$

Sachant que $\hat{\beta}_j^{(L)}$ et $\hat{\beta}_j^{(MCO)}$ doivent être de même signe, il est important que $\left(\left|\hat{\beta}_j^{(MCO)}\right| - n\lambda\right)$ soit positif. De ce fait, on utilise l'opérateur $(\cdot)^+$, qui retourne uniquement des valeurs non-négatives, soit

$$\hat{\beta}_j^{(L)} = sign\left(\hat{\beta}_j^{(MCO)}\right)\left(\left|\hat{\beta}_j^{(MCO)}\right| - n\lambda\right)^+. \tag{1.3.7}$$

□

En travaillant avec des variables explicatives orthonormées, il est également possible d'obtenir une solution réduite pour l'estimateur Ridge, soit

$$\hat{\boldsymbol{\beta}}^{(R)} = (1+\lambda)^{-1}\boldsymbol{X}^\top \mathbf{y} = (1+\lambda)^{-1}\hat{\boldsymbol{\beta}}^{(MCO)}. \tag{1.3.8}$$

Spécifions cependant que le rétrécissement pour l'estimateur Ridge s'applique sur tous les prédicteurs simultanément, ce qui nous empêche d'avoir des composantes nulles. De son côté, dans le cas orthonormé, le Lasso met tous les coefficients à 0 lorsque le $\max_j|\hat{\beta}_j^{(MCO)}| \leq n\lambda$, ce qui témoigne de sa supériorité par rapport à l'estimateur Ridge. Toutefois, en pratique, les variables explicatives sont corrélées entre elles. Ainsi, il faut comprendre comment trouver la solution du Lasso à l'aide d'algorithmes, par exemple, l'algorithme de descente du gradient.

Les algorithmes de descente du gradient sont très intéressants, car ils n'utilisent que la dérivée de premier ordre de la fonction objective ($\partial \boldsymbol{S}(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$). Ceci rend les calculs moins complexes et évite d'aller chercher les dérivées d'ordre supérieur, ce qui pourrait parfois être difficile à calculer. Dans l'algorithme de descente du gradient pour le Lasso, le processus descend dans la direction la plus abrupte possible compte tenu de sa valeur actuelle, jusqu'à ce qu'un minimum de la fonction objective (ou une barrière) soit atteint. L'algorithme réévalue alors sa direction afin de poursuivre sa descente (il choisit à nouveau la direction de la descente la plus rapide compte tenu du point actuel de la fonction à minimiser). Lorsque le processus rencontre une barrière, l'un des coefficients de régression est mis à 0. L'algorithme répète ces étapes et élimine un à un les coefficients, jusqu'à ce qu'un minimum de la fonction soit atteint. Le nombre de coefficients rejetés dépend finalement du paramètre d'ajustement λ, qui agit comme un poids sur la contrainte Lasso. Plus il est grand, plus la pénalité est lourde par rapport à la fonction objective et plus l'intention d'écarter les variables explicatives est grande. Lorsque $\lambda = 0$, on se retrouve avec un modèle complet, dont les coefficients $\hat{\boldsymbol{\beta}}$ sont ceux de l'estimateur MCO. En augmentant graduellement le paramètre λ, on passe du modèle complet au modèle nul. D'autres variations de ces algorithmes peuvent être retrouvées dans le livre de Hastie *et al.* (2015).

Les fonctions objectives du Ridge (1.3.2) et du Lasso (1.3.4) sont écrites sous forme lagrangienne. Ces méthodes peuvent être interprétées comme cherchant à minimiser la fonction objective $(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})$, sujet aux contraintes respectives $\|\boldsymbol{\beta}\|_1 \leq t$ et $\|\boldsymbol{\beta}\|^2 \leq t^2$. La représentation géométrique de l'algorithme de descente du gradient peut être visualisée sur la figure 1.1. Seulement deux variables explicatives sont considérées, ce qui mène à une contrainte $|\beta_1| + |\beta_2| < t$ sous forme de losange pour le Lasso et une contrainte $\beta_1^2 + \beta_2^2 < t^2$ sous forme circulaire pour le Ridge. Plus λ augmente, plus la région bleue du graphique diminue. Si l'un des deux coefficients atteint un coin du losange, alors ce coefficient est mis

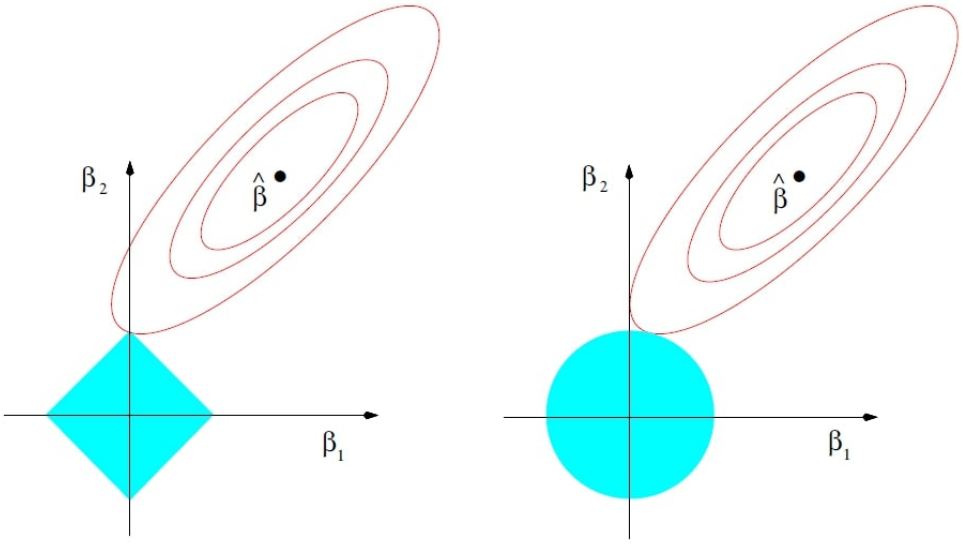


Figure 1.1. Visualisation géométrique du Lasso (gauche) et Ridge (droite). Les courbes de niveau rouges représentent la fonction objective $(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})$, alors que $\hat{\boldsymbol{\beta}}$ représente l'estimateur MCO. Le losange est la contrainte du Lasso et le cercle est la contrainte du Ridge.

à 0. En revanche, la forme circulaire du Ridge ne permet que le rétrécissement uniforme sur toutes les variables.

1.3.3. Choisir le paramètre d'ajustement par validation croisée

La régression pénalisée telle que définie à l'équation (1.3.1) possède un paramètre d'ajustement λ qui doit être sélectionné par l'utilisateur. Ce paramètre doit être choisi judicieusement, car différents choix mèneront à différentes estimations des coefficients. Tel que mentionné dans la remarque 1.3.4, si le paramètre λ est trop grand, les coefficients de l'estimateur Ridge seront rétrécis et la variance sera diminuée, mais le biais sera augmenté. Pour le Lasso, si le paramètre λ est trop petit, le modèle risque de comporter trop de variables alors que s'il est trop grand, le modèle sera trop simple. Il faut donc trouver un juste milieu pour le choix de ce paramètre; pour ce faire, nous utilisons la technique de validation croisée.

Afin de trouver la valeur optimale du paramètre λ parmi les candidats $\lambda \in \{\lambda_1, \ldots, \lambda_m\}$, il faut d'abord diviser aléatoirement le jeu de données en $K > 1$ groupes de tailles (plus ou moins) égales $F_1, \ldots F_K$, où K se situe généralement entre 5 et 10. Nous désignons l'ensemble de test comme étant $(\mathbf{x}_{ij}, \mathbf{y}_i), i \in F_k$ où $k \in \{1, \ldots, K\}$, et combinons ensuite les $K-1$ autres groupes de façon à former un ensemble d'entraînement $(\mathbf{x}_{ij}, \mathbf{y}_i), i \notin F_k$, $j = 1, \ldots, p$. Pour chacun des paramètres $\lambda \in \{\lambda_1, \ldots, \lambda_m\}$, nous ajustons un modèle $\hat{f}_\lambda^{(-k)}$

à l'aide de l'ensemble d'entraînement et nous utilisons ensuite ce modèle pour effectuer des prédictions dans l'ensemble test. Les erreurs quadratiques moyennes (MSE) des prédictions pour l'ensemble test F_k, obtenues avec un modèle $\hat{f}_\lambda^{(-k)}$, sont alors

$$MSE_k(\lambda) = |F_k|^{-1} \sum_{i \in F_k} (\mathbf{y}_i - \hat{f}_\lambda^{(-k)}(\mathbf{x}_{ij}))^2.$$

Ce processus est effectué un total de K fois, chacun des K groupes ayant la chance de jouer le rôle de l'ensemble test. De cette façon, nous obtenons K estimations différentes de l'erreur quadratique moyenne de prédiction pour chacune des valeurs de λ. La moyenne des K estimations de l'erreur de prédiction est ainsi calculée pour chaque valeur de λ,

$$MSE(\lambda) = K^{-1} \sum_{k=1}^{K} MSE_k(\lambda),$$

produisant une courbe d'erreur de validation croisée. On identifie le paramètre λ menant à la plus petite erreur de prédiction, soit

$$\lambda_{min} = \underset{\lambda \in \{\lambda_1, \ldots, \lambda_m\}}{\operatorname{argmin}} \; MSE(\lambda),$$

et on ajuste un nouveau modèle à l'aide de ce paramètre λ_{min}, mais cette fois en utilisant toutes les observations. Cette méthode très simple sera utilisée dans les chapitres ultérieurs pour trouver la valeur optimale de λ.

Chapitre 2

Le Lasso Linéaire : Résolution d'un modèle de position

2.1. Introduction

Dans le chapitre 1, nous avons vu que l'estimateur des moindres carrés ordinaires $\hat{\boldsymbol{\beta}}^{(MCO)}$ est un excellent estimateur pour plusieurs raisons. Par contre, en régression linéaire, nous sommes souvent confrontés à des modèles comportant plusieurs variables explicatives, ce qui entraîne souvent de la multicolinéarité. De plus, il est également possible de se retrouver avec un problème de grande dimensionnalité ($p > n$). C'est alors qu'on observe que l'hypothèse du rang de la matrice de design $\boldsymbol{X}$ est violée; on se retrouve donc avec deux problèmes, soient $\hat{\boldsymbol{\beta}}^{(MCO)} \to \infty$ et $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}}^{(MCO)})_{jj} \to \infty$, tel que discuté dans la section 1.2.2. Une solution possible est d'éliminer les variables qui sont fortement corrélées. Par contre, les méthodes traditionnelles de sélection de variables sont sensibles aux données aberrantes et computationnellement coûteuses en grande dimension ($p > n$). De ce fait, on se tourne vers des méthodes de régression pénalisée telles que le Ridge ou le Lasso. Ces méthodes permettent de rétrécir les coefficients $\hat{\boldsymbol{\beta}}$ et de réduire la variance estimée $\hat{\mathbb{V}}(\hat{\boldsymbol{\beta}})_{jj}$ des coefficients. En plus de rétrécir les coefficients, le Lasso élimine les variables fortement corrélées. Toutefois, le Lasso possède deux problèmes potentiels qui peuvent survenir selon Zou et Hastie (2005).

(1) Si $p > n$, le Lasso sélectionne au plus n variables, avant de saturer dû à la nature convexe du problème. Il s'agit alors de trouver une façon de gérer la grande dimensionnalité.

DÉMONSTRATION. Lorsque $p > n$, le rang de la matrice de design est au plus égal à n, de sorte que la dimension de son espace nul est au moins égale à $p - n$. Dénotons tout vecteur de cet espace nul par z. Alors, en tout point réalisable $\boldsymbol{\beta}$, on peut toujours se déplacer dans cet espace $p - n$ vers les axes de coordonnées de l'espace ambiant p pour arriver à un point $\boldsymbol{\beta} + z$ où (au plus) n β_j sont non nuls, et la fonction

objective du Lasso donnée par

$$\begin{aligned}\|\boldsymbol{y} - \boldsymbol{X}(\boldsymbol{\beta} + z)\|^2 + \lambda\|\boldsymbol{\beta} + z\|_1 &= \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta} + z\|_1 \\ &< \|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|_1\end{aligned}$$

a diminué. Il y a alors une infinité de solutions puisque pour tout vecteur de coefficients $\boldsymbol{\beta}$, $\boldsymbol{\beta} + z$ est également une solution. □

(2) S'il existe un groupe de deux variables explicatives possédant une grande corrélation appariée, le Lasso sélectionne une variable au hasard sans se soucier du choix de la variable.

Pour adresser ces deux problèmes, nous allons nous tourner vers une méthode dérivée du Lasso, introduite par Fraser et Bédard (2022), qui s'intitule le Lasso Linéaire. Cette méthode puise son nom dans la façon géométrique d'interpréter la régression Lasso. Le vecteur $\mathbf{y}$, contenant les réponses, sera vu comme le point focal de l'espace, alors que tous les autres vecteurs de variables explicatives graviteront autour de ce vecteur réponse. Les angles formés par le vecteur réponse et les vecteurs de variables explicatives constituent la base de la théorie du modèle de position utilisé.

Le Lasso Linéaire est un processus d'optimisation qui vise à extraire l'information sur y contenue dans les variables explicatives, et à la représenter sur une seule ligne dans l'espace. Ceci rendra le problème *dimension free*. La résolution du Lasso Linéaire sera simple, car la théorie portant sur les modèles linéaires normaux permettra d'utiliser l'estimateur MCO pour les coefficients estimés. La sélection de variables sera rapide, car les variables explicatives qui ont une grande corrélation avec $\mathbf{y}$ seront priorisées dans le modèle final.

2.2. Notation

Soit y, la variable aléatoire d'intérêt, et $x_1, \ldots, x_p$, les p variables aléatoires explicatives. Supposons n observations pour chacune des $1 + p$ variables, ce qui mène au jeu de données complet $(\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p)$ de dimension $n \times (1+p)$. Chaque vecteur constituant le jeu de données est de longueur $n^{1/2}$, puisque le jeu de données est centré et standardisé par hypothèse. De cette façon, il est possible d'obtenir les corrélations entre les différents vecteurs en calculant le produit croisé de leur version unitaire. Le coefficient de corrélation entre deux vecteurs est simplement le cosinus de l'angle entre ces deux vecteurs centrés. Les corrélations peuvent varier de -1 à 1. Par exemple, la corrélation entre $\mathbf{y}$ et $\mathbf{x}_1$ est $(\mathbf{y}/n^{1/2}) \cdot (\mathbf{x}_1/n^{1/2}) = \mathbf{y} \cdot \mathbf{x}_1/n$. Soit c, le vecteur qui contient la valeur des corrélations entre $\mathbf{y}$ et $\mathbf{x}_j$; alors c satisfait

$$c = (c_j) = (\mathbf{y} \cdot \mathbf{x}_1/n, \ldots, \mathbf{y} \cdot \mathbf{x}_p/n)^\top, \text{ pour } j \in \{1 \ldots, p\}.$$

De façon similaire, pour calculer la corrélation entre les différents prédicteurs, il suffit de calculer le produit interne des p vecteurs unitaires $\mathbf{x}_j/n^{1/2}$, ce qui donne

$$C = (C_{ij}) = \begin{bmatrix} 1 & C_{12} & \dots & C_{1p} \\ C_{21} & 1 & \dots & C_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ C_{p1} & C_{p2} & \dots & 1 \end{bmatrix}.$$

Ainsi, la matrice des corrélations pour le jeu de données $(\mathbf{y}, \mathbf{x}_1, \dots, \mathbf{x}_p)$ peut être exprimée sous la forme suivante

$$\tilde{C} = \begin{bmatrix} 1 & c^\top \\ c & C \end{bmatrix}. \tag{2.2.1}$$

2.2.1. Représentation géométrique

Une fois le jeu de données centré et standardisé, il faut bien établir la représentation géométrique derrière le Lasso Linéaire. Pour ce faire, nous travaillons avec les vecteurs unitaires, soient $\mathbf{u_y} = \mathbf{y}/\sqrt{n}$ et $\mathbf{u}_j = \mathbf{x}_j/\sqrt{n}$, pour $j = 1, \dots p$. Supposons que le vecteur $\mathbf{u_y}$ est placé à l'origine et pointe vers le haut. Tous les vecteurs associés aux variables explicatives gravitent autour de $\mathbf{u_y}$ et sont également centrés à l'origine. La valeur c_j nous donne alors la coordonnée de la projection dans la base $\mathbf{u_y}$. Définissons maintenant la droite $\mathcal{L}\mathbf{y}$, qui se superpose au vecteur $\mathbf{u_y}$ et est perpendiculaire à l'hyperplan $\mathcal{L}^\perp\mathbf{y}$. Les $\mathbf{u}_j$ qui sont corrélés positivement avec $\mathbf{u_y}$ sont donc au-dessus de l'hyperplan $\mathcal{L}^\perp\mathbf{y}$, alors que les $\mathbf{u}_j$ qui sont corrélés négativement avec $\mathbf{u_y}$ pointent vers le bas et sont situés sous l'hyperplan $\mathcal{L}^\perp\mathbf{y}$. Ce cas en 3 dimensions peut être visualisé dans la figure 2.1.

Afin d'établir l'approche du Lasso Linéaire, il sera utile d'appliquer une standardisation de signe aux vecteurs qui sont corrélés négativement avec $\mathbf{u_y}$. Ceci implique que lorsque $c_j < 0$, on utilise $-\mathbf{x}_j$ à la place de $\mathbf{x}_j$. De ce fait, on remarque sur la figure 2.1 que tous les vecteurs unitaires pointent vers le haut et se trouvent dans la moitié supérieure de l'espace. Ceci permet d'alléger la visualisation du modèle et ne change pas la nature du problème à résoudre. Notons que dans le cas particulier où les prédicteurs sont indépendants (c'est-à-dire non corrélés entre eux, avec une matrice C diagonale), cette standardisation de signe mène à des coefficients de régression entièrement positifs. Cet ajustement nous permettra de bien visualiser l'hyperplan $\mathcal{L}^\perp\mathbf{y}$ qui est translaté le long de la droite $\mathcal{L}\mathbf{y}$.

L'hyperplan $\mathcal{L}^\perp\mathbf{y}$ a une fonction similaire à celle du paramètre d'ajustement λ dans la technique du Lasso régulier. Spécifiquement, on peut imaginer que l'hyperplan $\mathcal{L}^\perp\mathbf{y}$ est centré à l'origine et qu'il est translaté graduellement le long de la droite $\mathcal{L}\mathbf{y}$ pour éliminer les variables explicatives séquentiellement. Plus l'hyperplan est haut sur la figure 2.1, plus il y a de variables explicatives qui seront éliminées. En d'autres mots, si un vecteur $\mathbf{u}_j$ se retrouve en-dessous de l'hyperplan $\mathcal{L}^\perp\mathbf{y}$ (nous supposons que l'hyperplan $\mathcal{L}^\perp\mathbf{y}$ est translaté),

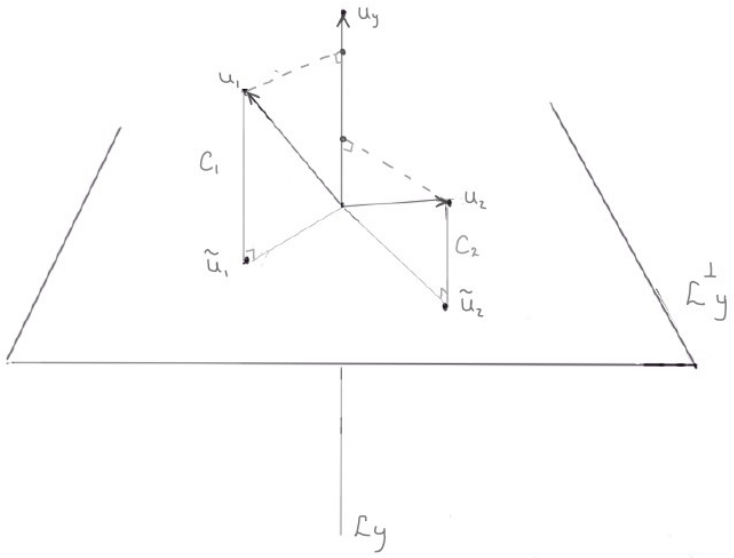


Figure 2.1. Le vecteur de réponse unitaire $\mathbf{u_y}$ et deux vecteurs de prédicteurs unitaires, $\mathbf{u}_1$ et $\mathbf{u}_2$, de même que leur projection associée c_1 et c_2 sur la droite $\mathcal{L}\mathbf{y}$. L'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ coupe $\mathcal{L}\mathbf{y}$ à l'origine.

alors la variable explicative j est retirée du modèle. De la même manière, nous observons que la variable explicative ayant la plus petite corrélation avec $\mathbf{y}$ sera éliminée du modèle en premier. Naturellement, on remarque que la fonction de l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ est semblable à celle du paramètre d'ajustement λ, car plus le paramètre λ est grand, plus il y a de variables qui sont éliminées dans le Lasso régulier. Nous discuterons plus en détails de cette analogie dans la section 2.6 et nous verrons comment spécifier une condition d'arrêt pour le déplacement de l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$.

La représentation géométrique nous permet de visualiser deux aspects importants. D'une part, si l'angle entre $\mathbf{u_y}$ et $\mathbf{u}_1$ est petit, c'est que la corrélation c_1 entre $\mathbf{y}$ et $\mathbf{x}_1$ est très grande, donc l'information apportée par ce prédicteur est pertinente pour expliquer la variable d'intérêt. D'autre part, si deux variables explicatives sont très corrélées avec la variable réponse, mais peu corrélées entre elles (le cas indépendant, par exemple), alors on sait que ces deux variables sont pertinentes pour le modèle. De ce fait, l'analyse sera concentrée autour de la variable réponse, mais fera également intervenir les relations entre les différentes paires de variables explicatives. Dans ce qui suit, nous faisons l'hypothèse que les différents termes de corrélation de la matrice $\tilde{C}$ sont fixes, ce qui signifie que les directions des vecteurs unitaires dans notre représentation géométrique sont constantes et ne peuvent être modifiées une fois observées. Nous verrons dans la prochaine section que les variables aléatoires $y, x_1, \ldots, x_p$ sont alors des variables prenant des valeurs sur les droites $\mathcal{L}\mathbf{y}, \mathcal{L}\mathbf{x}_1, \ldots, \mathcal{L}\mathbf{x}_p$ se confondant

avec les vecteur $\mathbf{u_y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$. Il faut maintenant trouver la distribution conjointe des variables aléatoires $y, x_1, \ldots, x_p$, afin de pouvoir éventuellement mener une inférence sur la variable d'intérêt y. Ceci nous permettra ultimement d'exprimer le problème en une seule dimension.

2.3. Modèle de position

Rappelons d'abord que chacun des vecteurs $\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p$ est centré à 0, possède une variance égale à 1 et est standardisé par signe (les vecteurs pointent au-dessus de $\mathcal{L}^{\perp}\mathbf{y}$). Dans notre représentation géométrique de la figure 2.1, nous avons établi qu'il était plus pratique d'utiliser les vecteurs unitaires $\mathbf{u_y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$ qui leurs sont associés. Ces vecteurs sont tous placés à l'origine, pointent dans différentes directions, et sont supposés fixes. Afin d'établir la distribution conjointe des variables aléatoires $y, x_1, \ldots, x_p$, nous devons maintenant imposer un cadre distributionnel sur notre espace. Puisque les corrélations sont supposées fixes, ceci sous-entend un contexte stochastique commun pour nos $1 + p$ variables. En fait, dans notre contexte de régression linéaire, chaque variable $y, x_1, \ldots, x_p$ peut être vue comme étant une combinaison linéaire d'une distribution normale latente en n dimensions sur un espace vectoriel sous-jacent. Ces fonctions linéaires prennent alors des valeurs sur les droites $\mathcal{L}\mathbf{y}, \mathcal{L}\mathbf{x}_1, \ldots, \mathcal{L}\mathbf{x}_p$, qui formeront le système de coordonnées de l'espace des variables aléatoires $y, x_1, \ldots, x_p$.

Spécifiquement, supposons un modèle stochastique sous-jacent avec n variables latentes, $Z_1, \ldots, Z_n$, qui sont conjointement distribuées selon une normale multivariée standard, $\mathcal{MN}(0_n, I_{n\times n})$. Nous exprimons alors chacune de nos $1+p$ variables aléatoires comme étant une combinaison linéaire de ce modèle latent. Le théorème suivant développe la distribution conjointe des variables aléatoires $y, x_1, \ldots, x_p$.

Théorème 2.3.1. *Supposons que les vecteurs unitaires $\mathbf{u_y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$ provenant des données fixent les directions dans l'espace et que la variable aléatoire x_1 est le coefficient de la projection orthogonale du vecteur Z sur le vecteur $\mathbf{u}_1$. Supposons également que les droites $\mathcal{L}\mathbf{y}, \mathcal{L}\mathbf{x}_1, \ldots, \mathcal{L}\mathbf{x}_p$ passent par l'origine et suivent la même direction que $\mathbf{u_y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$. Alors, on a*

$$y, x_1, \ldots, x_p \sim \mathcal{MN}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & c^{\top} \\ c & C \end{bmatrix}\right), \tag{2.3.1}$$

où c est le vecteur des corrélations entre $\mathbf{y}$ et chaque $\mathbf{x}_j$ et C la matrice des corrélations entre les paires de variables explicatives.

Démonstration. On exprime les $1 + p$ variables aléatoires $y, x_1, \ldots, x_p$, comme une combinaison linéaire des variables latentes $Z_1, \ldots, Z_n$. La variable aléatoire x_1, qui prend des valeurs le long de la droite $\mathcal{L}\mathbf{x}_1$, peut donc être exprimée en fonction des variables latentes

$Z_1, \ldots, Z_n$, comme suit $x_1 = (\mathbf{x}_1^\top / \sqrt{n}) \cdot (Z_1, \ldots, Z_n)^\top$. La variable x_1 est alors une projection du vecteur de variables latentes $(Z_1, \ldots, Z_n)^\top$ sur la droite $\mathcal{L}\mathbf{x}_1$. La distribution marginale de x_1 satisfait alors

$$x_1 \sim \frac{\mathbf{x}_1^\top}{\sqrt{n}} \, \mathcal{MN}(0_n, I_{n\times n}) = \mathcal{N}\left(\mathbf{u}_1^\top \cdot 0_n, \mathbf{u}_1^\top I_{n\times n} \mathbf{u}_1\right) = \mathcal{N}(0, 1).$$

où 0_n désigne un vecteur de dimension n composé uniquement de 0. De ce fait, des conclusions similaires peuvent être tirées pour chacune des variables aléatoires $y, x_1, \ldots, x_p$, étant donné que les droites $\mathcal{L}\mathbf{y}, \mathcal{L}\mathbf{x}_1, \ldots, \mathcal{L}\mathbf{x}_p$ fixent la direction de ces variables aléatoires dans l'espace. Chacune de ces $1+p$ variables aléatoires est donc marginalement distribuée selon une normale standard, le long de la droite associée à la variable aléatoire en question.

En utilisant le modèle latent, il devient alors facile d'exprimer la distribution conjointe des $1 + p$ variables

$$y = \left(\frac{\mathbf{y}^\top}{\sqrt{n}}\right) \cdot (Z_1, \ldots, Z_n)^\top, x_1 = \left(\frac{\mathbf{x}_1^\top}{\sqrt{n}}\right) \cdot (Z_1, \ldots, Z_n)^\top, \ldots, x_p = \left(\frac{\mathbf{x}_p^\top}{\sqrt{n}}\right) \cdot (Z_1, \ldots, Z_n)^\top.$$

Par les propriétés de la loi normale, cette distribution conjointe satisfait

$$\begin{aligned}
\begin{bmatrix} n^{-1/2}\mathbf{y}^\top \\ n^{-1/2}\mathbf{x}_1^\top \\ \vdots \\ n^{-1/2}\mathbf{x}_p^\top \end{bmatrix} \mathcal{MN}(0_n, I_{n\times n}) &= \mathcal{MN}\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, n^{-1} \begin{bmatrix} \mathbf{y}^\top \\ \mathbf{x}_1^\top \\ \vdots \\ \mathbf{x}_p^\top \end{bmatrix} I_{n\times n} \begin{bmatrix} \mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p \end{bmatrix}\right) \\
&= \mathcal{MN}\left(\begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & c_1 & \ldots & c_p \\ c_1 & C_{11} & \ldots & C_{1p} \\ \vdots & \vdots & \ddots & \vdots \\ c_p & C_{p1} & \ldots & C_{pp} \end{bmatrix}\right) \\
&= \mathcal{MN}\left(\begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & c^\top \\ c & C \end{bmatrix}\right).
\end{aligned}$$

Ainsi, on a bien

$$y, x_1, \ldots, x_p \sim \mathcal{MN}\left(0_{1+p}, \tilde{C}\right).$$

□

Nous avons supposé l'existence de variables latentes $Z_1, \ldots, Z_n$ suivant une distribution multivariée $\mathcal{MN}(0_n, I_{n\times n})$. En fixant la direction des vecteurs unitaires $\mathbf{u}_\mathbf{y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$ associés aux données dans l'espace et en conditionnant implicitement sur ces directions, nous avons exprimé chacune de nos $1 + p$ variables comme une combinaison linéaire des variables latentes. En raison des corrélations entre les vecteurs, il était évident que la variance de

la distribution conjointe de $y, x_1, \ldots, x_p$ deviendrait $\tilde{C}$. Nous pouvons imaginer que ces variables se situent sur les droites $\mathcal{L}\mathbf{y}, \mathcal{L}\mathbf{x}_1, \ldots, \mathcal{L}\mathbf{x}_p$ se confondant avec les vecteurs unitaires $\mathbf{u}_\mathbf{y}, \mathbf{u}_1, \ldots, \mathbf{u}_p$. Ces $1+p$ droites passant par l'origine sont fixées et corrélées entre elles, ce qui forme une espèce de système de coordonnées distortionné pour nos $1+p$ variables $y, x_1, \ldots, x_p$. Ce modèle de position jouera un grand rôle dans l'inférence du Lasso Linéaire, ce qui nous permettra de comprendre comment prédire la variable d'intérêt.

2.4. Inférence du modèle

La structure du modèle étant bien établie, nous souhaitons maintenant effectuer une inférence sur la variable réponse y. Afin d'obtenir une bonne approximation de cette variable, nous utiliserons l'information contenue dans les variables explicatives à propos de y.

Proposition 2.4.1. *Soit les deux variables aléatoires x et y avec une distribution conjointe normale de la forme*

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim \mathcal{MN}\left(\begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}, \begin{bmatrix} \sigma_{x,x} & \sigma_{x,y} \\ \sigma_{y,x} & \sigma_{y,y} \end{bmatrix} \right),$$

où $\sigma_{x,y} = \sigma_{y,x}$ est le terme de covariance. En conditionnant par rapport à la variable x, on obtient la distribution suivante pour la variable y, soit

$$y|x \sim \mathcal{N}(\sigma_{y,x}\sigma_{x.x}^{-1}x \ , \ \sigma_{y,y} - \sigma_{y,x}\sigma_{x.x}^{-1}\sigma_{x,y}). \tag{2.4.1}$$

Démonstration. Voir l'annexe A.3.

□

En appliquant le résultat (2.4.1) à la distribution conjointe (2.3.1), nous obtenons,

$$y|x_1, \ldots, x_p \sim \mathcal{N}\left(c^\top C^{-1} \begin{bmatrix} x_1 \\ \vdots \\ x_p \end{bmatrix}, 1 - c^\top C^{-1} c \right). \tag{2.4.2}$$

Nous pouvons imaginer que cette distribution est située le long de la droite $\mathcal{L}\mathbf{y}$. L'équation (2.4.2) représente la distribution de la variable réponse y étant donné l'information apportée par les des variables explicatives $x_1, \ldots, x_p$. La moyenne de la distribution conditionnelle en (2.4.2) peut alors agir à titre de prédiction pour y; cette prédiction est alors donnée par le point $c^\top C^{-1}(x_1, \ldots, x_p)^\top$ sur la droite $\mathcal{L}\mathbf{y}$. Cette valeur prédite sera appelée la teneur en y (qui est contenue dans les prédicteurs).

Remarque 2.4.2. *La valeur prédite le long de $\mathcal{L}\mathbf{y}$, ou la teneur en y, est la moyenne de la distribution conditionnelle explicitée en (2.4.2), soit*

$$c^\top C^{-1}(x_1, \ldots, x_p)^\top = (x_1, \ldots, x_p)C^{-1}c,$$

ce qui est algébriquement équivalent à la prédiction du modèle MCO, puisque

$$\hat{\boldsymbol{\beta}}^{(MCO)} = C^{-1}c = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y},$$

où $\mathbf{X}$ *est la matrice de design et* $\mathbf{y}$ *le vecteur réponse.*

La remarque 2.4.2 nous précise qu'il existe une équivalence algébrique entre les prévisions du Lasso Linéaire et celles des moindres carrés ordinaires.

La variance de la distribution conditionnelle, $1 - c^\top C^{-1}c$, représente la variabilité autour de la prévision qui est non expliquée par le modèle linéaire. Pour pouvoir prédire y de la façon la plus précise possible, il faut maintenant comprendre quelles sont la moyenne et la variance de la teneur en y.

