

Régularisation et sélection de variables par le biais de la vraisemblance pénalisée

Mohammed El Anbari El Anbari

▶ **To cite this version:**

Mohammed El Anbari El Anbari. Régularisation et sélection de variables par le biais de la vraisemblance pénalisée. Mathématiques générales [math.GM]. Université Paris Sud - Paris XI; Université Cadi Ayyad (Marrakech, Maroc), 2011. Français. NNT : 2011PA112297 . tel-00661689

HAL Id: tel-00661689

https://theses.hal.science/tel-00661689

Submitted on 20 Jan 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

UNIVERSITÉ PARIS-SUD 11

جامعة القاضي عياض

UNIVERSITE CADI AYYAD
FACULTE DES SCIENCES SEMLALIA

THÈSE

Présentée pour obtenir

LE GRADE DE DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ CADI AYYAD ET DE L'UNIVERSITÉ PARIS-SUD 11

Spécialité : Mathématiques

par

Mohammed EL ANBARI

Regularization and variable selection using penalized likelihood

Soutenue le 14 Décembre 2011 après avis des rapporteurs

M. Stéphane GIRARD
M. Stéphane CANU
M. Idir OUASSOU

devant la Commission d'examen composée de :

M. Stéphane GIRARD	(Rapporteur)
M. Idir OUASSOU	(Rapporteur)
M. Brahim OUHBI	(Examinateur)
M. Gilles CELEUX	(Directeur de thèse)
M. Abdallah MKHADRI	(Directeur de thèse)
M. Mohamed TISSAFI IDRISSI	(Président du jury)

Remerciements

Mes premiers remerciements vont d'abord à mes directeurs de thèse Abdallah Mkhadri et Gilles Celeux, qui m'ont fait découvrir le domaine de l'apprentissage statistique en DESA puis en thèse. Je vous remercie pour votre confiance, votre soutien permanent, votre disponibilité et vos conseils qui m'ont permis de progresser durant ces années, en particulier dans les moments difficiles. Vous avez su me faire profiter de vos expériences et de votre goût pour la recherche. J'ai pris beaucoup de plaisir à travailler avec vous et j'espère que cette fin de thèse n'est pas un point final à notre collaboration.

J'adresse mes sincères remerciements à Stéphane Girad, Stéphane Canu et Idir Ouassou pour l'intérêt qu'ils ont porté à mon travail en acceptant de rapporter ma thèse. Je remercie également Brahim Ouhbi de m'avoir fait l'honneur de sa présence dans le jury et Mohamed Tissafi Idrissi pour avoir accepté de le présider.

Un merci tout particulier à Christian P. Robert et Jean-Michel Marin (les grands bayésiens), c'est un grand honneur pour moi de travailler avec vous et j'espère que notre collaboration continuera !

Un grand merci à Valérie Lavigne dont l'efficacité et la gentillesse ont rendu agréables et simples les nombreuses formalités administratives de la fin de thèse. Merci beaucoup Valérie !

Je remercie aussi également et chaleureusement le programme Maroc-Stic - en le nom de Michel Mauny et Driss Aboutajdine - qui m'a octroyé une bourse pour effectuer ma thèse en cotutelle entre les universités Paris-Sud 11 et Cadi Ayyad.

Enfin, je tiens à remercier mes proches, famille et amis : mes parents ; mes soeurs et frères Souad, Abdelouahed, Khalid et Mina ; ma femme Laila ; mes amis en France et au Maroc, chacun par son nom ; toute la famille El Anbari, toute la famille Amrani et toute la famille Barrak.

Table des matières

. **Présentation générale** **1**

Chapitre 1. Revue bibliographique sur la sélection de variables en grande dimension **5**

1.1. Introduction 5
1.2. Méthodes de régularisation 6
1.2.1. La régression *Ridge* 6
1.2.2. La régression *Lasso* (least absolute shrinkage and selection operator) 6
1.2.3. La régression *adaptive Lasso* 7
1.2.4. La régression *Dantzig Selector* 7
1.2.5. Les régressions *Scad* et *Mcp* 8
1.2.5.1. La régression *Scad* (smoothly clipped absolute deviation) . 8
1.2.5.2. La régression *Mcp* (minimax concave penalty) 9
1.2.6. Méthodes de régularisation pour des données corrélées 9
1.2.6.1. La régression *Elastic-net* 9
1.2.6.2. La régression *Weighted fusion* 10
1.2.7. Méthodes de régularisation pour des données ordonnées 10
1.2.7.1. La régression *Fused Lasso* 10
1.2.7.2. La régression *Smooth Lasso* 11
1.3. Chemin de régularisation 11
1.3.1. *Coordinate Descent* pour le *Lasso* 12
1.3.2. *Coordinate Descent* pour L'*adaptive Lasso* 14
1.3.3. Dantzig Selector 14
1.3.4. *Coordinate Descent* pour *Scad* et *Mcp* 15
1.3.4.1. *Coordinate Descent* pour *Scad* 15
1.3.4.2. *Coordinate Descent* pour *Mcp* 15
1.3.4.3. Convergence 15
1.3.5. *Coordinate Descent* pour *Elastic-net* 16

1.3.6. *Coordinate Descent* pour Weighted Fusion et SLasso 16
1.4. Application des méthodes décrites sur la base de données PAC 17
1.4.1. Description des données 17
1.4.2. Sélection des paramètres de lissage 17
1.4.3. Performances des méthodes sur la base de données PAC 17
1.5. Discussion 20
1.A. Code R commenté 27
1.A.1. Lasso 27
1.A.2. L'adaptive Lasso 28
1.A.3. Scad 29
1.A.4. Mcp 29
1.A.5. Weighted Fusion 30
1.A.6. Smooth Lasso 31

Chapter 2. Penalized regression combining the L_1 norm and a correlation based penalty 33

2.1. Introduction 33
2.2. Methodology 35
2.2.1. The L_1CP criterion 35
2.2.2. The L_1CP procedure 36
2.3. The grouping effect and tuning parameters selection 38
2.3.1. The grouping effect 38
2.3.2. Tuning parameters selection 39
2.4. Numerical experiments 39
2.4.1. Examples $1-6$ 42
2.4.2. Example 7 42
2.5. Real data sets Experiments 45
2.5.1. US crim data 45
2.5.2. GC-Retention PAC data 45
2.6. Discussion 47

Chapter 3. The adaptive Gril estimator with a diverging number of parameters 51

3.1. Introduction 51
3.2. Different Gril estimators 54
3.2.1. Doubly regularized techniques 54
3.2.2. A computational algorithm 55
3.2.3. Statistical properties of Different Gril estimators 56
3.3. The adaptive Gril estimator 57
3.3.1. The grouping effect of AdaCnet 57
3.3.2. Model selection consistency for AdaGril when p diverges 58

3.3.2.1. Mean Sparsity Inequality . . . 58
3.3.2.2. Oracle properties . . . 59
3.3.3. Sparsity inequality for AdaGril estimator . . . 60
3.4. Computation and tuning parameters selection . . . 62
3.5. Numerical study . . . 62
3.6. Discussion . . . 67
3.A. Proofs . . . 67
3.A.1. Proof of Lemma 3.3.1 . . . 67
3.A.2. Proof of Theorem 3.3.1 . . . 69
3.A.3. Proof of Theorem 3.3.2 . . . 71
3.A.4. Proof of Theorem 3.3.3 . . . 74
3.A.5. Proof of Theorem 3.3.4 . . . 76

Chapter 4. Regularization in regression: comparing Bayesian and frequentist methods in a poorly informative situation 81

4.1. Introduction . . . 81
4.2. Zellner's g-priors . . . 83
4.3. Mixtures of g-priors . . . 85
4.4. Numerical comparisons . . . 87
4.4.1. Regularization methods . . . 88
4.4.2. Numerical experiments on simulated datasets . . . 89
4.4.3. Real datasets . . . 98
4.5. Conclusion . . . 100

Chapter 5. Regularization and variable selection using the SF-MCP 105

5.1. Introduction . . . 105
5.2. SLasso and MNet methods . . . 107
5.3. The SF-Mcp method . . . 108
5.4. Numerical experiments . . . 109
5.4.1. Simulation study . . . 109
5.4.2. Real dataset . . . 110
5.5. conclusion . . . 112

Chapter 6. Conclusions et Perspectives 115

6.1. Bilan . . . 115
6.2. Perspectives . . . 117

Présentation générale

La grande dimension des bases de données, est devenu une caractéristique de nombreux domaines de la statistique contemporaine. Nous pouvons rencontrer de telles situations en génomique, en protéomique, en imagerie biomédicale, en imagerie fonctionnelle par résonance magnétique (RMN), en classification des tumeurs, en traitement du signal, en analyse d'image et en finance. Pour ce genre de bases de données, le nombre de variables peut être beaucoup plus grand que la taille de l'échantillon.

En modélisation statistique, il est souvent souhaitable d'avoir une bonne qualité de prédiction combinée à une représentation parcimonieuse. Comme mentionné dans les exemples cités ci-dessus, les bases de données modernes sont généralement caractérisées par un grand nombres de variables explicatives, d'où la parcimonie est devenue un objectif d'une grande importance.

Le *Lasso* (Tibshirani, 1996) a ouvert une nouvelle porte à la sélection de variables en régression linéaire. En imposant une contrainte sur la norme l_1 des coefficients, le *Lasso* permet de faire de la sélection automatique de variables en estimant certaines composantes du vecteur de régression par 0.

Bien qu'il ait connu le succès dans de nombreuses situations, le *Lasso* peut produire des résultats moins satisfaisante dans certaines situations : (1) le nombre de variables explicatives est beaucoup plus grand que le nombre d'observations, (2) les variables explicatives sont fortement corrélées et forment des "groupes".

Dans cette thèse, nous nous plaçons dans le cadre de la régression linéaire. L'objectif des travaux de cette thèse est la proposition de méthodes de sélection de variables dans le cadre des modèles linéaires. Nous traitons ce problème de deux points de vues : un point de vue fréquentiel et un point de vue bayésien.

D'un point de vue fréquentiel, nous proposons des techniques de modélisation, qui produisent des modèles parcimonieux et permettent de résoudre les problèmes (1) et (2), soulignés ci-dessus et dont souffre la méthode *Lasso*.

D'un point de vue bayésien, nous proposons une approche globale de sélection de variables en régression construite sur les lois a priori g de Zellner dans une approche similaire mais non identique à celle de Liang et al. (2008).

Les chapitres peuvent être lus indépendamment. Les preuves et les calculs techniques sont reportés dès que possible en annexe du chapitre correspondant pour faciliter la lecture de ce manuscrit.

Chapitre 1 : Revue bibliographique sur la sélection de variables en grande dimension

Dans ce chapitre, nous passons en revue les principales techniques de régularisation et de sélection de variables qui ont été proposées ces dernières années pour la régression linéaire. D'abord nous rappelons les définitions des techniques et les situations pour lesquelles elles sont adaptées. Puis, nous discutons les principaux algorithmes disponibles permettant de retrouver l'ensemble des solutions (chemin de régularisation) pour une grille de valeurs des paramètres de lissage. Nous donnons un intérêt spécial à l'algorithme *Coordinate Descent*, vu sa simplicité et son adaptation pour plusieurs techniques de régularisation. Ensuite, une base de données du domaine chimique comportant un nombre de variables plus grand que la taille de l'échantillon est utilisée pour illustrer les différentes techniques présentées. Enfin, nous donnons en annexe des codes commentés utilisant des librairies sous le logiciel `R`, décrivant les différentes étapes d'estimation des paramètres pour les méthodes exposées.

Chapitre 2 : Penalized regression combining the l_1 norm and a correlation based penalty

Ce travail a été réalisé sous la supérvision de mon directeur de thèse Abdallah Mkhadri. La sélection de variables peut être difficile, en particulier dans les situations où un grand nombre de variables explicatives est disponible, avec la présence possible de corrélations élevées entre elles, comme le cas des données d'expression génétique par exemple. Dans ce chapitre, nous proposons une nouvelle méthode de régression linéaire pénalisée, appelée L_1CP, pour estimer les paramètres inconnus et sélectionner les variables importantes simultanément . De plus, elle encourage un effet de groupe : les variables fortement corrélées ont tendance à être toutes incluses ou toutes exclues du modèle. La méthode est fondée sur les moindres carrés pénalisés, où la pénalité utilisée est une combinaison de la pénalité l_1 et de la pénalité CP proposée par Tutz and Ulbricht (2009). Le terme l_1 de la pénalité permet de donner un modèle parcimonieux, et donc interprétable, tandis que l'autre partie de la pénalité contient un terme qui fait un lien explicite entre la force de pénalisation et la corrélation entre les variables explicatives. La méthode proposée est particulièrement utile lorsque le nombre de variables explicatives est beaucoup plus grand que la taille de l'échantillon. D'un point de vue algorithmique, nous montrons que l'ensemble des solutions (chemin de régularisation) peut être obtenu efficacement en utilisant les algorithmes disponibles pour le Lasso, en montrant que la méthode proposée est équivalente à un problème Lasso sur des données transformées des données de départ. Nous comparons les performances de la méthode proposée avec des méthodes concurrentes sur une série d'exemples simulés et de données réelles.

Chapitre 3 : The adaptive Gril estimator with a diverging number of parameters

Ce travail a été réalisé sous la supérvision de mon directeur de thèse Abdallah Mkhadri. Dans ce chapitre, nous considérons le problème d'estimation et de sélection de variable en grande dimension en régression. Nous supposons que le nombre de variables p est une fonction divergente de la taille de l'échantillon n ($p = p(n)$ et $p \to \infty$ lorsque $n \to \infty$). Pour de telles situations, nous proposons l'ensemble d'estimateurs *Adaptive Generalized Ridge-Lasso* (AdaGril) qui sont une généralisation de l'estimateur *Adaptive elastic-net*. Les estimateurs proposés combinent les avantages de l'*Adaptive Lasso* avec l'effet d'un terme quadratique permettant de résoudre le problème de la colinéarité qui est une caractéristique des données de grande dimension. D'un point de vue théorique, nous montrons d'abord l'effet de groupement de la méthode AdaCnet (un type d'estimateur AdaGril) dans le cas particulier de corrélations égales entre toutes les variables. Ensuite, nous montrons que sous de faibles conditions de régularité, les estimateurs AdaGril possèdent la propriété *Oracle* au sens de Donoho and Johnstone, 1994; Fan and Li, 2001; Fan and Peng, 2004), assurant par conséquent une performance optimale en grande dimension. Donc, les estimateurs proposés réalisent deux objectifs importants simultanément : ils permettent de résoudre le problème de colinéarité tout en possédant la propriété *Oracle*. Nous montrons ensuite que les estimateurs proposés possèdent une inégalité de parcimonie (*sparsity inequality* (SI)), i.e., inégalité avec un majorant qui est fonction du nombre de composantes non nulles du *vrai* coefficient de régression. Cette inégalité est obtenue en imposant une condition du type RE (*Restricted Eigenvalue*), similaire à celle demandée pour obtenir une inégalité de parcimonie pour le *Lasso* (cf. Bickel et al. (2009)). D'un point de vue pratique, des exemples simulés pour lequels nous permettons au nombre de variables de dépendre de la taille de l'échantillon, sont utilisés pour comparer les performances de l'ensemble des estimateurs proposés avec des méthodes concurrentes.

Chapitre 4 : Regularization in regression : comparing Bayesian and frequentist methods in a poorly informative situation

Ce travail a été réalisé sous la supérvision de mon directeur de thèse Gilles Celeux, et en collaboration avec Jean-Michel Marin et Christian P. Robert. Dans ce chapitre nous traitons le problème de sélection de variables en régression d'un point de vue bayésien. Nous proposons une approche globale de sélection bayésienne de variables construite sur des lois a priori g de Zellner dans une approche similaire mais non identique à celle de Liang et al. (2008). Nous proposons deux lois a priori g de Zellner qui ne nécessitent aucune calibration. La première est la loi a priori de Jeffery qui n'est pas *stable par translation* (les performances sont affectées par l'ajout d'une valeur constante à toute les observations), tandis que la deuxième permet de résoudre ce problème en imposant au moins une variable dans chaque modèle. Avoir des formes explicites des lois marginales est un avantage majeur de l'utilisation d'une loi a priori g de Zellner. A travers une série d'exemples simulés et des bases de données réelles, nous comparons les approches de régularisation bayésienne et fréquentielle dans un contexte peu informatif où le nombre de variables est presque égal au nombre d'observations.

Chapitre 5 : Regularization and variable selection using the SF-MCP

Ce travail a été réalisé sous la supérvision de mon directeur de thèse Abdallah Mkhadri. En plus de la limite des performances de la méthode dans les deux situations (1) et (2) cités dessus, le Lasso produit des estimations avec un biais qui n'est pas négligeable pour les grandes valeurs du coefficients de régression. Pour résoudre ce problème de biais, des alternatives au *Lasso* sont proposées : la méthode SCAD (Fan and Li, 2001) et la méthode MCP (Zhang, 2010). En outre, certains types de données peuvent avoir un certain ordre sur les variables explicatives, ce qui est le cas par exemple en protéomique, où les variables sont les longeurs d'ondes de la lumière. Dans ce chapitre, nous proposons une méthode de sélection de variables capable de traiter des bases de données, possédant des variables ordonnées, tout en réduisant le biais produit lors de l'estimation des coefficients. La méthode est basée sur les moindres carrés pénalisés, où la pénalité utilisée est une combinaison de la pénalité MCP et une pénalité du type l_2 entre les composantes successives du vecteur de régression. Nous appliquons l'algorithme *Coordinate Descent* pour retrouver l'ensemble des solutions de la méthodes proposée pour une grille de valeurs des paramètres de lissage. Des exemples simulés et une base de données réelles sont utilisés pour comparer les performances de la méthode proposée avec des méthodes concurrentes.

Le chapitre 3 est un article à paraître dans Communications in Statistics-Theory and Methods (Marcel Dekker), le chapitre 4 est à paraître dans Bayesian Analysis (Revue internationale de la société savante des Statisticiens Bayesiens, le chapitre 2 est soumis à Synkhaya B (Revue de la société savante des Statisticiens Indiens). Le chapitre 5 est une communication aux journées : *Applied Stochastic Models and Data Analysis* (ASMDA2011, Rome, Italy, 7 - 10 June 2011).

Chapitre 1

Revue bibliographique sur la sélection de variables en grande dimension

1.1 Introduction

Dans de nombreuses et importantes applications statistiques modernes, le nombre de variables ou de paramètres p est beaucoup plus grand que le nombre d'observations n. En radiologie ou en imagerie biomédicale par exemple, nous pouvons collecter, pour une image donnée, un nombre de mesures beaucoup plus petit que le nombre inconnu de pixels. Les bases de données en grande dimension se posent aussi fréquemment en génomique. L'étude d'expression génétique est un exemple typique : nous disposons d'un nombre relativement faible d'observations (en dizaines), tandis que le nombre total de gènes est en milliers.
Donc, dans de nombreux domaines de recherche, les scientifiques doivent travailler avec des matrices de données avec p variables et n observations où $n << p$. Dans ce chapitre, nous considérons un tel problème dans le cas des modèles linéaires. Nous voulons estimer le paramètre inconnu $\boldsymbol{\beta} \in \mathbb{R}^p$ du modèle linéaire suivant :

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{1.1}$$

où $\mathbf{y} \in \mathbb{R}^n$, est la variable réponse, $\mathbf{X} = (\mathbf{x}_1|...|\mathbf{x}_p)$ est la matrice des variables explicatives de dimension $n \times p$, et $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ est le vecteur des erreurs stochastiques. Nous pouvons vouloir estimer $\boldsymbol{\beta}$, soit pour prédire $\mathbf{y}$ pour des valeurs futures de $\mathbf{X}$, soit pour examiner la relation entre la variable réponse $\mathbf{y}$ et les variables explicatives $\mathbf{X}$. Pour le premier but, la qualité de prédiction est importante, tandis que pour le deuxième but la complexité du modèle est plus importante. Il est connu que, dans ce contexte, la régression des moindres carrés ordinaire (OLS) ne donne pas de bonnes performances ni en termes de qualité de prédiction, ni en termes de complexité de modèle. Plusieurs méthodes de régression régularisées ont été développées ces dernières décennies pour surmonter les défauts de la régression OLS,

commençant avec la régression Ridge (Hoerl and Kennard, 1970), suivi de la régression Bridge (Frank and Friedman, 1993), la régression Garotte (Breiman, 1995), la régression Lasso (Tibshirani, 1996), la régression Scad (Fan and Li, 2001), la régression Elastic-net (Zou and Hastie, 2005), la régression Dantzig Selector (Candes and Tao, 2007), la régression Mcp (Zhang, 2010) et plusieurs autres.