Proposition 2.4.3. *La teneur en* y, $c^\top C^{-1}(x_1, \ldots, x_p)^\top$, *possède une distribution le long de la droite* $\mathcal{L}\mathbf{y}$ *satisfaisant*

$$c^\top C^{-1}(x_1, \ldots, x_p)^\top \sim \mathcal{N}(0, c^\top C^{-1}c). \tag{2.4.3}$$

Démonstration. La teneur en y est une fonction linéaire d'une distribution normale avec moyenne 0 et matrice de covariance C. L'espérance de la teneur en y satisfait donc

$$\mathbb{E}\left(c^\top C^{-1}(x_1, \ldots, x_p)^\top\right) = c^\top C^{-1}\mathbb{E}\left((x_1, \ldots, x_p)^\top\right) = 0,$$

et sa variance se développe comme suit

$$\begin{aligned}\mathbb{V}\left(c^\top C^{-1}(x_1, \ldots, x_p)^\top\right) &= c^\top C^{-1}\mathbb{V}\left((x_1, \ldots, x_p)^\top\right) C^{-1}c \\ &= c^\top C^{-1}CC^{-1}c \\ &= c^\top C^{-1}c.\end{aligned}$$

Alors, la distribution de la teneur en y satisfait

$$c^\top C^{-1}(x_1, \ldots, x_p)^\top \sim \mathcal{N}(0, c^\top C^{-1}c).$$

□

Le terme $c^\top C^{-1}c$ représente la variabilité de la prédiction en utilisant le modèle complet avec toutes les variables explicatives $x_1, \ldots, x_p$. Ainsi, la teneur en y est distribuée le long de la droite $\mathcal{L}\mathbf{y}$ et satisfait

$$c^\top C^{-1}(x_1, \ldots, x_p)^\top \sim \mathcal{N}(0, c^\top C^{-1}c) = \left\{c^\top C^{-1}c\right\}^{-1/2}\mathcal{N}(0, 1). \tag{2.4.4}$$

Selon cette expression, la teneur en y serait alors une fraction de la distribution marginale de y. Naturellement, il suffit que cette fraction se rapproche le plus possible de 1 pour que la teneur en y se rapproche le plus possible de la distribution marginale de y, soit une $\mathcal{N}(0, 1)$.

La distribution de la teneur en y en (2.4.4) représente donc la portion de la distribution marginale de y que l'on peut reproduire le long de $\mathcal{L}\mathbf{y}$ à l'aide de l'information contenue dans l'ensemble de variables explicatives (modèle complet). Le but maintenant serait d'obtenir la meilleure approximation possible de cette distribution, tout en utilisant le moins grand nombre de variables explicatives possible, puisqu'on veut évidemment que notre modèle de régression linéaire soit le plus parcimonieux possible. Dans la prochaine section, nous verrons qu'en choisissant un sous-ensemble de variables explicatives de façon judicieuse, nous pouvons maximiser la quantité $\left\{c^\top C^{-1} c\right\}^{-1/2}$ calculée à l'aide des variables sélectionnées et nous rapprocher le plus possible d'une $\mathcal{N}(0,1)$ le long de $\mathcal{L}\mathbf{y}$.

2.5. Rôle des variables explicatives

La distribution de la teneur en y, détaillée à l'équation (2.4.4), fait intervenir toutes les variables explicatives $x_1, \ldots, x_p$. Cette distribution peut être vue comme une fraction de la distribution marginale de y, c'est-à-dire la portion de cette loi marginale pouvant être recouvrée à l'aide des variables explicatives contenues dans le modèle linéaire. Toutefois, on sait qu'en régression linéaire, nous n'avons pas nécessairement besoin de toutes les variables explicatives pour expliquer adéquatement la variable d'intérêt y. De façon similaire au Lasso, le Lasso Linéaire cherchera un sous-modèle comportant s variables explicatives, où $s < p$. Ainsi, nous souhaitons que le sous-ensemble J_s de variables explicatives $x_{j_1}, \ldots, x_{j_s}$ nous offre le meilleur modèle possible, de sorte à ce que la distribution de la teneur en y se rapproche le plus possible d'une $\mathcal{N}(0,1)$. Pour un sous-ensemble J_s de variables explicatives, où $s = \{j_1, \ldots, j_s\}$, la teneur en y sera distribuée sur la droite $\mathcal{L}\mathbf{y}$ selon l'équation

$$c_s^\top C_s^{-1} (x_{j_1}, \ldots, x_{j_s})^\top \sim \left\{c_s^\top C_s^{-1} c_s\right\}^{-1/2} \mathcal{N}(0,1). \tag{2.5.1}$$

Maintenant, dans le but de sélectionner les variables les plus appropriées pour le modèle de régression linéaire, nous introduisons un paramètre $\delta = (\delta_1, \ldots, \delta_p)$ qui contrôle la présence ($\delta = 1$) ou l'absence ($\delta = 0$) de chacune des p variables explicatives. De ce fait, la distribution de la valeur qui est prédite sur la droite $\mathcal{L}\mathbf{y}$ à l'aide de l'ensemble de variables explicatives J_s satisfait

$$c_\delta^\top C_\delta^{-1} \mathbf{x}_\delta \sim \mathcal{N}(0, c_\delta^\top C_\delta^{-1} c_\delta), \tag{2.5.2}$$

où $\mathbf{x}_\delta = (x_{1\delta_1}, \ldots, x_{p\delta_p})^\top$ et où les composantes ayant un indice de 0 selon cette notation sont écartées du vecteur. Un raisonnement similaire est appliqué aux notations c_δ et C_δ, menant à un vecteur et une matrice de dimensions $\sum_j \delta_j$ et $(\sum_j \delta_j \times \sum_j \delta_j)$, respectivement. La distribution des valeurs prédites à l'aide de l'ensemble δ intègre à 1. Plus le terme de variance de cette distribution est élevé (c'est-à-dire près de 1), plus l'ensemble δ contient de l'information à propos de la variable réponse y.

L'équation (2.5.2) peut être réécrite sous la même forme que l'équation (2.5.1), c'est-à-dire

$$c_\delta^\top C_\delta^{-1}\mathbf{x}_\delta \sim \left\{c_\delta^\top C_\delta^{-1}c_\delta\right\}^{-1/2}\mathcal{N}(0,1) = \sigma(\delta)\mathcal{N}(0,1), \tag{2.5.3}$$

où $\sigma(\delta)$ correspond à une fraction de la distribution marginale de la variable d'intérêt y. On peut alors visualiser le problème de sélection de variables comme un problème d'inférence dans lequel le paramètre d'intérêt est y et le paramètre de nuisance est δ, menant au modèle statistique

$$f(y;\delta) = \sigma(\delta)\phi(y), \tag{2.5.4}$$

où $\phi(\cdot)$ est la densité d'une normale standard de moyenne 0 et de variance unitaire. Notons que la densité $f(y;\delta)$ n'intègre typiquement pas à 1, puisqu'elle représente la fraction de la distribution marginale de y contenue dans l'ensemble de prédicteurs δ. La fonction de vraisemblance est alors donnée par

$$L(y,\delta) = \sigma(\delta)\phi(y),$$

ce qui correspond simplement à une fraction de la vraisemblance marginale de y. L'objectif devient donc de maximiser $\sigma(\delta)$ par rapport au paramètre δ, de manière à éliminer le paramètre de nuisance, tout en ayant simultanément accès à la plus grande fraction possible de la vraisemblance marginale pour y. En d'autres mots, il faut choisir les variables explicatives qui maximisent la fraction $\sigma(\delta) = \left\{c_\delta^\top C_\delta^{-1}c_\delta\right\}^{-1/2}$, ce qui maximisera la vraisemblance. Pour ce faire, nous ajouterons éventuellement un paramètre de position γ dans la fonction de vraisemblance. Ce paramètre sera supposé fixe et il jouera la même rôle que le paramètre λ dans le Lasso régulier.

2.6. Réduction substantielle des variables explicatives

Dans cette section, nous souhaitons proposer une approche de sélection de variables basée sur le modèle statistique $\sigma(\delta)\phi(y)$. Notre objectif est de supprimer les prédicteurs qui ne contribuent pas beaucoup au terme de teneur en y, en maximisant la fonction de vraisemblance $L(y,\delta)$. Nous allons aussi établir le lien entre le Lasso régulier et le Lasso Linéaire.

2.6.1. Cas indépendant

Pour faciliter la compréhension, nous commençons par analyser le cas où les variables explicatives sont indépendantes, c'est-à-dire $C = I_p$. La standardisation de signe nous permet de placer tous les vecteurs dans la partie supérieure de l'espace $\mathcal{L}^+\mathbf{y}$. Ainsi, dans ce contexte d'indépendance, les coefficients de régression β_j sont tous plus grands que 0. Nous pouvons

alors supposer que la fonction à minimiser pour le Lasso régulier est

$$\frac{1}{2n}(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})+\lambda\sum_{j=1}^{p}\beta_j,$$

où la valeur absolue ne fait plus partie de la fonction objective. Sachant que les variables explicatives sont indépendantes, chaque nouvelle variable explicative ajoutée au modèle apporte une nouvelle information sur la variable y. Lorsque $\lambda = 0$, la fonction $(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})$ est minimisée à $\beta_j = c_j$, pour $j = 1,\ldots,p$. Le prédicteur avec le plus petit coefficient β_j est alors celui qui apporte le moins d'information sur y. Lorsque λ augmente, le Lasso régulier force les plus petits β_j à 0, tel que discuté dans la section 1.3.2. En d'autres mots, la procédure écarte les variables qui sont les plus faiblement corrélées avec y.

Dans le cas indépendant, la distribution de la teneur en y devient $\mathcal{N}(0, c_\delta^\top c_\delta)$ le long de la droite $\mathcal{L}\mathbf{y}$. Les estimés des coefficients sont alors $\hat{\boldsymbol{\beta}}_\delta = c_\delta = (c_{1\delta_1},\ldots,c_{p\delta_p})$, où un indice nul indique que la variable j est exclue du vecteur. Sachant que $\hat{\boldsymbol{\beta}}_\delta = c_\delta$, la somme résiduelle des carrés devient

$$\begin{aligned}
(\mathbf{y}-\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta)^\top(\mathbf{y}-\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta) &= \mathbf{y}^\top\mathbf{y}-2\mathbf{y}^\top\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta+(\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta)^\top\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta\\
&= \mathbf{y}^\top\mathbf{y}-2(\boldsymbol{X}_\delta^\top\mathbf{y})^\top\hat{\boldsymbol{\beta}}_\delta+\hat{\boldsymbol{\beta}}_\delta^\top(\boldsymbol{X}_\delta^\top\boldsymbol{X}_\delta)\hat{\boldsymbol{\beta}}_\delta\\
&= n-2nc_\delta^\top c_\delta+nc_\delta^\top I_{(\sum\delta_j\times\sum\delta_j)}c_\delta\\
&= n-2nc_\delta^\top c_\delta+nc_\delta^\top c_\delta\\
&= n(1-c_\delta^\top c_\delta),
\end{aligned}$$

où $\boldsymbol{X}_\delta = (\mathbf{x}_{1\delta_1},\ldots,\mathbf{x}_{p\delta_p})$ est la matrice $\boldsymbol{X}$ dans laquelle les colonnes comportant un indice nul ont été retirées. Nous remarquons que le fait de minimiser la somme des carrés résiduelle par rapport au paramètre δ est équivalent à maximiser la variance $\sigma^2(\delta) = c_\delta^\top c_\delta$ par rapport à δ. De ce fait, plutôt que de minimiser la fonction

$$\frac{1}{2n}(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})+\lambda\sum_{j=1}^{p}\beta_j,$$

on peut maximiser la fonction

$$\sigma^2(\delta)-\gamma\|\hat{\beta}_\delta\|_1 = c_\delta^\top c_\delta-\gamma\sum_{j=1}^{p}c_{j\delta_j}. \tag{2.6.1}$$

Lorsque $\gamma = 0$, l'équation (2.6.1) atteint son maximum quand tous les prédicteurs sont inclus dans le modèle, soit $c^\top c$. Par contre, à mesure que γ augmente, les prédicteurs sont graduellement retirés du modèle, en commençant par le prédicteur le plus faiblement corrélé avec la variable réponse. Supposons que nous ordonnions les corrélations de sorte à avoir

$c_1 > c_2 > \ldots > c_p$; alors, le $(p-k+1)$ prédicteur x_k sera retiré lorsque

$$\begin{aligned}
\sum_{j=1}^{k-1} c_j^2 - \gamma \sum_{j=1}^{k-1} c_j &> \sum_{j=1}^{k} c_j^2 - \gamma \sum_{j=1}^{k} c_j \\
\sum_{j=1}^{k-1} c_j^2 - \gamma \sum_{j=1}^{k-1} c_j &> \sum_{j=1}^{k-1} c_j^2 + c_k^2 - \gamma \sum_{j=1}^{k} c_j - \gamma c_k \\
\gamma c_k &> c_k^2 \\
\gamma &> c_k.
\end{aligned}$$

On remarque que si γ est plus grand que c_k, le prédicteur x_k est retiré et le coefficient $\hat{\beta}_k$ prend une nouvelle valeur, soit $\hat{\beta}_k = 0$. De ce fait, tel qu'observé dans le Lasso régulier, nous partons du modèle complet avec p prédicteurs et nous retirons séquentiellement les prédicteurs x_j qui sont les moins corrélés avec y.

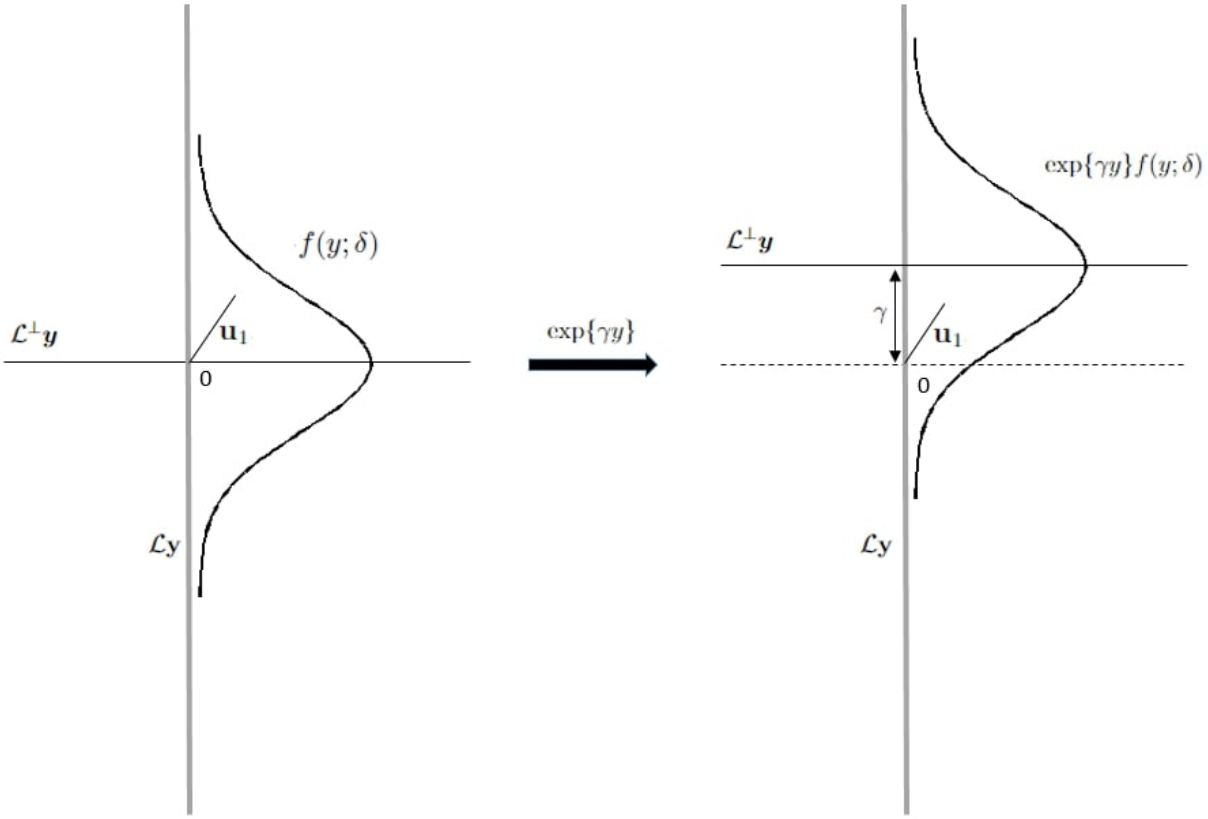


Figure 2.2. Représentation visuelle de la quantité $\exp\{\gamma y\}$ agissant sur la densité $f(y;\delta)$, ce qui crée une translation de l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ vers le haut (déplacement de γ). Lorsque l'hyperplan est au-dessus d'un vecteur unitaire, par exemple $\mathbf{u}_1$, la variable x_1 est retirée du modèle.

Pour représenter visuellement le processus de sélection de variables, nous supposons que le modèle statistique $f(y;\delta) = \sigma(\delta)\phi(y) = \left\{c_\delta^\top c_\delta\right\}^{-1/2}\phi(y)$ est couché sur la droite $\mathcal{L}\mathbf{y}$. Ceci

correspond au schéma gauche de la figure 2.2. Nous multiplions par la suite $f(y;\delta)$ par la quantité $\exp\{\gamma y\}$, de sorte à obtenir une translation de notre courbe vers le haut (de magnitude γ). De façon spécifique,

$$\begin{aligned}\exp\{\gamma y\} f(y;\delta) &= \sigma(\delta)\phi(y)\exp\{\gamma y\} \\ &\propto \sigma(\delta)\exp\{-y^2/2\}\exp\{\gamma y\} \\ &\propto \sigma(\delta)\exp\{-(y-\gamma)^2/2\},\end{aligned}$$

où la quantité $\exp\{\gamma y\}$ génère un déplacement de la distribution dans la direction positive le long de la droite $\mathcal{L}\mathbf{y}$. La fonction de densité normale est alors centrée en γ plutôt qu'en 0 et γ joue le rôle d'un paramètre de position dans le modèle statistique. On remarque que lorsque γ est suffisamment grand, l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ est translaté au-dessus du vecteur $\mathbf{u}_1$. Nous nous retrouvons ainsi dans la situation où $\gamma > \hat{\beta}_1 = c_1$; il faut alors exclure la variable x_1. Ceci peut être visualisé dans le schéma de droite de la figure 2.2.

Lorsque l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ subit un déplacement de γ vers le haut, la magnitude de ce déplacement agit de la même manière que le paramètre λ dans le Lasso régulier; elle élimine les prédicteurs les moins corrélés avec $\mathbf{y}$. Ceci est une étape cruciale dans la méthode de sélection de variables selon la méthode du Lasso Linéaire. Par contre, dans cette section, nous n'avons pas encore tenu compte de la corrélation qu'il pourrait y avoir entre les prédicteurs. C'est pourquoi, dans la prochaine section, nous introduisons une étape additionnelle dans la méthode du Lasso Linéaire.

2.6.2. Prédicteurs corrélés

Contrairement à la section précédente, nous allons maintenant supposer que les prédicteurs sont corrélés, impliquant que la matrice des corrélations C n'est plus diagonale. Nous rappelons que le vecteur des coefficients $\hat{\boldsymbol{\beta}} = C^{-1}c$ est algébriquement équivalent à celui des moindres carrés ordinaires. La standardisation de signe nous garantit que les corrélations entre $\mathbf{y}$ et $\mathbf{x}_j$ sont supérieures ou égales à 0 pour $j = 1,\ldots,p$, mais ceci ne garantit pas que les coefficients soient positifs. Il est certainement possible que certains coefficients soient légèrement négatifs. Cette situation peut se présenter lorsque deux ou plusieurs prédicteurs présentent une certaine redondance dans leur teneur en y. Il est alors possible que certains coefficients associés à ces prédicteurs soient positifs et que d'autres soient légèrement négatifs, de sorte à éviter de dédoubler l'information expliquant y.

La fonction objective à minimiser pour le Lasso régulier est celle explicitée en (1.3.4). Lorsque λ augmente, certains coefficients de régression sont retirés du modèle. En raison de la corrélation entre les prédicteurs, la variable associée au plus petit coefficient de régression n'est pas nécessairement rejetée en premier, contrairement à ce qui a été observé dans la section 2.6.1. Il est donc difficile de savoir exactement dans quel ordre les prédicteurs sont

éliminés. C'est pourquoi le Lasso régulier utilise un algorithme tel que la descente du gradient pour minimiser la fonction objective. En pratique, il est préférable d'utiliser un algorithme de sous-gradient ou l'algorithme du LARS.

Dans le cas du Lasso Linéaire, nous avons établi que la prévision le long de la droite $\mathcal{L}\mathbf{y}$, pour un sous-ensemble de prédicteurs δ, est de $c_\delta^\top C_\delta^{-1}\mathbf{x}_\delta$. Il suffit alors de comprendre comment choisir le paramètre δ pour pouvoir faire de la sélection de variables. Sachant que $\hat{\boldsymbol{\beta}}_\delta = C_\delta^{-1}c_\delta$, on minimise la somme des carrés résiduelle $(\mathbf{y}-\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta)^\top(\mathbf{y}-\boldsymbol{X}_\delta\hat{\boldsymbol{\beta}}_\delta)$ par rapport à δ. Nous obtenons

$$\begin{aligned}(\mathbf{y}-\boldsymbol{X}_\delta C_\delta^{-1}c_\delta)^\top(\mathbf{y}-\boldsymbol{X}_\delta C_\delta^{-1}c_\delta) &= \mathbf{y}^\top\mathbf{y} - 2\mathbf{y}^\top\boldsymbol{X}_\delta C_\delta^{-1}c_\delta + (\boldsymbol{X}_\delta C_\delta^{-1}c_\delta)^\top\boldsymbol{X}_\delta C_\delta^{-1}c_\delta \\ &= \mathbf{y}^\top\mathbf{y} - 2(\boldsymbol{X}_\delta^\top\mathbf{y})^\top C_\delta^{-1}c_\delta + c_\delta^\top C_\delta^{-1}(\boldsymbol{X}_\delta^\top\boldsymbol{X}_\delta)C_\delta^{-1}c_\delta \\ &= n - 2nc_\delta^\top C_\delta^{-1}c_\delta + nc_\delta^\top C_\delta^{-1}C_\delta C_\delta^{-1}c_\delta \\ &= n - 2nc_\delta^\top C_\delta^{-1}c_\delta + nc_\delta^\top C_\delta^{-1}c_\delta \\ &= n(1 - c_\delta^\top C_\delta^{-1}c_\delta). \end{aligned} \tag{2.6.2}$$

Tel que discuté dans la section 2.6.1, pour minimiser la fonction objective (2.6.2), il faut maximiser la quantité $c_\delta^\top C_\delta^{-1}c_\delta$ par rapport au paramètre δ. Ceci est cohérent avec le fait que $c_\delta^\top C_\delta^{-1}\mathbf{x}_\delta \sim \mathcal{N}(0, c_\delta^\top C_\delta^{-1}c_\delta)$ et qu'il faille choisir les variables explicatives maximisant la fraction $\left\{c_\delta^\top C_\delta^{-1}c_\delta\right\}^{-1/2}$, de sorte à se rapprocher le plus possible d'une $\mathcal{N}(0,1)$. C'est pourquoi il faut trouver le modèle comportant la plus grande variance $c_\delta^\top C_\delta^{-1}c_\delta$ assujettie à la pénalité $\gamma\sum_{j=1}^p|\hat{\beta}_{j\delta_j}| = \gamma\sum|C_\delta^{-1}c_\delta|$. Sachant que les estimés $C_\delta^{-1}c_\delta$ ne dépendent que de δ, il s'agit de trouver un compromis entre la maximisation de la variance $c_\delta^\top C_\delta^{-1}c_\delta$ et la minimisation de la cardinalité $\sum_{j=1}^p\delta_j$.

Nous procédons de la même manière que dans le cas indépendant, c'est-à-dire que nous perturbons la densité $f(y,\delta)$ par un facteur de $\exp\{\gamma y\}$. Nous obtenons l'équation

$$f(y,\delta)\exp\{\gamma y\} \propto \left\{c_\delta^\top C_\delta^{-1}c_\delta\right\}^{-1/2}\exp\{-(y-\gamma)^2/2\},$$

le long de la droite $\mathcal{L}\mathbf{y}$. Lorsque γ augmente, la distribution glisse le long de la droite $\mathcal{L}\mathbf{y}$ comme auparavant. Ceci est donc équivalent à retirer le k^e prédicteur du modèle lorsque $\gamma > \beta_k$; toutefois, retirer les prédicteurs en fonction de la taille des coefficients $(C_\delta^{-1}c_\delta)$ n'est pas évident en pratique, puisqu'il faudrait inverser la matrice C_δ à chaque étape.

Rappelons-nous qu'un coefficient β_j peut être interprété comme étant la projection, sur la droite $\mathcal{L}\mathbf{y}$, d'une portion du vecteur unitaire $\mathbf{u}_j$ sur $\mathcal{L}\mathbf{x}_j$ (disons $\tilde{\mathbf{u}}_j$). Cette portion de vecteur $\tilde{\mathbf{u}}_j$ étant elle-même difficile à identifier, nous pallions à ce problème en considérant le prochain ordre d'approximation, c'est-à-dire que nous écartons les coefficients selon la pente de $\tilde{\mathbf{u}}_j$ sur $\mathcal{L}\mathbf{x}_j$, associé au coefficient β_j sur la droite $\mathcal{L}\mathbf{y}$. De ce fait, plus $\mathcal{L}\mathbf{y}$ et $\mathcal{L}\mathbf{x}_j$ sont près l'un de l'autre, plus la projection β_j sur $\mathcal{L}\mathbf{y}$ sera grande et plus il deviendra difficile d'écarter

le prédicteur x_j. Le processus de sélection des variables est donc similaire à celui qui a été proposé dans le contexte d'indépendance. Ainsi, en nous tournant vers les vecteurs unitaires, lorsque γ se retrouve au-dessus de $\mathbf{u}_j$, on retire la variable j du modèle. Ceci nous mène à une approche semblable à celle représentée dans la figure 2.2; le Lasso Linéaire s'avère donc être une approche unidimensionnelle (sans itération), puisque tout se passe sur le long de la droite $\mathcal{L}\mathbf{y}$.

2.7. Procédure

Nous avons maintenant compris que pour effectuer la sélection des variables dans notre modèle de régression linaire, il s'agit de trouver un compromis entre la maximisation de la variance $c_\delta^\top C_\delta^{-1} c_\delta$ et la minimisation de la cardinalité $\sum_{j=1}^p \delta_j$. Pour arriver à nos fins, le Lasso Linéaire s'effectuera en deux étapes. La première étape consistera à retirer un grand nombre de variables explicatives du modèle, successivement ou simultanément tout dépendant des contextes, en nous basant uniquement sur le vecteur des corrélations c. Afin d'ajuster plus finement notre modèle, la deuxième étape consistera à retirer successivement, parmi les variables restantes, celle menant à la plus petite diminution de la variance $c_\delta^\top C_\delta^{-1} c_\delta$. En retirant ainsi les variables une à une, nous serons en mesure de proposer le sous-modèle optimal en utilisant la validation croisée, tel que discuté dans la section 1.3.3.

2.7.1. Première étape du Lasso Linéaire

Dans l'étape initiale du Lasso Linéaire, les corrélations c_j sont calculées pour le modèle complet, nous permettant ainsi de retirer les m variables explicatives ayant les plus petites corrélations. Ces m variables ne seront pas introduites dans la procédure un-à-un afin de minimiser le temps de calcul. Nous nous retrouvons alors avec un sous-modèle comportant $p - m$ prédicteurs, où

$$m = \{\#j \in \{1, \dots, p\} : c_j < \gamma\} \tag{2.7.1}$$

et où γ est le paramètre d'ajustement spécifié par l'utilisateur.

On remarque ainsi que le Lasso Linéaire est une méthode qui gère bien un contexte de grande dimension ($p > n$). Il suffit alors d'éliminer au moins $p - n$ variables explicatives, en nous basant sur la taille des corrélations c_j. De ce fait, on se retrouve maintenant avec une méthode qui performe bien dans un contexte de petite ou grande dimension. Ceci permet alors de régler un des problèmes du Lasso régulier tel que discuté dans la section 2.2. De plus, l'une des forces de la procédure basée sur les c_j est que l'ordonnancement des modèles de différentes dimensions ne nécessite pas d'itérations. Notons que l'ordonnancement des sous-modèles ne serait pas nécessairement le même si on maximisait directement la variance $c_\delta^\top C_\delta^{-1} c_\delta$ pour une cardinalité donnée, mais cette approche nous permet de sauver un temps de computation énorme. Par contre, celle-ci ne tient pas compte du fait que deux prédicteurs

corrélés pourraient comporter une certaine redondance dans leur teneur en y. Pour pallier à ce problème, nous nous tournons vers la deuxième étape du Lasso Linéaire.

2.7.2. Deuxième étape du Lasso Linéaire

L'étape précédente nous permet d'obtenir un modèle plus ou moins compact en nous basant uniquement sur la valeur des c_j. Ainsi, une fois le modèle réduit à $p-m$ variables, il est possible de le raffiner en utilisant la procédure un-à-un explicitée dans l'algorithme 2.7.1.

Algorithme 2.7.1. Procédure un-à-un

Entrée : Le vecteur réponse $\mathbf{y}$, les p prédicteurs, la valeur m et la liste $\mathcal{M}$ des indices des m variables comportant les plus petites corrélations avec $\mathbf{y}$.

1: Calculer le coefficient $\hat{\boldsymbol{\beta}}_{(j)}$ de chaque modèle, de la taille p à la taille $p-m$, en partant du modèle complet et en écartant à chaque fois le prédicteur ayant la plus petite corrélation avec $\mathbf{y}$ dans $\mathcal{M}$. Ajouter chaque coefficient dans la liste

$$\mathcal{B} = \{1 \leq j \leq m : \hat{\boldsymbol{\beta}}_{(j)} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \mathbf{y}\}.$$

Pour $k = 0, \ldots, p-m-1$:

2: Initialiser $\delta^{(k)}$ de sorte à ce que $\delta_j = 0$ pour $j \in \mathcal{M}$, et $\delta_j = 1$ pour $j \notin \mathcal{M}, j = 1, \ldots, p$. La cardinalité de $\delta^{(k)}$ est alors $\sum \delta_j = p - m - k$.

3: Calculer la variance $\sigma^2(\delta^{(k)}) = c_{\delta^{(k)}}^\top C_{\delta^{(k)}}^{-1} c_{\delta^{(k)}}$ pour le modèle avec les $p-m-k$ prédicteurs restants.

4: Calculer

$$\min \left\{ \sigma^2(\delta^{(k)}) - \sigma^2(\delta_*^{(1)}), \ldots, \sigma^2(\delta^{(k)}) - \sigma^2(\delta_*^{(p-m-k)}) \right\},$$

où $\delta_*^{(1)}, \ldots, \delta_*^{(p-m-k)}$ sont les modèles de cardinalité $p-m-k-1$, dans lesquels on enlève un seul prédicteur à la fois. Noter le prédicteur $\mathbf{x}_j$ qui mène à la plus petite réduction de variance.

4.1: Calculer le coefficient $\hat{\boldsymbol{\beta}}_{(m+k+1)}$ sans la variable $\mathbf{x}_j$ identifiée à l'étape précédente et l'ajouter dans la liste $\mathcal{B}$.

4.2: Retirer $\mathbf{x}_j$ de la matrice de design $\boldsymbol{X}$ et ajouter l'indice j dans $\mathcal{M}$.

5: Poser $k = k + 1$ et retourner à l'étape 2.

Sortie : La liste des coefficients $\mathcal{B} = \left\{ \hat{\boldsymbol{\beta}}_{(1)}, \hat{\boldsymbol{\beta}}_{(2)}, \ldots, \hat{\boldsymbol{\beta}}_{(p)} \right\}$.

La liste $\mathcal{B}$ que nous retourne l'algorithme 2.7.1 contient les vecteurs de coefficients MCO pour les sous-modèles de différentes tailles. Nous passons alors de la dimension p pour $\hat{\boldsymbol{\beta}}_{(1)}$ à la dimension 1 pour $\hat{\boldsymbol{\beta}}_{(p)}$. Afin d'identifier le meilleur modèle parmi ces p sous-modèles, nous effectuons une validation croisée, tel que discuté dans la section 1.3.3. Au lieu de choisir le λ optimal parmi certains candidats $\lambda \in \{\lambda_1, \ldots, \lambda_p\}$ comme dans le Lasso régulier, la

procédure de validation croisée du Lasso Linéaire calculera l'erreur quadratique moyenne de prédiction associée aux différentes tailles $\mathcal{T} \in \{1, \ldots, p\}$ des modèles générés par le Lasso Linéaire. Nous sélectionnerons alors la taille de modèle $\mathcal{T}$ menant à la plus petite erreur de prédiction, soit

$$\mathcal{T}_{min} = \underset{\mathcal{T} \in \{1,\ldots,p\}}{\operatorname{argmin}} \, MSE(\mathcal{T}),$$

où $MSE(\mathcal{T})$ est l'erreur quadratique moyenne de prédiction obtenue par validation croisée et associée à une certaine taille de modèle. Plus de précisions sont fournies dans l'algorithme 2.7.2.