La suite de chapitre est organisée de la manière suivante : dans la Section 1.2, nous rappelons la définition d'un nombre de méthodes de régularisation. La Section 1.3 donne un aperçu sur les algorithmes permettant de retrouver les solutions des problèmes régularisés exposés dans la Section 1.2. Une base de données du domaine chimique pour laquelle $n << p$ est utilisée pour illustrer les méthodes à la Section 1.4. Un code commenté, utilisant des librairies sous le logiciel `R` est donné en annexe.

1.2 Méthodes de régularisation

Dans la suite de ce chapitre, nous supposons que $\mathbf{y}$ est centré et que chaque variable explicative $\mathbf{x}_j = (x_{1j}, ..., x_{nj})^t$ est normalisée :

$$\sum_{i=1}^{n} y_i = 0, \ \sum_{i=1}^{n} x_{ij} = 0 \ \text{ et } \ \frac{1}{n}\sum_{i=1}^{n} x_{ij}^2 = 1, \ \text{ pour } \ j = 1, ..., p. \tag{1.2}$$

1.2.1 La régression *Ridge*

Pour obtenir une meilleure prédiction, Hoerl and Kennard (1970) ont proposé la régression *Ridge*. C'est la solution du problème pénalisé suivant :

$$\hat{\boldsymbol{\beta}}_{\text{Ridge}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \frac{\lambda}{2}\sum_{j=1}^{p}\beta_j^2, \tag{1.3}$$

où λ est un paramètre de lissage. La régression *Ridge* rétrécit l'estimateur des moindres carrés ordinaire vers 0. Ce faisant, elle donne un estimateur biaisé, mais avec une variance plus petite que celle de l'estimateur des moindres carrés ordinaire. La régression *Ridge* permet de rétrécir les coefficients, mais elle n'en annule aucun, donc elle ne donne pas des modèles interprétables.

1.2.2 La régression *Lasso* (least absolute shrinkage and selection operator)

La méthode *Lasso* est certainement la plus populaire des méthodes de régularisation, elle a été proposée par Tibshirani (1996). Cette méthode minimise la somme des carrés résiduels en imposant une contrainte sur la norme l_1 des coefficients. Elle s'écrit sous la

forme penalisée suivante :

$$\hat{\boldsymbol{\beta}}(\text{Lasso}) = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|, \tag{1.4}$$

où λ est un paramètre de lissage. Elle a le double effet de rétrécir les coefficients de $\boldsymbol{\beta}$, permettant de diminuer le biais à l'instar de la méthode *Ridge*, mais surtout de faire de la selection automatique de variables, en annulant certains coefficients de $\boldsymbol{\beta}$, pour des valeurs suffisamment grandes du paramètre λ.

1.2.3 La régression *adaptive Lasso*

L'adaptive Lasso a été proposé par Zou (2006). C'est une version pondérée du Lasso classique. C'est une solution du problème penalisé suivant :

$$\hat{\boldsymbol{\beta}}_{\text{AdaLasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \hat{\omega}_j |\beta_j|, \tag{1.5}$$

où $\hat{\boldsymbol{\omega}} = (\hat{\omega}_1, ..., \hat{\omega}_p)^t$ est un vecteur de poids donnés par un estimateur initial et λ est un paramètre de lissage. Elle résout le problème de consistance en sélection du *Lasso*, en pénalisant différemment les coefficients de $\boldsymbol{\beta}$.

1.2.4 La régression *Dantzig Selector*

Candes and Tao (2007) ont proposé le *Dantzig Selector* comme une alternative au *Lasso*. C'est la solution du problème d'optimisation suivant :

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|_1 \quad \text{sujet à} \quad \|\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)\|_\infty \leq \lambda, \tag{1.6}$$

où $\lambda \geq 0$ est un paramètre de lissage. La contrainte $\|\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)\|_\infty \leq \lambda$ peut être vue comme une relaxation de l'équation normale dans la régression linéaire classique. Ce critère peut être écrit sous la forme équivalente :

$$\min_{\beta \in \mathbb{R}^p} \|\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)\|_\infty \quad \text{sujet à} \quad \|\beta\|_1 \leq \lambda. \tag{1.7}$$

Ici $\|.\|_\infty$ est la norme l_∞ : $\|\mathbf{a}\|_\infty = \max_{k=1,...,p} |a_k|$, pour tout vecteur $\mathbf{a} = (a_1, ..., a_p)^t \in \mathbb{R}^p$. Sous cette dèrnière forme le *Dantzig Selector* ressemble au *Lasso*, en remplaçant la perte des moindres carrés par la norme l_∞ de son gradient. Mais cette remarque ne veut pas dire que les deux méthodes produisent les mêmes solutions.

1.2.5 Les régressions *Scad* et *Mcp*

Lorsque p est grand et le nombre de variables pertinentes est petit, Fan and Li (2001) et Zhang (2010) ont remarqué que :

1. Pour écarter les variables parasites, le paramètre de lissage λ du problème *Lasso* doit être grand, ce qui provoque un biais dans les variables retenues.
2. La diminution de λ pour réduire le biais cause l'introduction de beaucoup de variables parasites.

Pour résoudre ces problèmes, Fan and Li (2001) ont proposé l'estimateur *Scad*, tandis que Zhang (2010) a proposé l'estimateur *Mcp*.

1.2.5.1 La régression *Scad* (smoothly clipped absolute deviation)

Etant donné $\lambda \geq 0$ et $\gamma > 2$, considérons la fonction définie sur $[0, \infty)$ par :

$$q_{\lambda,\gamma}(t) = \begin{cases} \lambda t, & \text{si } t \leq \lambda \\ \frac{\gamma\lambda t - 0.5(t^2+\lambda^2)}{\gamma-1}, & \text{si } \lambda < t \leq \lambda\gamma \\ \frac{\lambda^2(\gamma^2-1)}{2(\gamma-1)}, & \text{si } t > \gamma\lambda. \end{cases} \tag{1.8}$$

Soit le critère pénalisé :

$$N(\boldsymbol{\beta}, \lambda, \gamma) = \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \sum_{j=1}^{p} q_{\lambda,\gamma}(|\beta_j|). \tag{1.9}$$

L'estimateur Scad est défini par :

$$\hat{\boldsymbol{\beta}}_{\text{Scad}} = \arg\min_{\boldsymbol{\beta}} N(\boldsymbol{\beta}, \lambda, \gamma). \tag{1.10}$$

Il consiste donc à remplacer la pénalité l_1 dans le *Lasso* par la pénalité (1.8). Pour avoir une idée du fonctionnement de cette pénalité, il est intéressant de considérer sa dérivée :

$$q'_{\lambda,\gamma}(t) = \begin{cases} \lambda, & \text{si } t \leq \lambda \\ \frac{\gamma\lambda - t}{\gamma-1}, & \text{si } \lambda < t \leq \lambda\gamma \\ 0, & \text{si } t > \gamma\lambda. \end{cases} \tag{1.11}$$

Cette pénalité permet de :

1. Pénaliser plus sévèrement que le *Lasso* les petits coefficients, conduisant à des modèles plus parcimonieux.
2. Pénaliser moins les grands coefficients, ce qui réduit leur biais.

1.2.5.2 La régression *Mcp* (minimax concave penalty)

Pour $\lambda \geq 1$ et $\gamma > 1$, considérons la fonction définie sur $[0, \infty)$ par :

$$p_{\lambda,\gamma}(t) = \begin{cases} \lambda t - \frac{t^2}{2\gamma}, & \text{si } t \leq \gamma\lambda, \\ \frac{1}{2}\gamma\lambda^2, & \text{si } t > \gamma\lambda. \end{cases} \tag{1.12}$$

Soit le critère pénalisé :

$$M(\boldsymbol{\beta}, \lambda, \gamma) = \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \sum_{j=1}^{p} p_{\lambda,\gamma}(|\beta_j|). \tag{1.13}$$

L'estimateur Mcp est défini par :

$$\hat{\boldsymbol{\beta}}_{\text{Mcp}} = \arg\min_{\boldsymbol{\beta}} M(\boldsymbol{\beta}, \lambda, \gamma). \tag{1.14}$$

Il consiste donc à remplacer la pénalité l_1 dans le *Lasso* par la pénalité (1.12), qui a pour dérivée :

$$p'_{\lambda,\gamma}(t) = \begin{cases} \lambda - \frac{t}{\gamma}, & \text{si } t \leq \gamma\lambda, \\ 0, & \text{si } t > \gamma\lambda. \end{cases} \tag{1.15}$$

La pénalité *Mcp* commence en appliquant le même taux de pénalisation que le *Lasso*, mais elle le relaxe d'une manière continue, jusqu'à ce que $t > \gamma\lambda$, où le taux de pénalisation devient nul.

1.2.6 Méthodes de régularisation pour des données corrélées

1.2.6.1 La régression *Elastic-net*

Zou and Hastie (2005) ont constaté que le *Lasso* posséde d'autres limitations :

1. Le nombre de variables sélectionnées par le *Lasso* est limité par la taille de l'échantillon, donc ce nombre ne peut pas dépasser n.
2. Le *Lasso* a tendance à sélectionner une seule variable parmi un groupe de variables très corrélées.
3. Les performances de prédiction du *Lasso* ne sont pas aussi bonnes que la régression *Ridge* s'il existe une forte corrélation entre les variables explicatives.

Pour résoudre ces 3 problèmes, Zou and Hastie (2005) ont proposé *l'elastic-net*. Cette méthode combine les deux pénalités l_1 et l_2. C'est la solution du problème pénalisé suivant :

$$\hat{\boldsymbol{\beta}}_{\text{Enet}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda\left(\alpha\sum_{j=1}^{p}|\beta_j| + \frac{1}{2}(1-\alpha)\sum_{j=1}^{p}\beta_j^2\right), \tag{1.16}$$

où $\lambda \geq 0$ et $\alpha \in [0, 1]$ sont deux paramètres de lissage. Avec un bon équilibre des deux pénalités, *l'elastic-net* parvient à combiner les points forts du *Lasso* et du *Ridge*, tout en minimisant leurs inconvénients. Notons que les régressions *Ridge* et *Lasso*, sont deux cas particuliers de l'estimateur *elastic-net* correspondants respectivement à $\alpha = 0$ et $\alpha = 1$.

1.2.6.2 La régression *Weighted fusion*

Par les mêmes arguments qui ont motivé *l'Elastic-net*, Daye and Jeng (2009) ont proposé la méthode *Weighted fusion* :

$$\hat{\boldsymbol{\beta}}_{\text{Wfusion}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \frac{\mu}{p} \sum_{j=1}^{p-1} \sum_{j>i} \omega_{ij} (\beta_i - s_{ij}\beta_j)^2, \tag{1.17}$$

avec $\omega_{ij} = \frac{|\rho_{ij}|^\gamma}{1-|\rho_{ij}|}$, $\rho_{ij} = \mathbf{x}_i^t \mathbf{x}_j$ est la corrélation empirique entre les variables $\mathbf{x}_i$ et $\mathbf{x}_j$, $s_{ij} = \text{sign}(\rho_{ij})$ le signe de ρ_{ij} et $\lambda \geq 0$, $\mu \geq 0$, $\gamma > 0$ sont des paramètres de lissage. Les auteurs montrent que ce critère s'écrit sous la forme suivante :

$$\hat{\boldsymbol{\beta}}_{\text{Wfusion}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \mu \boldsymbol{\beta}^t \mathbf{Q} \boldsymbol{\beta}, \tag{1.18}$$

où

$$p \times \mathbf{Q} = \begin{pmatrix} \sum_{k\neq 1} w_{1k} & -s_{12}w_{12} & \cdots & -s_{1p}w_{1p} \\ -s_{12}w_{12} & \sum_{k\neq 2} w_{2k} & \cdots & \vdots \\ \vdots & \vdots & \ddots & -s_{p-1,p}w_{p-1,p} \\ -s_{1p}w_{1p} & \cdots & -s_{p-1,p}w_{p-1,p} & \sum_{k\neq p} w_{pk} \end{pmatrix}. \tag{1.19}$$

A noter que $\mathbf{Q}$ est symétrique et semi-définie positive. Soit $\mathbf{R}$ sa décomposition de Cholesky :

$$\mathbf{Q} = \mathbf{R}^t \mathbf{R}. \tag{1.20}$$

Cette remarque sera très utile pour décrire l'algorithme calculant $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$.

1.2.7 Méthodes de régularisation pour des données ordonnées

1.2.7.1 La régression *Fused Lasso*

Tibshirani et al. (2005) ont propsé le *Fused Lasso*. Cette méthode est utile lorsqu'il y a une structure d'ordre entre les variables. Elle s'écrit sous la forme pénalisée suivante :

$$\hat{\boldsymbol{\beta}}_{\text{FLasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \mu \sum_{j=2}^{p} |\beta_j - \beta_{j-1}|. \tag{1.21}$$

La première pénalité encourage la parcimonie des coefficients, tandis que la deuxième pénalité encourage la parcimonie de leurs différences. Le critère (1.21) conduit à un problème de programmation quadratique. Lorsque le nombre de variables p est grand, le problème devient difficile à résoudre.

1.2.7.2 La régression *Smooth Lasso*

Cette méthode a été proposée par Hebiri and van De Geer (2010) pour les situations où les coefficients de régression successifs varient lentement ou bien lorsque les variables sont ordonnées. C'est la solution au problème pénalisé suivant :

$$\hat{\boldsymbol{\beta}}_{\text{SLasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \mu \sum_{j=2}^{p} (\beta_j - \beta_{j-1})^2. \tag{1.22}$$

De la même manière que *Weighted fusion*, ce critère s'écrit sous la forme :

$$\hat{\boldsymbol{\beta}}_{\text{SLasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \mu \boldsymbol{\beta}^t \mathbf{R}^t \mathbf{R} \boldsymbol{\beta}, \tag{1.23}$$

où

$$\mathbf{R} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \ddots & \vdots \\ 0 & 1 & -1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -1 \end{pmatrix}. \tag{1.24}$$

1.3 Chemin de régularisation

Les statisticiens sont intéressés par l'obtention d'un estimateur $\hat{\boldsymbol{\beta}}$ pour une grille de valeurs de λ, allant d'une valeur maximale $\lambda_{\max}$ pour laquelle tous les coefficients pénalisés sont nuls, jusqu'à $\lambda = 0$ ou une valeur minimale $\lambda_{\min}$ pour laquelle le modèle devient trop grand ou cesse d'être identifiable. Nous parlons donc de *chemin de régularisation* $\{\hat{\boldsymbol{\beta}}(\lambda)\}_{\lambda_{\min} \leq \lambda \leq \lambda_{\max}}$.

En 2001, l'algorithme LARS (Efron et al., 2004) a fourni un moyen pour calculer efficacement le *chemin de régularisation* du Lasso avec un coût de calcul des moindres carrés ordinaires. De 2001 jusqu'à présent, d'autres algorithmes de calcul de chemin de régularisation ont apparu pour une variété de problèmes similaires au Lasso : Grouped lasso (Yuan and Lin, 2006), support-vector machine (Hastie et al., 2004), elastic net (Zou and Hastie, 2005), quantile regression (Li and Zhu, 2008), logistic regression and glms (Park and Hastie, 2007), Dantzig selector (James et al., 2009). Cependant, beaucoup d'algorithmes parmi eux ne possédent pas la propriété de la linéarité par morceaux de l'algorithme LARS, ce qui peut poser un certain nombre de problèmes de calcul.

Dernièrement, un grand intérêt a été donné à l'algorithme *Coordinate Descent* : (Shevade and Keerthi, 2003; Krishnapuram et al., 2005; Genkin et al., 2007; Friedman et al., 2007; Wu and Lange, 2008; Meier et al., 2008). Cet algorithme consiste à optimiser chaque paramètre séparément, tout en fixant les autres, et répéter la procédure jusqu'à convergence.

Pour sa simplicité et son adaptation à différentes techniques de régularisation, cet algorithme est devenu très compétitif avec le célèbre LARS. Pour ces raisons, nous allons nous concentrer dans ce chapitre sur cet algorithme pour retrouver les chemins de régularisation. Pour cela, nous aurons besoin de la fonction appelée le seuil doux (*soft threshold*) (Donoho and Johnstone, 1994) définie pour $\lambda \geq 0$ par :

$$S(t,\lambda) = \text{sign}(t)(|t| - \lambda)^+ \tag{1.25}$$

$$= \begin{cases} \hat{\beta} - \lambda, & \text{si } t > \lambda, \\ 0, & \text{si } |t| \leq \lambda, \\ \hat{\beta} + \lambda, & \text{si } t < -\lambda. \end{cases} \tag{1.26}$$

1.3.1 *Coordinate Descent* pour le *Lasso*

Cet algorithme a été proposé par Friedman et al. (2007). Considérons le problème de régression linéaire simple d'une variable réponse $\mathbf{y} = (y_1, ..., y_n)^t$ sur une variable explicative normalisée $\mathbf{x} = (x_1, ..., x_n)^t$. Notons z l'estimateur des moindres carrées ordinaires de β. Alors z est solution de :

$$-\mathbf{x}^t\mathbf{y} + \mathbf{x}^t\mathbf{x}z = 0. \tag{1.27}$$

Utilisant le fait que $\mathbf{x}^t\mathbf{x} = n$, nous avons :

$$z = \frac{1}{n}\sum_{i=1}^{n} x_i y_i. \tag{1.28}$$

D'une autre part, le problème *Lasso* posséde une solution analytique simple. C'est un seuil doux (*soft threshold*) de l'estimateur des moindres carrés ordinaire (1.28) :

$$\hat{\boldsymbol{\beta}}(\text{Lasso}) = S(z,\lambda) = \text{sign}(z)(|z| - \lambda)^+. \tag{1.29}$$

En effet, dans ce cas l'estimateur $\hat{\boldsymbol{\beta}}(\text{Lasso})$ est le minimiseur de la fonction objectif :

$$\begin{aligned} g(\beta) &= \frac{1}{2n}\sum_{i=1}^{n}(y_i - x_i\beta)^2 + \lambda|\beta| \\ &= \frac{1}{2n}\sum_{i=1}^{n} y_i^2 - \frac{1}{n}\beta\sum_{i=1}^{n} x_i y_i + \frac{1}{2n}\beta^2\sum_{i=1}^{n} x_i^2 + \lambda|\beta| \\ &= \frac{1}{2n}\sum_{i=1}^{n} y_i^2 - z\beta + \frac{1}{2}\beta^2 + \lambda|\beta|. \end{aligned} \tag{1.30}$$

La minimisation de (1.30) est équivalente à la minimisation de :

$$f(\beta) = -z\beta + \frac{1}{2}\beta^2 + \lambda|\beta|. \tag{1.31}$$

Si $z > 0$, alors nous devons avoir $\beta \geq 0$, car sinon nous pourrions prendre son opposé et obtenir une valeur inférieure pour la fonction objectif f. De même, si $z < 0$, alors nous devons choisir $\beta \leq 0$.
Cas 1 : Si $z > 0$. Puisque $\beta \geq 0$,

$$f(\beta) = -z\beta + \frac{1}{2}\beta^2 + \lambda\beta,$$

en différenciant par rapport à β, et en mettons l'égalité à 0, nous obtenons $\beta = z - \lambda$ et cela n'est possible que si le terme de droite est positif, alors dans ce cas, la solution est :

$$\hat{\boldsymbol{\beta}}(\text{Lasso}) = (z - \lambda)^+ = \text{sign}(z)(|z| - \lambda)^+.$$

Cas 2 : Si $z < 0$. Ceci implique que nous devons avoir $\beta \leq 0$, et ainsi

$$f(\beta) = -z\beta + \frac{1}{2}\beta^2 - \lambda\beta,$$

La différenciation par rapport à β et à la mise de l'égalité à zéro, nous donnent

$$\hat{\boldsymbol{\beta}}(\text{Lasso}) = z + \lambda = \text{sign}(z)(|z| - \lambda).$$

Mais, encore une fois, pour s'assurer que cela est possible, nous devons avoir $\beta \leq 0$, ce qui est réalisé en prenant

$$\hat{\boldsymbol{\beta}}(\text{Lasso}) = \text{sign}(z)(|z| - \lambda)^+.$$

Dans les deux cas, on obtient la forme désirée (1.29).

Dans le cas où $p \geq 2$, l'idée de l'algorithme *Coordinate Descent* est de fixer le paramètre de lissage λ du Lagrangien (1.4) et d'optimiser un seul paramètre à chaque fois, en gardant tous les autres fixes à leurs valeurs courantes.

Soit $\tilde{\boldsymbol{\beta}}_k^{(m)}$ l'estimateur courant de $\boldsymbol{\beta}_k$ pour la valeur λ du paramètre de lissage. Pour isoler $\boldsymbol{\beta}_j$, le critère (1.4) peut être écrit sous la forme :

$$I(\tilde{\boldsymbol{\beta}}^{(m)}, \boldsymbol{\beta}_j) = \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{k\neq j} x_{ik}\tilde{\boldsymbol{\beta}}_k^{(m)} - x_{ij}\boldsymbol{\beta}_j\right)^2 + \lambda\sum_{k\neq j}|\tilde{\boldsymbol{\beta}}_k^{(m)}| + \lambda|\boldsymbol{\beta}_j|. \tag{1.32}$$

Ceci peut être vu comme un problème Lasso univarié, dans lequel le vecteur des *résidus partiels* $\tilde{r}_{ij} = y_i - \tilde{y}_i^{(j)} = y_i - \sum_{k\neq j} x_{ik}\tilde{\boldsymbol{\beta}}_k^{(m)}$ est la variable réponse. Ce problème possède donc une solution explicite, et ceci permet d'avoir la mise à jour suivante :

$$\tilde{\boldsymbol{\beta}}_j^{(m+1)} \leftarrow S\left(\frac{1}{n}\sum_{i=1}^{n} x_{ij}\tilde{r}_{ij}, \lambda\right). \tag{1.33}$$

Le premier argument de $S(.)$ est le coefficient des moindres carrés simple de la régression du vecteur des *résidus partiels* sur la variable normalisée $\mathbf{x}_j$.

Algorithme 1. Etant donné la valeur courante $\tilde{\boldsymbol{\beta}}^{(m)}$ à l'itération m pour $m = 0, 1, ...,$ l'algorithme pour déterminer $\hat{\beta}$(Lasso) est le suivant :

i) Calcul du vecteur des *résidus partiels* :

$$\tilde{r}_{ij} = y_i - \sum_{k \neq j} x_{ik} \tilde{\boldsymbol{\beta}}_k^{(m)}, i = 1, ..., n. \tag{1.34}$$

ii) Calcul du coefficient des moindres carrés simple de la régression du vecteur (1.34) sur variable explicative normalisée $\mathbf{x}_j$:

$$\tilde{z}_j = \frac{1}{n} \sum_{i=1}^{n} x_{ij} \tilde{r}_{ij} = \frac{1}{n} \sum_{i=1}^{n} x_{ij} (y_i - \tilde{y}_i + x_{ij} \tilde{\beta}_j^{(m)}) = \frac{1}{n} \sum_{i=1}^{n} x_{ij} r_i + \tilde{\beta}_j^{(m)}, \tag{1.35}$$

où $\tilde{y}_i = \sum_{i=1}^{n} x_{ij} \tilde{\beta}_j^{(m)}$ est l'estimation courante de l'observation y_i et $r_i = y_i - \tilde{y}_i$ est le résidu courant.

iii) Mise à jour de $\tilde{\boldsymbol{\beta}}_j^{(m+1)}$ par :

$$\tilde{\boldsymbol{\beta}}_j^{(m+1)} \leftarrow S(\tilde{z}_j, \lambda) = \text{sign}(\tilde{z}_j)(|\tilde{z}_j| - \lambda)^+. \tag{1.36}$$

1.3.2 *Coordinate Descent* pour L'*adaptive Lasso*

Zou (2006) a montré que l'adaptive Lasso peut être vu comme un problème Lasso, en transformant les données originales. Donc, il est possible d'utiliser l'algorithme *Coordinate Descent* pour retrouver $\hat{\boldsymbol{\beta}}_{\text{AdaLasso}}$.

Algorithme 2. L'algorithme pour retrouver $\hat{\boldsymbol{\beta}}_{\text{AdaLasso}}$ est le suivant :

1. Construire la nouvelle matrice des variables explicatives $\mathbf{X}^{**} = (\mathbf{x}_1^{**}|...|\mathbf{x}_p^{**})$, avec $\mathbf{x}_j^{**} = \mathbf{x}_j / \hat{\omega}_j$ pour $j = 1, \ldots, p$.
2. Pour tout λ, résoudre le problème Lasso suivant en utilisant l'Algorithme 1 :

$$\hat{\boldsymbol{\beta}}^{**} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}^{**} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|.$$

3. L'estimateur $\hat{\boldsymbol{\beta}}_{\text{AdaLasso}}$ est donnée par : $\hat{\boldsymbol{\beta}}_{\text{AdaLasso}} = (\hat{\boldsymbol{\beta}}_1^{**} / \hat{\omega}_1, ..., \hat{\boldsymbol{\beta}}_p^{**} / \hat{\omega}_p)^t$.

1.3.3 Dantzig Selector

Pour tout λ donné, le critère d'optimisation du Sélecteur Dantzig (1.6) peut-être résolu en utilisant une procédure de programmation linéaire standard ; d'où le nom du Dantzig Selector, en l'honneur de George Dantzig, l'inventeur de la méthode du simplexe pour la programmation linéaire. James et al. (2009) ont proposé l'algorithme Dasso qui permet de construire un chemin de régularisation linéaire par morceaux par le biais d'un algorithme du type simplex séquentiel.

1.3.4 *Coordinate Descent* pour *Scad* et *Mcp*

Ces deux algorithmes ont été proposées par Breheny and Huang (2011).

1.3.4.1 *Coordinate Descent* pour *Scad*

Dans le cas unidimensionnel, l'estimateur *Scad* a la forme explicite suivante :

$$\hat{\beta}_{\text{Scad}} = f_{\text{Scad}}(z, \lambda, \gamma) = \begin{cases} S(z, \lambda), & \text{si } |z| \leq 2\lambda, \\ \frac{S(z, \gamma\lambda/(\gamma-1))}{1-1/(\gamma-1)}, & \text{si } 2\lambda < |z| \leq \gamma\lambda, \\ z, & \text{si } |z| > \gamma\lambda, \end{cases} \tag{1.37}$$

où z et S sont donnés par (1.28) et (1.25) respectivement. Soit $\tilde{\boldsymbol{\beta}}^{(m)}$ la valeur courante de $\hat{\boldsymbol{\beta}}_{\text{Scad}}$ à l'itération m pour $m = 0, 1, ...$ Donc la mise jour de $\hat{\boldsymbol{\beta}}_{\text{Scad}}$ est donnée par :

$$\tilde{\boldsymbol{\beta}}_j^{(m+1)} \leftarrow f_{\text{Scad}}(\tilde{z}_j, \lambda, \gamma),$$

où $\tilde{z}_j$ est calculé comme dans (1.35).

1.3.4.2 *Coordinate Descent* pour *Mcp*

Considérons le problème *Mcp* unidimensionnel. l'estimateur *Mcp* a la forme explicite suivante :

$$\hat{\beta}_{\text{Mcp}} = f_{\text{Mcp}}(z, \lambda, \gamma) = \begin{cases} \frac{S(z, \lambda)}{1-1/\gamma}, & \text{si } |z| \leq \gamma\lambda, \\ z, & \text{si } |z| > \gamma\lambda, \end{cases} \tag{1.38}$$

où z est donné par (1.28) et S est donnée par (1.25). Comme pour le Lasso, cette solution unidimensionnelle est employée par l'algorithme *Coordinate Descent* pour obtenir la prochaine mise à jour de la procédure de minimisation de la fonction objective. Soit $\tilde{\boldsymbol{\beta}}^{(m)}$ la valeur courante de $\hat{\boldsymbol{\beta}}_{\text{Mcp}}$ à l'itération m pour $m = 0, 1, ...$ La mise à jour de chaque coordonnée de $\hat{\boldsymbol{\beta}}_{\text{Mcp}}$ est donnée par :

$$\tilde{\boldsymbol{\beta}}_j^{(m+1)} \leftarrow f_{\text{Mcp}}(\tilde{z}_j, \lambda, \gamma),$$

où $\tilde{z}_j$ est calculé comme dans (1.35).

1.3.4.3 Convergence

Comme les fonctions *Scad* et *Mcp* ne sont pas convexes, alors il n'est pas garanti que l'algorithme de *Coordinate Descent* converge vers un minimum global. Cependant, il est possible pour les fonctions objectif (1.9) et (1.13) d'être convexes par rapport à la variable $\boldsymbol{\beta}$. En particulier, si c_* désigne la plus petite valeur propre de $n^{-1}\mathbf{X}^t\mathbf{X}$, Zhang (2011) a montré que le critère (1.13) est convexe en $\boldsymbol{\beta}$ si $\gamma > 1/c_*$, tandis que le critère (1.9) est convexe en $\boldsymbol{\beta}$ si $\gamma > 1 + 1/c_*$. Dans ce cas l'algorithme *Coordinate Descent* converge vers le minimum global de la fonction objectif.

1.3.5 *Coordinate Descent* pour *Elastic-net*

Pour l'estimateur Elastic-Net la mise à jour à l'étape $(m+1)$ est donnée par :

$$\tilde{\boldsymbol{\beta}}_j^{(m+1)} \leftarrow \frac{S(\tilde{z}_j, \lambda\alpha)}{1+\lambda(1-\alpha)}.$$

Cet algorithme a été proposé par van der Kooij (2007).

1.3.6 *Coordinate Descent* pour Weighted Fusion et SLasso

L'algorithme permettant de retrouver les estimateurs $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$ et $\hat{\boldsymbol{\beta}}_{\text{SLasso}}$ est basé sur la remarque intéressante suivante :

$$\hat{\boldsymbol{\beta}}_{\text{Wfusion}}, \hat{\boldsymbol{\beta}}_{\text{SLasso}} \in \arg\min_{\boldsymbol{\beta}} \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j| + \mu\boldsymbol{\beta}^t\mathbf{R}^t\mathbf{R}\boldsymbol{\beta}, \quad (1.39)$$

où $\mathbf{R}$ est donnée par (1.20) pour $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$ et par (1.24) pour $\hat{\boldsymbol{\beta}}_{\text{SLasso}}$.

En introduisant une augmentation des données de départ, les critères Weighted Fusion et SLasso peuvent être reformulés en un problème Lasso. Soient $\mathbf{X}^{**}$ et $\mathbf{y}^{**}$, la nouvelle matrice des variables explicatives et la nouvelle variable réponse respectivement :

$$\mathbf{X}^{**}_{(n+p)\times p} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\mu}\mathbf{R} \end{pmatrix}, \quad \mathbf{y}^{**}_{(n+p)} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}, \quad (1.40)$$

alors, il est facile de montrer que :

$$\hat{\boldsymbol{\beta}}_{\text{Wfusion}}, \hat{\boldsymbol{\beta}}_{\text{SLasso}} \in \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\sum_{i=1}^{n}\left(y_i^{**} - \sum_{j=1}^{p} x_{ij}^{**}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j|.$$

Ceci permet donc de retrouver les estimateurs $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$ et $\hat{\boldsymbol{\beta}}_{\text{SLasso}}$ par l'intermédiaire des algorithmes efficaces disponibles pour le Lasso. Pour retrouver ces deux estimateurs, la procédure est resumée comme suit :

---

1. Pour μ fixé, construire la nouvelle matrice des variables explicatives et la nouvelle variable réponse (1.40).
2. Pour tout λ, résoudre le problème Lasso :

$$\hat{\boldsymbol{\beta}}^{**} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2}\sum_{i=1}^{n}\left(y_i^{**} - \sum_{j=1}^{p} x_{ij}^{**}\beta_j\right)^2 + \lambda\sum_{j=1}^{p}|\beta_j|.$$

1.4 Application des méthodes décrites sur la base de données PAC

1.4.1 Description des données

Les données décrivent des indices de rétention GC (chromatographie en phase gazeuse) pour des composés aromatiques polycycliques, modélisés par des descripteurs moléculaires. Un ensemble de $n = 209$ composés aromatiques polycycliques (PAC) et $p = 467$ descripteurs moléculaires ont été utilisés dans cet exemple. Nous sommes donc dans une situation où $p >> n$. Dans cette partie, nous nous concentrons sur la démonstration de l'utilisation des méthodes de régularisation présentées dans ce chapitre sur cette base de données en utilisant des librairies **R**.

1.4.2 Sélection des paramètres de lissage

Tous les algorithmes présentés dans ce chapitre donnent un ensemble de solutions $\{\hat{\boldsymbol{\beta}}(\theta)\}$ pour une grille de valeurs du paramètre θ ($\theta = \lambda, \theta = (\lambda, \mu)$ ou $\theta = (\lambda, \mu, \gamma)$), laissant à l'utilisateur de choisir une solution particulière $\hat{\boldsymbol{\beta}}(\hat{\theta})$. Une approche générale consiste à utiliser l'erreur de prédiction pour faire ce choix. Si l'utilisateur est riche en données, il peut réserver un pourcentage (un tiers par exemple) de la base de données à cet effet. Il faut donc évaluer les performances de prévision pour chaque valeur θ, et choisir le modèle avec les meilleures performances. Alternativement, l'utilisateur peut utiliser la validation croisée pour choisir son modèle final. Dans ce qui suit nous donnons la description de la 10-*fold* :

Notons par $\mathbf{E}$ la base de données complète, et pour $\nu = 1, ..., 10$, notons par $\mathbf{E}$ - $\mathbf{E}^\nu$ et $\mathbf{E}^\nu$ les échantillons d'apprentissage et test respectivement. Pour tout θ et ν, nous cherchons l'estimateur $\hat{\beta}^{(\nu)}(\theta)$ de β en utilisant l'échantillon d'apprentissage. Construisons le critère de validation croisée défini par :

$$\mathrm{MSE}_{\mathrm{CV}}(\theta) = \sum_{\nu} \sum_{(y_k, \mathbf{x}_k) \in \mathbf{E}^\nu} \left\{ y_k - \mathbf{x}_k^t \hat{\beta}^{(\nu)}(\theta) \right\}^2 .$$

L'estimateur $\hat{\theta}$ de θ par validation croisée est celui qui minimise $\mathrm{MSE}_{\mathrm{CV}}(\theta)$.

1.4.3 Performances des méthodes sur la base de données PAC

Lasso Dans le présent document, toutes les méthodes suivent l'approche de Friedman et al. (2010) qui donne les solutions pour une grille de 100 valeurs pour λ qui sont equi-espacées sur l'échelle logarithmique. La Figure 1.1 donne le chemin de régularisation du Lasso, ainsi que l'erreur de validation croisée calculée par du 10-*fold*. Pour la base de données PAC, la valeur optimale du paramètre de lissage est $\hat{\lambda} = 0.91$, correspondant à une erreur $\mathrm{MSE}_{\mathrm{CV}} = 66.34$. L'estimateur $\hat{\boldsymbol{\beta}}$(Lasso) est l'intersection du chemin de régularisation (courbe gauche de la Figure 1.1) avec la ligne verticale d'abscisse $\hat{\lambda} = 0.91$. Pour cet exemple le *Lasso* produit 44 variables non nulles.

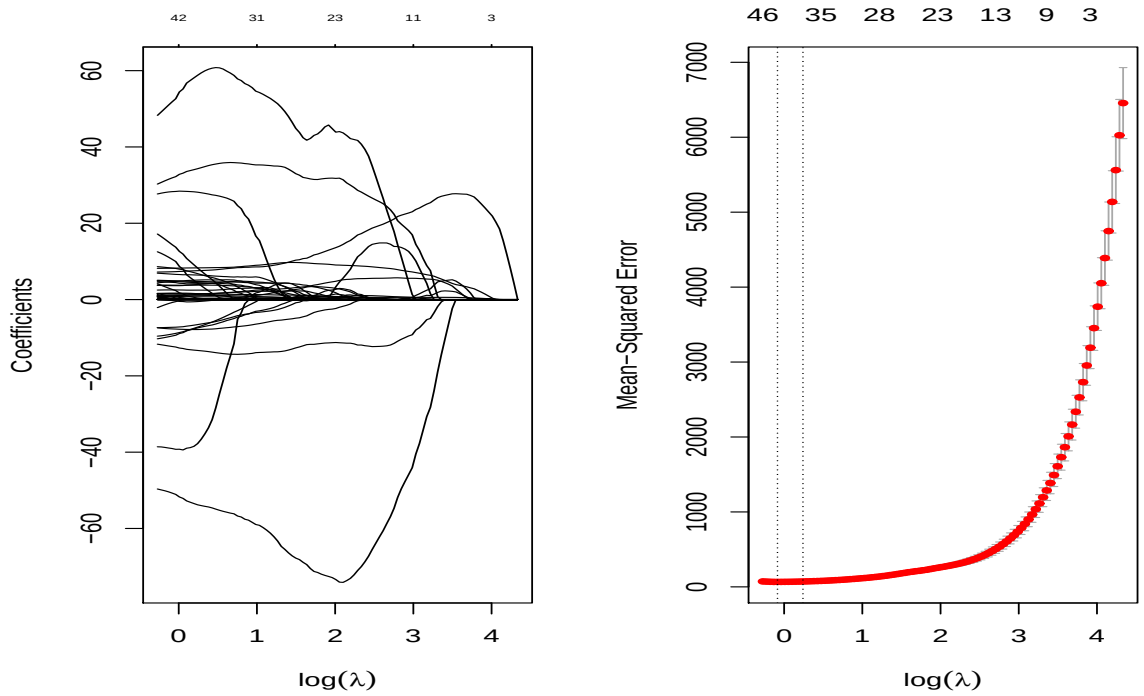


FIG. 1.1 – La régression Lasso pour la base de données PAC. La courbe de gauche représente $\hat{\boldsymbol{\beta}}$(Lasso) en fonction de $\log(\lambda)$, tandis que la courbe de droite représente l'erreur moyenne calculée par validation croisée, ainsi qu'un intervalle de confiance de l'erreur associée à son écart-type. La ligne verticale de gauche correspond à l'erreur minimale *mincv*, tandis que la ligne verticale de droite correspond à la plus grande valeur de λ telle que son erreur est inférieure ou égale à *mincv* + *sdmincv*, où *sdmincv* est l'écart-type de *mincv*. Les nombres en haut des deux courbes représentent la taille des modèles.

Adaptive Lasso Pour l'*Adaptive-Lasso* un choix usuel pour les poids est le suivant :

$$\hat{\omega}_j = \left(|\hat{\boldsymbol{\beta}}_j(\text{Lasso})| + 1/n\right)^{-\gamma}, \tag{1.41}$$

où γ est un paramètre de lissage. Nous faisons une validation croisée bidimensionnelle pour une grille de valeurs du paramètre de lissage $\theta = (\gamma, \lambda)$. Nous choisissons la grille $\{0.01, 0.5, 1, 5\}$ pour le paramètre γ et de la même manière que le *Lasso*, nous faisons tourner l'algorithme pour 100 valeurs de λ. La Figure 1.2 permet de visualiser les résultats obtenus pour $\gamma = 0.01$. L'erreur de prédiction optimale vaut $\text{MSE}_{\text{CV}} = 58.06$, elle atteinte pour $\hat{\theta} = (\hat{\gamma}, \hat{\lambda}) = (0.01, 0.81)$ et 45 variables sont introduite dans le modèle final.