En résumé, dans la première étape, des variables sont écartées du modèle basé sur leur corrélation avec la variable réponse; le nombre de variables écartées (ou ordonnées) lors de cette étape dépend d'un paramètre d'ajustement γ. Dans la seconde étape, un critère d'exclusion basé sur la variance de la distribution de la variable réponse est introduit pour retirer (ou ordonner) les variables restantes. Maintenant que les deux étapes du Lasso Linéaire sont bien établies, nous présentons l'algorithme dans son intégralité.

Algorithme 2.7.2. Algorithme du Lasso Linéaire

Entrée : Le vecteur réponse $\mathbf{y} \in \mathbb{R}^n$, la matrice de design $\boldsymbol{X}_{n\times p} = (\mathbf{x}_1, \ldots, \mathbf{x}_p)$, le paramètre d'ajustement γ, ainsi que le nombre de plis K et le nombre de cycles L pour la validation croisée.

1: Centrer et standardiser (par variance) les vecteurs $\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p$.

Étape 1 : Retirer les m variables explicatives

2: Calculer le vecteur des corrélations c entre $\mathbf{y}$ et chaque $\mathbf{x}_j$ et la matrice des corrélations C entre les paires de variables explicatives.

3: Inverser le signe des vecteurs $\mathbf{x}_j$ pour lesquels $c_j < 0$.

4: Calculer m tel que $m = \{\#j \in \{1, \ldots, p\} : c_j < \gamma\}$.

5: Noter les m prédicteurs $\mathbf{x}_j$ pour lesquels $c_j < \gamma$, en ordre croissant des c_j.

Étape 2 : Procédure un-à-un avec la validation croisée K-plis

6: Initialiser deux matrices de dimension $L \times p$, $\mathbf{M}$ et $\mathbf{V}$, qui contiendront respectivement les erreurs quadratiques moyennes et les variances associées à ces erreurs pour chaque taille de modèle $\mathcal{T}$ et chacune des L répétitions de la validation croisée.

Pour $l = 1, \ldots, L$:

7: Partager le jeu de données en K groupes de tailles égales, de façon aléatoire et sans remise.

8: Pour $k = 1, \ldots, K$,

8.1: Désigner le groupe k comme ensemble test, dénoté $\mathbf{y}^*_{test}$ et $\boldsymbol{X}^*_{test}$, et désigner les $K-1$ autres groupes comme ensemble d'entraînement.

8.2: Exécuter la procédure un-à-un de l'algorithme 2.7.1 sur l'ensemble d'entraînement, ce qui nous retourne la liste $\mathcal{B}$.

8.3: Pour chaque taille de modèle $\mathcal{T} \in \{1, \ldots, p\}$, calculer l'erreur quadratique moyenne

$$MSE_k(\mathcal{T}) = |\mathbf{y}^*_{test}|^{-1} \left\| \mathbf{y}^*_{test} - \tilde{\boldsymbol{X}}^*_{test} \hat{\boldsymbol{\beta}}_{(p+1-\mathcal{T})} \right\|^2, \ \mathcal{T} = 1, \ldots, p,$$

où $|\mathbf{y}^*_{test}|^{-1}$ est la cardinalité du vecteur $\mathbf{y}^*_{test}$ et les variables explicatives de $\tilde{\boldsymbol{X}}^*_{test}$ concordent avec les coefficients $\hat{\boldsymbol{\beta}}_{(p+1-\mathcal{T})}$.

8.4: Poser $k = k + 1$ et retourner à l'étape 8.1, de sorte à ce que chacun des K groupes ait la chance jouer le rôle de l'ensemble test.

9: Calculer la moyenne et la variance des K estimations de l'erreur de prédiction pour chaque taille de modèle $\mathcal{T} = 1, \ldots, p$,

$$MSE(\mathcal{T}) = K^{-1} \sum_{k=1}^{K} MSE_k(\mathcal{T}) \ \text{ et } \ V(\mathcal{T}) = (K-1)^{-1} \sum_{l=1}^{K} \left(MSE_k(\mathcal{T}) - MSE(\mathcal{T}) \right)^2.$$

10: Enregistrer $MSE(\mathcal{T})$ et $V(\mathcal{T})$ dans la ligne l des matrices $\mathbf{M}$ et $\mathbf{V}$ respectivement.

11: Poser $l = l+1$ et retourner à l'étape 7 (c'est-à-dire utiliser une nouvelle partition aléatoire des données pour chacun des L cycles de la validation croisée).

Étape 3 : Trouver le $\mathcal{T}_{min}$ et ajuster le modèle

12: Calculer la moyenne de chacune des p colonnes des matrices $\mathbf{M}$ et $\mathbf{V}$, de sorte à obtenir

$$\mathbf{M}_{\mathcal{T}} = L^{-1}\sum_{l=1}^{L}\mathbf{M}_{l\mathcal{T}} \quad \text{et} \quad \mathbf{S}_{\mathcal{T}} = \sqrt{L^{-1}\sum_{l=1}^{L}\mathbf{V}_{l\mathcal{T}}},\ \mathcal{T} = 1,\ldots,p.$$

13: Identifier la taille du sous-modèle ayant l'erreur quadratique moyenne la plus petite,

$$\mathcal{T}_{min} = \underset{\mathcal{T}\in\{1,\ldots,p\}}{\operatorname{argmin}}\ \mathbf{M}_{\mathcal{T}},$$

où l'erreur $\mathbf{M}_1$ correspond au sous-modèle comportant une seule variable et l'erreur $\mathbf{M}_p$ correspond au sous-modèle comportant p variables.

14: Appliquer l'algorithme 2.7.1 sur le jeu de données complet et sélectionner le modèle de taille $\mathcal{T}_{min}$.

Résultat : Un tableau des moyennes $\mathbf{M}_{\mathcal{T}}$ et des écarts-types $\mathbf{S}_{\mathcal{T}}$, le vecteur c_j des corrélations en ordre décroissant, la taille $\mathcal{T}_{min}$ et la liste $\mathcal{B}$ des coefficients MCO obtenus en appliquant l'algorithme 2.7.1 sur le jeu de données complet.

Nous remarquons que l'implémentation de l'algorithme 2.7.2 est plutôt simple, ne nécessitant aucun algorithme complexe (contrairement à la descente du gradient par exemple). L'implémentation de cet algorithme requiert toutefois le choix d'un paramètre d'ajustement γ. Selon les explorations menées dans Fraser et Bédard (2022), il semblerait que le choix par défaut $\gamma = 0{,}2$ mène à des résultats intéressants et à une implémentation efficace de la méthode. Un exemple est présenté dans la prochaine section pour nous aider à comprendre l'algorithme en pratique.

2.7.3. Exemple : Données diabète

Pour illustrer le fonctionnement de l'algorithme 2.7.2, on étudie l'exemple du diabète introduit par Efron *et al.* (2004). Les données sont obtenues à partir de $n = 442$ patients diabétiques; la réponse est une mesure quantitative de la progression de la maladie, un an après le début du diabète. Il y a $p = 10$ variables explicatives, soit l'âge, le sexe, l'indice de masse corporelle (`BMI`), la pression artérielle moyenne (`BP`) et six mesures du sérum sanguin ($\mathtt{S}_1, \ldots, \mathtt{S}_6$). La variable `SEXE` est codée comme une variable binaire et est standardisée au même titre qu'une variable continue.

Nous exécutons l'algorithme 2.7.2 en spécifiant le vecteur réponse $\mathbf{y}$, la matrice de design $\boldsymbol{X}$, le paramètre d'ajustement par défaut $\gamma = 0{,}2$, le nombre de cycles de validation croisée $L = 50$, ainsi que le nombre de plis par cycle $K = 13$. Dans le tableau 2.1, nous remarquons que les variables `SEXE`, $\mathtt{S}_2$ et `AGE` sont écartées à la première de étape de l'algorithme, puisque

Tableau 2.1. Corrélations c_j entre $\mathbf{y}$ et $\mathbf{x}_1, \ldots, \mathbf{x}_{10}$, en ordre décroissant

BMI	S_5	BP	S_4	S_3	S_6	S_1	AGE	S_2	SEXE
0,586	0,566	0,441	0,430	0,395	0,382	0,212	0,188	0,174	0,043

leur corrélation c_j est inférieure à $\gamma = 0{,}2$. Nous notons que le MSE est tout de même calculé pour les modèles incluant 3 variables. Par la suite, les variables restantes dans le modèle sont soumises à la deuxième étape du Lasso Linéaire, soit la procédure un-à-un, pour déterminer l'ordre dans lequel elles sortiront du modèle.

Tableau 2.2. MSE et écarts-types obtenus à l'aide du Lasso Linéaire appliqué au jeu de données du diabète.

Modèles (basés sur les données complètes)	Taille du modèle ($\mathcal{T}$)	$\mathbf{S}_\mathcal{T}$	$\mathbf{V}_\mathcal{T}$
Première étape			
BMI + S_5 + BP + S_1 + S_4 + S_6 + S_3 + AGE + S_6 + SEXE	10	0,5046	0,1142
BMI + S_5 + BP + S_1 + S_4 + S_6 + S_3 + AGE + S_2	9	0,5202	0,1119
BMI + S_5 + BP + S_1 + S_4 + S_6 + S_3 + AGE	8	0,5187	0,1114
Deuxième étape			
BMI + S_5 + BP + S_1 + S_4 + S_6 + S_3	7	0,5173	0,1105
BMI + S_5 + BP + S_1 + S_4 + S_6	6	0,5186	0,1103
BMI + S_5 + BP + S_1 + S_4	5	0,5167	0,1104
BMI + S_5 + BP + S_1	4	0,5203	0,1124
BMI + S_5 + BP	3	0,5282	0,1122
BMI + S_5	2	0,5454	0,1188
BMI	1	0,6783	0,1461

Les deux premières colonnes du tableau 2.2 nous indiquent les sous-modèles considérés ainsi que la taille de ces sous-modèles. La procédure nous permet d'obtenir un ordonnancement de modèles : nous passons d'un modèle à 10 variables jusqu'à un modèle ne comportant qu'une seule variable. La première variable à sortir du modèle durant la procédure un-à-un est S_3, celle-ci menant à la plus faible réduction de la variance. La procédure se poursuit alors, écartant les variables restantes une par une, jusqu'à en arriver à la variable BMI. Dans les colonnes 3 et 4, nous avons respectivement les erreurs quadratiques moyennes et les écarts-types provenant de la procédure de validation croisée. L'erreur quadratique moyenne minimale est de 0,5046 et correspond au modèle de taille $\mathcal{T}_{min} = 10$. Le modèle de taille 10 est celui comportant toutes les variables explicatives.

Il est intéressant de noter que la variable SEXE est la moins corrélée avec la variable réponse, mais pourtant elle semble être importante pour le modèle. Avec une corrélation

inférieure à $\gamma = 0{,}2$, la variable est rejetée du modèle à la première étape. Pourtant, la validation croisée nous indique tout de même que la variable `SEXE` est pertinente pour expliquer la progression de la maladie. Ce phénomène peut être lié au fait que la variable `SEXE` interagit avec une autre variable, ce qui force la procédure à la conserver dans le modèle final, malgré son rejet à la première étape. C'est pour cette raison qu'il est important d'évaluer l'ensemble des modèles proposés via les deux étapes de l'algorithme. Le vecteur des coefficients $\hat{\boldsymbol{\beta}}^{(MCO)}$ pour le sous-modèle retenu par le Lasso Linéaire est présenté dans le tableau 2.3.

Tableau 2.3. Estimés de $\hat{\boldsymbol{\beta}}^{(MCO)}$ pour le sous-modèle retenu par le Lasso Linéaire

Prédicteurs	$\hat{\boldsymbol{\beta}}^{(MCO)}$
`AGE`	0,010
`SEX`	20,801
`BMI`	5,418
`BP`	0,962
S_1	1,636
S_2	-1,604
S_3	-3,536
S_4	-7,741
S_5	-3,598
S_6	0,081

Notons que dans le processus de validation croisée, les modèles sont ajustés en utilisant des sous-ensembles des données initiales. Ces sous-modèles peuvent donc varier d'un ensemble d'apprentissage à un autre, et en particulier peuvent différer des modèles présentés dans le tableau 2.2 (obtenus en utilisant l'ensemble des observations disponibles). L'approche de validation croisée permet néanmoins de comparer des modèles de différentes tailles et d'identifier la taille optimale du modèle final pour le jeu de données du diabète.

En régression linéaire, nous voulons comprendre quelles sont les variables qui sont pertinentes dans notre modèle, ce qui nous amène à expliquer le mieux possible la variable d'intérêt. Avec l'algorithme du Lasso Linéaire, le vecteur des corrélations c_j nous permet d'obtenir rapidement un aperçu des variables importantes dans le modèle. Dans le cas des données de diabète, on sait que la variable `BMI` est celle qui contient le plus d'information sur la progression du diabète. Par contre, les corrélations c_j ne tiennent pas compte des relations entre les paires de variables explicatives et c'est pourquoi la procédure un-à-un nous permet de dresser un sous-modèle intéressant pour expliquer le jeu de données en question.

2.8. Discussion

Le Lasso Linéaire puise son nom dans la façon géométrique d'interpréter la régression Lasso. Le vecteur $\mathbf{y}$, contenant les réponses, est vu comme le point focal alors que tous les autres vecteurs de variables explicatives gravitent autour de ce vecteur réponse. Les vecteurs sont standardisés par signe (pointent au-dessus de $\mathcal{L}^{\perp}\mathbf{y}$). Le modèle de position nous permet d'établir une structure pour les variables aléatoires $y, x_1, \ldots, x_p$, par le biais d'une distribution conjointe. Par la suite, nous définissons la teneur en y et développons sa distribution. En considérant la réponse y comme le paramètre d'intérêt et en incorporant un paramètre de nuisance δ indiquant quelles variables sont sélectionnées dans le modèle ainsi qu'un paramètre d'ajustement γ contraignant la taille du modèle, nous réalisons alors qu'il faut maximiser $c_\delta^\top C_\delta^{-1} c_\delta$ sujet à une contrainte sur la cardinalité du modèle afin de recouvrer la plus grande portion possible de la distribution marginale de y. Dans cette optique, il est possible de visualiser l'effet du paramètre d'ajustement γ en imaginant que l'hyperplan $\mathcal{L}^{\perp}\mathbf{y}$ glisse le long de $\mathcal{L}\mathbf{y}$, écartant les prédicteurs dont les vecteurs se retrouvent entièrement sous l'hyperplan.

Le Lasso Linéaire est alors un algorithme qui s'implémente en deux étapes : la première étape consiste à enlever m prédicteurs en utilisant le vecteur des corrélations c; lorsqu'il ne reste que quelques variables explicatives, la seconde étape effectue une recherche plus approfondie en écartant les variables explicatives menant à la plus faible baisse du terme de variance $c_\delta^\top C_\delta^{-1} c_\delta$. Le sous-modèle sélectionné est celui dont la taille correspond à la plus petite erreur quadratique moyenne par validation croisée, tel qu'observé dans le tableau 2.2. On se retrouve avec un algorithme qui est *dimension free*, dans lequel les calculs sont relativement simples. Contrairement au Lasso régulier, qui ne peut choisir qu'au maximum n variables lorsque $p > n$, le Lasso Linéaire peut être appliqué avec des données de grande dimension. Il faut cependant que γ soit suffisamment grand, de sorte à ce que la deuxième étape soit implémentée avec un nombre de prédicteurs inférieur à n; en effet, l'estimateur $\hat{\boldsymbol{\beta}}^{(MCO)}$ ne peut être calculé si la dimension d'un sous-modèle est supérieure à n. De plus, rappelons que le point focal de l'analyse tourne autour de la variable réponse. Ainsi, s'il existe une paire de variables explicatives $\mathbf{x}_1$ et $\mathbf{x}_2$ possédant une grande corrélation appariée, il est possible qu'une de ces variables soit éliminée dans la première étape du Lasso Linéaire. Ceci nous permet d'adresser, d'une certaine façon, le second problème du Lasso régulier : dans le cas de prédicteurs corrélés, celui-ci sélectionne une variable au hasard sans se soucier du choix de la variable.

Il serait intéressant d'étudier le Lasso Linéaire plus en profondeur, de sorte à éventuellement optimiser sa performance via le choix des paramètres d'ajustement. Le choix $\gamma = 0{,}2$ dans la première étape de l'algorithme semble bien fonctionner dans les exemples de Fraser et Bédard (2022); qu'en est-il plus généralement? Cette valeur par défaut est-elle robuste

ou devrait-elle être ajustée selon les différents contextes? Est-ce que la première étape du Lasso Linéaire pourrait être suffisante dans le choix d'un sous-modèle performant ? Ce sont des questions auxquelles nous tenterons de répondre dans le chapitre 3.

Chapitre 3

Optimisation de l'algorithme du Lasso Linéaire

L'algorithme du Lasso Linéaire introduit dans Fraser et Bédard (2022) s'effectue en deux étapes et nous permet de travailler avec des données de petites et grandes dimensions. La première étape consiste à réduire le nombre de prédicteurs à l'aide d'un paramètre d'ajustement γ spécifié par l'utilisateur. La deuxième étape nous permet, à l'aide de la validation croisée K-plis répétée L fois, d'identifier la taille de sous-modèle optimale, $\mathcal{T}_{min}$, qui est associée à la plus petite valeur de MSE.

L'algorithme du Lasso Linéaire peut être optimisé via des études par simulation. Il serait important d'étudier l'impact du choix de paramètre d'ajustement γ sur la performance de l'algorithme. Par ailleurs, nous remarquons que la première étape du Lasso Linéaire est très rapide à exécuter, puisque les calculs sont extrêmement simples; il serait alors intéressant de comparer la performance de la première étape à celle de la seconde, ainsi qu'à la performance des deux étapes combinées. Nous sommes donc confrontés à tester plusieurs variations de l'algorithme du Lasso Linéaire pour ainsi optimiser sa performance et sa puissance.

3.1. Les paramètres K et L pour la validation croisée

La validation croisée K-plis, répétée L-fois, est utilisée dans l'algorithme du Lasso Linéaire pour choisir la taille optimale du modèle. À première vue, le choix du nombre de plis K et du nombre de cycles de validation croisée L peut sembler arbitraire. Le choix de ces paramètres est en fait à la discrétion de l'utilisateur. En particulier, il est évident qu'un grand nombre de répétitions L aura pour effet d'augmenter la précision de notre étude. Cependant, de tels choix augmentent considérablement le temps de computation, puisque nous augmentons le nombre d'itérations. C'est pourquoi il est important de clarifier quels sont les choix par défaut des paramètres de validation croisée K et L, avant de passer aux études par simulation.

3.1.1. Choisir le nombre de plis K

Les choix typiques pour le nombre de plis sont souvent $K = 5, 10$ ou n, selon Hastie *et al.* (2015). Lorsque $K = n$, la validation croisée porte le nom spécial de *leave-one-out* et fait en sorte que chacune des observations a la chance de jouer le rôle de l'ensemble test. Lorsque K est très grand ou proche de n, l'estimateur par validation croisée est approximativement sans biais pour l'erreur de prédiction réelle (attendue), mais peut avoir une variance élevée puisque les n ensembles d'entraînement sont très similaires les uns aux autres. De plus, le coût computationnel devient considérable, surtout si n est grand. Lorsque K est plus petit, par exemple $K = 5$, le coût computationnel diminue grandement, mais on observe l'effet contraire, soit un plus grand biais et une plus petite variabilité. Nous sommes alors confrontés à choisir un paramètre K qui n'introduit pas trop de biais pour l'erreur de prédiction, mais qui garde une faible variabilité.

À l'aide d'une étude par simulation sur des données artificielles, les auteurs Breiman et Spector (1992) ont comparé la méthode de validation croisée K-plis à la méthode *leave-one-out*. Différents plis $K \in \{2, 5, 10\}$ sont utilisés dans les simulations, ainsi que différents nombres d'observations n. Des conclusions intéressantes peuvent être tirées de cette étude :

(1) La validation croisée 10-plis performe mieux que la méthode du *leave-one-out* dans un contexte de sélection de modèle.
(2) Pour les modèles ayant plusieurs variables explicatives, le *leave-one-out* mène à un MSE plus petit que la validation croisée 2-plis et 5-plis.
(3) La validation croisée 5-plis est une méthode plus robuste que le *leave-one-out* en termes de prédiction et sélection de modèle.

Les auteurs remarquent également que la méthode *leave-one-out* est une approche computationnellement coûteuse, surtout pour des jeux de données ayant un grand n.

Kohavi (1995) compare le bootstrap, une technique de rééchantillonnage avec remise, et la validation croisée K-plis. Basé sur des études par simulation, l'auteur conclut que le nombre de plis $K = 10$ est généralement le meilleur compromis en pratique. Il suggère également de répéter la validation croisée L fois, ce qui permettra d'améliorer l'approche de validation croisée. Les auteurs Hastie *et al.* (2015) semblent en accord avec les conclusions de Breiman et Spector (1992) et Kohavi (1995); une validation croisée avec $K = 5$ ou 10 semble optimale pour les études statistiques.

3.1.2. Choisir le nombre de cycles de validation croisée L

La validation croisée répétée est utilisée dans l'algorithme 2.7.2 (Lasso Linéaire). Kohavi (1995) suggère d'effectuer un certain nombre de répétitions L afin d'améliorer l'analyse statistique et la robustesse des prédictions provenant de la validation croisée. À chacune des

répétitions, le jeu de données est redivisé aléatoirement, nous permettant de former K nouveaux groupes; l'algorithme entraîne le modèle à nouveau, ce qui nous permet d'améliorer les MSE provenant des prédictions. Par contre, plus le nombre de répétitions L est grand, plus le temps de calcul sera grand.

Le nombre de répétitions utilisé par les auteurs Fraser et Bédard (2022) est $L = 50$. Ce nombre peut sembler excessif, surtout lorsque la dimension p est grande. Il n'y pas de consensus dans la littérature sur le nombre optimal de répétitions L; le choix de ce paramètre est laissé à la discrétion de l'utilisateur. Une règle du pouce serait d'essayer $L = 10$ pour des petits jeux de données ($p < 20$). Toutefois, si on se trouve dans un contexte où p est grand, il sera préférable de ne pas dépasser $L = 10$; autrement, il pourrait devenir fort coûteux computationnellement d'implémenter l'algorithme du Lasso Linéaire (ainsi que tout algorithme compétiteur).

3.2. Améliorations à l'algorithme

En régression, nous sommes souvent confrontés à des variables explicatives catégorielles. Pour que l'algorithme du Lasso Linéaire fonctionne, les variables catégorielles doivent être recodées en variables binaires. Afin de faciliter l'expérience de l'utilisateur, nous allons introduire une fonction automatisant ce processus de binarisation. Dans ce qui suit, nous proposons également d'appliquer la règle de l'erreur standard (1se) dans le but d'obtenir un modèle final plus parcimonieux.

3.2.1. La règle du 1 erreur standard

La règle de l'erreur standard (1se) a été introduite par Breiman *et al.* (1984) dans un contexte d'arbre de décision, permettant d'obtenir un arbre de taille parcimonieuse. Cette notion est par la suite reprise par les auteurs Hastie *et al.* (2009) dans un contexte de validation croisée appliquée au Lasso régulier. En particulier, plutôt que de choisir λ_{min} parmi les candidats $\lambda \in \{\lambda_1, \ldots, \lambda_m\}$, nous utilisons plutôt le paramètre λ_{1se} dont la valeur de MSE se situe à (au plus) une erreur standard de celle de λ_{min}. Pour chacun des candidats $\lambda \in \{\lambda_1, \ldots, \lambda_m\}$, l'erreur standard peut être calculée avec l'équation suivante

$$SE(\lambda) = SD(\lambda)/\sqrt{K},$$

où $SD(\lambda)$ est l'écart-type des K estimations $MSE_k(\lambda)$ de l'erreur moyenne de prédiction. Au lieu de choisir le paramètre λ_{min} minimisant l'erreur quadratique moyenne, nous choisissons le paramètre $\lambda_{1se} > \lambda_{min}$, associé à un MSE se situant à (au plus) une erreur standard de l'erreur quadratique moyenne minimum. Sachant que λ_{1se} est plus grand λ_{min}, le Lasso régulier choisit alors un modèle final comportant moins de prédicteurs. Par exemple, les auteurs Friedman *et al.* (2010) introduisent la règle du 1se dans la fonction `glmnet` sur

le progiciel **R**, nous permettant de choisir un modèle plus parcimonieux que la règle du minimum pour le Ridge ou le Lasso régulier.

De la même manière, pour l'algorithme du Lasso Linéaire, il est possible de calculer l'erreur standard. Pour ce faire, nous utilisons les équations explicitées dans l'étape 9 de l'algorithme 2.7.2, soit

$$MSE(\mathcal{T}) = K^{-1}\sum_{k=1}^{K} MSE_k(\mathcal{T}) \text{ et } V(\mathcal{T}) = (K-1)^{-1}\sum_{l=1}^{K}\left(MSE_k(\mathcal{T}) - MSE(\mathcal{T})\right)^2,$$

où $MSE(\mathcal{T})$ et $V(\mathcal{T})$ sont respectivement la moyenne et la variance des K estimations $MSE_k(\mathcal{T})$ de l'erreur moyenne de prédiction. Pour chaque taille de modèle, $\mathcal{T} \in \{1, \ldots, p\}$, nous calculons simplement

$$SE(\mathcal{T}) = \frac{\sqrt{V(\mathcal{T})}}{\sqrt{K}},$$

ce qui constitue les erreurs standards associées aux différentes tailles de modèle. Pour trouver la taille optimale selon la règle du 1se, $\mathcal{T}_{1se}$, il suffit d'identifier la valeur $MSE(\mathcal{T})$ la plus proche de

$$MSE(\mathcal{T}_{min}) + SE(\mathcal{T}_{min}),$$

et de choisir la taille $\mathcal{T}_{1se} < \mathcal{T}_{min}$ correspondante.

Il a été montré que la règle du 1se ($\mathcal{T}_{1se}$) ne surpasse pas nécessairement la méthode traditionnelle de validation croisée ($\mathcal{T}_{min}$). Malgré le fait qu'elle ait été largement utilisée comme une alternative au $\mathcal{T}_{min}$, il n'y a pas de preuve confirmant que la règle du 1se est supérieure à la règle du minimum selon Chen et Yang (2021). Les auteurs de Chen et Yang (2021) ont remarqué que la méthode du 1se a tendance à être instable. Elle a tendance à surestimer le modèle avec 2 plis, mais à sous-estimer le modèle lorsqu'il est utilisé avec une validation croisée à 20 plis. Nous incluons malgré tout la méthode du 1se dans l'algorithme du Lasso Linéaire, puisque cette règle est une alternative intéressante dans la recherche d'un modèle parcimonieux. Selon Hastie *et al.* (2015), le sous-modèle se trouve souvent dans une dimension inférieure à p, nous donnant la motivation de conserver la règle du 1se dans l'algorithme du Lasso Linéaire.

3.2.2. Les variables catégorielles

Nous savons que les variables explicatives peuvent être continues, discrètes ou catégorielles. Les variables discrètes ou continues sont déjà bien gérées par l'algorithme, mais nous devons apporter quelques précisions au sujet des variables catégorielles. Celles-ci peuvent être nominales (bleu, vert ou brun, pour la couleur des yeux par exemple) ou ordinales (mesure de satisfaction de 1 à 5, par exemple). Afin d'éviter d'induire des contraintes non

désirées entre les différentes catégories d'une variable, nous codons une variable x_j comportant r catégories à l'aide de $r-1$ variables binaires. Nous pouvons par la suite ajuster $r-1$ coefficients de régression associés aux $r-1$ catégories de cette variable. L'algorithme du Lasso Linéaire peut alors étudier la pertinence de chacun de ces coefficients dans le modèle, et ensuite regrouper les différentes catégories de façon appropriée dans le but d'interpréter le modèle correctement.

À cet effet, la fonction `model.matrix` est incluse dans l'algorithme du Lasso Linéaire pour éviter que l'utilisateur n'ait à recoder les r catégories des variables catégorielles manuellement. De plus, dans l'éventualité où les catégories sont codées en format texte (pour, contre ou indécis, par exemple), la fonction `model.matrix` ajuste également le nom des $r-1$ variables générées, afin de bien refléter le nom des catégories introduites. L'exemple des notes de mathématiques introduit par Cortez et Silva (2008) contient plusieurs variables catégorielles. Cet exemple est également étudié par les auteurs Fraser et Bédard (2022) pour illustrer le fonctionnement du Lasso Linéaire. Dans cet exemple, nous utilisons la règle du 1se, ainsi que la fonction `model.matrix` pour recoder les variables catégorielles.

3.2.3. Exemple : Notes de mathématiques

Le jeu de données des notes de mathématiques introduit par Cortez et Silva (2008) est constitué de plusieurs variables explicatives nominales, binaires ou continues. Il y a 13 variables binaires, dont le sexe de l'étudiant, son domicile (urbain ou rural), la taille de sa famille (≤ 3 ou > 3), l'accès à internet (oui ou non), etc. Il y a 4 variables nominales et 15 variables continues, telles que les notes obtenues dans le passé. La liste détaillée des variables explicatives, ainsi que leur type, se trouvent à l'annexe A.4. La variable réponse est la note obtenue au dernier trimestre (`G3`). Les variables binaires et nominales sont codées sous format texte, alors en ayant un grand nombre de variables catégorielles, il devient intéressant d'utiliser la fonction `model.matrix` sur le progiciel **R** pour pouvoir rapidement recoder les variables. On se retrouve ainsi avec $p = 41$ variables explicatives. Nous effectuons 10 répétitions d'une validation croisée à 5 plis et étudions les valeurs $\gamma \in \{0; 0{,}2\}$. Notons qu'une valeur $\gamma = 0$ teste la procédure un-à-un uniquement, alors que $\gamma = 0{,}2$ représente le paramètre par défaut spécifié dans Fraser et Bédard (2022).

Les figures 3.1 et 3.2 nous permettent de bien visualiser le processus de sélection et de cerner les variables qui sont incluses dans le modèle final. La figure 3.1 trace les valeurs de MSE de la procédure un-à-un en fonction des variables sorties du modèle. Le point rouge indique le MSE minimal, qui correspond à une taille de modèle $\mathcal{T}_{min} = 5$; de façon similaire, le point vert représente le MSE associé à $\mathcal{T}_{1se} = 1$. La ligne rouge sur la figure nous permet de voir facilement quelles sont les variables retenues dans le modèle final selon la règle du minimum ($\mathcal{T}_{min}$).

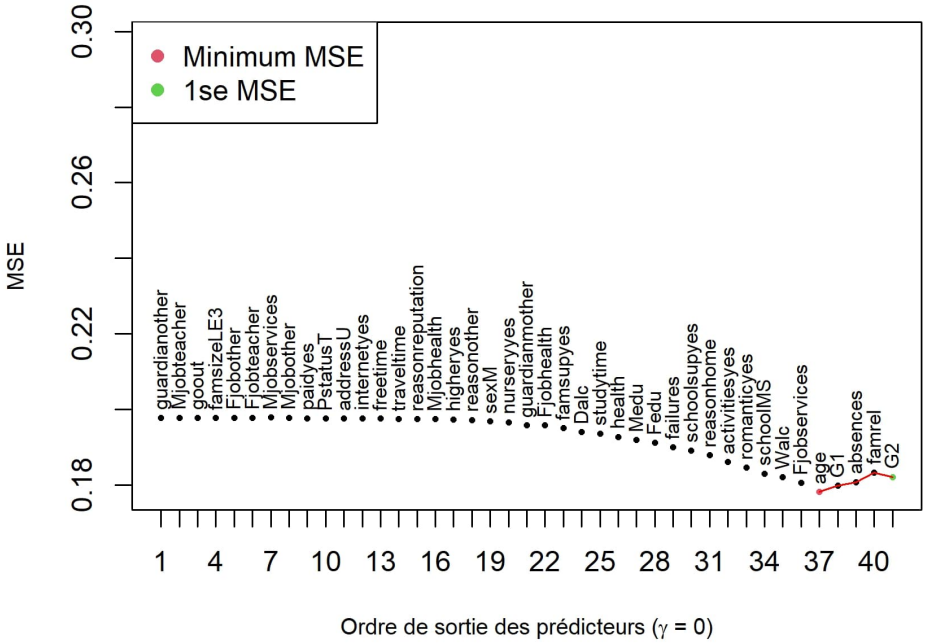


Figure 3.1. *MSE* en fonction des variables sorties du modèle pour la validation croisée à 5 plis répétée 10 fois avec un paramètre d'ajustement de $\gamma = 0$. La ligne rouge indique le modèle final.