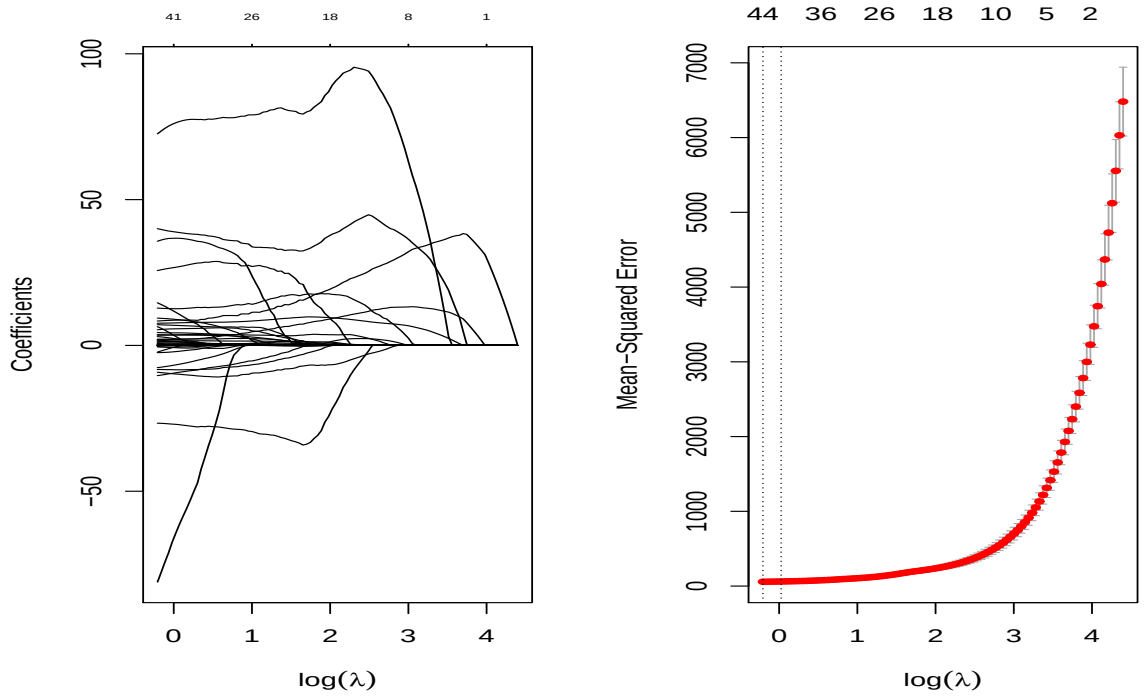


FIG. 1.2 – La régression Adaptive Lasso pour la base de données PAC. La courbe de gauche représente $\hat{\boldsymbol{\beta}}_{\text{AdaLasso}}$ en fonction de $\log(\lambda)$, tandis que la courbe de droite représente l'erreur moyenne calculée par validation croisée, ainsi qu'un intervalle de confiance de l'erreur associée à son écart-type. Les résultats tracés ici sont pour $\gamma = 0.01$.

Scad Le paramètre de lissage de la régression *Scad* est $\theta = (\gamma, \lambda)$. Afin de trouver les meilleurs performances, nous devons faire de la validation croisée bidimensionnelle. Nous faisons tourner l'algorithme pour $\gamma \in \{3, 10, 20, 100\}$ et pour 100 valeurs de λ choisies par défaut. Les résultats obtenus sont résumés graphiquement sur la Figue 1.3. La meilleure

prédiction est $\text{MSE}_{\text{CV}} = 43.66$, elle est atteinte pour $\hat{\theta} = (\hat{\gamma}, \hat{\lambda}) = (100, 0.76)$, et nous avons 38 variables sélectionnées dans le modèle final.

Mcp Pour la régression *Mcp*, le paramètre de lissage est $\theta = (\gamma, \lambda)$. Nous devons faire donc de la validation croisée bidimensionnelle pour retrouver l'estimateur optimal. La Figure 1.4 montre le chemin de régularisation et l'erreur de prédiction pour $\gamma \in \{3, 10, 20, 100\}$. Il faut souligner ici que pour certaines valeurs du paramètre (γ, λ), la fonction objectif peut ne pas être convexe, ce qui peut ralentir la convergence de l'algorithme. La Figure 1.4 montre les chemins de régularisation et l'erreur de prédiction pour les 4 valeurs de γ. Le paramètre de lissage optimal est $\hat{\theta} = (\hat{\gamma}, \hat{\lambda}) = (100, 0.76)$, l'erreur de prédiction est $\text{MSE}_{\text{CV}} = 43.43$ et le nombre de variables non nulles est 36.

Elastic-net Pour l'*Elastic-net* le paramètre de lissage est $\theta = (\alpha, \lambda)$. En pratique, nous choisissons une petite grille de valeurs de α, à savoir $\alpha \in \{0.1, 0.5, 0.9, 1\}$. Ensuite, pour tout α fixé, l'algorithme *coordinate descent* produit le chemin de régularisation de l'*Elastic-net*. L'autre paramètre de lissage λ est choisi par validation croisée. La valeur finale de α est celle qui minimise l'erreur de validation croisée. La Figure 1.5 donne les chemins de régularisation de l'*Elastic-net*, ainsi que l'erreur de validation croisée pour les 4 valeurs de α. La valeur optimale du paramètre de lissage est $\hat{\theta} = (\hat{\alpha}, \hat{\lambda}) = (0.5, 1.52)$. L'erreur de prédiction est $\text{MSE}_{\text{CV}} = 57.74$ et le modèle final contient 56 variables.

Weighted Fusion Pour la régression *Weighted Fusion*, le paramètre de lissage est $\theta = (\lambda, \mu, \gamma)$. En pratique, nous prenons $\mu \in \{0.01, 0.1, 1, 10, 100\}$ et $\gamma \in \{0.5, 2.5, 5, 10, 25\}$ et 100 valeurs pour λ. Pour sélectionner le paramètre de lissage optimal, nous devons calculer l'erreur de prédiction par validation croisée pour une grille tri-dimensionnelle. Pour la base de données PAC, l'erreur de prédiction est atteinte pour $\hat{\theta} = (\hat{\lambda}, \hat{\mu}, \hat{\gamma}) = (1.16, 0.1, 25)$, elle vaut $\text{MSE}_{\text{CV}} = 86.87$ pour 51 variables introduites dans le modèle final.

Smooth Lasso Pour la régression *Smooth Lasso*, le paramètre de lissage est $\theta = (\lambda, \mu)$. Nous faisons tourner une validation croisée bidimensionnelle pour 100 valeurs du paramètre λ et pour $\mu \in \{0.01, 0.1, 1, 10, 100\}$. Pour la base de données PAC, la Figure 1.7 donne les chemins de régularisation et l'erreur de prédiction pour $\mu \in \{0.01, 0.1, 1, 10\}$. La valeur optimale du paramètre de lissage est $\hat{\theta} = (\hat{\lambda}, \hat{\mu}) = (1.05, 10)$, donnant une erreur de prédiction $\text{MSE}_{\text{CV}} = 72.29$ et 37 variables non nulles.

1.5 Discussion

Dans ce chapitre, nous avons présenté un certain nombre de méthodes de régularisation. Ensuite, nous avons décrit les algorithmes permettant de retrouver les solutions. Nous nous sommes surtout intéressés à l'algorithme *Coordinate Descent* pour sa simplicité et son adaptation à un bon nombre de techniques de régularisation. Enfin, nous avons illustré ces techniques par un exemple de données réelles provenant du domaine de la chimie.

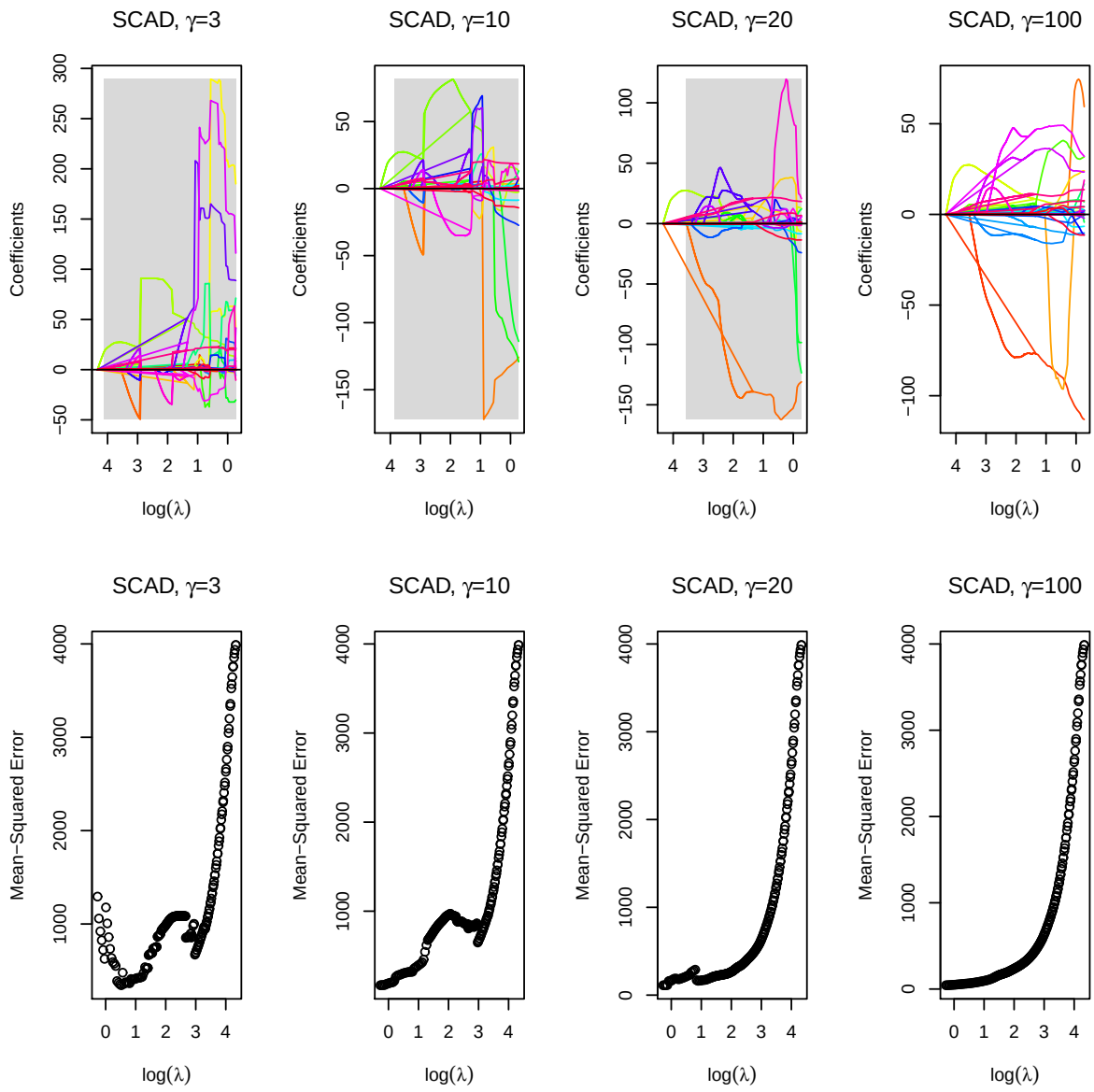


FIG. 1.3 – La régression *Scad* pour la base de données PAC. Les chemins de régularisation et l'erreur de prédiction pour différentes valeurs du paramètre γ.

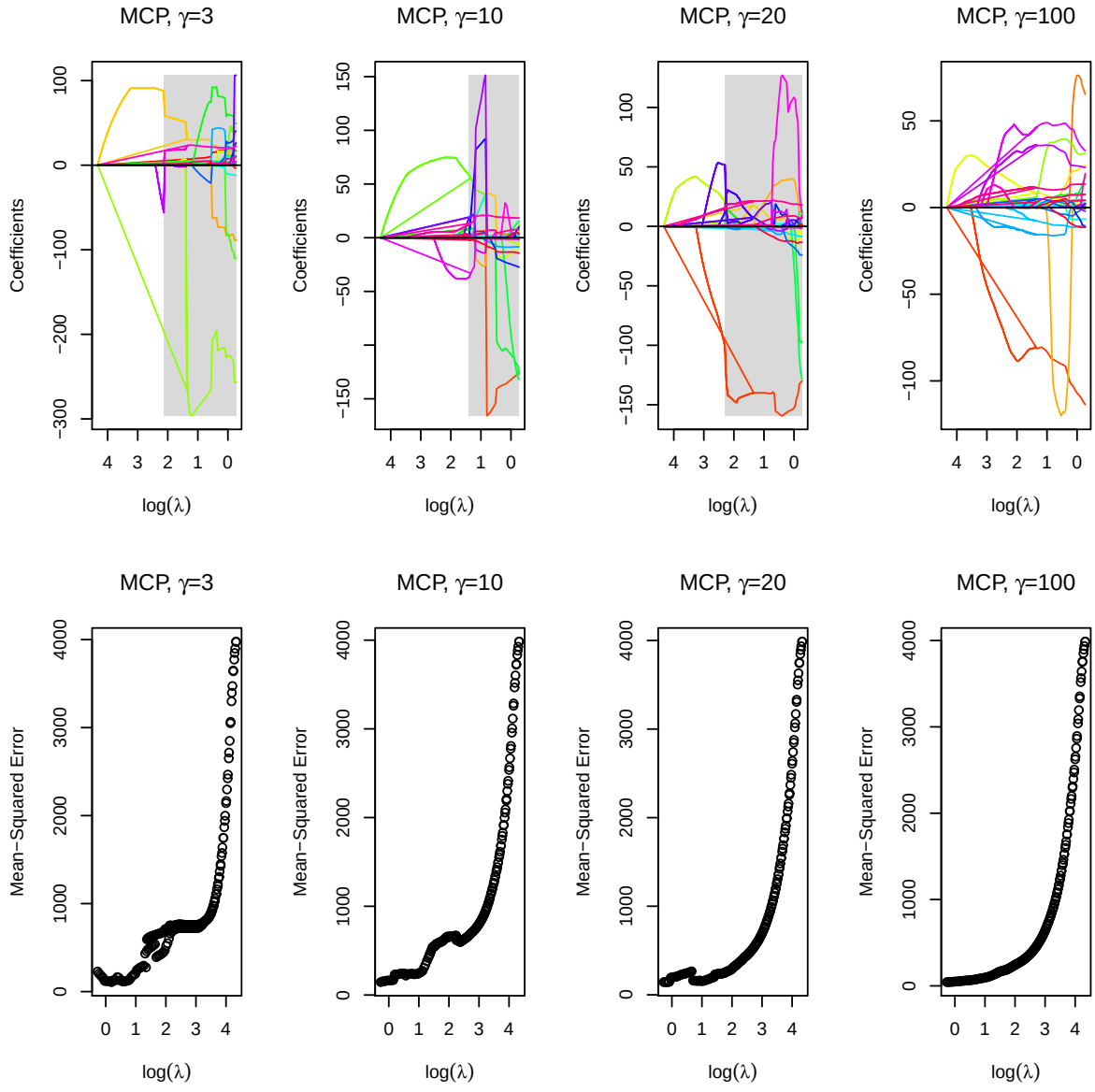


FIG. 1.4 – La régression *Mcp* pour la base de données PAC. Notons que les chemins de régularisation sont continus à l'intérieur des régions localement convexes, mais ils sont dicontinus et irréguliers à l'extérieur.

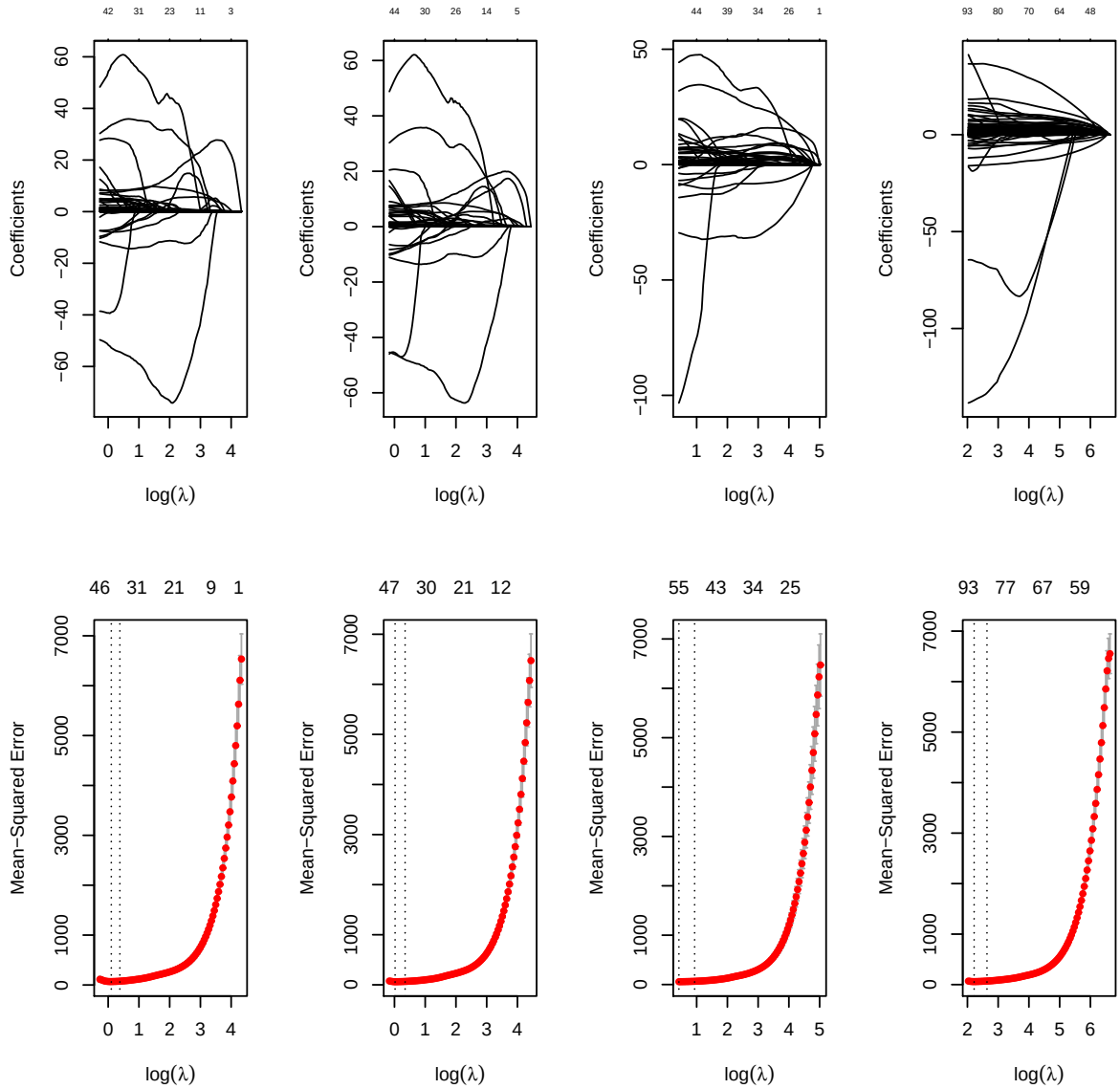


FIG. 1.5 – La régression Elastic-net pour la base de données PAC. Les courbes du haut représentent $\hat{\boldsymbol{\beta}}_{\text{Enet}}$ en fonction de $\log(\lambda)$ pour $\alpha \in \{0.1, 0.5, 0.9, 1\}$, tandis que les courbes du bas représentent l'erreur moyenne calculée par validation croisée en fonction de $\log(\lambda)$ pour les même valeurs de α.