Tableau 3.1. Modèles retenus d'après la procédure un-à-un pour $\gamma \in \{0; 0{,}2\}$

γ	$\mathcal{T}$	Modèle retenu
0	min	`age` + `G1` + `absences` + `famrel` + `G2`
	1se	`G2`
0,2	min	`G1` + `G2`
	1se	`G2`

Dans la figure 3.1, la procédure un-à-un est exécutée sur le modèle complet, puisque $\gamma = 0$. Toutefois, lorsqu'un γ de 0,2 est spécifié, le temps de calcul est grandement réduit; il y a 37 variables qui sont retirées basé sur la taille de leur corrélation avec la variable réponse. Les modèles pour chacun des paramètres γ (0 et 0,2) et chacun des types de *MSE* (min ou 1se) se trouvent dans le tableau 3.1.

Tel que mentionné dans Cortez et Silva (2008), il y a un grand nombre de variables explicatives qui sont peu pertinentes dans la modélisation de la variable réponse, ce qui est cohérent avec les observations du tableau 3.1. Les notes du premier (`G1`) et second (`G2`) trimestres semblent bien expliquer la note du dernier trimestre, ce qui a également été observé par Cortez et Silva (2008). De plus, le modèle choisi par la règle du 1se est plus

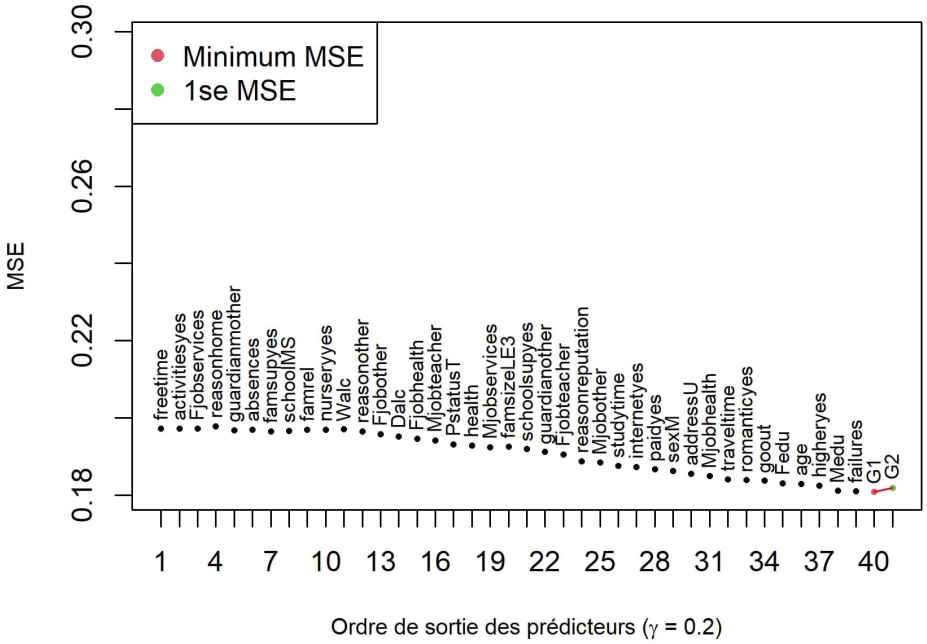


Figure 3.2. *MSE* en fonction des variables sorties du modèle pour la validation croisée à 5 plis répétée 10 fois avec un paramètre d'ajustement de $\gamma = 0{,}2$. La ligne rouge indique le modèle final.

parcimonieux que celui obtenu par la règle du minimum, tel que mentionné dans la section 3.2.1.

Les ajouts à la fonction du Lasso Linéaire permettent d'améliorer la méthode et de la rendre plus facile d'utilisation. La règle du 1se nous permet de trouver un modèle plus parcimonieux que la méthode standard du minimum. La figure 3.1, par exemple, nous aide à mieux situer le modèle final pour la procédure un-à-un, et ce pour le minimum et la méthode 1se. De plus, l'ajout de la fonction `model.matrix` permet d'automatiser le processus de recodage des variables catégorielles. Les modèles finaux ne conservent aucune variable catégorielle, ce qui mène à une interprétation simple du modèle puisqu'il n'y a aucun regroupement de catégories à effectuer.

3.3. Grande dimensionnalité

La grande dimensionnalité des données est un enjeu actuel qui touche plusieurs domaines et il est fréquemment possible d'observer des données dont la dimension p est beaucoup plus grande que la taille échantillonnale n. Nous savons que la situation où $p > n$ est problématique pour l'estimateur MCO, tel que discuté dans la section 1.2.2.

Il est alors intéressant de noter que si la dimensionnalité était réduite de p à $d \leq n$, n'importe quel estimateur serait facilement utilisable par la suite. Une méthode très intéressante développée par Fan et Lv (2008) est celle du *Sure Independence Screening* (SIS). Dans un premier temps, le SIS permet de retirer rapidement un grand nombre de prédicteurs en gardant, par la suite, uniquement les prédicteurs importants. Ceci peut être effectué en étudiant la corrélation entre la variable réponse $\mathbf{y}$ et les variables explicatives $\mathbf{x}_j$. Les variables ayant les plus grandes corrélations absolues sont conservées dans le modèle et les autres rejetées. Nous passons alors d'une grande dimension p à une dimension modérée d, qui est généralement choisie de sorte à être plus petite ou égale à n. Dans un deuxième temps, avec le nombre de paramètres réduit, il est possible de calculer les estimateurs voulus, puisque la dimension est maintenant $d \leq n$.

Dans le Lasso Linéaire, les variables faiblement corrélées avec la variable réponse sont conservées et étudiées dans la validation croisée. Toutefois, lorsque $p > n$, il devient impossible de calculer l'estimateur MCO pour toutes les tailles de modèle. Il devient alors nécessaire d'éliminer un certain nombre de prédicteurs, de sorte à retrouver un nombre de variables inférieur à n. L'approche découlant naturellement du Lasso linéaire est évidemment d'éliminer les prédicteurs les plus faiblement corrélés avec la variable réponse, ce qui est cohérent avec le SIS. Une fois cette étape complétée, un paramètre γ est sélectionné et l'algorithme est appliqué sur les variables restantes.

En pratique, nous avons observé qu'il est préférable de diminuer grandement la dimension, puisque si d est près de n, la matrice de design $\boldsymbol{X}$ demeure parfois singulière. Dans ce qui suit, nous utilisons donc l'approche proposée par Fan et Lv (2008), soit de réduire la dimension à $d = \lfloor n/\log(n) \rceil$ prédicteurs, où $\lfloor \cdot \rceil$ est l'entier le plus proche. Lorsque $p > n$, l'algorithme du Lasso Linéaire conserve alors les d prédicteurs les plus corrélés avec $\mathbf{y}$ et rejette les autres. Par la suite, nous appliquons la procédure habituelle sur les variables restantes. Voici un résumé des étapes à suivre :

(1) Lorsque $p \geq n$, calculer $d = \lfloor n/\log(n) \rceil$, où $\lfloor \cdot \rceil$ est l'entier le plus proche.
(2) Retirer les $p - d$ variables du modèle basé sur les corrélations avec $\mathbf{y}$. C'est ainsi qu'on peut passer d'une dimension p à une dimension d, puisqu'il est important que d soit inférieur ou égal à n.
(3) Choisir un paramètre γ et appliquer le Lasso Linéaire sur les d variables restantes.

Dans le cas $p > n$, nous verrons que les valeurs des corrélations c_j sont très grandes, ce qui nous empêche d'utiliser le paramètre d'ajustement par défaut $\gamma = 0{,}2$. Il devient alors plus intuitif de référer au nombre de prédicteurs classifiés lors de la première étape du Lasso Linéaire, en se rappelant que la notation m définie en (2.7.1) est utilisée pour dénoter ce nombre, soit $m = \{\#j \in \{1, \ldots, p\} : c_j < \gamma\}$. Si nous souhaitons utiliser uniquement la procédure un-à-un, il suffit alors de choisir un paramètre γ qui concorde avec $m = 0$. À

l'inverse, si nous voulons classifier toutes les variables lors de la première étape du Lasso Linéaire, nous choisissons plutôt $m = d$. L'exemple qui suit nous permet d'étudier le Lasso Linéaire sur des données d'expression génique.

3.3.1. Exemple : Données d'expression génique

Les données d'expression génique sont souvent utilisées pour prédire la progression d'une maladie, ou encore la maladie elle-même. Par exemple, il est courant de chercher à identifier les gènes expliquant le mieux la progression d'une maladie spécifique. Dans les jeux de données d'expression génique, la quantité de gènes p est souvent très grande; les gènes sont également fortement corrélés entre eux.

Dans l'étude de Scheetz *et al.* (2006), les auteurs avaient pour but d'étudier le niveau d'expression du gène TRIM32 lié à une maladie génétique appelée syndrome de Bardet-Biedl dans l'œil des mammifères. Des rats de laboratoire ont été étudiés afin de connaître l'expression et la régulation des gènes dans l'œil de ces mammifères. Le jeu de données original de Scheetz *et al.* (2006) est constitué de 120 rats et de 31 000 gènes différents. Toutefois, le jeu de données fourni publiquement contient uniquement les 500 variables explicatives les plus corrélées avec la variable réponse $\mathbf{y}$. Nous aimerions comprendre quels sont les gènes qui pourraient prédire la progression de la maladie en utilisant le gène TRIM32 comme variable réponse.

L'histogramme dans la figure 3.3 nous indique que plusieurs des gènes utilisés comme variables explicatives possèdent une très grande corrélation avec TRIM32; en effet, 500 des 31 000 prédicteurs possèdent une corrélation avec la variable réponse qui est supérieure à 0,64. Il est alors pratique d'utiliser la méthode décrite ci-dessus, nous permettant de ne conserver que $d = \lfloor n/\log(n) \rceil$ prédicteurs et, par la suite, d'appliquer l'algorithme 2.7.2 sur les variables restantes. Nous remarquons également que le paramètre d'ajustement par défaut de $\gamma = 0{,}2$ n'est clairement pas adapté pour ce jeu de données, puisque sur la figure 3.3, les corrélations sont très élevées.

Sachant que $n = 120$, nous obtenons $d = \lfloor 120/\log(120) \rceil = 25$. Dès lors, nous appliquons les étapes (1) à (3) avec une validation croisée à 10 plis et 10 cycles afin de trouver le sous-modèle expliquant le mieux la progression de la maladie. Parmi les 25 variables restantes, la plus petite corrélation c_j est de 0,73 et la plus grande est de 0,78. À des fins de comparaison, nous implémentons l'algorithme 2.7.2 avec $\gamma \in \{0{,}73; 0{,}74; 0{,}75; 0{,}76, 0{,}77, 0{,}78\}$ ce qui concorde avec $m \in \{0, 9, 15, 19, 22, 25\}$. Les modèles obtenus pour $\gamma = 0{,}73$ et $\gamma = 0{,}78$ sont illustrés à la figure 3.4.

Le Lasso Linéaire arrive à bien cerner la grande dimensionnalité en éliminant rapidement les 30 975 prédicteurs les moins importants pour ce jeu de données, nous laissant seulement avec $d = 25$. Le modèle final selon la règle du minimum avec $\gamma = 0{,}73$ est indiqué par

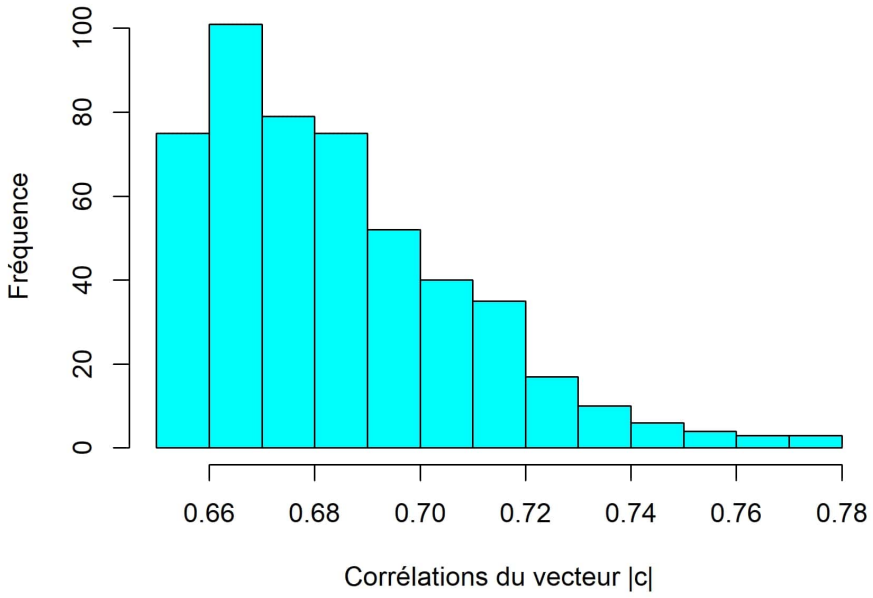


Figure 3.3. Histogramme des corrélations entre la variable réponse et les 500 variables explicatives les plus corrélées avec **y** pour le jeu de données des rats.

Tableau 3.2. Erreur quadratique moyenne (MSE) pour les différents γ pour l'exemple de gènes.

γ	m	$\mathcal{T}_{min}$	MSE
0,73	0	9	**0,2769**
0,74	9	19	0,2953
0,75	15	19	0,2940
0,76	19	18	0,2898
0,77	22	20	0,2982
0,78	25	20	0,2934

la ligne rouge sur la figure 3.4 (graphique de gauche) et comporte 9 gènes; le modèle final selon la règle du 1se est indiqué par le point vert et conserve seulement 5 gènes comme prédicteurs. Lorsque $\gamma = 0{,}78$ (graphique de droite), les modèles sont plus complexes avec 20 et 11 prédicteurs conservés respectivement pour la règle du minimum et 1se. Dans le tableau 3.2, le MSE minimum est celui avec la taille de modèle $\mathcal{T}_{min} = 9$ pour $\gamma = 0{,}73$ (ou $m = 0$). Ainsi, la procédure un-à-un semble être mieux adaptée pour ces données.

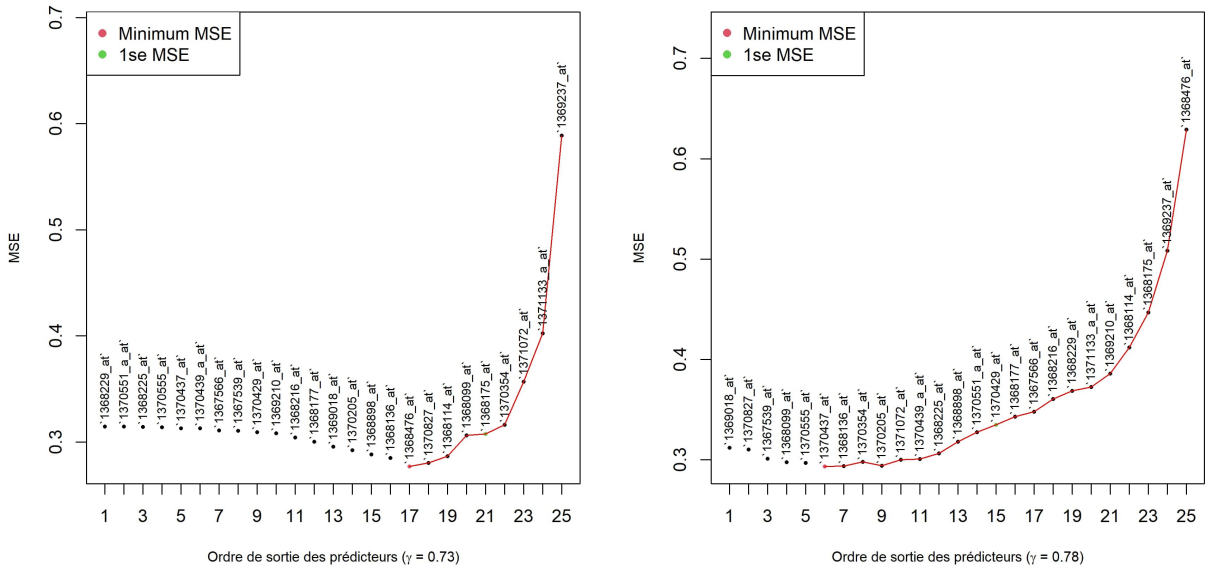


Figure 3.4. *MSE* en fonction des variables sorties du modèle pour la validation croisée à 10 plis répétée 10 fois avec un paramètre d'ajustement de $\gamma = 0{,}73$ et $\gamma = 0{,}78$ selon la règle $d = \lfloor n/\log(n) \rceil$. La ligne rouge indique le modèle final.

À première vue, nous pourrions penser qu'un plus grand nombre de gènes devraient être inclus dans le modèle, puisque plus de 500 prédicteurs sont fortement corrélés avec la variable réponse. Toutefois, nous savons que les maladies sont multifactorielles, et qu'en ayant des milliers de gènes, il est important de savoir quels gènes sont les plus pertinents. Un bon point de départ pour des scientifiques étudiant le syndrome de Bardet-Biedl serait de commencer par étudier les gènes identifiés par l'algorithme du Lasso Linéaire. Les 9 gènes spécifiés par l'algorithme pourraient potentiellement expliquer la progression du gène TRIM32 dans l'œil des mammifères.

3.4. Réglage du paramètre d'ajustement γ

La valeur du paramètre d'ajustement γ utilisée par défaut est de 0,2. Cette valeur a été testée par Fraser et Bédard (2022) et donne des résultats comparables au Lasso régulier pour deux exemples avec de vraies données. Pour les données d'expression génétique de grande dimension, nous en sommes venus à la conclusion qu'il était préférable d'éliminer $p - d = p - \lfloor n/\log(n) \rceil$, comme dans la méthode du *Sure Independence Screening* de Fan et Lv (2008), et ensuite d'appliquer l'algorithme 2.7.2. L'objectif de cette section est de comprendre dans quelles circonstances le paramètre d'ajustement $\gamma = 0{,}2$ constitue un candidat intéressant pour le Lasso Linéaire lorsque $n > p$.

3.4.1. Les exemples de simulation

Pour arriver à nos fins, nous simulons des jeux de données à l'aide du progiciel **R**. Ces simulations sont construites de sorte à aisément calibrer l'algorithme du Lasso Linéaire. Supposons n et p qui sont fixes et le modèle de régression linéaire usuel,

$$\mathbf{y} = \boldsymbol{X}^*\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \tag{3.4.1}$$

où $\boldsymbol{\beta}^*$ est un vecteur contenant les coefficients connus, $\boldsymbol{X}^*$ est la matrice de design générée de façon aléatoire et $\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n)$. Selon les auteurs Hastie *et al.* (2015), le sous-modèle idéal se trouve généralement dans une dimension inférieure au nombre de variables explicatives disponibles, nous suggérant de choisir des exemples avec peu de coefficients non nuls. Nous utilisons alors les exemples de Tibshirani (1996) en variant le nombre d'observations n; $n \in \{50, 100\}$ pour l'exemple 1 et 2, ainsi que $n \in \{200, 400\}$ pour l'exemple 3. Dans tous les cas, nous posons $\sigma = 3$.

- Exemple 1 : Le vecteur de coefficients $\boldsymbol{\beta}_1^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$ sera testé puisque c'est un exemple comportant peu de coefficients de grande taille.
- Exemple 2 : Le vecteur de coefficients $\boldsymbol{\beta}_2^* = (0{,}85; \ldots; 0{,}85)^\top$ possède 8 coefficients non nuls de petite taille.
- Exemple 3 : Le vecteur de coefficients $\boldsymbol{\beta}_3^* = (\underbrace{0, \ldots, 0}_{10}, \underbrace{2, \ldots, 2}_{10}, \underbrace{0, \ldots, 0}_{10}, \underbrace{2, \ldots, 2}_{10})^\top$ est un exemple intéressant, car il possède plusieurs coefficients non nuls de grande taille. Cet exemple sera un bon test pour le Lasso Linéaire, puisqu'il sera difficile de discerner les coefficients nuls des coefficients non nuls.

Dans un même ordre d'idées, la matrice de design $\boldsymbol{X}^*$ est construite selon deux approches différentes. Dans la première, nous utilisons des prédicteurs gaussiens standards i.i.d, soit $\boldsymbol{X}^* \sim \mathcal{MN}(0_p, I_p)$. Dans la deuxième, nous utilisons des prédicteurs provenant d'une normale avec matrice de variance-covariance autorégressive de dimension p, $\boldsymbol{X}^* \sim \mathcal{MN}(0_p, \Sigma_{AR})$, où

$$\Sigma_{AR} = \begin{pmatrix} 1 & \rho & \rho^2 & \ldots & \rho^{p-1} \\ \rho & 1 & \rho & \ldots & \rho^{p-2} \\ \rho^2 & \rho & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \rho \\ \rho^{p-1} & \ldots & \rho^2 & \rho & 1 \end{pmatrix}.$$

Nous devons également générer aléatoirement le vecteur d'erreurs $\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n)$; pour ce faire, nous spécifions une valeur de $\sigma = 3$ et obtenons $\boldsymbol{\varepsilon}$. Ainsi, en spécifiant le nombre d'observations n, le nombre de prédicteurs p, la matrice de design simulée $\boldsymbol{X}^*$, ainsi que le vecteur de coefficients $\boldsymbol{\beta}^*$, nous pouvons calculer le vecteur réponse $\mathbf{y}$ à l'aide de l'équation

(3.4.1). Pour chacun de nos trois exemples, nous utilisons les deux matrices simulées $\boldsymbol{X}^* \sim \mathcal{MN}(0_p, I_p)$ et $\boldsymbol{X}^* \sim \mathcal{MN}(0_p, \Sigma_{AR})$, ainsi que les deux tailles échantillonnales spécifiées; ceci totalise 4 combinaisons par exemple.

Le paramètre ρ de la matrice Σ_{AR} est fixé à 0,5. Les valeurs $K = 10$ et $L = 5$ sont utilisées pour l'étape de validation croisée du Lasso Linéaire. Ces valeurs demeurent constantes durant toutes les simulations.

3.4.2. Critères de performance

Une fois les exemples bien documentés, nous aimerions vérifier si le paramètre d'ajustement $\gamma = 0{,}2$ est optimal, en termes de performance, pour le Lasso Linéaire. Pour atteindre notre objectif, nous devons naturellement établir des critères de performance. Les critères choisis se baseront sur la performance prédictive, c'est-à-dire la distance entre $\mathbf{y}$ et $\hat{\mathbf{y}}$, ainsi que sur le nombre de coefficients nuls ou non nuls correctement classifiés. Nous rapporterons également le temps de computation du Lasso Linéaire.

Pour ce faire, nous générons 500 jeux de données pour chacune des combinaisons de $(\boldsymbol{X}^*, n)$. Pour chaque jeu de données, nous testons 51 paramètres d'ajustement, soit

$$\gamma \in \{0; 0{,}01; 0{,}02; \ldots; 0{,}49; 0{,}50\}. \tag{3.4.2}$$

Nous incluons la valeur $\gamma = 0$, puisque ce paramètre d'ajustement nous permet d'étudier le fonctionnement individuel de la procédure un-à-un du Lasso Linéaire. Nous prenons la décision d'exclure les valeurs supérieures à 0,50, puisqu'en pratique, les corrélations entre le vecteur réponse et les prédicteurs dépassent rarement 0,50 lorsque $n > p$.

Les 500 jeux de données sont produits selon l'approche suivante: nous générons d'abord 500 matrices de design $\{\boldsymbol{X}_1^*, \ldots, \boldsymbol{X}_{500}^*\}$; par la suite, en utilisant les autres paramètres à notre disposition, nous générons les vecteurs réponse correspondants, $\{\mathbf{y}_1^*, \ldots, \mathbf{y}_{500}^*\}$, à l'aide de (3.4.1). Pour chacune des paires $(\boldsymbol{X}_k^*, \mathbf{y}_k^*)$, nous effectuons le Lasso Linéaire en testant tous les paramètres d'ajustement spécifiés dans (3.4.2). Pour chacun de ces paramètres, l'algorithme du Lasso Linéaire nous retourne $MSE(\mathcal{T}_{min})$, que nous notons à des fins de comparaison. De plus, l'algorithme nous retourne également l'estimateur $\hat{\boldsymbol{\beta}}^{(MCO)}$ associé à $\mathcal{T}_{min}$, qui nous permet par la suite de calculer les critères VP et FP. La valeur de vrais positifs (VP) est définie comme

$$VP_k(\gamma) = \{\#j \in \{1, \ldots, p\} : \beta_j^* \neq 0 \text{ et } \hat{\beta}_j^{(MCO)} \neq 0\},$$

et la valeur de faux positifs (FP),

$$FP_k(\gamma) = \{\#j \in \{1, \ldots, p\} : \beta_j^* = 0 \text{ et } \hat{\beta}_j^{(MCO)} \neq 0\},$$

où $k = 1, \ldots, 500$. Finalement, nous retournons le temps de calcul nécessaire pour chacun des paramètres γ.

Les tables des sections suivantes rapportent la moyenne des $k = 500$ valeurs de $MSE(\mathcal{T}_{min})$. Pour chaque critère, nous identifions par la suite la valeur de γ optimale,

$$\gamma_{min} = \underset{\gamma \in \{0;0,01;0,02;\ldots;0,49;0,50\}}{\operatorname{argmin}} MSE(\mathcal{T}_{min}).$$

Nous calculons également la moyenne des k valeurs de $VP_k(\gamma)$ et $FP_k(\gamma)$, pour ainsi obtenir $VP(\gamma)$ et $FP(\gamma)$. La valeur de vrais positifs $VP(\gamma)$ doit s'approcher le plus possible du nombre de coefficients non nuls de l'exemple donné, tandis que le valeur de faux positifs $FP(\gamma)$ doit plutôt s'approcher de 0. En ce qui concerne le temps de computation, nous calculons la moyenne du temps de calcul pour les 500 jeux de données. Les 4 critères explicités ci-dessus (MSE, VP, FP, temps de computation) nous permettront de vérifier si $\gamma = 0{,}2$ constitue le meilleur choix parmi tous les candidats (3.4.2).

3.4.3. Résultats du temps de computation

Le temps de computation est un aspect essentiel pour évaluer la performance d'un algorithme en régression linéaire. Pour être capable de se mesurer aux algorithmes modernes de régression linéaire, il est souhaitable que le Lasso Linéaire ait un temps de calcul aussi court que possible. Lorsque $\gamma \to 0$, nous savons que le temps de calcul est grand, car le Lasso Linéaire exécute la procédure un-à-un. Lorsque $\gamma \to 1$, nous savons que la totalité des variables explicatives est ordonnée lors de la première étape du Lasso Linéaire, ce qui nous mène à un faible temps de calcul; par contre, la corrélation entre les prédicteurs n'est pas prise en compte dans cet ordonnancement des sous-modèles. On s'attend donc à ce que le graphique du temps (en secondes) en fonction du paramètre d'ajustement γ soit de forme exponentielle. Ainsi, pour trouver le paramètre d'ajustement optimal sur le graphique, nous identifions le γ pour lequel le temps de calcul devient relativement constant. Les moyennes du temps de calcul (en secondes) des 500 simulations des exemples 1 et 2 sont présentées dans la figure 3.5 et celles de l'exemple 3 sont présentées dans la figure 3.6.

Les conclusions sont très similaires pour les deux graphiques de la figure 3.5. Nous remarquons que les courbes du temps de calcul deviennent relativement constantes à partir du paramètre d'ajustement 0,2 (sauf pour l'exemple 2 avec $\mathcal{MN}(0_p, \Sigma_{AR})$). Une augmentation du paramètre d'ajustement par incrément de 0,01 n'est alors pas justifiée, puisque nous ne réduisons pas significativement le temps de calcul. De plus, la procédure un-à-un sur le jeu de données complet ($\gamma = 0$) n'est pas une bonne option en termes de temps de calcul. Par ailleurs, le temps de calcul est plus grand lorsque le nombre d'observations est de 100, ce qui est tout à fait normal.

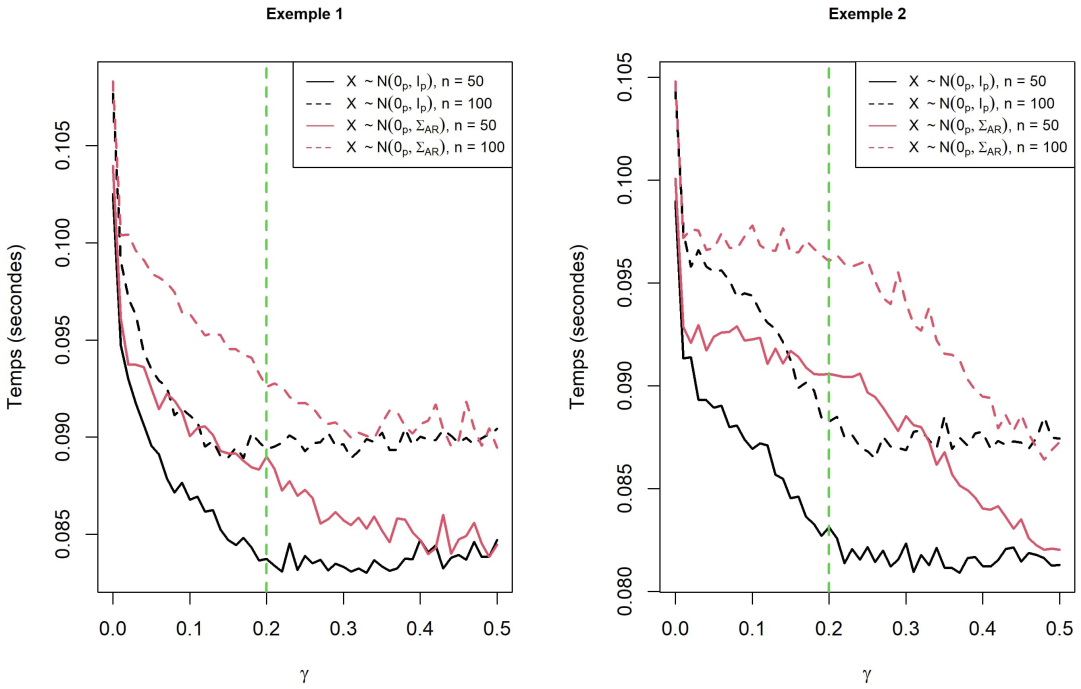


Figure 3.5. Moyennes du temps de calcul (en secondes) des 500 simulations pour les exemples 1 et 2. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

Les temps de calcul pour l'exemple 3 sont illustrés dans la figure 3.6. Nous remarquons que les courbes du temps de calcul deviennent relativement constantes à partir du paramètre d'ajustement 0,2, comme sur la figure 3.5. Nous renforçons le point que la procédure un-à-un sur le jeu de données complet n'est une bonne idée en termes de temps de calcul. En ayant 40 coefficients, dont 20 qui sont non nuls, la procédure un-à-un prend en moyenne 2 fois plus de temps que si le paramètre d'ajustement était de 0,2.

Pour les trois exemples, nous remarquons que le fait de rouler uniquement la procédure un-à-un ($\gamma = 0$) n'est pas une bonne idée d'un point de vue computationnel. La procédure un-à-un prend beaucoup de temps à faire tourner, alors il est important qu'elle soit jumelée avec la première étape de l'algorithme 2.7.2, dans laquelle on ordonne plusieurs prédicteurs basé sur la taille de leur corrélation c. De manière générale, le paramètre $\gamma = 0{,}2$ semble être un choix judicieux en termes de temps de computation, puisque les courbes ont tendance à se stabiliser à partir de cette valeur.

3.4.4. Résultats des critères MSE

Nous étudions maintenant l'effet du paramètre d'ajustement γ sur la performance du Lasso Linéaire en termes de l'erreur quadratique moyenne (MSE) de prédiction. Pour chacun

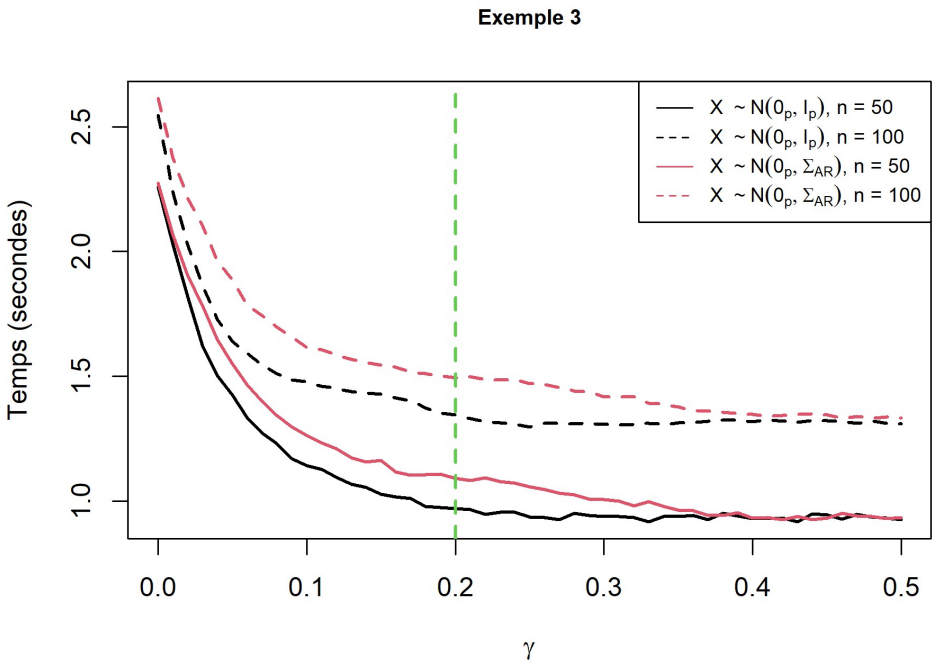


Figure 3.6. Moyennes du temps de calcul (en secondes) des 500 simulations pour l'exemple 3. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

des exemples $\boldsymbol{\beta}_1^*$, $\boldsymbol{\beta}_2^*$ et $\boldsymbol{\beta}_3^*$, nous présentons un tableau résumant l'information obtenue à propos de γ_{min} pour toutes les combinaisons possibles $(\boldsymbol{X}^*, n)$. Nous présentons également, pour chacun des exemples $\boldsymbol{\beta}_1^*$, $\boldsymbol{\beta}_2^*$ et $\boldsymbol{\beta}_3^*$, des figures illustrant l'évolution des courbes MSE en fonction du paramètre d'ajustement.