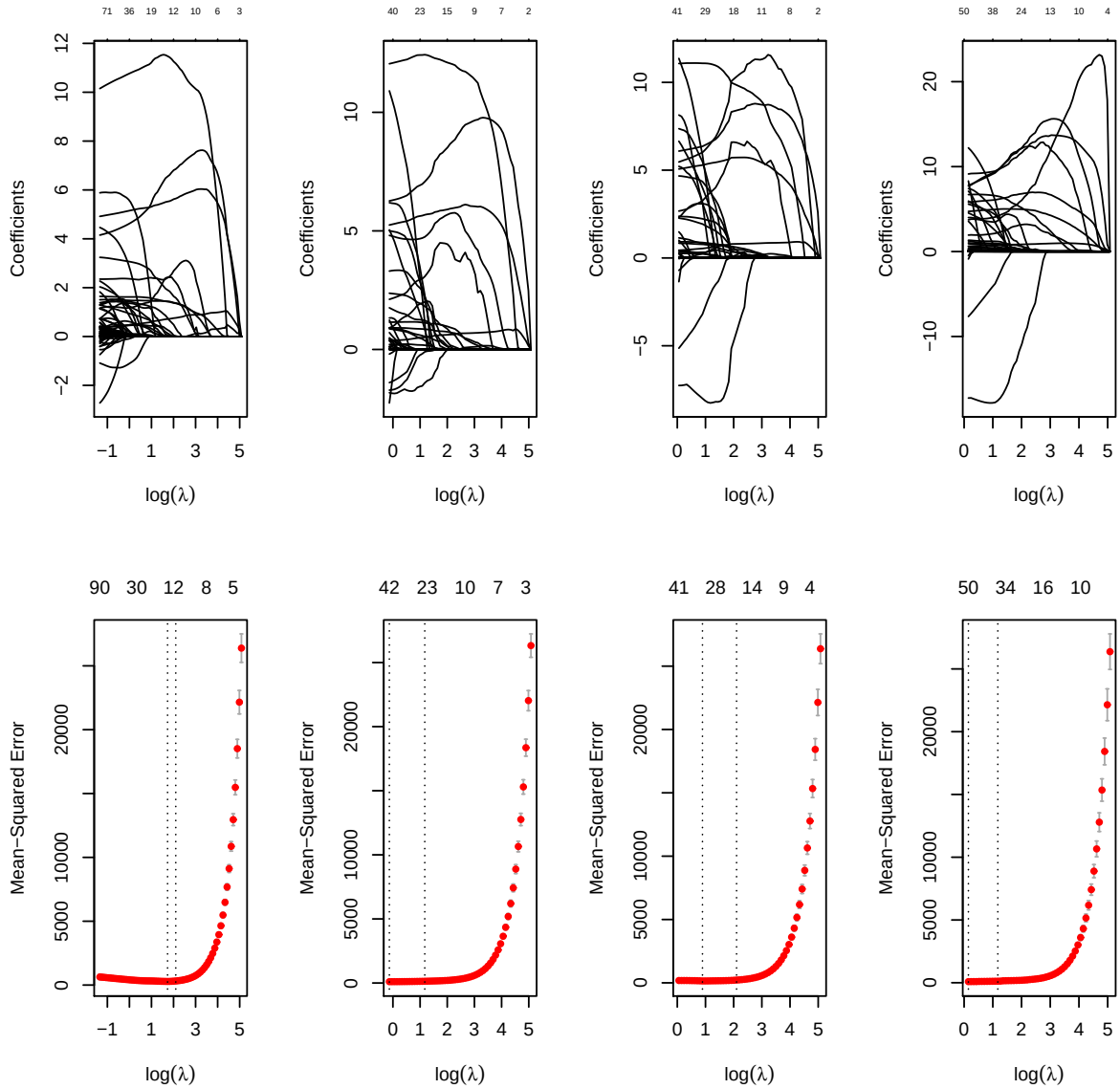


FIG. 1.6 – La régression *Weighted Fusion* pour la base de données PAC. Les courbes du haut représentent $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$ en fonction de $\log(\lambda)$, tandis que les courbes du bas représentent l'erreur moyenne calculée par validation croisée. Les résultats tracés ici sont pour $\mu = 0.1$ et de gauche à droite pour $\gamma = 0.5, 2.5, 5, 25$.

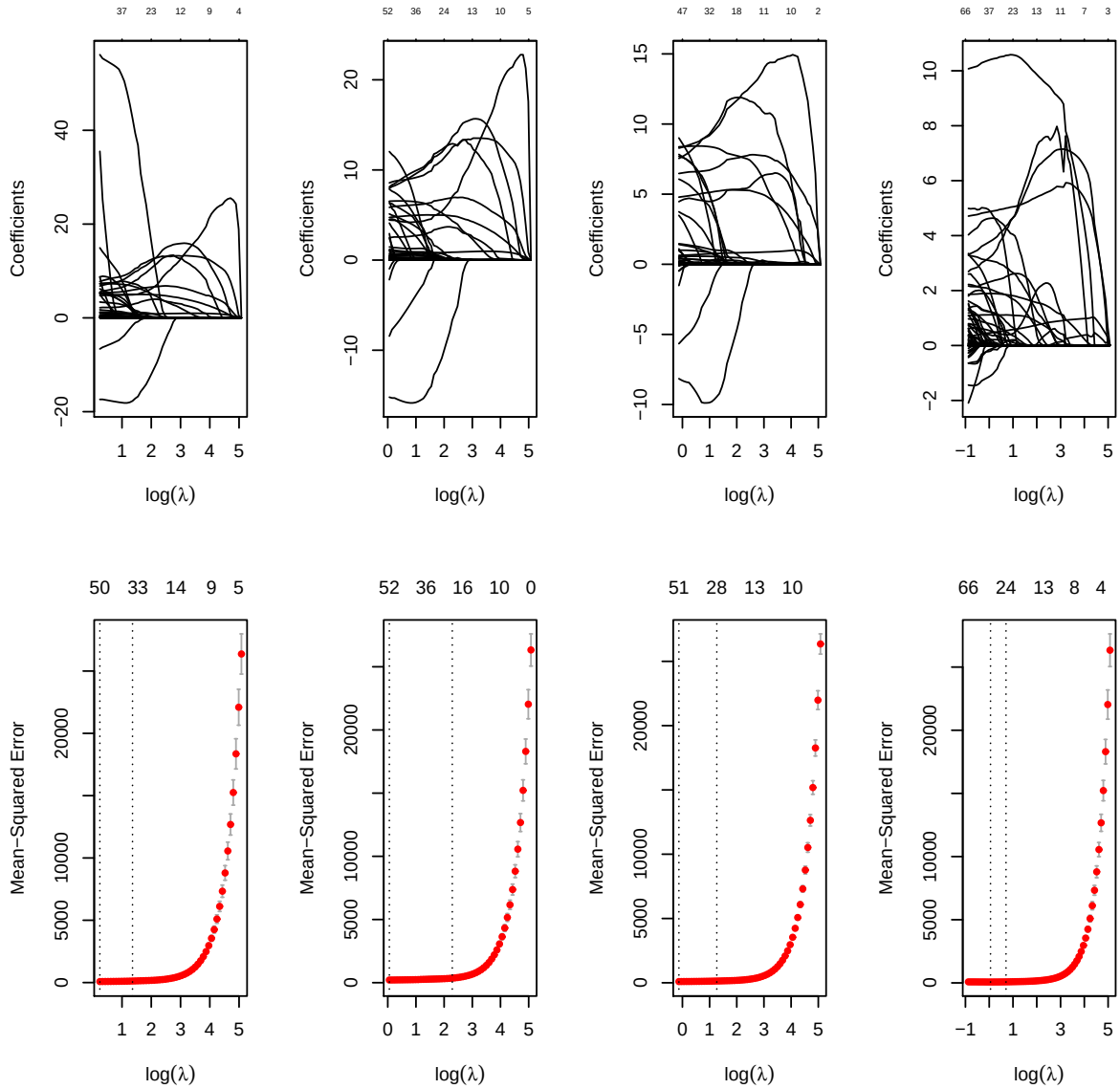


FIG. 1.7 – La régression *Smooth Lasso* pour la base de données PAC. Les courbes du haut représentent $\hat{\boldsymbol{\beta}}_{\text{Slasso}}$ en fonction de $\log(\lambda)$, tandis que les courbes du bas représentent l'erreur de prédiction calculée par validation croisée. De gauche à droite, les résultats tracés ici sont pour $\mu = 0.01, 0.1, 1, 10$.

Il faut dire que notre présentation n'est pas exhaustive. D'autres technique de régularisation ont été proposées :

La régression *Bridge* La régression Bridge a été proposée par Frank and Friedman (1993) et elle a suscité plus d'attention de la part de Fu (1998). C'est la solution du problème pénalisé suivant :

$$\hat{\boldsymbol{\beta}}_{\text{Bridge}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^{\gamma}, \tag{1.42}$$

où $\gamma \geq 1$ et $\lambda \geq 0$ sont deux paramètres de lissage. Elle inclut les régressions Ridge et Lasso comme deux cas particuliers, pour $\gamma = 2$ et $\gamma = 1$ respectivement.

La régression *Garrote* La motivation du Lasso est venue d'une proposition intéressante de Breiman (1995). Le *Garotte* de Breiman, qui est le minimiseur du problème d'optimisation :

$$\frac{1}{2} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} c_j \hat{\boldsymbol{\beta}}_j^{ols} x_{ij} \right)^2 \text{ sujet à } \; c_j \geq 0 \;\; , \;\; \sum_{j=1}^{p} c_j \leq t, \tag{1.43}$$

où $\hat{\boldsymbol{\beta}}^{ols}$ est l'estimateur des moindres carrés ordinaire. Le *Garotte* rétrécit l'estimateur des moindres carrés par des coefficients positifs dont la somme est contrainte. C'est une méthode de régularisation prometteuse. Cependant, elle n'est pas adaptée pour le cas où $p > n$ car elle nécessite le calcul de l'estimateur des moindres carrés ordinaire, qui n'est pas stable et peut ne pas être unique dans ce cas. Yuan and Lin (2007) considèrent et étudient les performances du *Garrote* lorsque l'estimateur initial est l'estimateur Ridge (1.3). Ce choix permet de traiter le cas où $p > n$. Yuan et Lin proposent aussi un algorithme de type LARS pour approcher l'estimateur *Garrote*.

La régression *Grouped Lasso* Pour certaines bases de données, nous pouvons savoir d'avance que l'ensemble des variables explicatives est divisé en groupes. L'exemple le plus intéressant est l'analyse de la variance (ANOVA) à plusieurs facteurs, dans laquelle chaque facteur peut avoir plusieurs niveaux et peut être exprimé par un groupe de variables nominales. Pour se confronter avec de telles situations Yuan and Lin (2006) ont proposé le *Grouped Lasso* :

$$\hat{\boldsymbol{\beta}}_{\text{Grplasso}} = \arg\min_{\boldsymbol{\beta}} \frac{1}{2n} \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{l=1}^{L} \sqrt{\sum_{j \in G_l} \beta_j^2}. \tag{1.44}$$

où λ est un paramètre de lissage, L est un nombre entier dans $\{1, ..., p\}$, et $(G_l)_{l \in \{1,...,L\}}$ est une partition de $\{1, ..., p\}$. Dans cette définition, L désigne le nombre de groupes et G_l l'ensemble des indices des variables contenues dans le groupe l avec $l = 1, ..., L$.

La régression *Oscar* Bondell and Reich (2008) ont proposé la méthode Oscar. Elle permet elle aussi de sélectionner les variables tout en les regroupant dans des *clusters*. C'est la solution du problème d'optimisation suivant :

$$\min_{\beta\in\mathbb{R}^p} \frac{1}{2n}\sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\beta_j\right)^2 \quad \text{sujet à} \quad \sum_{j=1}^{p}|\beta_j| + c\sum_{j<k}\max\{|\beta_j|,|\beta_k|\} \leq \lambda, \qquad (1.45)$$

où $c \geq 0$ et $\lambda > 0$ sont des paramètres de lissage. La contrainte utilisée est une combinaison pondérée de la norme l_1 et une norme l_∞ par paires entre les coefficients. La norme l_1 encourage des modèles clairsemés, tandis que norme l_∞ par paires encourage l'égalité des coefficients. Pour pratiquer cette méthode, un programme sous Matlab est disponible sur le lien : `http://www4.stat.ncsu.edu/~bondell/software.html`.

1.A Code R commenté

Pour calculer les estimateurs, nous chargeons les packages suivants :

```
> library(glmnet)
> library(ncvreg)
```

La package `glmnet` a été conçu pour faire un modèle linéaire généralisé par le biais du maximum de vraisemblance pénalisée. Le chemin de régularisation est calculé pour le Lasso ou la pénalité elastic-net pour une grille de valeurs du paramètre de régularisation λ. Un point fort de ce package est de d'inclure les modèles de régression linéaire, logistique et multinomiale, Poisson, et les modèles de Cox.

Nous chargeons les données PAC. Cette base de données contient une variable réponse `y` et une matrice de variables explicatives `X`.

```
> library(chemometrics)
> data(PAC)
> X<-PAC$X
> y<-PAC$y
```

1.A.1 Lasso

Par défaut, `glmnet` calcule la solution du Lasso pour 100 valeurs du paramètre de lissage λ depuis une valeur $\lambda_{\max}$ annulant tous les coefficients jusqu'à une valeur $\lambda_{\min}$ proche de zéro, où le vecteur estimé est proche des moindres carrés. L'algorithme ne descend pas nécessairement jusqu'à $\lambda = 0$ pour des problèmes de stabilité.

Nous appelons maintenant les fonctions `glmnet` et `cv.glmnet` pour faire la régression Lasso ($\alpha = 1$), et réaliser la 10-*fold* validation croisée pour sélectionner le paramètre de lissage λ.

```
>   fit_lasso<-glmnet(X,y,alpha = 1)
>   fit_cv_lasso<-cv.glmnet(X,y,alpha=1, nfolds=10)
```

Nous pouvons produire la Figure 1.1 pour visualiser le chemin de régularisation $\{\hat{\boldsymbol{\beta}}(\text{Lasso})(\lambda)\}_{\lambda_{\min}\leq\lambda\leq\lambda_{\max}}$ et l'erreur de validation croisée pour la grille des 100 valeurs de λ.

```
>   plot(fit_lasso,xvar ="lambda",label = FALSE,col=T,xlab=expression(log(lambda)))
>   plot(fit_cv_lasso,xlab=expression(log(lambda)))
```

Nous pouvons maintenant déterminer la valeur optimale de λ, et par conséquent notre estimateur $\hat{\boldsymbol{\beta}}(\text{Lasso})$.

```
>   lambda.opt<-fit_cv_lasso$lambda.min
>   fit_coef_lasso<-predict(fit_lasso,type="coef",s=lambda.opt)
```

1.A.2 L'adaptive Lasso

Nous donnons d'abord une valeur au paramètre γ.

```
> gamma=0.5
```

Calculons les poids (1.41) en utilisant l'estimateur *Lasso*.

```
> w0<-1/(abs(fit_coef_lasso) + (1/length(y)))
> poids.lasso<-w0^(gamma)
```

Nous pouvons maintenant faire la régression *Adaptive Lasso* et calculer l'erreur de prédiction pour une grille de 100 valeurs choisies par défaut.

```
> fit_adalasso <- glmnet(x, y, penalty.factor =poids.lasso)
> fit_cv_adalasso<-cv.glmnet(x, y,penalty.factor=poids.lasso)
```

Nous pouvons produire la Figure 1.2 pour visualiser les performances en fonction de $\log\lambda$.

```
> par(mfrow=c(1,2))
> plot(fit_adalasso,xvar ="lambda",label = FALSE,col=T,xlab=expression(log(lambda)))
> plot(fit_cv_adalasso,xlab=expression(log(lambda)))
```

Nous calculons maintenant l'erreur de prédiction minimale et la valeur du paramètre de lissage λ pour laquelle elle est atteinte.

```
> min(fit_cv_adalasso$cvm)
> fit_cv_adalasso$lambda.min
```

Nous pouvons maintenant avoir notre estimateur $\hat{\boldsymbol{\beta}}_{\text{Adalasso}}$.

```
> fit_coef_adalasso<-predict(fit_adalasso,type="coef",s=fit_cv_adalasso$lambda.min)
```

1.A.3 Scad

Nous fixons le paramètre de lissage γ à 60 par exemple. Cherchons les chemins de régularisation et l'erreur de prédiction pour les 100 valeurs de λ choisies par défaut par le logiciel.

```
> fit.scad<-ncvreg(X,y,family="gaussian",penalty="SCAD",gamma=60)
> cvfit.scad<-cv.ncvreg(X,y,family="gaussian",penalty="SCAD",gamma=60,nfolds=10)
```

Nous pouvons visulaiser graphiquement la partie du tracé de la Figure 1.3 correspondant à $\gamma = 60$ (chemin de régularisation, coefficients et erreur de prédiction).

```
> par(mfrow=c(1,2))
> plot(fit.scad,ylab="Coefficients", log.l=T, shade=TRUE,main=expression(paste("scad,
+ ",gamma,"=",60)))
> plot(log(cvfit.scad$lambda),cvfit.scad$error,xlab=expression(log(lambda)),
+ ylab="Mean-Squared Error",main=expression(paste("scad, ",gamma,"=",60)))
```

Calculons maintenant l'erreur de prédiction optimale et la valeur du paramètre de lissage λ.

```
> min(cvfit.scad$error)
> cvfit.scad$lambda[cvfit.scad$cv]
```

L'estimateur $\hat{\boldsymbol{\beta}}_{\text{Scad}}$ est obtenu de la manière suivante :

```
> fit_coef_scad <- cvfit.scad$beta[,cvfit.scad$cv]
```

1.A.4 Mcp

Nous suivons les mêmes étapes de la régression *Scad* pour obtenir l'estimateur *Mcp*. Il faut juste changer dans les fonctions principales le champ `penalty=SCAD` par `penalty=MCP`.

```
> fit.mcp<-ncvreg(X,y,family="gaussian",penalty="MCP",gamma=60)
> cvfit.mcp<-cv.ncvreg(X,y,family="gaussian",penalty="MCP",gamma=60,nfolds=10)

> fit_coef_mcp <- cvfit.mcp$beta[,cvfit.mcp$cv]
> min(cvfit.mcp$error)

> par(mfrow=c(1,2))
> plot(fit.mcp,ylab="Coefficients", log.l=T, shade=TRUE,main=expression(paste("MCP,
+ ",gamma,"=",60)))
> plot(log(cvfit.mcp$lambda),cvfit.mcp$error,xlab=expression(log(lambda)),
+ylab="Mean-Squared Error",main=expression(paste("MCP, ",gamma,"=",60)))

> fit_coef_mcp <- cvfit.mcp$beta[,cvfit.mcp$cv]
```

1.A.5 Weighted Fusion

Nous donnons d'abord des valeurs aux paramètres de lissage γ et μ.

```
> gamma=0.5
> mu=0.1
```

Calculons maintenant la matrice (1.19).

```
> cor.mat <- cor(X)
> abs.cor.mat <- abs(cor.mat)
> sign.mat <- sign(cor.mat) - diag(2, nrow(cor.mat))
> Wmat <- (abs.cor.mat^gamma - 1 * (abs.cor.mat == 1))/
+          (1 -abs.cor.mat * (abs.cor.mat != 1))
> weight.vec <- apply(Wmat, 1, sum)
> fusion.penalty <- -sign.mat * (Wmat + diag(weight.vec))
```

Calculons maintenant la décomposition de Cholesky (1.20).

```
> R<-chol(fusion.penalty, pivot = TRUE)
```

Transformons *Weighted Fusion* en un problème *Lasso* sur les données augmentées (1.40).

```
> p<-dim(X)[2]
> xstar<-rbind(X,sqrt(mu)*R)
> ystar<-c(y,rep(0,p))
```

Faisons du *Lasso* maintenant sur les données augmentées.

```
> fit_wfusion<-glmnet(xstar,ystar)
> fit_cv_wfusion<-cv.glmnet(xstar,ystar)
```