3.4.4.1. Exemple 1 : $\boldsymbol{\beta}_1^*$

Dans cet exemple, nous étudions le coefficient $\boldsymbol{\beta}_1^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$, un exemple comportant peu de coefficients de grande taille. Les résultats portant sur le critère du MSE peuvent être retrouvés dans le tableau 3.3. De plus, les figures illustrant l'évolution des courbes MSE en fonction du paramètre d'ajustement sont illustrées dans la figure 3.7.

Pour les graphiques obtenus obtenus à partir d'une matrice de design $\mathcal{MN}(0_p, I_p)$, nous avons des courbes qui croissent rapidement à partir de $\gamma = 0$ et qui ensuite se stabilisent. Par conséquent, même si $\gamma > 0$ est avantageux d'un point de vue computationnel, il semblerait plus approprié de favoriser la procédure un-à-un dans cet exemple. À l'inverse, pour la matrice de design $\mathcal{MN}(0_p, \Sigma_{AR})$, les courbes ont plutôt une allure convexe. Le minimum se situe proche de 0,2 pour $n = 50$; pour $n = 100$, la courbe augmente drastiquement à partir

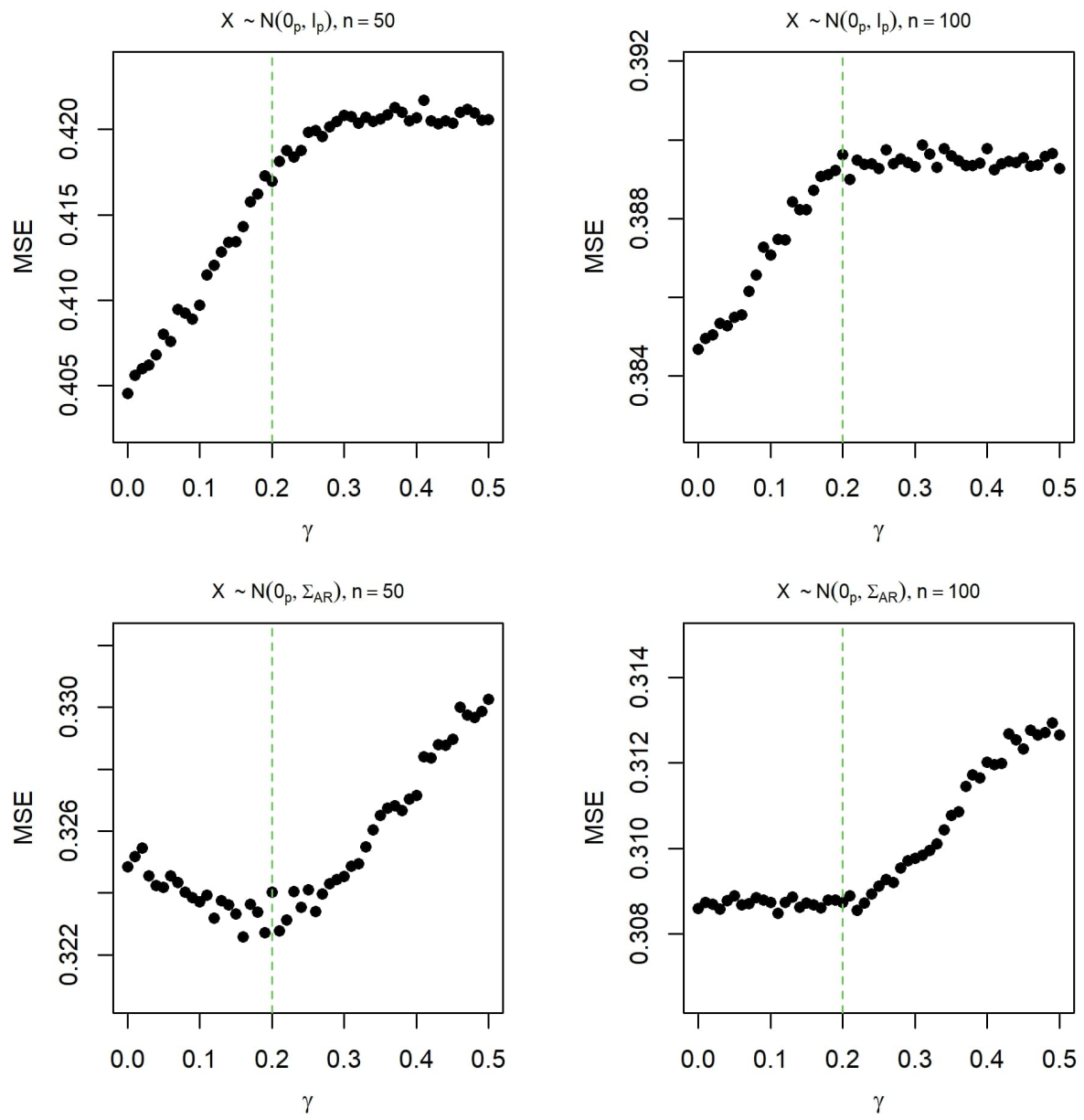


Figure 3.7. Erreur quadratique moyenne (MSE) de prédiction basée sur les 500 jeux de données pour l'exemple 1. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

de 0,2. Selon le tableau 3.3, γ_{min} se situe à 0 pour $\mathcal{MN}(0_p, I_p)$ et aux valeurs 0,16 et 0,11 $(n = 50, n = 100)$ pour $\mathcal{MN}(0_p, \Sigma_{AR})$.

Ces résultats peuvent s'expliquer par la présence de corrélation dans la deuxième matrice de design. En effet, cette corrélation a pour effet de stabiliser le bruit pour une observation donnée et un prédicteur donné. Ceci mène à des résultats intuitifs/sensés. Par opposition,

le fait de générer des éléments indépendants provenant d'une normale avec une grande variance relativement à la taille des coefficients a pour effet de masquer la corrélation entre les prédicteurs importants et les autres. Ceci est naturellement un scénario catastrophe pour le Lasso Lineaire, qui puise sa force dans l'existence d'une corrélation significative entre les prédicteurs importants et la variable réponse. Il serait éventuellement intéressant de creuser davantage à ce sujet en étudiant davantage de scénarios, mais ceci pourra être abordé dans un projet de recherche ultérieur.

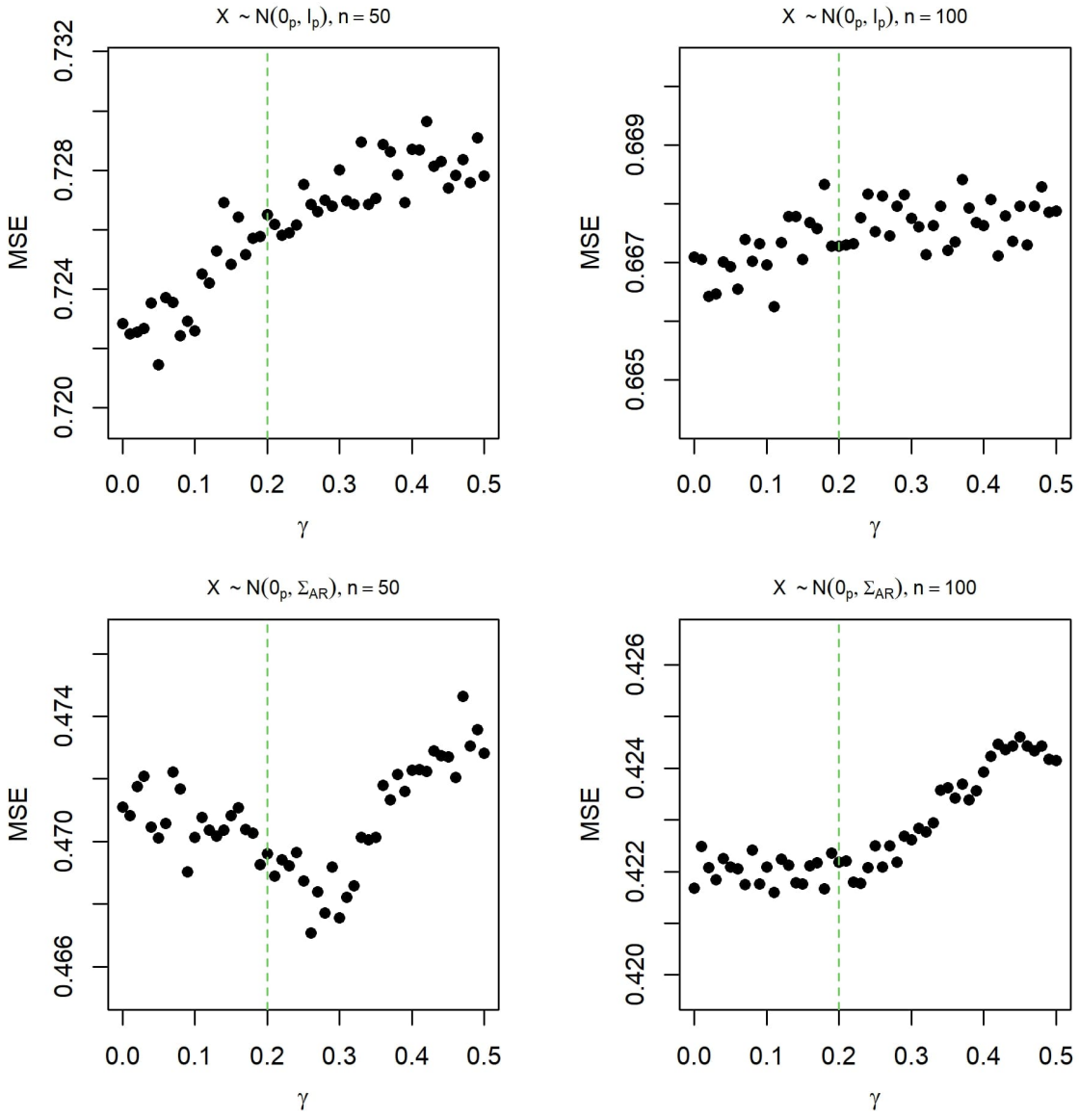


Figure 3.8. Erreur quadratique moyenne (MSE) de prédiction basée sur les 500 jeux de données pour l'exemple 2. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

3.4.4.2. Exemple 2 : $\boldsymbol{\beta}_2^*$

Dans cet exemple, nous étudions le coefficient $\boldsymbol{\beta}_2^* = (0{,}85; \ldots; 0{,}85)^\top$, un exemple modifié de Tibshirani (1996) comportant 8 coefficients identiques de petite taille. Les résultats portant sur le MSE de prédiction se trouvent dans le tableau 3.3, ainsi qu'à la figure 3.8.

La figure 3.8 nous permet de visualiser l'évolution de l'erreur quadratique moyenne de prédiction en fonction du paramètre d'ajustement. Cet exemple est particulier, car il comporte 8 coefficients qui sont tous non nuls, mais de petite taille. Nous remarquons que nous avons des courbes similaires, mais moins prononcées que celles de l'exemple 1. Pour la matrice $\mathcal{MN}(0_p, I_p)$, $\gamma_{min} \approx 0{,}1$ et pour $\mathcal{MN}(0_p, \Sigma_{AR})$, $\gamma_{min} \approx 0{,}2$.

Tableau 3.3. Erreur quadratique moyenne (MSE) de prédiction basée sur les 500 jeux de données pour les trois exemples.

Exemple	Combinaisons		$MSE(\mathcal{T}_{min})$		
	$\boldsymbol{X}^*$	n	γ_{min}	$\overline{MSE}$	SD
1	$\mathcal{MN}(0_p, I_p)$	50	0,00	0,405	0,105
		100	0,00	0,385	0,064
	$\mathcal{MN}(0_p, \Sigma_{AR})$	50	0,16	0,323	0,0846
		100	0,11	0,309	0,051
2	$\mathcal{MN}(0_p, I_p)$	50	0,05	0,722	0,134
		100	0,11	0,666	0,085
	$\mathcal{MN}(0_p, \Sigma_{AR})$	50	0,26	0,467	0,118
		100	0,11	0,422	0,067
3	$\mathcal{MN}(0_p, I_p)$	200	0,00	0,114	0,015
		400	0,00	0,106	0,011
	$\mathcal{MN}(0_p, \Sigma_{AR})$	200	0,00	0,047	0,007
		400	0,09	0,044	0,004

3.4.4.3. Exemple 3 : $\boldsymbol{\beta}_3^*$

Dans cet exemple, nous étudions le coefficient $\boldsymbol{\beta}_3^* = (\underbrace{0, \ldots, 0}_{10}, \underbrace{2, \ldots, 2}_{10}, \underbrace{0, \ldots, 0}_{10}, \underbrace{2, \ldots, 2}_{10})^\top$, un exemple modifié de Tibshirani (1996) comportant des coefficients de taille 2. Cet exemple contient 40 prédicteurs, contrairement aux autres exemples qui n'en ont que 8. Les résultats

portant sur le MSE de prédiction se trouvent dans le tableau 3.3, ainsi que dans la figure 3.9.

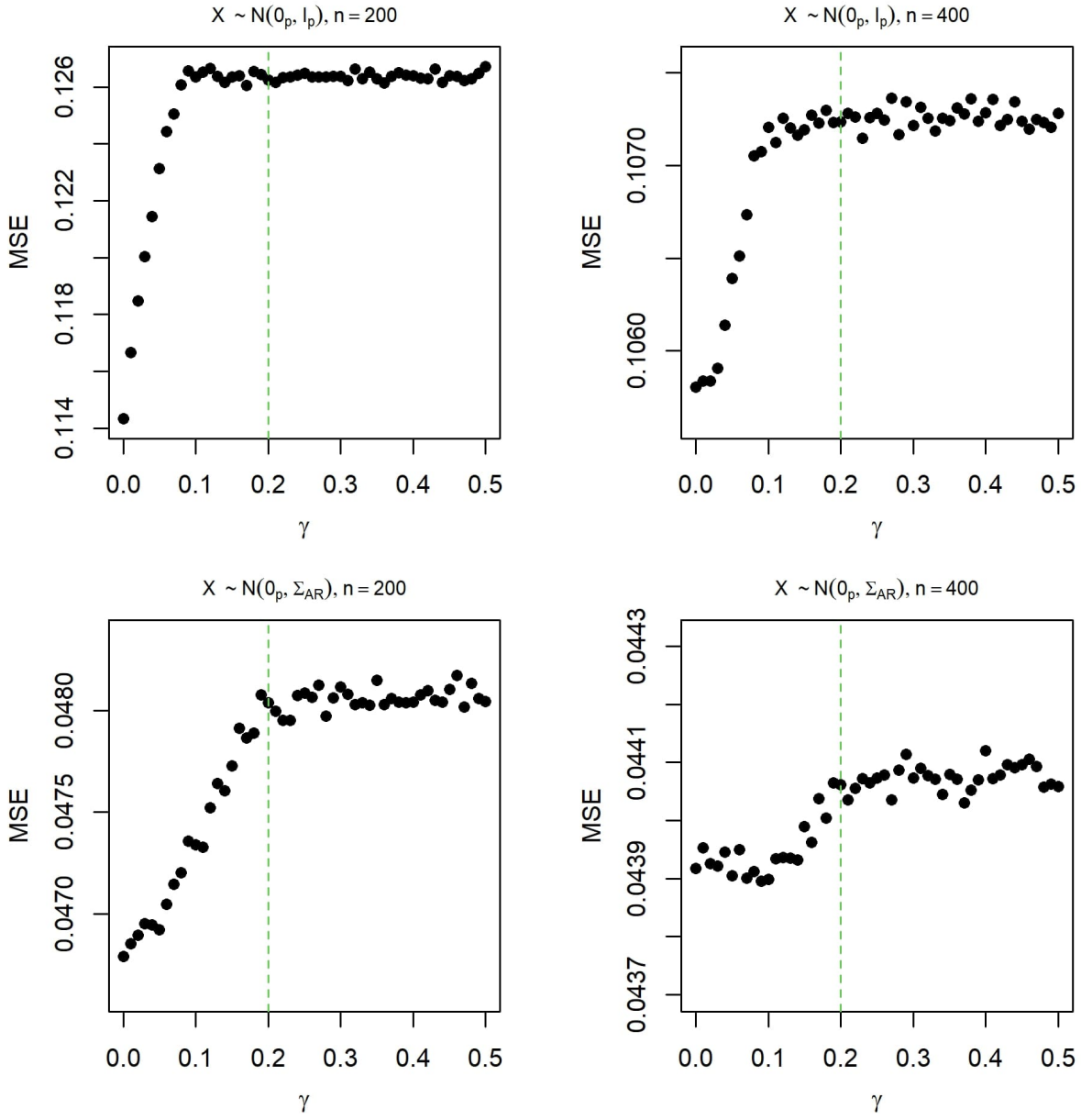


Figure 3.9. Erreur quadratique moyenne (MSE) de prédiction basée sur les 500 jeux de données pour l'exemple 3. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

La figure 3.9 nous permet de visualiser l'évolution de l'erreur quadratique moyenne de prédiction en fonction du paramètre d'ajustement. Cet exemple est plus difficile, puisqu'il y a 20 coefficients non nuls et 20 coefficients nuls. Nous observons que plus le paramètre γ est petit, plus le MSE est petit. Dans cet exemple, la procédure un-à-un sur le jeu de données

complet ($\gamma = 0$) semble être adéquate pour $n = 200$; pour $n = 400$, nous obtenons plutôt une valeur de γ_{min} près de 0,05. Les paramètres γ qui sont supérieurs à 0,1 ne semblent pas adéquats pour l'exemple 3.

Pour les trois exemples, nous remarquons que le critère du MSE nous mène à des conclusions intéressantes. Pour l'exemple 1, le γ minimum se situait autour de 0 ou 0,2 dépendamment du choix de la matrice de design générée. Pour l'exemple 2, nous avons des conclusions similaires à celles de l'exemple 1. Finalement, pour l'exemple 3, le paramètre γ idéal semble se situer dans l'intervalle [0; 0,1]. Nous discutons de ces résultats en détail dans la section 3.4.6.

3.4.5. Résultats des critères VP et FP

Nous étudions maintenant l'effet du paramètre d'ajustement γ sur la performance du Lasso Linéaire en termes de vrais positifs (VP) et de faux positifs (FP). Pour chacun des exemples $\boldsymbol{\beta}_1^*$, $\boldsymbol{\beta}_2^*$ et $\boldsymbol{\beta}_3^*$, nous présentons des figures illustrant l'évolution des courbes VP et FP en fonction du paramètre d'ajustement. Pour tracer les courbes, la moyenne des vrais positifs et des faux positifs des 500 jeux de données est utilisée. Nous voulons évidemment que la valeur de VP s'approche le plus possible du nombre de coefficients non nuls, tandis que la valeur de FP devrait plutôt s'approcher de 0.

Pour l'exemple 1, les résultats portant sur les critères VP et FP se trouvent dans la figure 3.10. Nous rappelons que l'exemple 1 comporte 3 coefficients non nuls et 5 coefficients nuls. Le Lasso Linéaire performe bien dans cet exemple; en effet, la valeur de VP est proche de 3 pour toutes les combinaisons et la valeur de FP est proche de 1. La valeur de FP a tendance à augmenter à partir de $\gamma = 0{,}2$. Nous notons également qu'en augmentant la taille échantillonnale, la valeur de FP diminue alors que celle de VP augmente, ce qui est désirable.

Pour l'exemple 2, les résultats portant sur les critères VP et FP se trouvent dans la figure 3.11. Nous rappelons que l'exemple 2 comporte 8 coefficients non nuls et aucun coefficient nul. Le Lasso Linéaire performe également bien dans cet exemple. Remarquons qu'il est logique que FP soit à 0, puisque nous n'avons que des coefficients non nuls. Pour $n = 50$, la valeur de VP est près de 6 pour tous les paramètres d'ajustement; pour $n = 100$, la valeur de FP s'approche de 8 pour tous les γ. Dans tous les cas, la valeur de VP augmente légèrement avec la valeur du paramètre d'ajustement γ. Finalement, tel qu'attendu, une augmentation du nombre d'observations mène à une valeur de VP plus élevée.

Pour l'exemple 3, les résultats portant sur les critères VP et FP se trouvent dans la figure 3.12. Nous rappelons que $\boldsymbol{\beta}_3^*$ a 20 coefficients non nuls et qu'il possède également 20 coefficients nuls. Cet exemple difficile représente un plus grand défi pour le Lasso Linéaire;

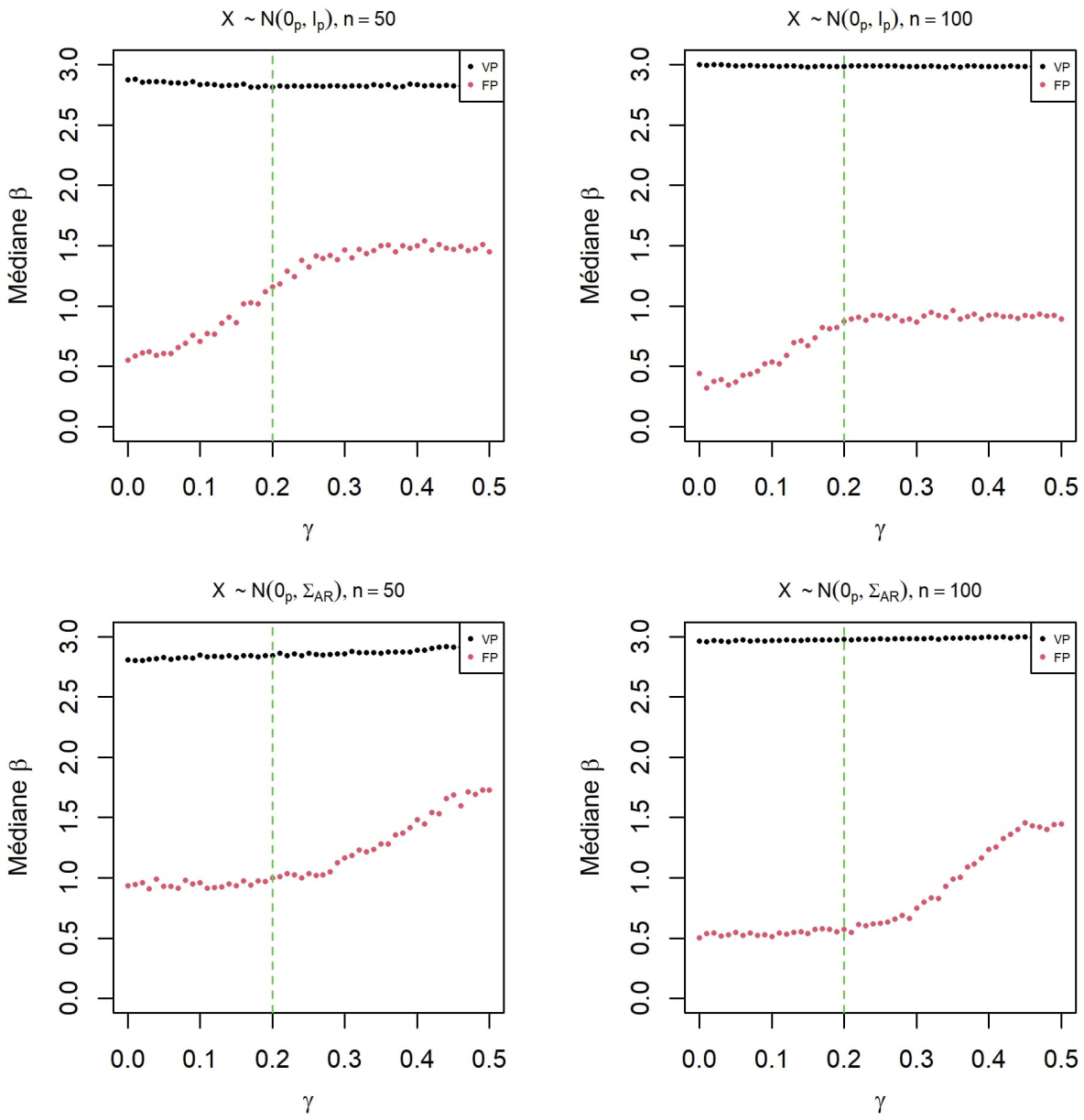


Figure 3.10. Moyennes des valeurs de vrais positifs (VP) et faux positifs (FP) basées sur les 500 jeux de données pour le coefficient $\boldsymbol{\beta}_1^*$. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

en effet, dans le cas de la matrice de design générée d'une normale avec composantes indépendantes, le nombre de faux positifs augmente plus rapidement que dans l'exemple 1. Par ailleurs, les valeurs de FP s'améliorent significativement lorsqu'on passe de $n = 200$ à $n = 400$. Nous notons également que lorsque γ augmente, la valeur de FP demeure relativement constante à partir de $\gamma = 0{,}1$. Nous remarquons qu'en général, les valeurs de VP semblent être très bonnes pour toutes les combinaisons de $\boldsymbol{X}^*$ et n.

Nous mentionnons tout de même que pour de grandes valeurs de γ, l'ordonnancement en fonction des corrélations c nous force parfois à conserver plus de prédicteurs que nécessaire. En effet, les prédicteurs possédant une valeur de c_j supérieure à certains prédicteurs importants finiront par être conservés dans le modèle. Ceci mène à une moyenne de VP qui est élevée, mais parallèlement à des valeurs de FP qui tendent à croître avec γ.

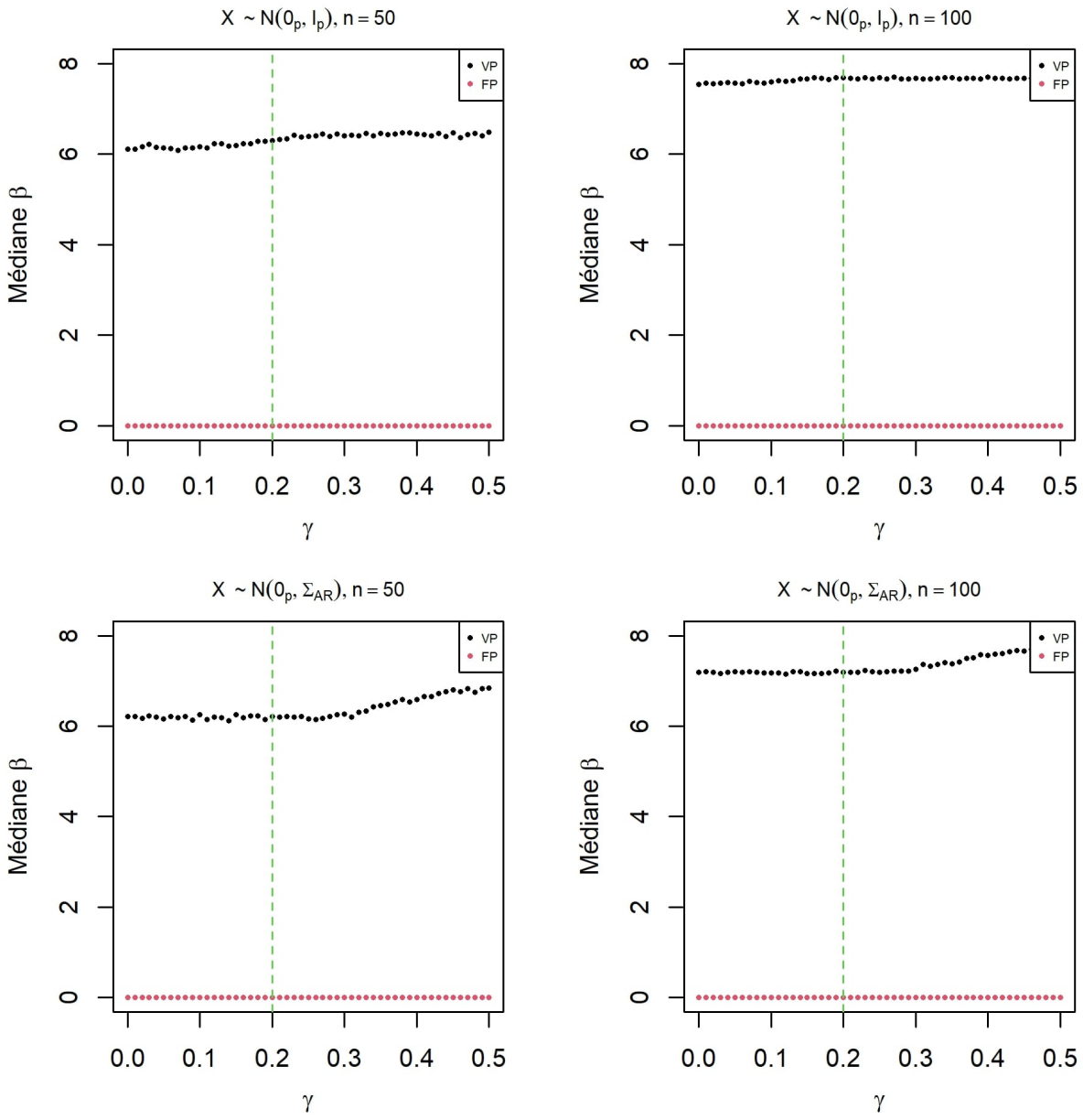


Figure 3.11. Moyennes des valeurs de vrais positifs (VP) et faux positifs (FP) basées sur les 500 jeux de données pour le coefficient $\boldsymbol{\beta}_2^*$. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

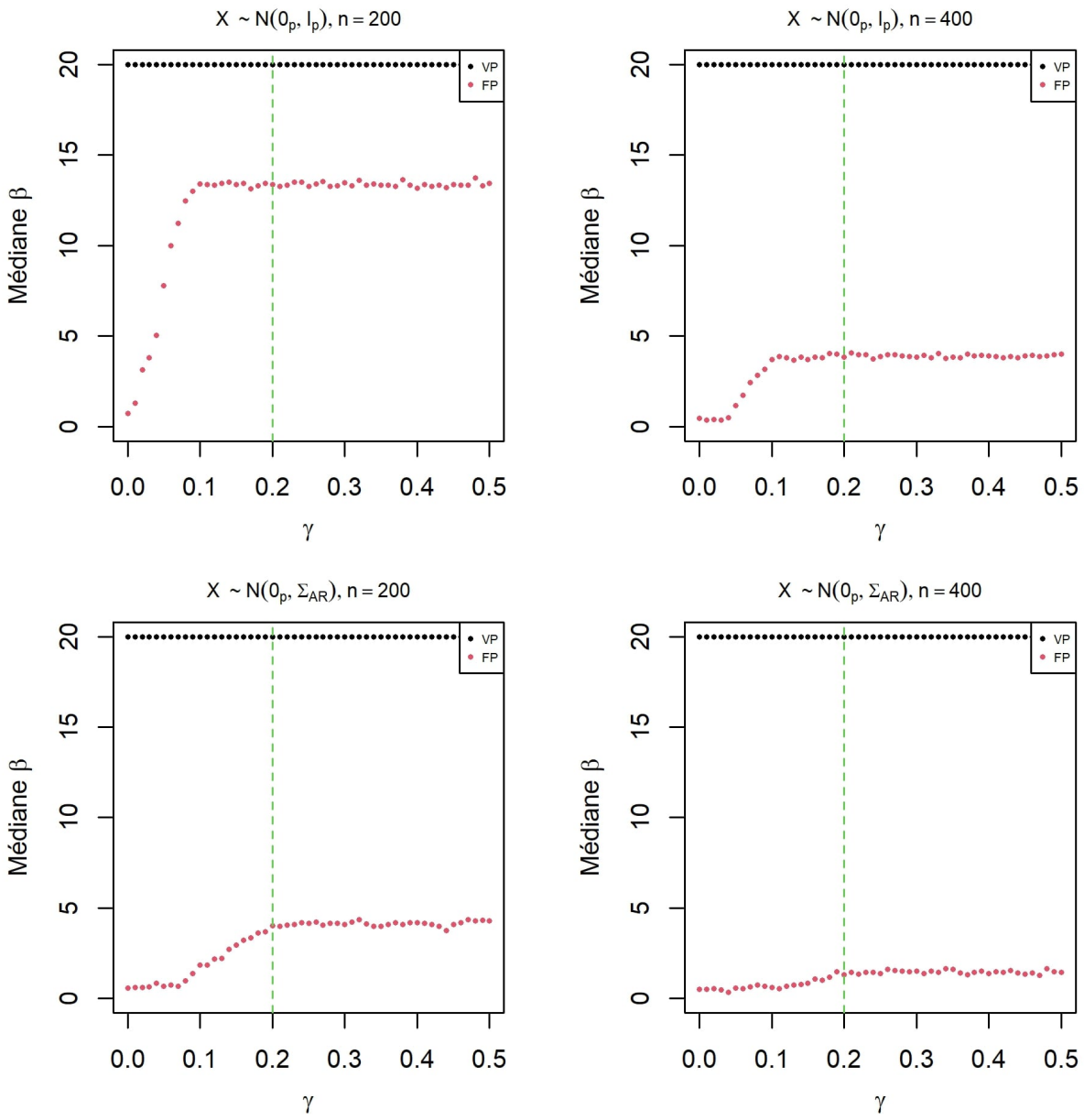


Figure 3.12. Moyennes des valeurs de vrais positifs (VP) et faux positifs (FP) basées sur les 500 jeux de données pour le coefficient β_3^*. La ligne pointillée verte indique le paramètre d'ajustement par défaut identifié par Fraser et Bédard (2022).

3.4.6. Interprétation des résultats

Dans les sections précédentes, des simulations ont été utilisées pour étudier l'optimalité du paramètre $\gamma = 0{,}2$ dans le contexte du Lasso Linéaire. Après avoir étudié différents exemples comportant différentes paramétrisations, nous pouvons conclure qu'empiriquement, le paramètre $\gamma = 0{,}2$ n'est pas optimal dans toutes les situations. En effet, nous avons observé que les choix $\gamma = 0$ et $\gamma = 0{,}1$ sont parfois supérieurs à cet ajustement par défaut.

Le temps de calcul était largement en faveur de $\gamma \to 1$, ce qui était facilement observable sur les figures 3.5 et 3.6. À l'inverse, lorsque $\gamma \to 0$, la procédure un-à-un prenait beaucoup de temps, et ce pour tous les exemples. De plus, nous remarquons qu'à partir de $\gamma = 0{,}2$, le temps de calcul devient pratiquement constant sur toutes les figures.

Pour les exemples 1 et 3, nous avons observé des similarités en termes de MSE. Lorsque le paramètre d'ajustement augmente, les courbes ont également tendance à augmenter, pour ensuite se stabiliser. Dans l'exemple 1, nous avons obtenu une valeur de γ_{min} près de 0 pour la matrice $\mathcal{MN}(0_p, I_p)$, ce qui a également été observé dans l'exemple 3. La valeur de γ_{min} est plutôt à 0,2 dans l'exemple 1 pour la matrice $\mathcal{MN}(0_p, \Sigma_{AR})$. Dans ce même exemple, la valeur de FP augmente à partir de $\gamma \approx 0{,}2$ alors que dans l'exemple 3, c'est plutôt à partir de $\gamma \approx 0{,}1$.

À travers notre étude, nous avons remarqué que le Lasso Linéaire arrive pratiquement toujours à inclure les variables pertinentes dans le modèle final, quelle que soit la valeur de γ utilisée. Toutefois, les contextes les plus difficiles à gérer pour le Lasso surviennent lorsqu'un prédicteur important n'affiche qu'une faible corrélation avec la variable réponse (ou à tout le moins une corrélation moindre que celle d'autres prédicteurs jugés moins significatifs). Ce genre de situation peut se présenter, entre autres, lorsqu'il y a beaucoup de bruit dans le jeu de données. Nous remarquons qu'avec $\sigma = 3$, nous introduisons une erreur relativement importante comparativement à la taille des coefficients $\boldsymbol{\beta}$. Dans ce genre de contexte, le Lasso Linéaire aura tendance à surajuster le modèle. En effet, plus γ est grand, plus nombreux sont les prédicteurs qui sont ordonnés en fonction de leur corrélation avec la variable réponse. Ceci implique que si on veut conserver un prédicteur significatif dont la corrélation avec la réponse est de c, nous devrons automatiquement conserver tous les prédicteurs possédant une corrélation supérieure à cette valeur dans le modèle. Dans certaines situations, cette façon de procéder peut mener à un surajustement du modèle.

C'est ce qu'on observe également dans l'exemple du diabète à la section 2.7.3. Dans cet exemple, la variable `SEXE` a la plus petite corrélation c_j, mais la validation croisée nous indique que cette variable devrait être incluse dans le modèle final. La taille du modèle, $\mathcal{T}_{min}$, est donc égale au nombre de prédicteurs disponibles. Ceci est bien sûr une situation extrême, mais qui est répétée de la même façon à l'exemple 3. Si la corrélation d'une variable avec $\mathbf{y}$ est petite, mais que la variable est importante, la validation croisée du Lasso Linéaire conservera cette variable, ce qui nous assure une bonne valeur de VP, mais également une valeur de FP qui s'éloigne de 0 lorsque γ croît. De plus, lorsque n n'est pas assez grand comparativement à p, des termes de corrélations dans c pourraient être artificiellement élevés; par conséquent, le modèle choisi par le Lasso Linéaire avec de grandes valeurs de γ aura tendance à être surajusté (à garder trop de prédicteurs, pour éviter de rejeter des prédicteurs importants qui pourraient apparaître comme étant moins corrélés avec $\mathbf{y}$ que d'autres). Dans ces situations,

des modèles plus parcimonieux peuvent être obtenus en utilisant de petites valeurs de γ et, à la limite, $\gamma = 0$ avec la procédure un-à-un. C'est exactement ce que nous observons dans l'exemple 3. Par contre, comme dans n'importe quelle méthode, si nous voulons choisir $\gamma = 0$, nous devons payer un prix en termes de temps de calcul.

À la lumière de nos résultats, le Lasso Linéaire offre une bonne performance lorsque $n >> p$. En effet, nous concluons que $\gamma = 0{,}1$ semble constituer un choix prudent qui est adapté à une grande variété d'exemples. Lorsque nous choisissons $\gamma = 0$, nous avons un temps de calcul trop grand, mais une meilleure précision; lorsque $\gamma = 0{,}2$, nous sommes plutôt contraints à accepter plus d'erreurs, mais un temps de calcul beaucoup moins grand. Dans le chapitre 4, nous continuons à comparer différents γ, en incluant cette fois la règle du 1se pour le Lasso Linéaire.

3.5. Implémentation

L'algorithme du Lasso Linéaire est disponible sur le site de GitHub et ouvert au grand public. Nous rappelons que l'algorithme 2.7.2 est une version optimisée du code introduit par Fraser et Bédard (2022). Nous avons également inclus les améliorations mentionnées au début du chapitre 3 (règle du 1se et variables catégorielles) dans la fonction du Lasso Linéaire. Pour accéder à cette fonction, il faut installer le *package* `linlasso` sur le progiciel **R**[1] :

```
devtools::install_github("yanwatts/linlasso")
```

Le *package* `linlasso` contient plusieurs fonctions, mais la fonction qui nous intéresse est celle qui performe le Lasso Linéaire (`LL`) en entier. Nous reprenons l'exemple du diabète introduit dans la section 2.7.3. Nous spécifions la mesure quantitative de la progression de la maladie dans le vecteur **y** et nous plaçons les 10 variables explicatives dans la matrice `x`. Par la suite, nous spécifions le paramètre d'ajustement **`gamma`**, le nombre de plis `K` et le nombre de cycles de validation croisée `L`. L'argument **`plot`** nous permet d'inclure le graphique des variables rejetées du modèle. En utilisant les mêmes paramètres (γ, K, L) que dans la section 2.7.3, nous avons le code suivant

```
model.LL = LL(y = diabetes[,11], x = diabetes[,-11], gamma = 0.2, K
   = 13, L = 50, plot = F)
```

Le vecteur des corrélations positives c, tel qu'observé dans le tableau 2.1, nous est retourné en ordre décroissant lorsque l'argument **`c.pos`** est spécifié :

```
model.LL$c.pos
  BMI     S5     BP     S4     S3     S6     S1    AGE     S2    SEX
```

[1]Nous supposons que le progiciel de **`devtools`** Wickham *et al.* (2022) a été préalablement installé

```
0.586 0.566 0.441 0.430 0.395 0.382 0.212 0.188 0.174 0.043
```

Nous savons essentiellement qu'avec $\gamma = 0{,}2$, les variables `SEX`, `S2`, `AGE` seront les premières à être écartées du modèle (lors de la première étape) et, par la suite, nous exécutons la procédure un-à-un sur les variables restantes. L'erreur quadratique moyenne, les écarts-types et les erreurs standards nous sont retournés dans la table suivante (similaire à la table 2.2),

```
model$table.MSE
```

Rejected variables in order	Length	MSE.CV	SD.CV	SE.CV
SEX	10	0.5055	0.1078	0.0299
S2	9	0.5213	0.1074	0.0298
AGE	8	0.5199	0.1073	0.0298
S3	7	0.5181	0.1063	0.0295
S6	6	0.5194	0.1062	0.0294
S4	5	0.5175	0.1064	0.0295
S1	4	0.5212	0.1084	0.0301
BP	3	0.5290	0.1082	0.0300
S5	2	0.5458	0.1113	0.0309
BMI	1	0.6821	0.1451	0.0402

et ce pour toutes les tailles de modèles (10 à 1). La présentation avec `table.MSE` est toutefois différente de celle dans le tableau 2.2. Lorsqu'il y a plusieurs variables explicatives, la première colonne est très large, ce qui nuit à l'interprétation du modèle. Toutefois, la première colonne de `table.MSE` est cohérente avec la figure 3.13. Le coefficient des moindres carrées ordinaires $\hat{\boldsymbol{\beta}}^{(MCO)}$, en utilisant la taille de modèle $\mathcal{T}_{min}$, peut être trouvé à l'aide de `beta.min`,

```
model$beta.min
```

	[,1]
AGE	0.009746087
SEX	20.801686649
BMI	5.417986348
BP	0.961537161
S1	1.635507013
S2	-1.604358783
S3	-3.536258970
S4	-7.741258450
S5	-3.598309289
S6	0.080981765

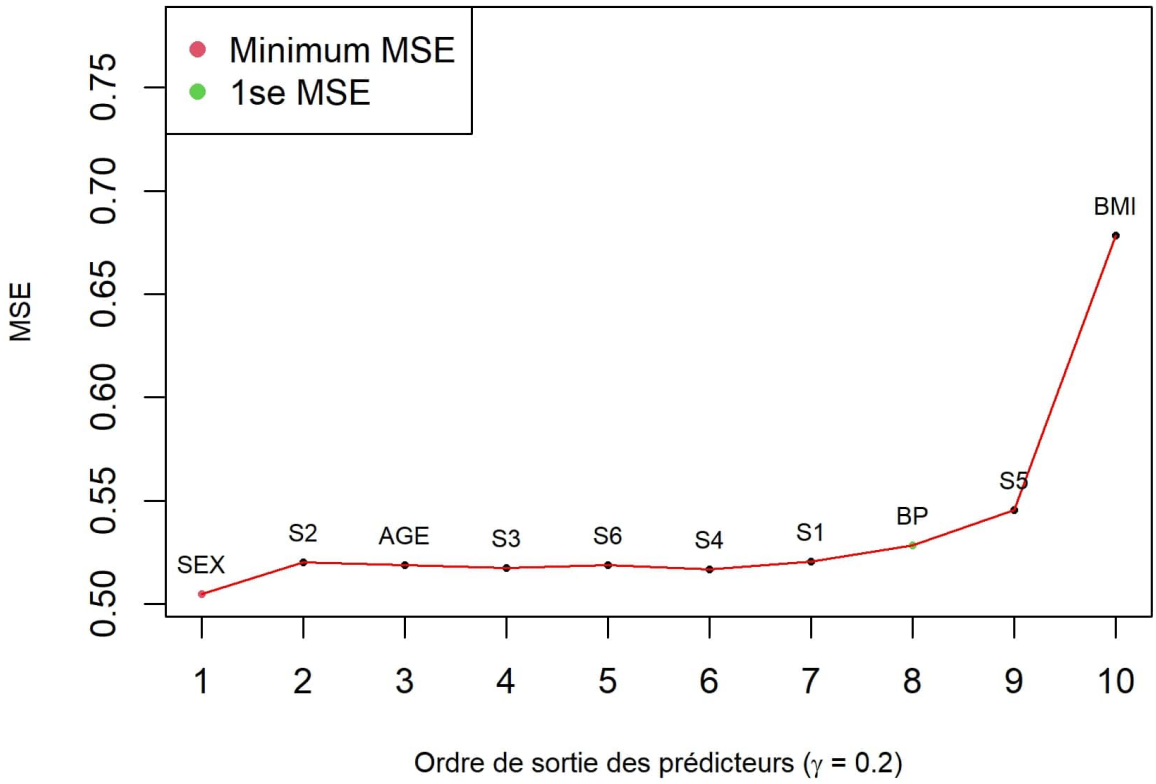


Figure 3.13. Présentation du graphique avec l'argument **`plot`** dans la fonction `LL`

et, de façon similaire, nous avons accès au coefficient des moindres carrés ordinaires basé sur la taille $\mathcal{T}_{1se}$ avec l'argument **`beta.1se`**,

```
model$beta.1se
             [,1]
AGE   0.000000000
SEX   0.000000000
BMI   5.281323909
BP   -0.005810611
S1    0.000000000
S2    0.000000000
S3    0.000000000
S4    0.000000000
S5    3.642028932
```

```
S6    0.000000000
```

L'implémentation complète peut être étudiée en profondeur à l'adresse suivante : https://github.com/yanwatts/linlasso.

Chapitre 4

Comparaison à des méthodes alternatives

L'algorithme du Lasso Linéaire est optimisé et prêt à être utilisé dans divers contextes de régression linéaire. Nous avons conclu dans le chapitre 3 que γ près de 0 était une bonne valeur pour les jeux de données en petite dimension, malgré le temps de calcul beaucoup plus grand. Toutefois, pour les jeux de données en grandes dimension $p > n$, nous utilisons les trois étapes décrites à la section 3.3.

Il existe, dans la littérature, d'autres algorithmes nous permettant d'identifier un sous-modèle et, par la suite, d'en fournir l'estimation $\hat{\boldsymbol{\beta}}$. Il serait alors intéressant de comparer la performance du Lasso Lineaire à ces algorithmes compétiteurs.

4.1. Méthodes alternatives

Dans la section 1.3, la méthode du Ridge est présentée comme une méthode intuitive et simple qui rétrécit les coefficients $\hat{\beta}_j^{(MCO)}$ lorsqu'il y a présence de multicolinéarité dans la matrice de design $\boldsymbol{X}$. Malgré ses vertus, la régression Ridge n'effectue pas de sélection de modèle, ce qui nous oblige à prendre toutes les variables dans le modèle final. À l'inverse, le Lasso régulier nous permet de faire de la sélection de variables, en plus de rétrécir les coefficients $\hat{\beta}_j^{(MCO)}$. Par contre, le Lasso régulier ne possède pas de solution analytique, contrairement au Ridge (voir la proposition 1.3.3). Le Ridge et le Lasso régulier sont des méthodes spécifiques faisant partie de la grande famille des méthodes de régression pénalisée. Les méthodes qui sont présentées dans cette section font également partie de cette famille.

4.1.1. Filet élastique

Dans la section 2.1, nous avons mentionné que le Lasso régulier a deux problèmes : il sélectionne au plus n variables lorsque $p > n$ et il a de la difficulté à sélectionner un prédicteur parmi un groupe de prédicteurs fortement corrélés. Ces limitations sont également discutées par les auteurs Zou et Hastie (2005); ceux-ci proposent alors la méthode du Filet élastique dans l'espoir de régler ces problèmes. Cette méthode est basée sur l'union des pénalités du

Ridge et Lasso et vise à tirer le meilleur des deux approches. La pénalité du Filet élastique, $\mathcal{P}_\alpha(\boldsymbol{\beta})$, introduit un paramètre $\alpha \in [0,1]$ servant à pondérer les pénalités du Lasso et du Ridge. En supposant toujours que le vecteur réponse et les vecteurs de variables explicatives sont centrés et standardisés, alors la fonction objective à optimiser pour le Filet élastique satisfait

$$\hat{\boldsymbol{\beta}}^{(FE)} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \mathcal{P}_\alpha(\boldsymbol{\beta}) \right\}, \tag{4.1.1}$$

où $\mathcal{P}_\alpha(\boldsymbol{\beta})$ est une fonction de pénalité et $\lambda \geq 0$ est un paramètre d'ajustement. Plus précisément, la fonction de pénalité prend la forme suivante

$$\mathcal{P}_\alpha(\boldsymbol{\beta}) = (1-\alpha)\frac{1}{2}\|\beta\|^2 + \alpha\|\beta\|_1 = \sum_{j=1}^{p} \left[\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j| \right]; \tag{4.1.2}$$

celle-ci est alors une généralisation des fonctions de pénalité Ridge ($\alpha = 0$) et du Lasso régulier ($\alpha = 1$). Lorsque le paramètre $\alpha \in (0,1)$, nous retrouvons la méthode du Filet élastique. Ce compromis nous permet d'exécuter une sélection de variables avec le Lasso, tout en rétrécissant les coefficients avec le Ridge. Lorsque α passe de 0 à 1, nous observons généralement un plus grand nombre de coefficients mis à 0.

Le choix du paramètre α dépend ultimement du jeu de données étudié. En effet, on peut penser qu'un α de 0,5 nous permettrait de profiter simultanément des avantages offerts par le Ridge et le Lasso. Cependant, lorsque p est grand, il sera avantageux d'utiliser un paramètre $\alpha \to 1$ afin d'éliminer un grand nombre de prédicteurs. À l'inverse, lorsque nous soupçonnons que les prédicteurs sont très corrélés, il devient préférable d'utiliser un paramètre $\alpha \to 0$. À des fins pratiques, il est habituellement souhaitable de considérer une suite de paramètres $\alpha \in (0,1)$; le paramètre α est par la suite choisi par validation croisée, en sélectionnant celui qui mène à la plus petite erreur quadratique moyenne. Rappelons toutefois qu'il faut également choisir le paramètre λ par validation croisée. Nous cherchons alors la meilleure combinaison (α, λ) pour un jeu de données spécifique, ce qui peut s'avérer computationnellement coûteux comparativement au Lasso et au Ridge, où il n'y a qu'un paramètre (λ) à choisir par validation croisée.

Les auteurs Bühlmann et Van De Geer (2013) et Zou et Hastie (2005) se penchent sur la supériorité de la méthode du Filet élastique. Contrairement au Lasso, le Filet élastique est capable de choisir plus de n variables lorsque $p > n$. Les résultats numériques obtenus avec cette méthode semblent aussi mener à une erreur quadratique moyenne plus faible dans le cas de prédicteurs corrélés lorsque comparé au Lasso. De plus, en termes de sélection de variables, le Filet élastique identifie correctement plus de variables que le Lasso. Pour ces raisons, nous allons comparer la performance du Lasso Linéaire à celle du Filet élastique.

4.1.2. Lasso adaptatif

Dans le Ridge, nous avons observé que lorsque $\lambda \to \infty$, le biais augmente. Ceci est également vrai pour le Lasso; le biais est contrôlé par la valeur du paramètre λ. En ajoutant un poids w_j au paramètre λ, nous trouvons alors $\lambda_j = w_j\lambda$, pour $j = 1, \ldots, p$. La motivation derrière cette pondération est d'assigner de petits poids aux coefficients ayant de grandes valeurs, ce qui permet de réduire le biais du Lasso. Cet ajout mène à une meilleure estimation des coefficients pour les prédicteurs, ce qui à son tour permet de faire une meilleure sélection de variables. Pour ces raisons, Zou (2006) introduit le Lasso adaptatif, qui possède les propriétés oracles : une sélection cohérente des prédicteurs et la normalité asymptotique. En d'autres mots, les propriétés oracles signifient que le modèle choisit aussi bien les variables que si le vrai modèle était connu.

La propriété de la sélection cohérente est tout particulièrement d'intérêt pour nous, car elle stipule que le modèle sélectionné choisit les coefficients non nuls avec probabilité 1. Zou (2006) montre que le Lasso adaptatif effectue une sélection cohérente, ce qui n'est pas le cas du Lasso. Pour satisfaire cette propriété de sélection cohérente, il est important que l'estimateur utilisé dans le Lasso adaptatif converge en probabilité vers la vraie valeur du paramètre lorsque la taille de l'échantillon n tend vers l'infini. Ainsi, les coefficients de régression utilisés dans le Lasso adaptatif peuvent être les coefficients du Lasso ou ceux du Ridge. Le poids utilisé par Zou (2006) est

$$w_j = \frac{1}{|\hat{\beta}_j|^{\gamma_{AL}}}, \tag{4.1.3}$$

où $\gamma_{AL} > 0$ est un paramètre d'ajustement et $\hat{\beta}_j \in \{\hat{\beta}_j^{(MCO)}, \hat{\beta}_j^{(R)}\}$. Pour des raisons de simplicité, nous utilisons le coefficient MCO dans le restant du mémoire pour les poids w_j. Nous remarquons alors que le Lasso adaptatif se programme en deux étapes : nous estimons tout d'abord $\hat{\boldsymbol{\beta}}^{(MCO)}$ à l'aide du jeu de données complet et, par la suite, nous trouvons

$$\hat{\boldsymbol{\beta}}^{(LA)} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} w_j |\beta_j| \right\}, \ j = 1, \ldots, p, \tag{4.1.4}$$

où w_j est le poids stipulé dans l'équation (4.1.3). Des valeurs typiques pour le paramètre γ_{AL} sont 0,5, 1 et 2. Il est également possible de procéder par validation croisée pour trouver le paramètre d'ajustement optimal γ_{AL}. Par contre, cette méthode ne peut être utilisée lorsque $p > n$ car elle nécessite, dans un premier temps, l'estimation des coefficients MCO pour les prédicteurs. En dépit de ceci, le Lasso adaptatif demeure une alternative très intéressante au Lasso régulier puisqu'il a les propriétés oracles. Il sera intéressant de comparer cette variante du Lasso avec l'algorithme du Lasso Linéaire.

4.1.3. Algorithmes de pénalisation (SCAD + MCP)

Nous savons qu'en utilisant une grande valeur de λ dans une régression pénalisée, nous pouvons rétrécir les coefficients, surtout ceux de grande taille. Par contre, en augmentant λ, nous augmentons également le biais d'une méthode comme celle du Lasso. Ainsi, le Lasso adaptatif permet de réduire le biais du Lasso régulier par l'ajout d'un certain poids w_j, ce qui permet de gérer les coefficients de grande taille et de réduire le biais. Rappelons que le Lasso adaptatif s'effectue en deux étapes : tout d'abord par le calcul d'un coefficient initial comme MCO ou Ridge, qui est par la suite inséré dans l'équation à minimiser (4.1.4) pour trouver $\hat{\boldsymbol{\beta}}^{(LA)}$. Cette approche est bien sûr inutilisable en grandes dimension. Une autre façon de pénaliser les coefficients de grande taille tout en palliant à ce problème de grandes dimensions est de se tourner vers les méthodes suivantes : le *Smoothly Clipped Absolute Deviations* (SCAD) de Fan et Li (2001b) et le *Minimax Concave Penalty* (MCP) de Zhang (2010).

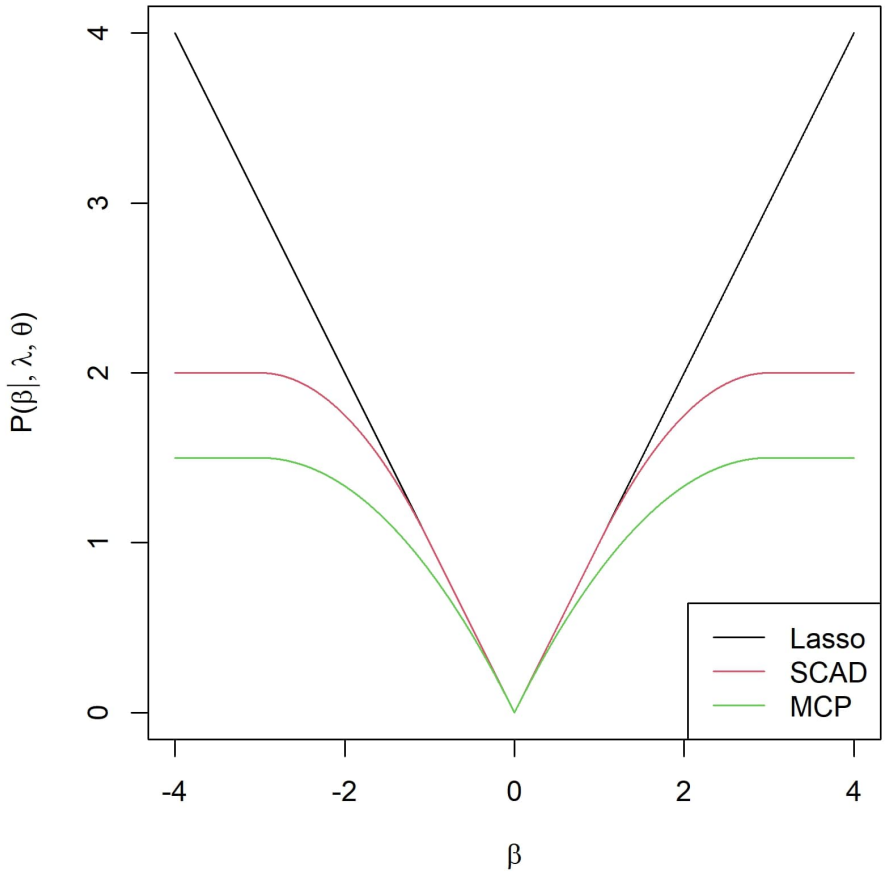


Figure 4.1. Pénalités du Lasso, SCAD et MCP pour $\lambda = 1$ et $\theta = 3$

Contrairement au Lasso, les pénalités du SCAD et MCP prennent différentes formes en fonction de la valeur de λ; spécifiquement, nous avons $\mathcal{P}(\boldsymbol{\beta}|\lambda,\theta) \neq \lambda\mathcal{P}(\boldsymbol{\beta})$, où $\mathcal{P}(\boldsymbol{\beta}|\lambda,\theta)$ est une pénalité concave et le paramètre d'ajustement θ contrôle la concavité de la pénalité. La pénalité SCAD de Fan et Li (2001b) est la suivante

$$\mathcal{P}(\boldsymbol{\beta}|\lambda,\theta) = \begin{cases} \lambda|\boldsymbol{\beta}| & \text{si } |\boldsymbol{\beta}| \leq \lambda, \\ \dfrac{2\theta\lambda|\boldsymbol{\beta}| - \boldsymbol{\beta}^2 - \lambda^2}{2(\theta-1)} & \text{si } \lambda < |\boldsymbol{\beta}| < \theta\lambda, \\ \dfrac{\lambda^2(\theta+1)}{2} & \text{si } |\boldsymbol{\beta}| \geq \theta\lambda, \end{cases}$$

où $\gamma > 2$. Nous remarquons évidemment que la pénalité SCAD prend la forme de la pénalité Lasso jusqu'à ce que $|\boldsymbol{\beta}| = \lambda$. Par la suite, la pénalité SCAD devient une fonction quadratique jusqu'à $|\boldsymbol{\beta}| = \theta\lambda$ et devient finalement une fonction constante pour $|\boldsymbol{\beta}| > \theta\lambda$. La dérivée de la pénalité du SCAD peut s'écrire comme

$$\mathcal{P}'(\boldsymbol{\beta},\lambda,\theta) = \begin{cases} \lambda & \text{si } |\boldsymbol{\beta}| \leq \lambda, \\ \dfrac{\theta\lambda - |\boldsymbol{\beta}|}{\theta-1} & \text{si } \lambda < |\boldsymbol{\beta}| < \theta\lambda, \\ 0 & \text{si } |\boldsymbol{\beta}| \geq \theta\lambda. \end{cases}$$

Nous précisons que la dérivée n'existe pas à $\boldsymbol{\beta} = 0$. Cette dérivée illustre bien que la pénalité s'atténue pour les coefficients de grande taille, ce qui permet de contrôler le biais.

La pénalité MCP de Zhang (2010) satisfait

$$\mathcal{P}(\boldsymbol{\beta}|\lambda,\theta) = \begin{cases} \lambda|\boldsymbol{\beta}| - \dfrac{\boldsymbol{\beta}^2}{2\theta} & \text{si } |\boldsymbol{\beta}| \leq \theta\lambda, \\ \dfrac{\theta\lambda^2}{2} & \text{si } |\boldsymbol{\beta}| > \theta\lambda, \end{cases}$$

où $\theta > 1$. La dérivée de la pénalité MCP peut également s'écrire comme

$$\mathcal{P}'(\boldsymbol{\beta},\lambda,\theta) = \begin{cases} \left(\lambda - \dfrac{|\boldsymbol{\beta}|}{\theta}\right) sign(\boldsymbol{\beta}) & \text{si } |\boldsymbol{\beta}| \leq \theta\lambda, \\ 0 & \text{si } |\boldsymbol{\beta}| > \theta\lambda. \end{cases}$$

Nous précisons que la dérivée n'existe pas à $\boldsymbol{\beta} = 0$. La pénalité MCP, comme celle du SCAD, débute avec la pénalité du Lasso pour des coefficients de petite taille. Contrairement au SCAD, la pénalité MCP atténue le taux de pénalisation directement. Ces conclusions peuvent également être observées sur la figure 4.1. Les auteurs nous rappellent que SCAD et MCP peuvent être utilisés en grande dimension et possèdent également les propriétés oracles, comme le Lasso adaptatif. Ainsi, ces deux pénalités seraient de bons compétiteurs à inclure dans notre étude de performance.

4.2. Simulations en petites dimensions

Afin de tester les méthodes en petites dimensions, nous devons construire des simulations à l'aide du progiciel **R**. Nous utilisons des critères de performance similaires à ceux du chapitre 3 nous permettant de facilement comparer le Lasso Linéaire aux méthodes alternatives de la section 4.1. Comme à l'équation (3.4.1), nous devons générer aléatoirement la matrice de design $\boldsymbol{X}^*$, ainsi que $\boldsymbol{\varepsilon} \sim \mathcal{N}(0_n, \sigma^2 I_n)$. Contrairement au chapitre 3, la matrice de design $\boldsymbol{X}^*$ sera uniquement générée d'une normale multivariée $\mathcal{MN}(0_p, \Sigma_{AR})$ sauf pour l'exemple 3. Le vrai vecteur de coefficients $\boldsymbol{\beta}^*$ dépend de l'exemple utilisé.

4.2.1. Les exemples de simulation

Nous sommes confrontés à des exemples similaires à ceux de la section 3.4.1. Dans chaque exemple, nos données sont simulées et des modèles sont par la suite ajustés; par la suite, nous notons certains critères de performance. Voici le détail pour les différents scénarios. Le paramètre σ est fixé à 3 pour les trois exemples et le nombre d'observations est de $n = 100$ pour les exemples 1 et 2 et de $n = 400$ pour l'exemple 3.

- Exemple 1 : Le vecteur de coefficients $\boldsymbol{\beta}^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$ sera testé, un exemple qui a également été étudié dans les simulations du chapitre 3. Différents paramètres ρ sont utilisés pour Σ_{AR}, soit $\rho \in \{0{,}15; 0{,}5; 0{,}9\}$, ce qui dénote respectivement une petite, moyenne et grande corrélation entre les prédicteurs.
- Exemple 2 : Le vecteur de coefficients $\boldsymbol{\beta}^* = (10; 5; 2; 1; 0{,}5; 0; 0; 0)^\top$ est un exemple contenant des coefficients de grande, moyenne et petite tailles. Le paramètre ρ est fixé à 0,5 pour cet exemple.
- Exemple 3 : Cet exemple est tiré de Zou et Hastie (2005), où le vecteur de coefficients est $\boldsymbol{\beta}^* = (\underbrace{3, \ldots, 3}_{15}, \underbrace{0, \ldots, 0}_{25})^\top$. Contrairement aux exemples précédents, la matrice de design $\boldsymbol{X}^* = (\mathbf{x}_1^*, \ldots, \mathbf{x}_{40}^*)$ sera générée comme ceci
  - $\mathbf{x}_j^* = Z_1 + \varepsilon_j^x$, pour $j = 1, \ldots, 5$
  - $\mathbf{x}_j^* = Z_2 + \varepsilon_j^x$, pour $j = 11, \ldots, 10$
  - $\mathbf{x}_j^* = Z_3 + \varepsilon_j^x$, pour $j = 11, \ldots, 15$
  - $\mathbf{x}_j^* \sim \mathcal{MN}(0_n, I_{n\times n})$ de taille n, pour $j = 16, \ldots, 40$

  où Z_1, Z_2, Z_3 sont tirés d'une normale standard $\mathcal{MN}(0_n, I_{n\times n})$ de taille n et les termes ε_j^x sont indépendants et identiquement distribués selon une normale $\mathcal{MN}(0_n,\ 0{,}01 I_{n\times n})$ pour les prédicteurs $j = 1, \ldots, 15$. Ce scénario nous intéresse puisqu'il crée trois groupes et une méthode se démarquera si elle est capable de choisir correctement les coefficients nuls et non nuls malgré les groupes de prédicteurs corrélés.