Nous pouvons visualiser le chemin de régularisation et l'erreur de prédiction de la Figure 1.6 correspondant à $\gamma = 0.5$ et $\mu = 0.1$.

```
> par(mfrow=c(1,2))
> plot(fit_wfusion,xvar ="lambda",label = FALSE,col=T,xlab=expression(log(lambda)))
> plot(fit_cv_wfusion,xlab=expression(log(lambda)))
```

L'erreur de prédiction minimale et la valeur du paramètre de lissage λ correspondant peuvent être obtenues.

```
> min(fit_cv_wfusion$cvm)
> lambda.opt<-fit_cv_wfusion$lambda.min
```

Nous pouvons maintenat obtenir l'estimateur $\hat{\boldsymbol{\beta}}_{\text{Wfusion}}$ correspondant à $\gamma = 0.5$ et $\mu = 0.1$.

```
> fit_coef_wfusion<-predict(fit_wfusion,type="coef",s=lambda.opt)
```

1.A.6 Smooth Lasso

Nous fixons une valeur pour μ.

```
> mu=0.01
```

Calculons la matrice (1.24).

```
> p <- ncol (X)
> R<-matrix(0,nc=p,nr=p)
> for(i in 2 :p){
+ R[i,i-1]<- 1
+ R[i,i]<- -1
+ }
```

Transformons notre problème en un problème *Lasso* en augmentant les données de départ. Nous obtenons (1.40).

```
> xstar<- rbind(X,sqrt(mu)*R)
> ystar<-c(y,rep(0,p))
```

Appliquons du *Lasso* sur les nouvelles données.

```
> fit_slasso<-glmnet(xstar,ystar)
> fit_cv_slasso<-cv.glmnet(xstar,ystar)
```

Nous pouvons calculer l'erreur de prédiction minimale et la valeur correspondante du paramètre de lissage λ.

```
> min(fit_cv_slasso$cvm)
> fit_cv_slasso$lambda.min
```

Nous pouvons maintenat obtenir l'estimateur $\hat{\boldsymbol{\beta}}_{\text{Slasso}}$ minimisant l'erreur de prédiction pour $\mu = 0.01$.

```
> fit_coef_slasso<-predict(fit_slasso,type="coef",s=fit_cv_slasso$lambda.min)
```

Chapter 2

Penalized regression combining the L_1 norm and a correlation based penalty

Résumé : La sélection de variables peut être difficile, en particulier dans les situations où un grand nombre de variables explicatives est disponible, avec la présence possible de corrélations élevées comme dans le cas des données d'expression génétique. Dans cet article, nous proposons une nouvelle méthode de régression linéaire pénalisée, appelée L_1CP, pour simultanément estimer les paramètres inconnus et sélectionner les variables importantes. De plus, elle encourage un effet de groupe: les variables fortement corrélées ont tendance à être toutes incluses ou toutes exclues du modèle. La méthode est fondée sur les moindres carrés pénalisés avec une pénalité qui, comme la pénalité l_1, rétrécit certains coefficients exactement vers zéro. En outre, cette pénalité contient un terme qui lie explicitement la force de pénalisation à la corrélation entre les variables explicatives.

2.1 Introduction

Consider the problem of general linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \tag{2.1}$$

where $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_p)$ is a nxp matrix of p predictor vectors of dimension p, $\boldsymbol{\varepsilon}$ is a random error vector with $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$ and $\boldsymbol{\beta}^*$ is a vector of unknown parameters which are to be estimated. The ordinary least squares (OLS) estimator $\hat{\boldsymbol{\beta}}$ is obtained by minimizing the residual sum of squares. It is often not good enough for two reasons: bad prediction accuracy caused by a large variance of the coefficients although there are unbiased and the problem of interpretation in the presence of large number of variables.

Traditionally, the latter problem is reduced by variable selection or some method of dimension reduction like principal component regression or partial least squares. It often

leads to the solution with good prediction accuracy as well as to a parsimonious model. While the first problem is resolved by using regularization methods which scarify some bias to reduce variance such as ridge regression (RR) (Hoerl and Kennard, 1970) and a more recent regularization based correlation penalty (CP) (Tutz and Ulbricht, 2009), which explicitly uses the correlation between predictors in the L_2 norm penalty term. This can be done also via coefficients shrinkage or by forcing some of the coefficients to zero by using a L_1 penalty such as Lasso (Tibshirani, 1996; Chen et al., 1998) instead of quadratic penalty of RR and CP. However, RR and CP do shrinkage but not variable selection while Lasso combines shrinkage and variable selection in a single procedure.

The Lasso is a popular and computationally efficient technique for automatically performing both variable selection and regularization in linear regression model. Lasso generates sparse models, which are more interpretable. This method was made particularly appealing by the advent of the LARS algorithm (Efron et al., 2004) which provided a highly efficient mean to simultaneously produce the set of Lasso fits for all values of the tuning parameter. However, a limitation of Lasso, in particular for $p > n$, is that it selects at most n variables before it puts all coefficients to nonzero as remarked by Zou and Hastie (2005). A second limitation is that group of variables can not enter in the same time with Lasso.

Zou and Hastie (2005) proposed Elastic-Net (ENET), which also has the property of sparsity, to solve the above problem. It uses both the Ridge and Lasso constraints. Hence, it performs the variable selection of the LARS algorithm with Lasso modification and the shrinkage of ridge. ENET has also the ability to include group of variables which are highly correlated. It is remarked that ENET is particularly adapted to high dimensional setting ($n << p$). However, as remarked in our experimental studies, ENET seems to be slightly less reliable if the correlation between variables is not so extreme (i.e. $|\rho| \leq 0.95$). Moreover, ENET does not take into account the information correlation of data in the L_2 norm penalty.

In the same spirit, Bondell and Reich (2008) proposed a method called Oscar, which is based on the penalized least squares with a penalty function combining the L_1 and the pairwise L_∞ norms. The Oscar forces some of the grouped coefficients identically equal, encouraging correlated covariates that have similar effect on the response to form clusters represented by the same coefficients. However, the computation of the Oscar estimates is based on a sequential quadratic programming algorithm which can be slow for large p. On the other hand, in order to obtain the grouping effect of CP in combination with variable selection, Tutz and Ulbricht (2009) proposed blockwise boosting procedure (BB) which updates at each step the coefficient of more than one variable. But, in practical implementation, the step length factor and the stopping number of iterations have to be determined. This sometimes may be difficult and affect the sparsity of the solution as well as the speed of convergence of the algorithm.

In this paper, we propose an alternative regularization procedure, based on the penalized least squares for variable selection in linear regression problem, which combines Lasso and CP penalties. We call it the L_1CP. Similar to the ENET method, the L_1CP performs automatic variable selection and allow to select or remove highly correlated vari-

ables (grouping effect). Additionally, the CP penalty contains a term which explicitly links strength of penalization to the correlation between variables. In contrast to the Elastic-Net, our approach seems to be adapted to both small and high dimensional settings : $p \leq n$ and $n << p$.

The remainder of this paper is organized as follow. In Section 2.2, we formulate the L_1CP as a constrained least squares problem using a combination of L_1 and CP penalties. We discuss its computational issues in the same section. The grouping effect that is caused by the L_1CP penalty and the selection of tuning parameters are considered in Section 2.3. Section 2.4 is devoted to numerical experimentations on simulated data which show that L_1CP compares favorably to the existing shrinkage and variable selection techniques in terms of both prediction error and identification of relevant variables. Finally, the L_1CP is applied to the US crim and GC-Retention data sets in Section 2.5. We end the paper with a brief discussion in Section 2.6.

2.2 Methodology

In this section, we present the elements of the L_1CP procedure in terms of regularization and computational aspects.

2.2.1 The L_1CP criterion

Suppose that the predictors are $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})^t$ and response values y_i, for $i = 1, \ldots, n$ and the symbol t stands for the transpose. Using suitable location and scale transformations we can assume without loss of generality that the response is centered and the predictors are centered and standardized,

$$\sum_{i=1}^{n} y_i = 0, \quad \sum_{i=1}^{n} x_{ij} = 0 \quad \text{and} \quad \sum_{i=1}^{n} x_{ij}^2 = 1, \quad \text{for } j = 1, ..., p. \tag{2.2}$$

In the following, we assume that (2.2) holds. However, all numerical results are presented in the original scale of the data.

As described above, we define the L_1CP estimator as

$$\hat{\boldsymbol{\beta}}(\lambda_1, \lambda_2) = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 P_c(\boldsymbol{\beta}) \right\}, \tag{2.3}$$

where

$$P_c(\boldsymbol{\beta}) = \sum_{j=1}^{p-1} \sum_{i>j} \left\{ \frac{(\beta_i - \beta_j)^2}{1 - \rho_{ij}} + \frac{(\beta_i + \beta_j)^2}{1 + \rho_{ij}} \right\}, \tag{2.4}$$

$\|\boldsymbol{\beta}\|_1 = \sum_{j=1}^{p} |\beta_j|$, λ_1 and λ_2 are positive tuning parameters and $\rho_{ij} = \mathbf{x}_i^t \mathbf{x}_j$ denotes the (empirical) correlation between the ith and the jth predictors. Here the penalty $P_c(\boldsymbol{\beta})$ was introduced by Tutz and Ulbricht (2009) and is defined assuming that $\rho_{ij}^2 \neq 1$ for $i \neq j$.

The L_1 penalty encourages sparsity in the coefficients. While the correlation based penalty $P_c(\boldsymbol{\beta})$ will encourage grouping effect for highly correlated variables. In fact, it's easy to see that for strong positive correlation ($\rho_{ij} \approx 1$), the first term becomes dominant having the effect that estimates for β_i and β_j are similar ($\hat{\beta}_i \approx \hat{\beta}_j$). For strong negative correlation ($\rho_{ij} \approx -1$), the second term becomes dominant and $\hat{\beta}_i$ will be close to $-\hat{\beta}_j$. Moreover, assuming that $\rho_{ij}^2 \neq 1$ for $i \neq j$, the penalty (2.4) can be written in a simple quadratic form

$$P_c(\boldsymbol{\beta}) = \boldsymbol{\beta}^t \mathbf{W} \boldsymbol{\beta},$$

where $\mathbf{W} = (w_{ij})_{1 \leq i,j \leq p}$ is a positive definite matrix with general term

$$w_{ij} = \begin{cases} 2 \sum_{s \neq i} \frac{1}{1-\rho_{is}^2}, & i = j \\ -2 \frac{\rho_{ij}}{1-\rho_{ij}^2}, & i \neq j \end{cases} \tag{2.5}$$

Putting $\gamma = \lambda_1/(\lambda_1 + \lambda_2)$, then the optimization problem (2.3) is equivalent to

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|^2, \text{s.t } (1-\gamma)\|\boldsymbol{\beta}\|_1 + \gamma P_c(\boldsymbol{\beta}) \leq t \text{ for } t \geq 0. \tag{2.6}$$

The L_1CP penalty function $(1-\gamma)\|\boldsymbol{\beta}\|_1 + \gamma P_c(\boldsymbol{\beta})$ is a convex combination of the Lasso and correlation based penalty. It is strictly convex for all $\gamma \in [0, 1)$ and singular at 0. For $p = 2$, Figure 2.1 gives its penalty contour for two amounts of positive and negative correlations.

2.2.2 The L_1CP procedure

In this section we propose a modification of the Elastic-Net algorithm for finding a solution of the penalized least squares problem (2.3) of the L_1CP. The main idea is to transform the L_1CP problem into an equivalent Lasso problem on the augmented data (cf. (Zou and Hastie, 2005)). The optimization problem (2.3) can be rewritten as

$$\hat{\boldsymbol{\beta}}(\lambda_1, \lambda_2) = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \lambda_2 \boldsymbol{\beta}^t \mathbf{W} \boldsymbol{\beta} \right\}, \tag{2.7}$$

where $\mathbf{W}$, defined by (2.5), is a real symmetric positive-definite square matrix, assuming that $\rho_{ij}^2 \neq 1$, with Choleski decomposition $\mathbf{W} = \mathbf{L}\mathbf{L}^t$ and $\mathbf{L} = \mathbf{W}^{\frac{1}{2}}$ which always exists. Now, let

$$\mathbf{X}^*_{(n+p)\times p} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{L}^t \end{pmatrix}, \quad \mathbf{y}^*_{(n+p)} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix},$$

The L_1CP estimator is defined as

$$\hat{\boldsymbol{\beta}}^* = \arg\min_{\boldsymbol{\beta}^*} \|\mathbf{y}^* - \mathbf{X}^*\boldsymbol{\beta}^*\|_2^2 + \lambda_1 \|\boldsymbol{\beta}^*\|_1.$$

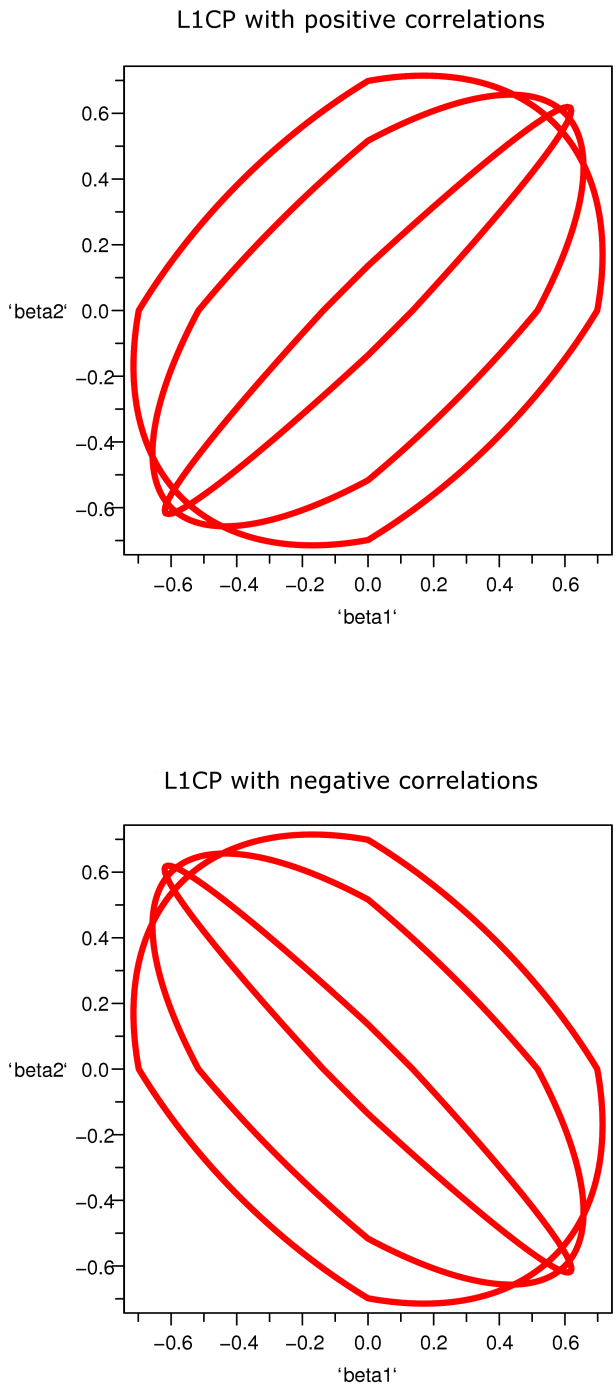


Figure 2.1: Top panel: Two-dimensional contour plots of $0.5\|\boldsymbol{\beta}\|_1 + 0.5P_c(\boldsymbol{\beta}) = 1$ for three amounts of positive correlation: $\rho = 0.5$, $\rho = 0.8$, and $\rho = 0.99$. Bottom panel: Two-dimensional contour plots of $0.5\|\boldsymbol{\beta}\|_1 + 0.5P_c(\boldsymbol{\beta}) = 1$ for three amounts of negative correlation: $\rho = -0.5$, $\rho = -0.8$, and $\rho = -0.99$.

We note that L_1CP estimator becomes naive Elastic-Net estimator when $\mathbf{W} = \mathbf{I}$. The latter result is a consequence of simple algebra (cf. ElAnbari and Mkhadri (2008)), and it motivates the following comments on the L_1CP method.

Remark 1. The L_1CP estimates can be computed via the Lasso modification of the LARS algorithm. For a fixed λ_2, it constructs at each step, which corresponds to a value of λ_1, an estimator based on the correlation between predictors and the current error term. Then for a fixed λ_2, we obtain the evolution of the L_1CP estimator coefficient values when λ_1 varies. It provides the coefficient regularization paths of the L_1CP estimator which are piecewise linear (Efron et al., 2004). Consequently, the L_1CP algorithm requires the same order of magnitude of computational effort as the OLS estimate via the Lasso modification of the LARS algorithm.

Remark 2. If $p > n$, it is well known that LARS and its Lasso version can select at most n variables before it puts all coefficients to nonzero. Now, applied LARS to augmented data $(\mathbf{y}^*, \mathbf{X}^*)$, the Lasso modification of the LARS algorithm is able to select all the p predictors in all situations. So the first limitation of the Lasso is overcome. Moreover, the variable selection is performed in a fashion similar to the Lasso.

2.3 The grouping effect and tuning parameters selection

2.3.1 The grouping effect

Grouping arises when the regression coefficients of a group of highly correlated variables tend to be equal (up to a change of sign if negatively correlated). Similar to ENET estimator, the L_1CP estimator has the natural tendency of grouping each pair of regression coefficients according to their correlations. It is too difficult to give, in a general case, an upper bound of the absolute difference between any pair (i, j) of the components of $\hat{\boldsymbol{\beta}}$ (the L_1CP estimates) as in Zou and Hastie (2005). We establish in the following Lemma the grouping effect of L_1CP in the identical correlation case.

Lemma 2.3.1. *[The identical correlation case] Given data* $(\mathbf{y}, \mathbf{X})$*, where* $\mathbf{X} = (\mathbf{x}_1|...|\mathbf{x}_p)$ *and parameters* (λ_1, λ_2)*, the response is centered and the predictors are standardized. Let* $\hat{\boldsymbol{\beta}}(\lambda_1, \lambda_2) = \hat{\boldsymbol{\beta}}$ *be the* L_1*CP estimate. If* $\hat{\beta}_i\hat{\beta}_j > 0$ *and* $\rho_{kl} = \rho$, *for all* (k, l) *, then*

$$\frac{1}{\|\mathbf{y}\|_2}\left|\hat{\beta}_j - \hat{\beta}_i\right| \leq \frac{1-\rho^2}{2(p+\rho-1)\lambda_2}\sqrt{2(1-\rho)}.$$

The proof is similar to the proof of Theorem 1 in Zou and Hastie (2005), details of the proof is are given in the appendix.

For the illustration of the grouping effect we use the following Example: we consider $n = 100$ observations described by 40 predictors. The true parameters are

$$\boldsymbol{\beta} = (\underbrace{1.85, ..., 1.85}_{5}, \underbrace{3, ..., 3}_{5}, \underbrace{4, ..., 4}_{5}, \underbrace{0, ..., 0}_{25})^t$$

and $\sigma = 1.5$. The predictors were generated as:

$$\mathbf{x}_i = Z_1 + 0.01\varepsilon_i, \quad Z_1 \sim N(0,1), \quad i = 1, ..., 5$$

$$\mathbf{x}_i = Z_2 + 0.01\varepsilon_i, \quad Z_2 \sim N(0,1), \quad i = 6, ..., 10$$

$$\mathbf{x}_i = Z_3 + 0.01\varepsilon_i, \quad Z_3 \sim N(0,1), \quad i = 11, ..., 15$$

$$\mathbf{x}_i \sim N(0,1), \quad i = 16, ..., 40,$$

where ε_i are independent identically distributed $N(0,1), i = 1, ..., 15$. In this model the three equally important groups have pairwise correlations $\rho \approx 0.99$, and there are 25 pure noise features. As can be seeing in Figure 2.2, for highly correlated variables the L_1CP presents the grouping effect property. While Lasso do not present any grouping effect.