Il y aura 500 jeux de données simulés comme au chapitre 3. Les 500 jeux de données simulés seront centrés et standardisés comme aux équations (1.0.1) et (1.0.2). Après cette étape, nous ajustons les méthodes suivantes : la méthode séquentielle, le Ridge, le Lasso régulier, le Filet élastique, le Lasso adaptatif, le SCAD, le MCP et le Lasso Linéaire. Une fois les coefficients $\hat{\boldsymbol{\beta}}$ calculés pour chacune des méthodes, nous calculons les critères de performance nous permettant de discerner la meilleure méthode.

Chacune de ces méthodes effectue une validation croisée à l'interne. Pour chacune de ces méthodes nous notons le MSE associé au modèle final choisi. Des diagrammes à boîtes afficheront les 500 MSE pour chacune de ces méthodes. La méthode ayant la plus petite erreur quadratique moyenne de prédiction se démarquera comme étant la meilleure, nous permettant de juger la capacité prédictive d'une certaine méthode.

Nous procédons autrement pour vérifier si les coefficients sont bien classifiés pour une certaine méthode. Ainsi, nous calculons le pourcentage de zéros :

$$\text{Correct} = \frac{100\%}{p - p_0}\left(\sum_{k=1}^{500}\sum_{j=1}^{p}\mathbb{1}\left(\hat{\beta}_{j(k)} = 0\right) \times \mathbb{1}\left(\beta^*_{j(k)} = 0\right)\right)$$

et

$$\text{Incorrect} = \frac{100\%}{p_0}\left(\sum_{k=1}^{500}\sum_{j=1}^{p}\mathbb{1}\left(\hat{\beta}_{j(k)} = 0\right) \times \mathbb{1}\left(\beta^*_{j(k)} \neq 0\right)\right)$$

où p_0 est le nombre de coefficients non nuls. Pour les trois exemples nous avons respectivement $p_0 = 3, 5, 15$. Le pourcentage de zéros corrects devrait se rapprocher de 100% et, à l'inverse, le pourcentage de zéros incorrects devrait s'approcher de 0%. Les trois critères explicités ci-dessus (MSE, Correct, Incorrect) nous permettront de vérifier quelle est la méthode la plus adaptée dans chacun des exemples.

4.2.2. Computation des méthodes

L'implémentation des méthodes présentées est facilitée grâce à l'existence de *packages* sur le progiciel **R**, ce qui nous permettra de fournir facilement des estimés $\hat{\boldsymbol{\beta}}$. Les librairies et fonctions utilisées pour chacune des méthodes se trouvent dans le tableau 4.1.

Nous utilisons la méthode de sélection séquentielle pour la régression linéaire, qui est une méthode traditionnelle pour choisir un modèle et qui est discutée dans la section 1.2.1. Pour arriver à nos fins, nous utilisons la fonction `stepAIC` dans la librairie `MASS`, qui choisit le modèle final ayant le plus petit AIC.

La librairie `glmnet` sur le progiciel **R** nous permet d'exécuter le Ridge, le Lasso, le Filet élastique et le Lasso adaptatif. Dans les cas du Lasso et du Ridge, il y a uniquement le paramètre λ à trouver par validation croisée, puisque $\alpha = 0$ (Ridge) et $\alpha = 1$ (Lasso). Pour la méthode du Filet élastique, nous testons une suite de 9 valeurs $\alpha \in \{0{,}1; 0{,}2; \ldots; 0{,}9\}$ et

Tableau 4.1. Libraries et fonctions sur le progiciel **R** pour les différentes méthodes.

Méthodes	Librarie	Fonction	Coefficient avec la règle du 1se	Autres paramètres*
Régression séquentielle (AIC (S))	MASS	stepAIC	-	-
Ridge	glmnet	cv.glmnet	Non	-
Lasso	glmnet	cv.glmnet	Oui	-
Filet élastique (Enet)	glmnet	cv.glmnet	Oui	$\alpha \in \{0{,}1; 0{,}2; \ldots; 0{,}9\}$
Lasso adaptatif (AL)	glmnet	cv.glmnet	Oui	$\gamma_{AL} \in \{0{,}5; 1; 2\}$
SCAD	ncvreg	cv.ncvreg	Non	$\theta = 3{,}7$
MCP	ncvreg	cv.ncvreg	Non	$\theta = 3$
Lasso Linéaire (LL)	linlasso	LL	Oui	$\gamma \in \{0; 0{,}1; 0{,}2; 0{,}3\}$

*Le paramètre λ est choisi à l'aide d'une validation croisée avec les paramètres par défaut de la fonction

pour le Lasso adaptatif, nous testons $\gamma_{AL} \in \{0{,}5; 1; 2\}$. Pour chacun de ces paramètres α et γ_{AL}, nous trouvons le paramètre λ à l'aide de la validation croisée de la fonction `cv.glmnet`. Les autres paramètres de la fonction, par exemple le nombre de plis, sont fixés aux valeurs par défaut dans la fonction `cv.glmnet`.

Pour SCAD et MCP, nous utilisons la librarie `ncvreg`. Nous utilisons les paramètres par défaut pour ces deux méthodes, qui sont respectivement de 3,7 et 3 . La fonction `cv.ncvreg` nous permet d'effectuer la validation croisée pour trouver le λ minimum. Contrairement à la fonction `cv.glmnet`, la fonction `cv.ncvreg` ne nous permet pas d'utiliser la règle du 1se.

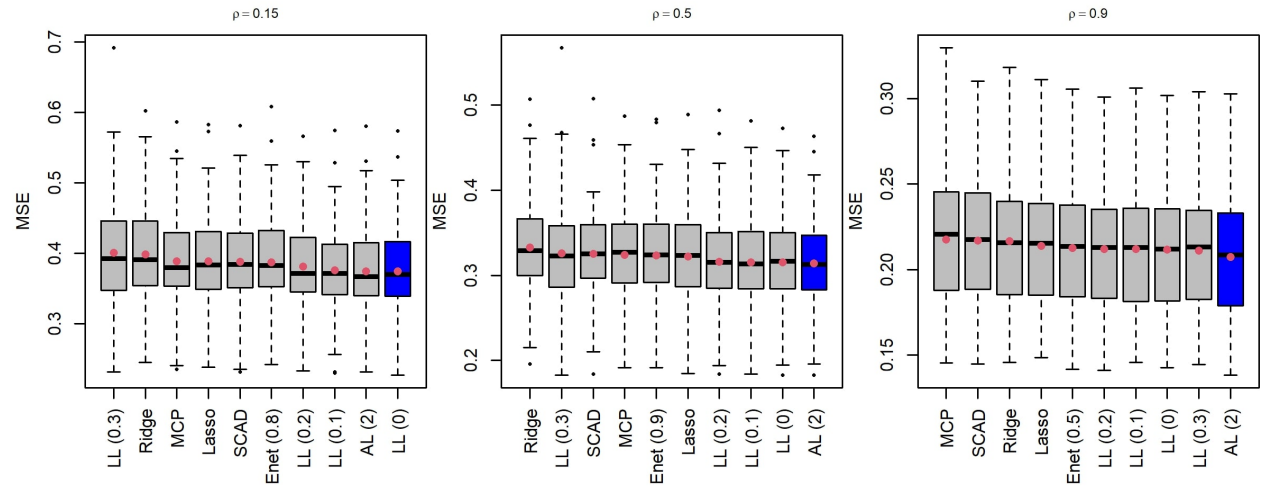
Suite aux conclusions de la section 3.4, nous comparons la performance du Lasso Linéaire aux méthodes alternatives en utilisant quatre paramètres : $\gamma \in \{0; 0{,}1; 0{,}2; 0{,}3\}$. Les paramètres pour la validation croisée répétée seront de $K = 10$ et $L = 5$.

4.2.3. Résultats

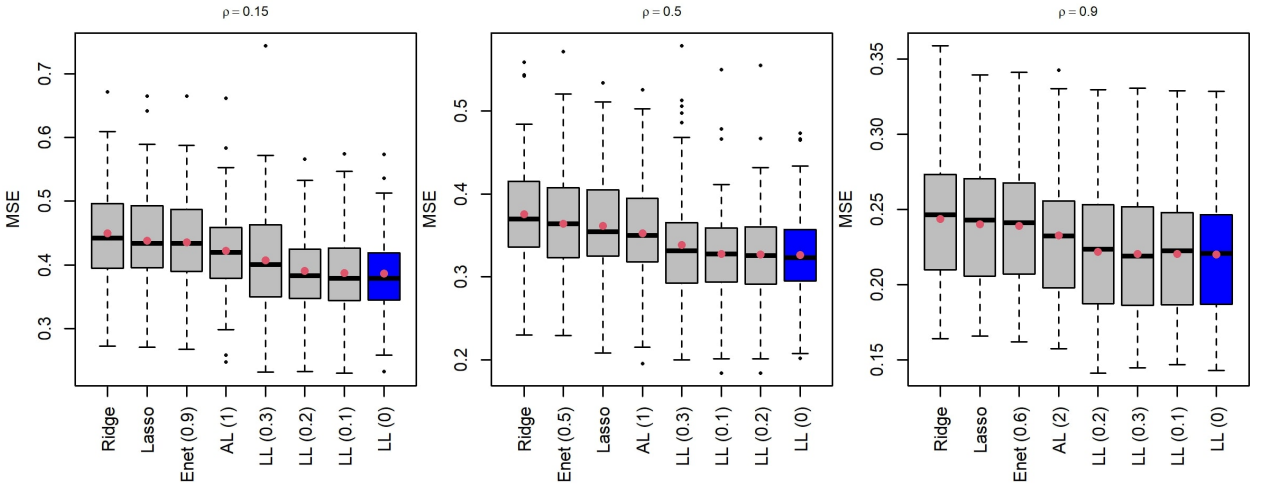
Les résultats obtenus pour les exemples nous permettront de comparer le Lasso Linéaire aux différentes méthodes alternatives. Pour chaque exemple et chaque matrice de design, nous présentons un graphique contenant les diagrammes à boîte des MSE de prédiction (obtenus à l'aide des 500 jeux de données) des différentes méthodes, ainsi qu'un tableau résumant tous les pourcentages de zéros corrects et incorrects. Pour le Filet élastique et le Lasso adaptatif, les méthodes ayant le plus petit MSE de prédiction parmi les suites de valeurs α et γ_{AL} seront présentées dans les figures. Par contre, dans le tableau des pourcentages, nous présentons tous les résultats (c'est-à-dire les résultats pour chacune des valeurs dans ces suites). Puisque la méthode séquentielle nous retourne des valeurs de MSE trop élevées, nous l'excluons des graphiques afin de faciliter la visualisation des autres méthodes; toutefois, les figures incluant cette méthode peuvent être consultées en annexe.

4.2.3.1. Exemple 1

Le vecteur de coefficients $\boldsymbol{\beta}^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$ est considéré; cet exemple a également été étudié dans les simulations du chapitre 3. Rappelons que différentes valeurs du paramètre ρ sont utilisées, soit $\rho \in \{0{,}15; 0{,}5; 0{,}9\}$; ces valeurs traduisent de petite, moyenne et grande corrélations entre les prédicteurs. Nous retrouvons les diagrammes à boîtes des MSE, en ordre décroissant de moyenne MSE, à la figure 4.2 pour la règle du minimum (ligne du haut) et la règle du 1se (ligne du bas). Le tableau 4.2 nous résume également les pourcentages de zéros corrects et incorrects pour les différentes méthodes.



(a) Régle du minimum.



(b) Règle du 1se.

Figure 4.2. Erreur quadratique moyenne (MSE) de prédiction en fonction des différentes méthodes pour chacun des $\rho \in \{0{,}15; 0{,}5; 0{,}9\}$ pour l'exemple 1 : $\boldsymbol{\beta}^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$. Le point rouge indique la moyenne et la boîte bleue indique la plus petite moyenne.

Pour le Filet élastique et le Lasso adaptatif, les paramètres α et γ_{AL} nous donnant le plus petit MSE de prédiction sont illustrés sur la figure 4.2. Nous remarquons rapidement que pour la règle du 1se, le Lasso Linéaire mène systématiquement au plus petit MSE parmi toutes les méthodes. Pour la règle du minimum, le Lasso Linéaire avec $\gamma = 0$ est la meilleure méthode pour $\rho = 0{,}15$, mais le Lasso adaptatif avec $\gamma_{AL} = 2$ l'emporte pour $\rho = 0{,}5$ et $\rho = 0{,}9$. Toutefois, dans ces deux cas, le Lasso Linéaire suit de près (basé sur différentes valeurs du paramètre γ). Ainsi, en termes de MSE, le Lasso Linéaire semble être une méthode adéquate, particulièrement (mais pas uniquement) lorsque $\gamma = 0$.

Tableau 4.2. Pourcentages de zéros corrects et incorrects pour les différentes méthodes de l'exemple 1 : $\boldsymbol{\beta}^* = (3; 1{,}5; 0; 0; 2; 0; 0; 0)^\top$.

Méthodes	$\rho = 0{,}15$				$\rho = 0{,}5$				$\rho = 0{,}9$			
	Min		1se		Min		1se		Min		1se	
	Cor.	Inc.	Cor.	Inc.	Cor.	Inc.	Cor.	Inc.	Cor.	Inc.	Cor.	Inc.
AIC (S)	82.28	0.07	-	-	79.80	0.00	-	-	77.92	16.73	-	-
Enet ($\alpha = 0{,}1$)	51.72	0.00	93.92	0.47	54.76	0.00	91.04	0.07	56.32	3.67	77.80	5.60
Enet ($\alpha = 0{,}2$)	12.36	0.00	44.20	0.00	16.72	0.00	37.00	0.00	26.48	0.93	12.60	0.00
Enet ($\alpha = 0{,}3$)	22.00	0.00	67.60	0.00	30.60	0.00	57.64	0.00	37.72	1.20	22.80	0.07
Enet ($\alpha = 0{,}4$)	30.64	0.00	77.92	0.13	37.00	0.00	69.20	0.00	44.76	1.67	32.08	0.00
Enet ($\alpha = 0{,}5$)	35.96	0.00	84.20	0.13	43.12	0.00	76.32	0.00	47.36	2.07	40.64	0.13
Enet ($\alpha = 0{,}6$)	40.88	0.00	88.40	0.13	47.64	0.00	81.28	0.00	50.96	2.40	48.80	0.40
Enet ($\alpha = 0{,}7$)	44.48	0.00	89.80	0.27	49.80	0.00	84.76	0.00	51.84	2.67	56.36	0.60
Enet ($\alpha = 0{,}8$)	45.60	0.00	92.04	0.20	51.96	0.00	86.64	0.07	54.76	3.00	62.28	1.40
Enet ($\alpha = 0{,}9$)	48.16	0.00	92.68	0.20	53.56	0.00	88.32	0.07	54.04	3.33	68.40	2.53
Lasso	50.48	0.00	93.72	0.33	53.76	0.00	89.96	0.13	55.32	3.53	73.32	3.60
AL ($\gamma_{AL} = 0.5$)	72.76	0.13	98.96	1.53	74.68	0.00	97.92	2.07	76.24	12.80	93.84	25.47
AL ($\gamma_{AL} = 1$)	80.08	0.13	99.40	2.20	79.08	0.07	99.12	4.53	77.80	15.00	94.92	29.87
AL ($\gamma_{AL} = 2$)	83.52	0.27	99.72	4.13	82.32	0.33	99.36	7.53	79.76	16.53	95.24	33.00
SCAD	74.40	0.20	-	-	76.16	0.40	-	-	75.40	20.33	-	-
MCP	83.08	0.33	-	-	82.04	0.67	-	-	75.64	20.87	-	-
LL ($\gamma = 0$)	91.60	0.27	99.92	6.00	88.36	0.67	99.64	12.67	78.72	19.27	95.80	35.80
LL ($\gamma = 0{,}1$)	91.72	0.20	99.92	5.93	88.56	0.33	99.68	12.40	78.04	19.20	95.76	35.80
LL ($\gamma = 0{,}2$)	88.84	0.20	99.68	6.27	87.40	0.13	99.72	12.07	78.88	18.40	95.72	35.53
LL ($\gamma = 0{,}3$)	86.72	0.20	99.32	6.00	84.32	0.13	98.72	12.47	79.92	18.80	95.84	35.93

Lorsqu'il y a peu de corrélation entre les prédicteurs ($\rho = 0{,}15$), basé sur la règle du minimum, le Lasso Linéaire offre une performance convaincante en détenant le plus haut pourcentage de zéros corrects; le paramètre $\gamma = 0{,}1$ nous donne le meilleur résultat avec 91,72%, suivi de près par $\gamma = 0$ avec 91,60%. De plus, le pourcentage de zéros incorrects est négligeable pour toutes les méthodes, donnant des résultats similaires et près de 0%. Pour ce qui est de la règle du 1se, le pourcentage de zéros corrects augmente d'environ 10 à 15% pour presque toutes les méthodes; dans le cas du Lasso Linéaire, le pourcentage de zéros

corrects s'approche de 100%. Tous les paramètres γ du Lasso Linéaire offrent des conclusions similaires. De manière générale, le pourcentage de zéros incorrects augmente pour toutes les méthodes sauf pour le Lasso et le Filet élastique, pour lesquelles il demeure près de 0%.

Lorsque $\rho = 0{,}5$, ce qui représente une corrélation moyenne entre les prédicteurs, le Lasso Linéaire avec $\gamma = 0{,}1$ nous offre encore une fois le plus grand pourcentage de zéros corrects pour la règle du minimum avec 88,56% suivi de très près par $\gamma = 0$ avec 88,36%. Comme avec $\rho = 0{,}15$, le Lasso Linéaire est la meilleure méthode, présentant les pourcentages de zéros corrects les plus élevés parmi toutes les méthodes. Les pourcentages de zéros incorrects sont semblables à ceux pour $\rho = 0{,}15$: toutes les méthodes semblent avoir un pourcentage près de 0. Basé sur la règle du 1se, le Lasso Linéaire semble toujours meilleur que les autres méthodes, présentant des pourcentages de zéros corrects très élevés. Notons toutefois que le pourcentage de zéros incorrects a également augmenté significativement par rapport à $\rho = 0{,}15$ pour toutes les méthodes sauf le Lasso et le Filet élastique.

Finalement lorsque $\rho = 0{,}9$, ce qui représente une grande corrélation entre les prédicteurs, le Lasso Linéaire offre une performance similaire à certaines méthodes comme le Lasso adaptatif, SCAD ou MCP, et est supérieur au Lasso et au Filet élastique. Basé sur la règle du minimum, le Lasso Linéaire avec $\gamma = 0{,}3$ est la méthode qui offre le plus grand pourcentage de zéros corrects avec 79,92%; les pourcentages sont d'ailleurs similaires pour tous les γ. Comparativement à $\rho = 0{,}15$ et $\rho = 0{,}5$, toutes les méthodes (sauf le Lasso et le Filet élastique) ont un plus grand pourcentage de zéros incorrects. Basé sur la règle du 1se, le Lasso Linéaire offre le plus grand pourcentage de zéros corrects, mais également le plus grand pourcentage de zéros incorrects, se situant près de 36%.

Avec de petite et moyenne corrélations entre les prédicteurs, le Lasso Linéaire offre des performances convaincantes, présentant les plus petits MSE de prédiction, ainsi que les plus grands pourcentages de zéros corrects, et ce peu importe la règle utilisée (minimum ou 1se). À l'inverse, le pourcentage de zéros incorrects est souvent plus élevé pour le Lasso Linéaire que pour les méthodes alternatives. Rappelons que les méthodes ayant un grand pourcentage de zéros incorrects sont très restrictives et estiment des coefficients non nuls à 0. De telles méthodes restrictives peuvent être bonnes, car elles permettent de retirer les prédicteurs peu importants; le risque est toutefois d'éliminer, par le fait même, des prédicteurs qui auraient dû être conservés. Ce phénomène est surtout accentué lorsque nous nous retrouvons dans une situation de grande corrélation appariée entre les prédicteurs. Toutefois, notons que les compétiteurs ne sont pas nécessairement supérieurs dans ce genre de situation. En effet, même si le Filet élastique a le plus bas pourcentage de zéros incorrects, il a également le plus bas pourcentage de zéros corrects, et ce dans presque toutes les situations; ceci nous indique qu'il conserve dans le modèle final des prédicteurs qui sont peu importants, ce qui nuit également à l'interprétabilité du modèle.

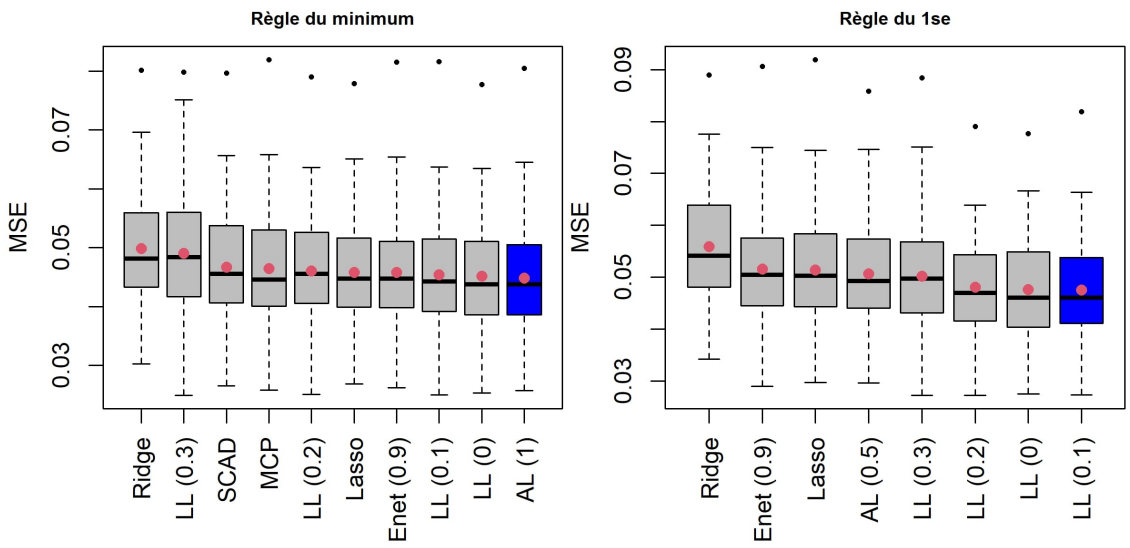


Figure 4.3. Erreur quadratique moyenne (MSE) de prédiction en fonction des différentes méthodes pour l'exemple 2 : $\boldsymbol{\beta}^* = (10; 5; 2; 1; 0{,}5; 0; 0; 0)^\top$. Le point rouge indique la moyenne et la boîte bleue indique la plus petite moyenne.

4.2.3.2. Exemple 2

Le vecteur de coefficients $\boldsymbol{\beta}^* = (10; 5; 2; 1; 0{,}5; 0; 0; 0)^\top$ est formé de coefficients de toutes les tailles. Le paramètre ρ est fixé à 0,5 dans cet exemple. Nous retrouvons à la figure 4.3 les diagrammes à boîte des MSE de prédiction, présentés en ordre décroissant, pour les règles du minimum et du 1se. Le tableau 4.3, pour sa part, résume les pourcentages de zéros corrects et incorrects pour les différentes méthodes.

Pour la règle du minimum, le Lasso adaptatif a le plus petit MSE de prédiction parmi toutes les méthodes, suivi de très près par la méthode du Lasso Linéaire ajusté avec $\gamma = 0$ et $\gamma = 0{,}1$. Dans cet exemple, nous observons que le MSE de prédiction obtenu avec le Lasso Linéaire croît légèrement à mesure que l'on augmente le paramètre γ, mais cette approche demeure tout de même efficace face aux compétiteurs. Pour la règle du 1se, le Lasso Linéaire avec $\gamma = 0{,}1$ détient le plus petit MSE suivi de très près par $\gamma = 0$ et $\gamma = 0{,}2$.

Pour la règle du minimum, nous remarquons que le Lasso adaptatif ($\gamma_{AL} = 2$) présente le plus grand pourcentage de zéros corrects, mais également le plus grand pourcentage de zéros incorrects. Le Lasso Linéaire suit le Lasso adaptatif avec un pourcentage de zéros corrects élevé, surtout lorsque $\gamma = 0{,}1$ et $\gamma = 0{,}2$. Pour la règle du 1se, nous notons la même problématique qu'à l'exemple 1 où le pourcentage de zéros incorrects est très élevé pour le Lasso Linéaire et pour plusieurs autres méthodes, à l'exception du Filet élastique, pour lequel le pourcentage de zéros incorrects est relativement bas. Toutefois, les pourcentages de zéros corrects détectés par le Filet élastique sont significativement plus bas que ceux des méthodes alternatives, ce qui indique que cette méthode a tendance à conserver un grand

Tableau 4.3. Pourcentages de zéros corrects et incorrects pour les différentes méthodes de l'exemple 2 : $\boldsymbol{\beta}^* = (10; 5; 2; 1; 0{,}5; 0; 0; 0)^\top$.

Méthodes	Min		1se	
	Cor.	Inc.	Cor.	Inc.
AIC (S)	80.27	12.40	-	-
Enet ($\gamma = 0{,}1$)	54.93	3.76	94.60	11.48
Enet ($\gamma = 0{,}2$)	14.00	1.04	47.93	2.48
Enet ($\gamma = 0{,}3$)	26.33	1.88	70.53	4.04
Enet ($\gamma = 0{,}4$)	35.40	2.56	80.80	5.96
Enet ($\gamma = 0{,}5$)	41.47	3.20	86.33	7.00
Enet ($\gamma = 0{,}6$)	45.60	3.32	89.53	8.12
Enet ($\gamma = 0{,}7$)	49.07	3.52	91.60	8.96
Enet ($\gamma = 0{,}8$)	49.33	3.56	93.60	10.04
Enet ($\gamma = 0{,}9$)	53.27	3.72	93.87	10.88
Lasso	53.73	4.00	94.67	11.28
AL ($\gamma_{AL} = 0.5$)	73.73	8.32	98.07	22.00
AL ($\gamma_{AL} = 1$)	79.47	11.08	98.53	25.80
AL ($\gamma_{AL} = 2$)	91.87	20.12	99.40	31.04
SCAD	65.80	10.24	-	-
MCP	72.53	12.92	-	-
LL ($\gamma = 0$)	80.20	13.36	98.87	29.56
LL ($\gamma = 0{,}1$)	82.80	12.44	99.33	29.88
LL ($\gamma = 0{,}2$)	82.00	11.48	99.47	29.68
LL ($\gamma = 0{,}3$)	80.07	10.52	99.33	31.20

nombre de prédicteurs, incluant des prédicteurs dont le coefficient est nul. Notons que le Lasso Linéaire performe particulièrement bien quant au pourcentage de zéros corrects, ce qui signifie qu'il propose des modèles parcimonieux, mais qu'il a parfois tendance à écarter des prédicteurs dont le coefficient est non nul.

4.2.3.3. Exemple 3

Cet exemple est pris directement de Zou et Hastie (2005). Ce scénario nous intéresse puisqu'il crée trois groupes dans la matrice de design générée. De plus, cet exemple ajoute un niveau de complexité puisqu'il contient 40 prédicteurs, dont 15 coefficients qui sont non nuls et 25 coefficients qui sont nuls. Nous rappelons que le nombre d'observations dans le jeu de données de l'exemple 3 est de 400, comparativement à 100 pour les exemples 1 et 2. Nous retrouvons les diagrammes à boîte des MSE, en ordre décroissant, à la figure 4.4 pour les règles du minimum et du 1se. Le tableau 4.4 résume les pourcentages de zéros corrects et incorrects pour les différentes méthodes.

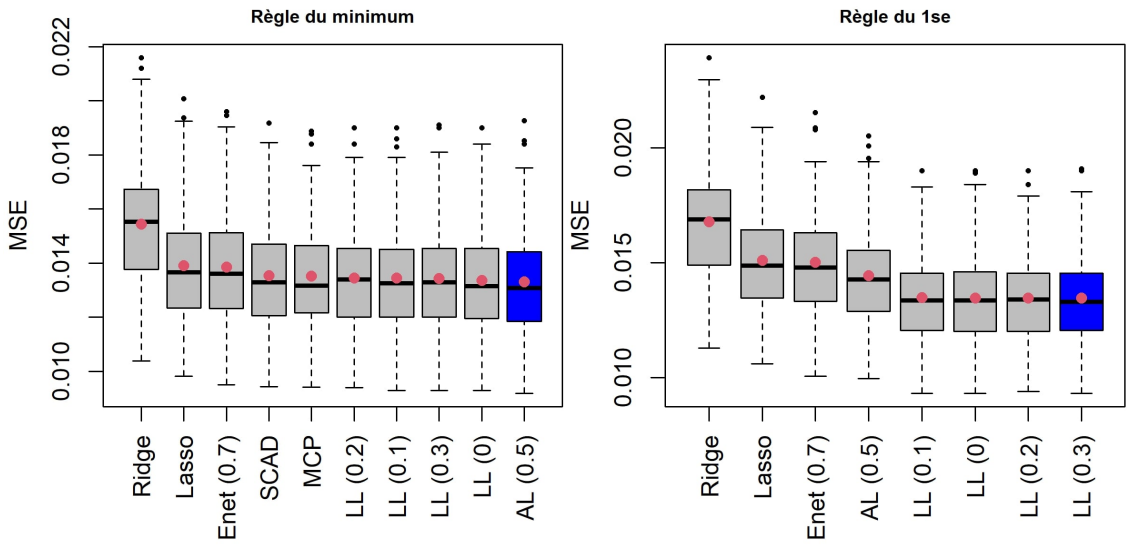


Figure 4.4. Erreur quadratique moyenne (MSE) de prédiction en fonction des différentes méthodes pour l'exemple 3. Le point rouge indique la moyenne et la boîte bleue indique la plus petite moyenne.

Le Lasso adaptatif avec $\gamma_{AL} = 0{,}5$ est la méthode qui produit le plus petit MSE de prédiction pour la règle du minimum, suivi de près par le Lasso Linéaire avec les différents γ. Pour la règle du 1se, le Lasso Linéaire avec $\gamma = 0{,}3$ détient le plus petit MSE, mais les autres valeurs de γ mènent à des MSE extrêmement près de cette valeur minimum. Nous remarquons que pour toutes les méthodes, sauf le Filet élastique, le pourcentage de zéros incorrects est très grand, se situant près de 80%. Ainsi, tel qu'anticipé, la plupart des méthodes offrent une performance médiocre dans cet exemple, à l'exception du Filet élastique, qui gère bien le groupage de variables. Pour la règle du 1se, le Filet élastique offre les meilleures performances, avec un très haut pourcentage de zéros corrects et un pourcentage de zéros incorrects se situant près de 0.

4.2.4. Interprétation des résultats

Dans les sections précédentes, des simulations ont été utilisées pour comparer le Lasso Linéaire aux méthodes alternatives de la section 4.1. Après avoir étudié trois exemples comportant différentes paramétrisations, nous pouvons conclure que le Lasso Linéaire est une excellente méthode en petites dimensions, particulièrement lorsqu'il y a une corrélation faible ou modérée entre les prédicteurs. À l'exemple 1, nous avons observé la supériorité du Lasso Linéaire dans divers contextes, sauf lorsque la corrélation entre les prédicteurs était trop élevée ($\rho = 0{,}9$). Toutefois, lorsque $\rho = 0{,}9$, le Lasso Linéaire avait des MSE de prédiction intéressants, et même meilleurs que la plupart des compétiteurs. Le seul désavantage dans ce contexte est que la méthode tendait à rejeter plus de prédicteurs (ceux

Tableau 4.4. Pourcentages de zéros corrects et incorrects pour les différentes méthodes de l'exemple 3.

Méthodes	Min		1se	
	Cor.	Inc.	Cor.	Inc.
AIC (S)	81.86	72.60	-	-
Enet ($\gamma = 0{,}1$)	74.17	30.49	99.25	32.71
Enet ($\gamma = 0{,}2$)	50.30	0.00	89.39	0.00
Enet ($\gamma = 0{,}3$)	62.93	0.00	96.52	0.00
Enet ($\gamma = 0{,}4$)	67.56	0.00	97.88	0.00
Enet ($\gamma = 0{,}5$)	70.30	0.00	98.42	0.00
Enet ($\gamma = 0{,}6$)	71.66	0.00	98.56	0.00
Enet ($\gamma = 0{,}7$)	72.62	0.00	98.79	0.00
Enet ($\gamma = 0{,}8$)	73.42	0.00	99.10	0.00
Enet ($\gamma = 0{,}9$)	74.08	0.00	99.10	0.00
Lasso	74.11	0.00	99.06	0.00
AL ($\gamma_{AL} = 0.5$)	99.87	77.63	100.00	79.08
AL ($\gamma_{AL} = 1$)	100.00	78.83	100.00	79.49
AL ($\gamma_{AL} = 2$)	100.00	79.28	100.00	79.73
SCAD	95.03	80.00	-	-
MCP	97.46	80.00	-	-
LL ($\gamma = 0$)	98.49	79.76	100.00	80.00
LL ($\gamma = 0{,}1$)	99.75	77.75	100.00	80.00
LL ($\gamma = 0{,}2$)	99.92	77.00	100.00	80.00
LL ($\gamma = 0{,}3$)	99.96	77.27	100.00	80.00

avec des coefficients nuls, mais également certains prédicteurs dont le coefficient était non nul). Par la suite à l'exemple 2, le Lasso Linéaire était la seconde meilleure méthode en termes de MSE de prédiction alors qu'à l'exemple 3, presque toutes les méthodes offraient une performance médiocre, avec des pourcentages de zéros incorrects trop élevés (à l'exception du Filet élastique). Une explication vraisemblable pour ce comportement est que lorsqu'une variable faisant partie d'un groupe est éliminée, les variables corrélées avec celle-ci sont également éliminées, ce qui résulte en un grand pourcentage de zéros incorrects pour certaines méthodes. Malgré les avantages du Filet élastique, il est important de rappeler que cette méthode nécessite l'estimation de deux hyperparamètres et fait appel à des procédures de calcul plus complexes.