2.3.2 Tuning parameters selection

In practice, it is important to select appropriate tuning parameters λ_2 and λ_1 in order to obtain a good prediction precision. Choosing the tuning parameters can be done via minimizing an estimate of the out-of-sample prediction error. If a validation set is available, this can be estimated directly. Lacking a validation set one can use ten-fold cross validation. Since there are two tuning parameters in the L_1CP, we need to cross-validate on a two dimensional surface. Typically we first pick a (relatively small) grid values for λ_2, say $(0, 0.01, 0.05, 0.1, 0.25, 0.5, 0.75, 0.9, 1, 10, 100)$. Then, for each λ_2, LARS algorithm produces the entire solution path of the L_1CP. The other tuning parameter is selected by ten-fold cross-validation (CV). The chosen λ_2 is the one giving the smallest CV prediction error.

An alternative is to use the uniform design approach of Fang and Wang (1994) to generate candidate points of (λ_1, λ_2). This method actually works for a tuning parameter with arbitrary dimension (cf. Wang et al. 2006). In our experimentations on simulated and real data, the selection of tuning parameters λ_1 and λ_2 is based on the minimization of CV criterion.

2.4 Numerical experiments

A simulation study was run to examine the performance of the L_1CP, under various conditions, with Lasso, Ridge Regression (RR), (CP) and Elastic-net (ENET). We note that BlockBoost is computationally extensive, so do not include it in our experimental studies(cf. ElAnbari and Mkhadri (2008)). The simulation setting is similar to the setting used in Wu et al. (2009) paper for the first four examples and the sixth one, while the five example is taken from Witten and Tibshirani (2009). The first two examples are modified from those of the original Lasso paper (Tibshirani, 1996) and the Elastic-Net paper Zou and Hastie (2005). The third example is taken from Yuan and Lin (2006) and the fifth and sixth examples consider the high dimensional setting where the dimension p can exceed

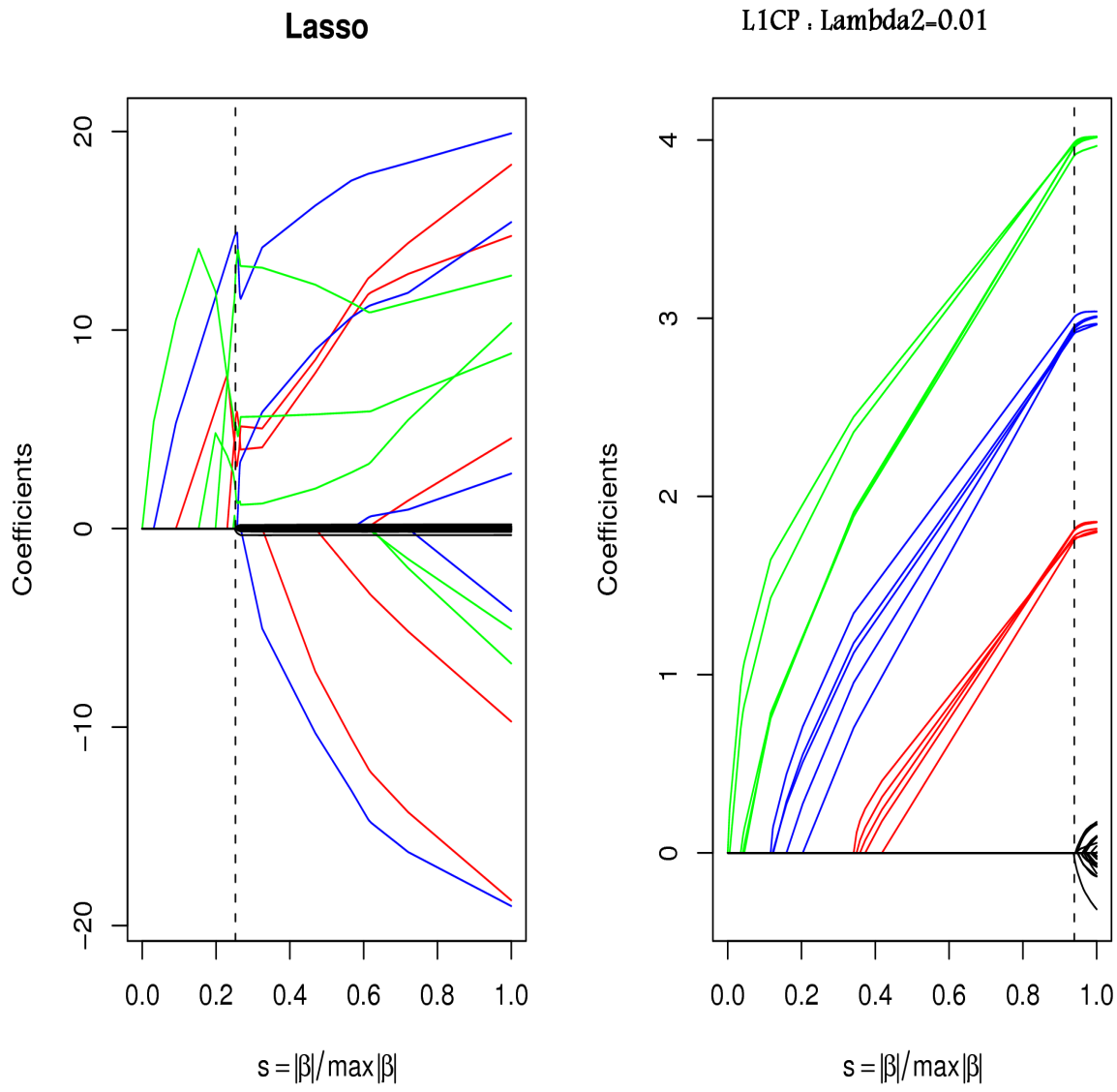


Figure 2.2: Lasso and L_1CP ($\lambda_2 = 0.01$) solution paths: L_1CP presents a the "grouped selection" effect.

the sample size n. To clarify more the performance of the L_1CP and ENET, we consider another seventh example, which is similar to Example 5 in Daye and Jeng (2009).

For each example, 100 data sets were simulated from the regression model,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \varepsilon, \quad \varepsilon \sim N(\mathbf{0}, \sigma^2\mathbf{I}_n),$$

consisting of a training set to fit the model, an independent validation set to select tuning parameters and an independent testing set to evaluate prediction.

Six simulated examples are generated according to the linear regression model defined above. The notation ././. denotes the number of observations in the training, validation and test data sets respectively. For each method and each simulation we compute the prediction accuracy $\text{MSE}_{\mathbf{y}}$ (Mean square error of the response), Hits and FP (respectively the number of relevant and irrelevant variables selected by the procedure) over 100 data sets. Standard are errors estimated by using bootstrap with $B = 500$ resamplings on the 100 mean-squared errors are given into brackets.

1. In Example one, we generated 100 data sets with sample sizes 20/20/200 and ten predictors. The true regression vector is $\boldsymbol{\beta} = (3, -1.5, 0, 0, 1, 0, 0, 0, 2, 0)^t$, so that only three positively and one negatively correlated predictors are truly relevant. We took $\sigma = 3$ and the correlation matrix is given by $\rho(\mathbf{x}_i, \mathbf{x}_j) = 0.5^{|i-j|}$. So, we are in a sparse predictor situation.

2. The second example was introduced in the the Elastic net paper. It is the same as the first one, except that $\beta_j = 0.85$ for all j so that all the variables are relevant.

3. The third example corresponds to $p = 20$ sparse grouped predictors simulated form $N(0, \Sigma)$ with the diagonal and off-diagonal components are fixed to 1 and 0.5, respectively. The true regression vector is $\boldsymbol{\beta} = (0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2)^t$ and $\sigma = 9$. So that only 10 predictors composed of two groups are truly relevant.

4. In this example, we simulate 20/20/200 observations with 40 predictors. The true regression parameter is such that

$$\beta = (\underbrace{3, ..., 3}_{15}, \underbrace{4, ..., 4}_{5}, \underbrace{0, ..., 0}_{20})^T$$

Let $\sigma = 6$, the predictors were generated as follows:

$$\mathbf{x}_i = Z_1 + \varepsilon_i^x, \quad Z_1 \sim N(0, 1), \quad i = 1, ..., 5$$
$$\mathbf{x}_i = Z_2 + \varepsilon_i^x, \quad Z_2 \sim N(0, 1), \quad i = 6, ..., 10$$
$$\mathbf{x}_i = Z_3 + \varepsilon_i^x, \quad Z_3 \sim N(0, 1), \quad i = 11, ..., 15$$
$$\mathbf{x}_i \sim N(0, 1), \quad i = 16, ..., 40$$

where ε_i^x are independent identically distributed (i.i.d.) $N(0, 0.16), i = 1, ..., 15$. Moreover, for $j = 16, \ldots, 40$, the random variables X_j's are i.i.d $N(0, 1)$. In this model of three equally important groups, the correlation ($\rho \approx 0.85$) between relevant predictors is high.

5. Each data set consists of 50/50/200 observations with 50 predictors, so this example corresponds to high dimensional setting where the number of predictors is equal to the sample size. The components of the vector parameter $\boldsymbol{\beta}$ are: $\beta_i = 2$ for $i < 9$ and $\beta_i = 0$ for $i \geq 9$. $\sigma = 2$ and $\rho_{ij} = 0.9 \times \mathbf{1}_{i,j\leq 9}$. The same example was considered by Whitten and Tibshirani (2008) where the test sample size was fixed to 400.

6. The sixth example, considered by Wu et al. (2009), corresponds to high dimensional simulation scenarios correlated groups with large $p = 200$ and small n. The X_i are simulated from $N(0, \Sigma)$ where the jk-th element of Σ is $0.5^{|j-k|}$ and $\boldsymbol{\beta} = (3\mathbf{1}_5, -1.5\mathbf{1}_5, \mathbf{1}_5, 2\mathbf{1}_5, \mathbf{0}_{180})$. So there are only 20 grouped relevant predictors and 180 noisy predictors.

7. In the last example, we simulate 100/100/200 observations with 40 predictors. We put $\sigma = 6$ and the true regression coefficients

$$\beta = (\underbrace{\beta_0, ..., \beta_0}_{15}, \underbrace{1.5, ..., 1.5}_{5}, \underbrace{0, ..., 0}_{20})^T$$

where $\beta \in \{-100, 0.85, 1.5, 3, 4, 100\}$. The predictors are generated as follows:

$$\mathbf{x}_i = Z_1 + 0.43\varepsilon_i^x, \quad Z_1 \sim N(0,1), \quad i = 1, ..., 5$$

$$\mathbf{x}_i \sim N(0,1), \quad i = 16, ..., 40$$

where $\varepsilon_i^x \sim N(0,1)$ and Z_1 are independent. So, the correlation between the first 15 predictors is approximately 0.85.

2.4.1 Examples $1-6$

For each method and each simulation we computed the prediction accuracy MSE_y over 100 data sets. Table 2.1 summarizes mean squared error of the estimation for the response $\mathbf{y}$ ($\text{MSE}_\mathbf{y}$) HITS and FP. In Table 2.1 the best performance is given in boldface.

In the first four examples, L_1CP performs the best in terms of prediction $\text{MSE}_\mathbf{y}$ followed by CP and ENET. While ENET is the winner in Examples 5 and 6 in terms of prediction followed by L_1CP and LASSO. Moreover, the L_1CP performs better than ENET and LASSO for model selection. In fact, L_1CP selects always all relevant variables to be included in the model, while ENET and LASSO may omit some of them as in Examples 2-5. In term of false positive, the performances of L_1CP are relatively comparable to those of ENET except for Examples 5 and 6 is superior.

2.4.2 Example 7

Median mean-squared errors for the simulated Example 7 for $\beta_0 \in \{-100, 0.85, 1.5, 3, 4, 100\}$, based on 100 replications with standard errors in brackets on the 100 mean-squared errors. The median number of HITS and false positives are also reported.

Example		MSE_y	HITS	FP
1	RR	10.86(0.17)	4	6
	CP	10.87(0.18)	4	6
	LASSO	10.66(0.17)	4	3
	ENET	10.61(0.17)	4	3
	L_1CP	**10.51**(0.20)	4	3
2	RR	9.93(0.09)	10	0
	CP	9.76(0.10)	10	0
	LASSO	10.58(0.13)	8	0
	ENET	10.00(0.12)	9	0
	L_1CP	**9.77**(0.11)	10	0
3	RR	96.11(1.49)	10	10
	CP	94.47(1.24)	10	10
	LASSO	103.47(1.54)	7	3
	ENET	95.48(1.20)	8	5
	L_1CP	**93.04**(0.15)	10	6
4	RR	61.73(1.32)	20	20
	CP	61.66(1.52)	20	20
	LASSO	60.60(1.29)	13	7
	ENET	55.14(0.96)	17	7
	L_1CP	**52.64**(1.25)	20	8
5	RR	48.95(0.93)	8	42
	CP	48.71(1.03)	8	42
	LASSO	43.66(0.73)	5	5
	ENET	**37.23**(0.51)	8	1
	L_1CP	41.36(0.52)	8	5
6	RR	191.65(3.37)	20	180
	CP	191.60(3.51)	20	180
	LASSO	169.00(2.58)	7	9
	ENET	**154.47**(2.97)	9	11.5
	L_1CP	158.67(3.08)	10	17

Table 2.1: Median mean-squared errors for the simulated examples of five methods based on 100 replications with standard errors estimated by using the bootstrap with $B = 500$ resamplings on the 100 mean-squared errors. The median number of HITS and false positives are also reported.

Values of β_0		MSE_y	HITS	FP
$\beta_0 = -100$	RR	22.72(0.47)	20	20
	CP	22.72(0.51)	20	20
	LASSO	17.04(0.74)	15	00
	ENET	27.38(1.25)	15	00
	L_1CP	**16.50**(0.77)	15	00
$\beta_0 = 0.85$	RR	9.58(0.22)	20	20
	CP	9.57(0.24)	20	20
	LASSO	11.84(0.45)	16	06
	ENET	10.15(0.20)	17	05
	L_1CP	**8.19**(0.44)	19	10
$\beta_0 = 1.5$	RR	11.46(0.38)	20	20
	CP	11.32(0.38)	20	20
	LASSO	13.95(0.48)	18	05
	ENET	11.94(0.45)	19	04
	L_1CP	**9.62**(0.34)	20	08
$\beta_0 = 3$	RR	15.23(0.48)	20	20
	CP	15.12(0.59)	20	20
	LASSO	14.52(0.46)	19	03
	ENET	**12.30**(0.40)	18	02
	L_1CP	12.57(0.31)	19	03
$\beta_0 = 4$	RR	18.46(0.62)	20	20
	CP	18.40(0.61)	20	20
	LASSO	15.80(0.45)	19	02
	ENET	**13.22**(0.44)	18	01
	L_1CP	14.38(0.57)	19	02
$\beta_0 = 100$	RR	22.82(1.12)	20	20
	CP	22.77(1.03)	20	20
	LASSO	16.88(0.64)	15	00
	ENET	25.59(0.90)	15	00
	L_1CP	**16.21**(0.66)	15	00

Table 2.2: Median mean-squared errors for the simulated Example 7 for $\beta_0 \in \{-100, 0.85, 1.5, 3, 4, 100\}$, based on 100 replications with standard errors estimated by using the bootstrap with $B = 500$ resamplings on the 100 mean-squared errors. The median number of HITS and false positives are also reported.

As we can see in Table 2.2, L_1CP has competitive prediction accuracies with all methods for small values of β_0, except perhaps when compared to ENET for $\beta_0 = 4$. Moreover, L_1CP has consistently smaller MSE_y than all other regularization methods when $|\beta_0| = 100$. However, the performance of ENET deteriorates greatly in the latter case with a difference of about 11 percent with L_1CP. Furthermore, the performance L_1CP in term of correct selection is slightly better than that of ENET for small values of β_0, and it is similar to those of ENET and Lasso when $|\beta_0| = 100$. While the performance of ENET in term of incorrect selection is slightly better than that of L_1CP especially for small values of β_0.

2.5 Real data sets Experiments

Here we examine the performance of the L_1CP for two real world data sets: the US crim and GC-retention indices of polycyclic aromatic compounds described by $p = 15$ and $p = 467$ explanatory predictors, respectively. The dimension p of the US crim data set is smaller than the sample size ($n = 47$), while the dimension of the PAC data exceeds largely the sample size $n = 209$.

2.5.1 US crim data

Criminologists are interested in the effect of punishment regimes on crime rates. This has been studied using aggregate data on 47 states of the USA for 1960. This data set contained the following 16 columns: percentage of males aged $14 - 24$ (M), indicator variable for a Southern state (So), mean years of schooling (Ed), police expenditure in 1960 (Po1), police expenditure in 1959 (Po2), labour force participation rate (LF), number of males per 1000 females (M.F), state population (Pop), number of non-whites per 1000 people (NW), unemployment rate of urban males $14-24$ (U1), unemployment rate of urban males $35 - 39$ (U2), gross domestic product per head (GDP), income inequality (Ineq), probability of imprisonment (Prob), average time served in state prisons (Time) and the outcome is the rate of crimes in a particular category per head of population (y). In order to investigate the performances of the L_1CP, the data set has been split 25 times into a training set of 23 observations and a test set of 24 observations. We have used CV criterion for selecting tuning parameters for all methods. The results are summarized in Table 2.3 and Figure 2.3. From Table 2.3, L1CP is the winner in term of prediction followed by ENET. However, it can be seen from Figure 2.3 that ENET and Lasso are more variable than all other methods. While L1CP has the best performance in terms of variability. In terms of model selection, L1CP and ENET are comparable: they select the same number 11 of predictors.

2.5.2 GC-Retention PAC data

The second data set we analyze is a subset of GC-retention indices of polycyclic aromatic compounds (y) which have being modeled by molecular descriptors (X). The data contains

Method	median MSE_y	median no. of selected variables
RR	0.092	15
CP	0.091	15
LASSO	0.092	8
ENET	0.087	11
L_1CP	**0.083**	11

Table 2.3: UScrim data - median test mean squared error over 25 random splits for different methods.

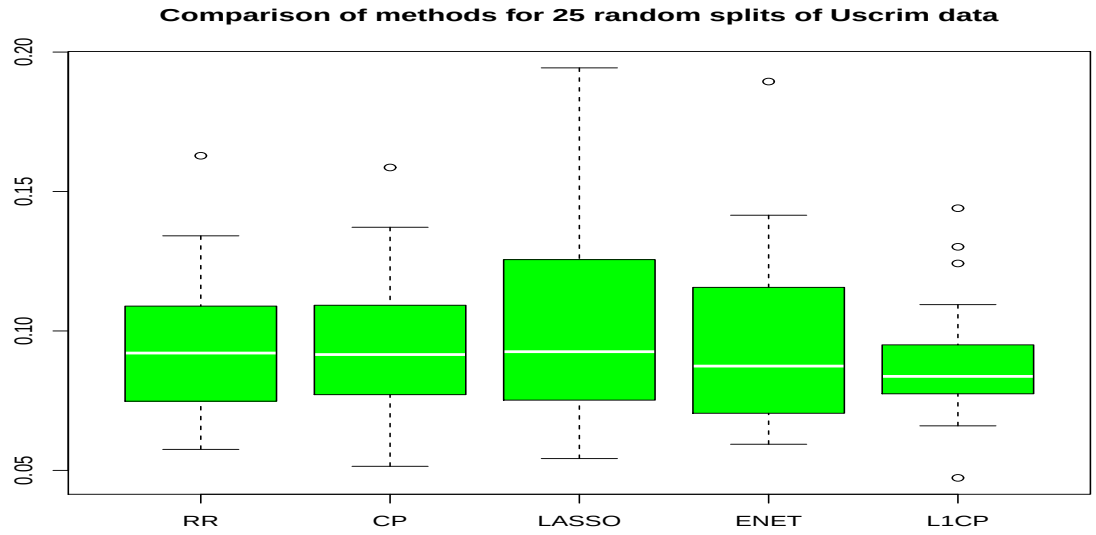


Figure 2.3: UScrim data - median test mean squared error over 25 random splits for different methods.

measurements on 467 variables for 209 objets. In contrast to the first example of US crim data, the dimension of the new example is much larger than the sample size (see R package *chemometrics* and Varmuza and Filzmoser (2009) for the description of data in chapter 4). We first randomly divide the data into a test data set of 105 observations, with the remainder making up the training data. As for the US crim data, we have repeated this procedure 25 times. The results of L_1CP and its competitors are given in Table 2.4 and Figure 2.4. CP has the best performance in term of prediction accuracy followed by L1CP. However, the latter presents less variability than all other methods as it can be seen from Figure 2.4. But, it selects more variables than Elastic Net.