Nous avons également remarqué certains comportements, ou caractéristiques, des méthodes alternatives. La régression séquentielle était inadéquate pour tous les exemples, car le MSE était beaucoup trop élevé (voir les figures A.1 et A.2). De même, la méthode du Ridge offrait des performances médiocres dans toutes les situations, affichant souvent le plus

grand MSE après la régression séquentielle. En effet, rappelons que la méthode du Ridge n'est pas tellement intéressante ici puisqu'elle ne fait aucune sélection de variables. De plus, les méthodes SCAD et MCP n'avaient pas une particulièrement bonne capacité prédictive et ne classifiaient pas particulièrement bien les coefficients nuls ou non nuls.

Nous rappelons que la règle du 1se est bien sûr une alternative à la règle du minimum; celle est plus sévère, en termes de taille du modèle, car elle retire plus de prédicteurs du modèle. Dans les exemples 1 et 2, nous voyons une augmentation du pourcentage de zéros incorrects, mais également une augmentation du pourcentage de zéros corrects lors de l'implémentation de cette approche. Le Lasso Linéaire bénéficie grandement de la règle du 1se, présentant des pourcentages de zéros corrects très élevés et fournissant une excellente capacité prédictive en ayant les MSE les plus bas. À l'inverse, nous avons observé que le Filet élastique menait à des pourcentages de zéros incorrects très faibles, mais également des pourcentages de zéros corrects très bas, aboutissant à des modèles beaucoup plus lourds.

Dans le chapitre 3, les conclusions semblaient supporter le choix $\gamma = 0{,}1$ pour le Lasso Linéaire, puisque ce paramètre était un bon compromis entre $\gamma = 0$ et $\gamma = 0{,}2$. L'interprétation des résultats de cette section nous confirme que $\gamma > 0{,}2$ n'est pas nécessairement optimal pour le Lasso Linéaire. En effet, nous avons observé une croissance du MSE de prédiction en fonction du paramètre γ. Les valeurs de vrais positifs (VP) et de faux positifs (FP), pour leur part, étaient souvent en faveur de $\gamma = 0$ ou $\gamma = 0{,}1$. Ainsi, il est toujours difficile de choisir un γ précis nous menant au meilleur modèle, mais $\gamma = 0{,}1$ semble un choix prudent en présentant un bon compromis entre le temps de calcul, la capacité prédictive et l'interprétabilité du modèle.

Tableau 4.5. Caractéristiques des différentes méthodes

Méthodes	Solution analytique	Sélection de variables	Grande dimension
Régression séquentielle	Oui	Oui	Non
Ridge	Oui	Non	Non
Lasso	Non	Oui	Non*
Filet élastique	Non	Oui	Oui
Lasso adaptatif	Non	Oui	Non
SCAD	Non	Oui	Oui
MCP	Non	Oui	Oui
Lasso Linéaire	Oui	Oui	Oui

Dans ce chapitre, nous remarquons que le Lasso Linéaire est une méthode qui se démarque positivement. La méthode est performante, particulièrement en dimensions modérées. Le tableau 4.5 nous rappelle que le Lasso Linéaire puise également sa force dans sa capacité à

effectuer des calculs simples pour discriminer entre les prédicteurs; il effectue une sélection de variables efficace et peut être utilisé en grandes dimensions. Il serait éventuellement intéressant de creuser davantage en comparant les méthodes alternatives au Lasso Linéaire en grandes dimensions, en étudiant davantage de scénarios, mais ceci pourra être abordé dans un projet de recherche ultérieur.

Conclusion

Dans ce mémoire, nous avons étudié en profondeur le Lasso Linéaire, une technique introduite par Fraser et Bédard (2022). Cette technique est intéressante dû à sa polyvalence : elle peut être utilisée en petites dimensions, lorsque le nombre d'observations est plus grand que le nombre de prédicteurs p et également en grandes dimensions, lorsque $p > n$. Sa solution analytique nous permet d'utiliser l'estimateur des moindres carrés ordinaires qui est sans biais à variance minimale. La méthode ne nécessite aucun algorithme complexe et est seulement assujettie à une simple validation croisée. De plus, nous essayons d'adresser avec le Lasso Linéaire les deux problèmes du Lasso régulier : sa sélection instable des prédicteurs fortement corrélés et sa gestion des grandes dimensions.

L'implémentation de l'algorithme du Lasso Linéaire est simple. Un paramètre d'ajustement γ est choisi dans un premier temps pour écarter les prédicteurs les moins corrélés avec la variable réponse et, par la suite, des prédicteurs sont retirés du modèle en fonction de la plus petite baisse de la variance $c_\delta^\top C_\delta^{-1} c_\delta$. Le sous-modèle sélectionné est celui dont la taille correspond à la plus petite erreur quadratique moyenne par validation croisée. Nous ajoutons la règle du 1se, qui est utilisée par d'autres algorithmes modernes. Cette alternative intéressante nous permet de choisir un modèle plus parcimonieux. Nous remarquons que le Lasso Linéaire bénéficie grandement de la règle du 1se en présentant des pourcentages de zéros corrects très élevés et également une excellente capacité prédictive. En grandes dimensions, nous nous inspirons de la technique du SIS pour choisir le modèle final. Nous éliminons les variables les moins corrélées du modèle et par la suite appliquons le Lasso Linéaire. Nous testons l'algorithme sur l'exemple de gènes pour prédire le niveau d'expression du gène TRIM32 lié à une maladie génétique appelée syndrome de Bardet-Biedl. Nous testons également la validité du paramètre d'ajustement par défaut $\gamma = 0{,}2$ sur des données artificielles pour valider son optimalité.

À la lumière de nos résultats, il semblerait qu'un paramètre d'ajustement γ plus faible devrait être priorisé dans nos simulations. Malgré des temps de calcul plus longs qu'avec des valeurs de γ plus élevées, les paramètres d'ajustement près de 0,1 semblent meilleurs en termes de prédiction et d'interprétabilité du modèle. Nous observons également un phénomène intéressant en petites dimensions; lorsque n n'est pas assez grand comparativement

à p, des termes de corrélations dans c pourraient être artificiellement élevés; par conséquent, le modèle choisi par le Lasso Linéaire avec de grandes valeurs de γ aura tendance à être surajusté. L'exemple du diabète nous fournit une bonne illustration de ce phénomène. Nous comparons finalement le Lasso Linéaire aux méthodes alternatives modernes et nous concluons également que $\gamma = 0{,}1$ semble constituer un choix prudent qui est adapté à une grande variété d'exemples. Nous notons que les simulations sont exécutées dans des contextes où $n >> p$. Lors de l'étude du paramètre d'ajustement, la valeur de faux positifs diminue lorsque n augmente. Ainsi, les conclusions doivent être interprétées avec prudence, mais $\gamma = 0{,}1$ dans nos études semble un excellent candidat. Lorsqu'il y a une corrélation faible ou moyenne entre les prédicteurs, le Lasso Linéaire performe bien dans les exemples étudiés. Toutefois, le Lasso Linéaire a tendance à rejeter plus de prédicteurs dans une situation de grande corrélation entre les prédicteurs. Nous avons remarqué que toutes les méthodes performaient moins bien dans une situation comportant des groupes de variables, à l'exception du Filet élastique.

Pour les recherches futures, il serait intéressant de comparer le Lasso Linéaire aux méthodes alternatives sur des données artificielles en grandes dimensions : SCAD, MCP, Filet élastique et SIS (jumelée avec une autre méthode). Nous pourrons alors comprendre quelle méthode est la mieux adaptée dans ce genre de contexte. Il serait également intéressant d'étudier une approche ascendante pour la procédure un-à-un. Nous pourrions possiblement réduire le temps de calcul en procédant de cette manière. De plus, une démonstration théorique pour confirmer le paramètre d'ajustement optimal pourrait être réalisée. Finalement, nous pourrions essayer de comprendre le comportement du Lasso Linéaire dans un contexte où le nombre d'observations n est proche de p. Comme l'a dit George Box, "Tous les modèles sont faux, mais certains sont utiles", ce qui signifie que bien que les modèles ne soient pas une représentation exacte de la réalité, ils peuvent être utilisés de manière pratique et bénéfique dans certains contextes. Nous croyons que le Lasso Linéaire est une méthode qui est polyvalente et simple à utiliser, ce qui la rend utile dans plusieurs contextes.

Références bibliographiques

Hirotogu Akaike : Information theory and an extension of the maximum likelihood principle. *In Springer Series in Statistics*, Springer Series in Statistics, pages 199–213. Springer New York, New York, NY, 1998.

Ravindra Bapat : *Linear algebra and linear models*. New York : Springer, 2020. ISBN 1498712169.

Leo Breiman, Jerome H. Friedman, Richard A. Olshen et Charles J. Stone : *Classification And Regression Trees*. Routledge, octobre 1984.

Leo Breiman et Philip Spector : Submodel Selection and Evaluation in Regression. The X-Random Case. *International Statistical Review / Revue Internationale de Statistique*, 60(3):291–319, 1992. ISSN 03067734, 17515823.

Peter Bühlmann et Sara Van De Geer : *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, 2013.

Yuchen Chen et Yuhong Yang : The one standard error rule for model selection: Does it work? *Stats*, 4(4):868–892, 2021. ISSN 2571-905X.

P Cortez et A M G Silva : Using data mining to predict secondary school student performance. *In* A Brito et & J Teixeira, éditeurs : *Proceedings of 5th Annual Future Business Technology Conference*, pages 5–12. Porto, 2008.

Bradley Efron, Trevor Hastie, Iain Johnstone et Robert Tibshirani : Least Angle Regression. *The Annals of Statistics*, 32(2):407–451, 2004. ISSN 00905364.

Jianqing Fan et Runze Li : Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001a. ISSN 01621459.

Jianqing Fan et Runze Li : Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001b. ISSN 01621459.

Jianqing Fan et Jinchi Lv : Sure Independence Screening for Ultrahigh Dimensional Feature Space. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 70 (5):849–911, 2008. ISSN 13697412, 14679868.

Don Fraser et Mylène Bédard : The Linear Lasso: A Location model approach. *Canadian Journal of Statistics*, 50(2):437–453, 2022.

Jerome Friedman, Trevor Hastie et Rob Tibshirani : Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1–22, 2010.

T. Hastie, R. Tibshirani et J.H. Friedman : *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2009. ISBN 9780387848846.

Trevor Hastie, Robert Tibshirani et Martin Wainwright : *Statistical Learning with Sparsity: The Lasso and Generalizations*. Chapman Hall/CRC, 2015. ISBN 1498712169.

Arthur E. Hoerl et Robert W. Kennard : Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 12(1):55–67, 1970. ISSN 00401706.

Ron Kohavi : A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *In Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'95, page 1137–1143, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc. ISBN 1558603638.

Todd E. Scheetz, Kwang-Youn A. Kim, Ruth E. Swiderski, Alisdair R. Philp, Terry A. Braun, Kevin L. Knudtson, Anne M. Dorrance, Gerald F. DiBona, Jian Huang, Thomas L. Casavant, Val C. Sheffield et Edwin M. Stone : Regulation of Gene Expression in the Mammalian Eye and Its Relevance to Eye Disease. *Proceedings of the National Academy of Sciences of the United States of America*, 103(39):14429–14434, 2006. ISSN 00278424.

Robert Tibshirani : Regression Shrinkage and Selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288, 1996. ISSN 00359246.

Hadley Wickham, Jim Hester, Winston Chang et Jennifer Bryan : *devtools: Tools to Make Developing R Packages Easier*, 2022. https://devtools.r-lib.org/, https://github.com/r-lib/devtools.

Cun-Hui Zhang : Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2):894–942, 2010. ISSN 00905364, 21688966.

Hui Zou : The Adaptive Lasso and Its Oracle Properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006. ISSN 01621459.

Hui Zou et Trevor Hastie : Regularization and Variable Selection via the Elastic Net. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, 67(2):301–320, 2005. ISSN 13697412, 14679868.

Annexe A

A.1. Démonstration du théorème 1.3.2

Il suffit de développer la fonction objective et de trouver son minimum. Cette fonction satisfait

$$\begin{aligned}\boldsymbol{S}(\boldsymbol{\beta}) &= (\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})^\top(\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta})+\lambda\|\boldsymbol{\beta}\|^2\\ &= \mathbf{y}^\top\mathbf{y}-\boldsymbol{\beta}^\top\boldsymbol{X}^\top\mathbf{y}-\mathbf{y}^\top\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\beta}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}+\lambda\|\boldsymbol{\beta}\|^2\\ &= \mathbf{y}^\top\mathbf{y}-2\boldsymbol{\beta}^\top\boldsymbol{X}^\top\mathbf{y}+\boldsymbol{\beta}^\top\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}+\lambda\boldsymbol{\beta}^\top\boldsymbol{\beta}.\end{aligned}$$

La dérivée s'écrit comme suit :

$$\frac{\partial\boldsymbol{S}(\boldsymbol{\beta})}{\partial\boldsymbol{\beta}} = -2\boldsymbol{X}^\top\mathbf{y}+2\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}+2\lambda\boldsymbol{\beta}.$$

Ainsi, la fonction $\boldsymbol{S}(\boldsymbol{\beta})$ est minimisée à $\hat{\boldsymbol{\beta}}^{(R)}$ tel que

$$-2\boldsymbol{X}^\top\mathbf{y}+2\boldsymbol{X}^\top\boldsymbol{X}\hat{\boldsymbol{\beta}}^{(R)}+2\lambda\hat{\boldsymbol{\beta}}^{(R)}=0 \implies \hat{\boldsymbol{\beta}}^{(R)}=(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbf{y}.$$

A.2. Démonstration de la proposition 1.3.3

La seule variable aléatoire dans l'expression $\hat{\boldsymbol{\beta}}^{(R)}=(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbf{y}$ étant $\mathbf{y}$, alors l'espérance et la variance de $\hat{\boldsymbol{\beta}}^{(R)}$ sont calculées par rapport à la distribution de $\mathbf{y}$:

$$\begin{aligned}\mathbb{E}(\hat{\boldsymbol{\beta}}^{(R)}) &= \mathbb{E}((\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbf{y})\\ &= (\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbb{E}(\mathbf{y})\\ &= (\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\boldsymbol{X}\boldsymbol{\beta}\\ &= (\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p-\lambda I_p)\boldsymbol{\beta}\\ &= \boldsymbol{\beta}-\lambda(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{\beta}.\end{aligned}$$

Selon la définition du biais en (1.1.4), nous obtenons

$$\boldsymbol{B}(\hat{\boldsymbol{\beta}}^{(R)}) = -\lambda(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{\beta},$$

ce qui montre que l'estimateur $\hat{\boldsymbol{\beta}}^{(R)}$ a un biais non nul. La variance est

$$\begin{aligned}\mathbb{V}(\hat{\boldsymbol{\beta}}^{(R)}) &= \mathbb{V}((\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbf{y}) \\ &= (\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\mathbb{V}(\mathbf{y})\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1} \\ &= \sigma^2(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}\boldsymbol{X}^\top\boldsymbol{X}(\boldsymbol{X}^\top\boldsymbol{X}+\lambda I_p)^{-1}.\end{aligned}$$

A.3. Démonstration de la proposition 2.4.1

Supposons que la distribution conjointe des variables aléatoires x et y est

$$\begin{bmatrix}x\\y\end{bmatrix} \sim \mathcal{MN}(\mu,\Sigma),$$

où

$$\mu=\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix} \quad \text{et} \quad \Sigma=\begin{bmatrix}\sigma_{x,x} & \sigma_{x,y}\\ \sigma_{y,x} & \sigma_{y,y}\end{bmatrix}.$$

La dimension du vecteur x est de $n_1\times 1$ et celle du vecteur y est de $n_2\times 1$. Le vecteur $(x,y)^\top$ aura alors une dimension de $n\times 1$, où $n=n_1+n_2$.

Lemme A.3.1. *Supposons la matrice bloc suivante*

$$\begin{bmatrix}A & B\\ C & D\end{bmatrix},$$

alors, son inverse est

$$\begin{bmatrix}A & B\\ C & D\end{bmatrix}^{-1} = \begin{bmatrix}(A-BD^{-1}C)^{-1} & -(A-BD^{-1}C)^{-1}BD^{-1}\\ -D^{-1}C(A-BD^{-1}C)^{-1} & D^{-1}+D^{-1}C(A-BD^{-1}C)^{-1}BD^{-1}\end{bmatrix}. \tag{A.3.1}$$

et son déterminant est

$$\begin{vmatrix}A & B\\ C & D\end{vmatrix} = |D|\cdot|A-BD^{-1}C|. \tag{A.3.2}$$

La distribution conjointe peut être écrite sous la forme suivante

$$f(x,y)=\frac{1}{\sqrt{(2\pi)^n|\Sigma|}}\exp\left\{-\frac{1}{2}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)^\top\Sigma^{-1}\left(\begin{bmatrix}x\\y\end{bmatrix}-\begin{bmatrix}\mu_x\\\mu_y\end{bmatrix}\right)\right\},$$

où Σ^{-1} est l'inverse de Σ. Nous appliquons l'inverse (A.3.1) pour la matrice bloc Σ. Sachant que Σ^{-1} est une matrice symétrique, nous obtenons

$$\begin{aligned}\Sigma^{-1} &= \begin{bmatrix}(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1} & -(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}\\ -\sigma_{y,y}^{-1}\sigma_{y,x}(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1} & \sigma_{y,y}^{-1}+\sigma_{y,y}^{-1}\sigma_{y,x}(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}\end{bmatrix} \\ &= \begin{bmatrix}(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1} & -(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}\\ -(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1}\sigma_{x,y}\sigma_{y,y}^{-1} & \sigma_{y,y}^{-1}+\sigma_{y,y}^{-1}\sigma_{y,x}(\sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x})^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}\end{bmatrix}.\end{aligned}$$

$$= \begin{bmatrix} \sigma_{x|y}^{-1} & -\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1} \\ -\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1} & \sigma_{y,y}^{-1} + \sigma_{y,y}^{-1}\sigma_{y,x}\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1} \end{bmatrix}$$

où $\sigma_{x|y} = \sigma_{x,x} - \sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x}$. Le déterminant de Σ est alors

$$|\Sigma| = |\sigma_{y,y}| \cdot |\sigma_{x|y}| \tag{A.3.3}$$

selon (A.3.2). La densité conjointe peut être réécrite sous la forme suivante

$$\begin{aligned} f(x,y) = \frac{1}{\sqrt{(2\pi)^n |\sigma_{y,y}| \cdot |\sigma_{x|y}|}} \exp\Bigg\{ & -\frac{1}{2}\Big((x-\mu_x)^\top \sigma_{x|y}^{-1}(x-\mu_x) \\ & - 2(x-\mu_x)^\top \sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y) \\ & + (y-\mu_y)^\top [\sigma_{y,y}^{-1} + \sigma_{y,y}^{-1}\sigma_{y,x}\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}](y-\mu_y)\Big)\Bigg\}. \end{aligned}$$

De plus, nous savons que la densité marginale de $f(y)$ est

$$\frac{1}{\sqrt{(2\pi)^{n_2}|\sigma_{y,y}|}} \exp\left\{-\frac{1}{2}\Big((y-\mu_y)^\top \sigma_{y,y}^{-1}(y-\mu_y)\Big)\right\}.$$

Nous savons que la loi de probabilité conditionnelle est

$$\begin{aligned} f(x|y) &= \frac{f(x,y)}{f(y)} \\ &= \frac{\frac{1}{\sqrt{(2\pi)^n|\Sigma|}} \exp\left\{-\frac{1}{2}\left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)^\top \Sigma^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)\right\}}{\frac{1}{\sqrt{(2\pi)^{n_2}|\sigma_{y,y}|}} \exp\left\{-\frac{1}{2}\Big((y-\mu_y)^\top \sigma_{y,y}^{-1}(y-\mu_y)\Big)\right\}} \\ &= \frac{1}{\sqrt{(2\pi)^{n-n_2}}} \sqrt{\frac{|\sigma_{y,y}|}{|\Sigma|}} \exp\Bigg\{ -\frac{1}{2}\left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right)^\top \Sigma^{-1} \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}\right) \\ &+ \frac{1}{2}\Big((y-\mu_y)^\top \sigma_{y,y}^{-1}(y-\mu_y)\Big)\Bigg\}. \end{aligned}$$

En insérant la distribution conjointe $f(x,y)$ développée, on obtient

$$\begin{aligned} f(x|y) = \frac{1}{\sqrt{(2\pi)^{n-n_2}}} \sqrt{\frac{|\sigma_{y,y}|}{|\Sigma|}} \exp\Bigg\{ & -\frac{1}{2}\Big((x-\mu_x)^\top \sigma_{x|y}^{-1}(x-\mu_x) \\ & - 2(x-\mu_x)^\top \sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y) \\ & + (y-\mu_y)^\top [\sigma_{y,y}^{-1} + \sigma_{y,y}^{-1}\sigma_{y,x}\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}](y-\mu_y)\Big) \\ & + \frac{1}{2}\Big((y-\mu_y)^\top \sigma_{y,y}^{-1}(y-\mu_y)\Big)\Bigg\}. \end{aligned}$$

Nous éliminons quelques termes pour obtenir

$$f(x|y) = \frac{1}{\sqrt{(2\pi)^{n-n_2}}}\sqrt{\frac{|\sigma_{y,y}|}{|\Sigma|}}\exp\Bigg\{-\frac{1}{2}\Big((x-\mu_x)^\top\sigma_{x|y}^{-1}(x-\mu_x) - 2(x-\mu_x)^\top\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y) + (y-\mu_y)^\top\sigma_{y,y}^{-1}\sigma_{y,x}\sigma_{x|y}^{-1}\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)\Big)\Bigg\}. \tag{A.3.4}$$

Le terme à l'intérieur de l'exponentielle peut être simplifié comme

$$\begin{aligned}
&-\frac{1}{2}\left[(x-\mu_x)-\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)\right]^\top\sigma_{x|y}^{-1}\left[(x-\mu_x)-\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)\right]\\
&=-\frac{1}{2}\left[x-(\mu_x+\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)\right]^\top\sigma_{x|y}^{-1}\left[x-(\mu_x+\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)\right]\\
&=-\frac{1}{2}\left[x-\mu_{x|y}\right]^\top\sigma_{x|y}^{-1}\left[x-\mu_{x|y}\right],
\end{aligned}$$

où $\mu_{x|y} = \mu_x + \sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)$. Sachant que $n_1 = n - n_2$, nous utilisons le résultat (A.3.3) dans l'équation (A.3.4), ce qui élimine le terme $|\sigma_{y,y}|$ et nous obtenons

$$f(x|y) = \frac{1}{\sqrt{(2\pi)^{n_1}|\sigma_{x|y}|}}\exp\left\{-\frac{1}{2}\left[x-\mu_{x|y}\right]^\top\sigma_{x|y}^{-1}\left[x-\mu_{x|y}\right]\right\}, \tag{A.3.5}$$

où $\mu_{x|y} = \mu_x+\sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y)$ et $\sigma_{x|y} = \sigma_{x,x}-\sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x}$. Alors, la distribution conditionnelle de $x|y$ est

$$x|y \sim \mathcal{N}(\mu_{x|y}, \sigma_{x|y}) = \mathcal{N}(\mu_x + \sigma_{x,y}\sigma_{y,y}^{-1}(y-\mu_y), \sigma_{x,x} - \sigma_{x,y}\sigma_{y,y}^{-1}\sigma_{y,x}).$$

Sans perte de généralité, nous pouvons obtenir l'équation conditionnelle pour la variable aléatoire y satisfaisant l'équation

$$y|x \sim \mathcal{N}(\mu_{y|x}, \sigma_{y|x}) = \mathcal{N}(\mu_y + \sigma_{y,x}\sigma_{x,x}^{-1}(x-\mu_x), \sigma_{y,y} - \sigma_{y,x}\sigma_{x,x}^{-1}\sigma_{x,y}).$$

Dans le contexte du Lasso Linéaire, nous savons que μ_x et μ_y sont des vecteurs nuls, alors

$$y|x \sim \mathcal{N}(\sigma_{y,x}\sigma_{x.x}^{-1}x\ ,\ \sigma_{y,y} - \sigma_{y,x}\sigma_{x.x}^{-1}\sigma_{x,y}).$$

A.4. Application aux données : Notes de mathématiques

Tableau A.1. Description des variables ainsi que leur type; `G3` est la variable réponse

Variables	Description*
`school`	école de l'étudiant (B : *Gabriel Pereira* ou *Mousinho da Silveira*)
`sex`	sexe de l'étudiant (B : femme ou homme)
`age`	âge de l'étudiant (C : 15 à 22 ans)
`adress`	adresse de l'étudiant (B : urbain ou rural)
`famsize`	taille de la famille (B : ≤ 3 ou > 3)
`Pstatus`	statut de cohabitation des parents (B : ensemble ou séparé)
`Medu`	éducation de la mère (C : 0 à 4[a])
`Fedu`	éducation du père (C : 0 à 4[a])
`Mjob`	emploi de la mère (N[b])
`Fjob`	emploi du père (N[b])
`reason`	raison du choix de l'école (N : proximité du domicile, réputation de l'école, préférence de cours ou autre)
`guardian`	tuteur de l'élève (N : mère, père ou autre)
`traveltime`	temps de trajet domicile-école (numérique : 1 - $<$ 15 min, 2 - 15 à 30 min, 3 - 30 min à 1 heure ou 4 - $>$ 1 heure).
`studytime`	temps d'étude hebdomadaire (C : 1 - $<$ 2 heures, 2 - 2 à 5 heures, 3 - 5 à 10 heures ou 4 - $>$ 10 heures)
`failures`	nombre d'échecs passés en classe (C : n si $1 \leq n < 3$, sinon 4)
`schoolsup`	soutien scolaire (B : oui ou non)
`famsup`	soutien éducatif familial (B : oui ou non)
`paid`	cours supplémentaires payants (B : oui ou non)
`activities`	activités parascolaires (B : oui ou non)
`nursery`	a fréquenté l'école maternelle (B : oui ou non)
`higher`	veut suivre un enseignement supérieur (B : oui ou non)
`internet`	accès à Internet à domicile (B : oui ou non)
`romantic`	relation romantique (B : oui ou non)
`famrel`	qualité des relations familiales (C : 1 - très mauvais à 5 - excellent)
`freetime`	temps libre après l'école (C : de 1 - très faible à 5 - très élevé)
`goout`	sortir avec des amis (C : de 1 - très faible à 5 - très élevé)
`Dalc`	consommation d'alcool pendant les heures de travail (C : de 1 - très faible à 5 - très élevé)
`Walc`	consommation d'alcool la fin de semaine (C : de 1 - très faible à 5 - très élevé)
`health`	état de santé actuel (C : de 1 - très faible à 5 - très élevé)
`absences`	nombre d'absences scolaires (C : 0 à 93)
`G1`	note du premier trimestre (C : 0 à 20)
`G2`	note du second trimestre (C : 0 à 20)
`G3`	note du dernier trimestre (C : 0 à 20)

*Description : B - Binaire, C - Continue, N - Nominale.

a 0 - aucune, 1 - enseignement primaire (4e année), 2 - de la 5e à la 9e année, 3 - enseignement secondaire ou 4 - enseignement supérieur.

b enseignant, lié aux soins de santé, services civils (par exemple, administration ou police), à domicile ou autre.

A.5. Résultats du chapitre 4 pour la régression séquentielle

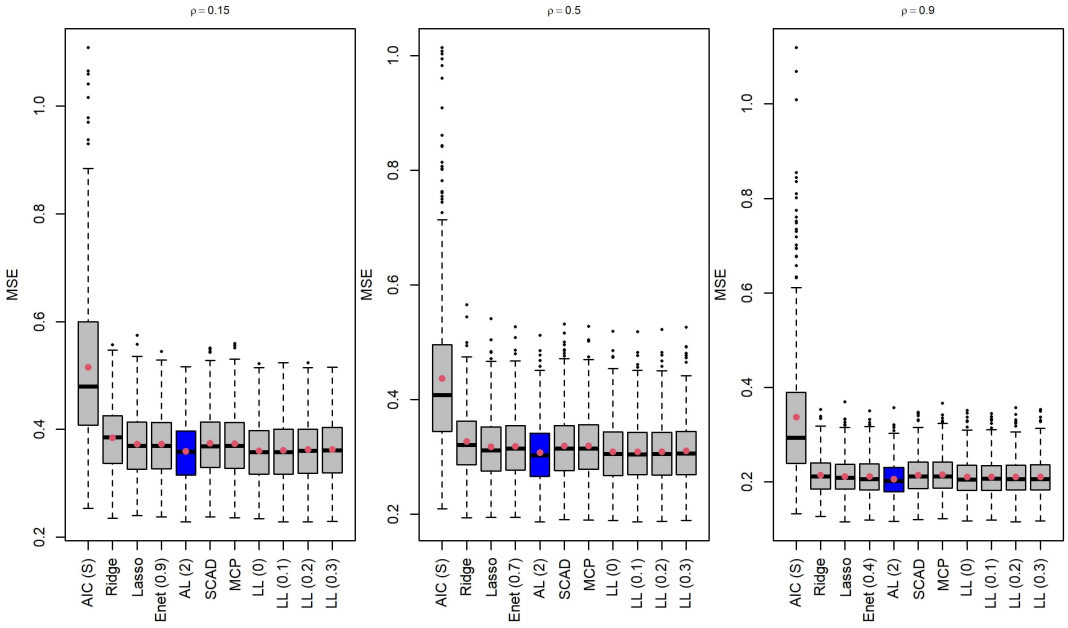


Figure A.1. *MSE* en fonction des différentes méthodes pour l'exemple 1. Le point rouge indique la moyenne et la boîte bleue indique la plus petite moyenne.

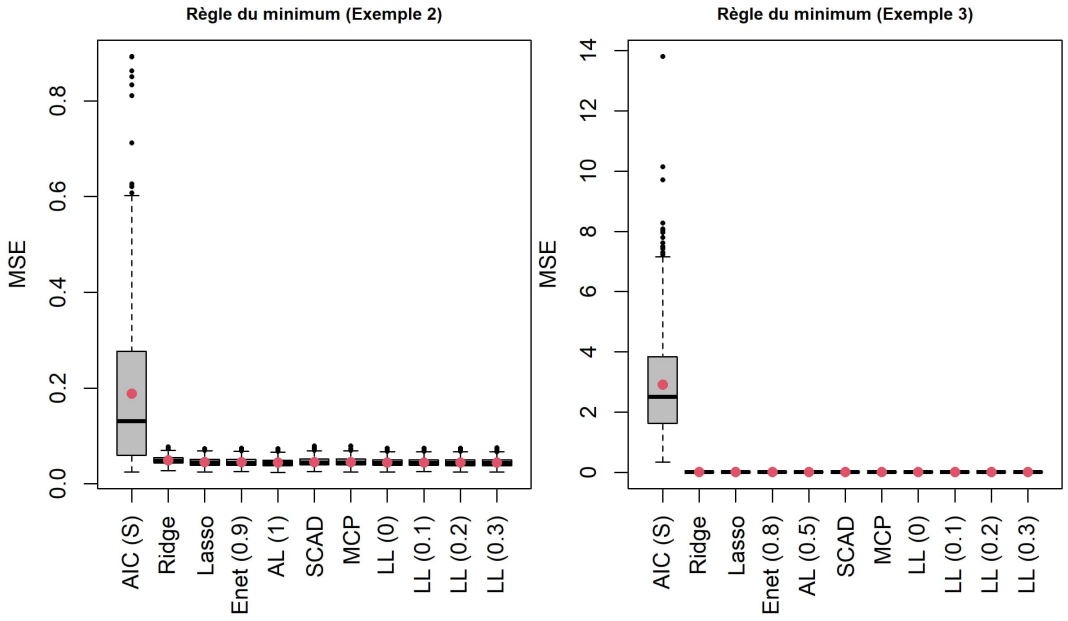


Figure A.2. *MSE* en fonction des différentes méthodes pour les exemples 2 et 3. Le point rouge indique la moyenne et la boîte bleue indique la plus petite moyenne.