Method	median $MSE_{\mathbf{y}}$	median no. of selected variables
RR	162.1572	467
CP	**104.4982**	467
LASSO	135.1979	59
ENET	130.7648	43
L_1CP	113.8402	63

Table 2.4: PAC data - median test mean squared error over 25 random splits for different methods.

2.6 Discussion

We have proposed the L_1CP procedure for simultaneous estimation and variable selection in linear regression problem. It is a regularization procedure based on the penalized least squares with a mixture of L_1 norm and a weighted L_2 norm penalties. Similar to the Elastic-Net method, the L_1CP encourages a grouping effect, where strongly correlated predictors tend to be in or out of the model together. Additionally, the weighted L_2 penalty explicitly links strength of penalization to the correlation between predictors. Due to the efficient path algorithm (LARS), our procedure enjoys the computational advantage of the Elastic-Net. Our simulations and empirical prediction results have shown good performance of our method in term of prediction accuracy, identification of relevant variables while encouraging a grouping effect.

The key contribution of the L_1CP is the identification of settings where the Elastic-Net fails to product good results. In fact, our empirical prediction results show that two setting between the sample size n and the dimension p should be distinguished. First if $p \leq n$, even in small and high dimension settings, the L_1CP has shown good empirical performance for both simulated and real world experiments in terms of prediction accuracy and model selection. Moreover, if $p \gg n$, L_1CP remains competitive and as ENET allows for the selection of more than n variables.

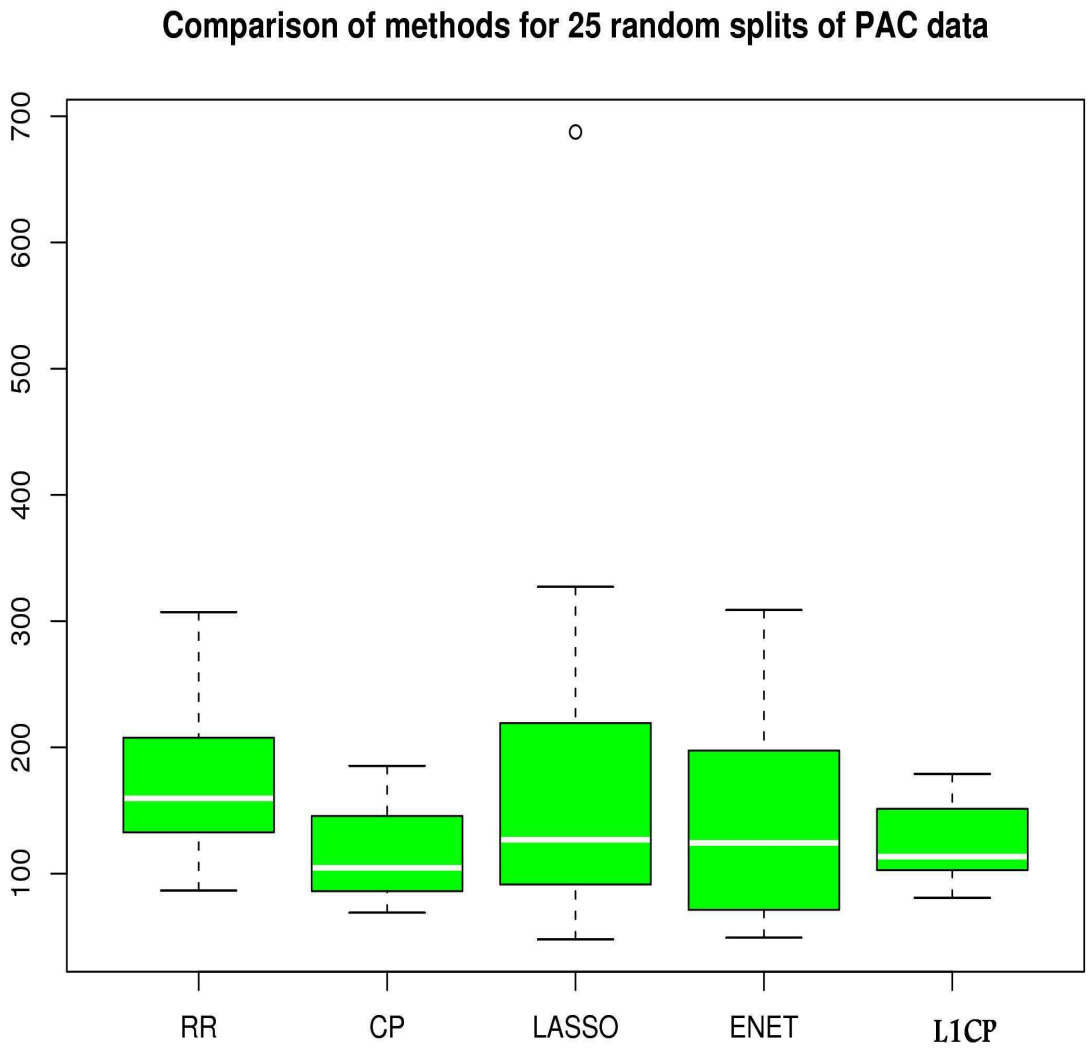


Figure 2.4: PAC data - median test mean squared error over 25 random splits for different methods.

Proof of Lemma 2.3.1

Proof. Let $\hat{\beta}(\lambda_1, \lambda_2) = \arg\min_\beta\{L(\lambda_1, \lambda_2, \beta)\}$. If $\hat{\beta}_i(\lambda_1, \lambda_2)\hat{\beta}_j(\lambda_1, \lambda_2) > 0$, then both $\hat{\beta}_i(\lambda_1, \lambda_2)$ and $\hat{\beta}_j(\lambda_1, \lambda_2)$ are non-zero, and we have $\text{sign}(\hat{\beta}_i(\lambda_1, \lambda_2)) = \text{sign}(\hat{\beta}_j(\lambda_1, \lambda_2))$. Then $\hat{\beta}(\lambda_1, \lambda_2)$ must satisfies

$$\frac{\partial L(\lambda_1, \lambda_2, \beta)}{\partial \beta}|_{\beta=\hat{\beta}(\lambda_1,\lambda_2)} = \mathbf{0}. \tag{2.8}$$

Hence we have

$$-2\mathbf{x}_i^T\{\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_1, \lambda_2)\} + \lambda_1 \text{sign}\{\hat{\beta}_i(\lambda_1, \lambda_2)\} + 2\lambda_2 \sum_{k=1}^{p} \omega_{ik}\hat{\beta}_k(\lambda_1, \lambda_2) = 0, \tag{2.9}$$

$$-2\mathbf{x}_j^T\{\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_1, \lambda_2)\} + \lambda_1 \text{sign}\{\hat{\beta}_j(\lambda_1, \lambda_2)\} + 2\lambda_2 \sum_{k=1}^{p} \omega_{jk}\hat{\beta}_k(\lambda_1, \lambda_2) = 0, \tag{2.10}$$

Subtracting equation (2.9) from (2.10) gives

$$(\mathbf{x}_j^T - \mathbf{x}_i^T)\{\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_1, \lambda_2)\} + \lambda_2 \sum_{k=1}^{p} (\omega_{ik} - \omega_{jk})\hat{\beta}_k(\lambda_1, \lambda_2) = 0,$$

which is equivalent to

$$\sum_{k=1}^{p} (\omega_{ik} - \omega_{jk})\hat{\beta}_k(\lambda_1, \lambda_2) = \frac{1}{\lambda_2}(\mathbf{x}_i^T - \mathbf{x}_j^T)\hat{\mathbf{r}}(\lambda_1, \lambda_2), \tag{2.11}$$

where $\hat{\mathbf{r}}(\lambda_1, \lambda_2) = \mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_1, \lambda_2)$ is the residual vector. Since $\mathbf{X}$ is standardized, then

$$\|\mathbf{x}_i - \mathbf{x}_j\|_2^2 = 2(1 - \rho_{ij}).$$

Because $\hat{\beta}(\lambda_1, \lambda_2)$ is the minimizer we must have

$$L\{\lambda_1, \lambda_2, \hat{\beta}(\lambda_1, \lambda_2)\} \leq L\{\lambda_1, \lambda_2, \beta = \mathbf{0}\},$$

i.e.

$$\|\hat{\mathbf{r}}(\lambda_1, \lambda_2)\|^2 + \lambda_2\hat{\beta}^T(\lambda_1, \lambda_2)\mathbf{W}\hat{\beta}(\lambda_1, \lambda_2) + \lambda_1\|\hat{\beta}(\lambda_1, \lambda_2)\|_1 \leq \|\mathbf{y}\|_2^2.$$

So $\|\hat{\mathbf{r}}(\lambda_1, \lambda_2)\|_2 \leq \|\mathbf{y}\|_2$. Then the equation (2.11) implies that

$$\frac{1}{\|\mathbf{y}\|_2} \mid \sum_{k=1}^{p} (\omega_{ik} - \omega_{jk})\hat{\beta}_k(\lambda_1, \lambda_2) \mid \leq \frac{1}{\lambda_2\|\mathbf{y}\|_2}\|\hat{\mathbf{r}}(\lambda_1, \lambda_2)\|\|\mathbf{x}_1 - \mathbf{x}_2\| \leq \frac{1}{\lambda_2}\sqrt{2(1 - \rho_{ij})}. \tag{2.12}$$

We have:

$$\omega_{ii} = -\sum_{s\neq i} \frac{\omega_{is}}{\rho_{is}} \qquad \text{and} \qquad \omega_{jj} = -\sum_{s\neq j} \frac{\omega_{js}}{\rho_{js}}. \tag{2.13}$$

Then

$$\sum_{k=1}^{p}(\omega_{ik}-\omega_{jk})\hat{\beta}_k(\lambda_1,\lambda_2)=\frac{-2}{1-\rho_{ij}}[\hat{\beta}_j(\lambda_1,\lambda_2)-\hat{\beta}_i(\lambda_1,\lambda_2)]+2SN \tag{2.14}$$

where

$$SN=\sum_{k\neq i,j}\frac{1}{1-\rho_{ki}^2}[\hat{\beta}_i(\lambda_1,\lambda_2)-\rho_{ki}\hat{\beta}_k(\lambda_1,\lambda_2)]+\frac{1}{1-\rho_{kj}^2}[\rho_{kj}\hat{\beta}_k(\lambda_1,\lambda_2)-\hat{\beta}_j(\lambda_1,\lambda_2)].$$

if

$$\rho_{ki}=\rho_{kj}=\rho,\quad \forall k=1,...,p,$$

we have

$$SN=\frac{p-2}{1-\rho^2}(\hat{\beta}_i(\lambda_1,\lambda_2)-\hat{\beta}_j(\lambda_1,\lambda_2)).$$

So using (2.12) we have:

$$\begin{aligned}\frac{1}{\|\mathbf{y}\|_2}\left|\hat{\beta}_j-\hat{\beta}_i\right| &\leq \frac{1-\rho^2}{2(p+\rho-1)\lambda_2}\sqrt{2(1-\rho)}\\ &\leq \frac{1}{\lambda_2}\sqrt{2(1-\rho)}.\end{aligned}$$

This completes the proof. □

Chapter 3

The adaptive Gril estimator with a diverging number of parameters

Résumé : Dans ce chapitre, nous proposons l'ensemble d'estimateurs *Adaptive Generalized Ridge-Lasso* (AdaGril), permettant des corriger les défauts théoriques (abscence de propriété Oracle) des estimateurs L1CP, Elastic net, smooth Lasso et Weithed Fusion parmi autres, tout en tenant compte de structures supplémentaires qui peuvent exister entre les variables explicatives, telles que présence de forte corrélation ou ordres entre les variables. La méthode est fondée sur l'introduction d'un poids adaptatif dans la norme l_1 et en considérant une norme l_2 générale.

3.1 Introduction

We consider the problem of variable selection and estimation for general linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \tag{3.1}$$

where $\mathbf{y} = (y_1, ..., y_n)^t$ is an n-vector of responses, $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_p)$ is a nxp design matrix of p predictor vectors of dimension n, $\boldsymbol{\beta}^*$ is a p-vector of unknown parameters which are to be estimated, t stands for the transpose and $\boldsymbol{\varepsilon}$ is a n-vector of (i.i.d.) random errors with mean 0 and variance σ^2. Without loss of generality we assume that the data are centered.

When p is large, selection of a small number of predictors that contribute to the response leads often to a parsimonious model. It amounts to assuming that $\boldsymbol{\beta}^*$ is sparse in the sense $s < p$ components are non-zero. Denote the set of non-zero values by $\mathcal{A} = \{j; \boldsymbol{\beta}_j^* \neq 0\}$. In this setting, variable selection can improve on both estimation accuracy and interpretation. Our goal is to determine the set $\mathcal{A}$ and to estimate the true corresponding coefficients.

Sparsity is associated to high dimensional data, where the number of predictors p is typically comparable or exceeds the sample size n. The problem occurs frequently in genomics and proteomics studies, functional MRI, tumor classification and signal processing

(Fan and Lv, 2008). In many of these applications, we would like to achieve both variable reduction and prediction accuracy.

Variable selection for high dimensional data has received a lot of attention recently. In the last decade interest has focused on penalized regression methods which implement both variable selection and coefficient estimation in a single procedure. The most well known of these procedures are Lasso (Tibshirani, 1996; Chen et al., 1998) and SCAD (Fan and Li, 2001), which have good computational and statistical properties.

In fact, there has been a large rapidly growing body of literature for the Lasso and SCAD studies over the past few years. (Osborne et al., 2000) derived the optimality conditions associated with the Lasso solution. Some theoretical statistical aspects of the Lasso estimator of the regression coefficients have been derived by (Knight and Fu, 2000) in finite dimension setting. Many other extensions for asymptotic and non asymptotic results can be found in (Zhao and Yu, 2006) and (Bunea et al., 2007), etc.

Various extensions and modifications of the Lasso have been proposed to ensure that on one hand, the variable selection process is consistent and on the other hand, the estimated regression coefficient has a fast rate of convergence. (Fan and Li, 2001) showed that the SCAD enjoys the oracle property, that is, the SCAD estimator can perform as well as the oracle if the penalization parameter is appropriately chosen. (Fan and Peng, 2004) studied the asymptotic behavior of SCAD when the dimensionality of the parameter diverges. (Fan and Li, 2001) showed that asymptotically the Lasso estimates produce non-ignorable bias. (Zou, 2006) showed that the Lasso has not the oracle property in finite parameter setting as conjectured in (Fan and Li, 2001). (Zhao and Yu, 2006) established the same result for $p > n$ case.

To overcome the bias problem of Lasso, (Zou, 2006) proposed the adaptive Lasso estimator (AdaLasso) defined by

$$\hat{\boldsymbol{\beta}}_{\text{AdaLasso}} = \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - X\boldsymbol{\beta}||_2^2 + \lambda \sum_{j=1}^{p} \hat{w}_j |\boldsymbol{\beta}_j|, \tag{3.2}$$

where the weights $\hat{w}_j = (|\hat{\boldsymbol{\beta}}_j^0|)^{-\gamma}$ $(j = 1, \ldots, p)$, with γ is a positive constant and $\hat{\boldsymbol{\beta}}^0$ is an initial consistent estimate of $\boldsymbol{\beta}^*$. We recall here that $\hat{\boldsymbol{\beta}}_{\text{Lasso}}$ is the solution to a similar equation (3.2) in which $\hat{w}_j = 1$ for all j.

The second most drawback of the Lasso (and also AdaLasso or ℓ_1 penalization methods) is its poor performance when there are highly correlated predictors. Under high dimensionality, the situation is particularly dire. (Zou and Hastie, 2005) showed that the Lasso estimates are instable when predictors are highly correlated. They proposed the Elastic Net (Enet) for variable selection, which combines ℓ_1 and ℓ_2 penalties. (ElAnbari and Mkhadri, 2008) proposed a procedure called Elastic Corr-Net (Cnet) which combines the ℓ_1 and the correlation based penalty of (Tutz and Ulbricht, 2009). (Daye and Jeng, 2009) proposed a slightly similar approach called the Weighted Fusion (WFusion). These two approaches can incorporate information redundancy among correlated predictors for estimation and variable selection. Numerical studies have shown that Cnet and WFusion

outperform the Lasso and Enet in certain situations. In the same setting, (Hebiri and van De Geer, 2010) considered the Smooth-Lasso procedure (S-Lasso), a modification of the Fused-Lasso procedure (Tibshirani et al., 2005), in which a second ℓ_1 Fused penalty is replaced by the smooth ℓ_2 norm penalty. The general formulation englobing all the four latter approaches, called the Generalized Ridge Lasso (Gril) estimator, can be defined by

$$\hat{\boldsymbol{\beta}}_{\text{Gril}}(\lambda_1, \lambda_2) = \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - X\boldsymbol{\beta}||_2^2 + \lambda_1||\boldsymbol{\beta}||_1 + \lambda_2\boldsymbol{\beta}^t\mathbf{Q}\boldsymbol{\beta}, \tag{3.3}$$

where $\mathbf{Q}$ is a positive semi-definite matrix. A similar formulation was cited in (Daye and Jeng, 2009) and (Hebiri and van De Geer, 2010) in regression problem and in (Clemmensen et al., 2011) in classification problem. Moreover, the computation of the estimates of the parameters of Gril procedure can be obtained efficiently via a modification of LARS algorithm (Efron et al., 2004).

The Gril estimator (Enet in particular) resolves the collinearity problem of Lasso, and AdaLasso estimator possesses the oracle property of SCAD. However, in high dimensional setting, the Gril misses the oracle property, while AdaLasso estimates are instable because of bias problem of Lasso. Recently, (Zou and Zhang, 2009) proposed the adaptive Elastic Net (AdaEnet) that combines the strengths of ℓ_2 norm and the adaptive weighted ℓ_1 shrinkage. They established the oracle property of the AdaEnet when the dimension diverges with the sample size. Independently, (Ghosh, 2011) proposed the same AdaEnet, but he specially focused on the grouped selection property of AdaEnet along with its model selection complexity.

Despite its popularity, Enet (an also AdaEnet) has been critiqued for being inadequate, notably in situations in which additional structural knowledge about predictors should be taken into account (**?**ElAnbari and Mkhadri, 2008; Daye and Jeng, 2009; Hebiri and van De Geer, 2010; Slawski et al., 2010; She, 2010). To this end, these authors complement ℓ_1-regularized with a second regularized based on the total variation or the quadratic penalty. The former aims at the explicit inclusion of structural knowledge about predictors, while the latter aims at taken into account some type of correlation between predictors. The experimental results of these alternatives have shown that Enet performs worse in grouping highly correlated predictors. But, similar to Enet, these new estimators are asymptotically biased because of the ℓ_1 component in the penalty and they cannot achieve selection consistency and estimation efficiency simultaneously.

Therefore, there is a need to develop methods that take into account of additional structural information of predictors and have the oracle property. In the same spirit of AdaEnet, we propose the adaptive Gril (AdaGril) that penalizes the least square loss using a mixture of weighted ℓ_2 norm and the adaptive weighted ℓ_1 penalty. We first highlight the grouped selection property for AdaCnet method (one type of AdaGril) in the equal correlation case, meaning that it selects or drops highly correlated predictors together. Under weak conditions, as in (Zou and Zhang, 2009), we study its asymptotic properties when the dimension diverges with the sample size. In particular, we show that the AdaGril enjoys the oracle property with a diverging number of predictors. Moreover, we show that AdaGril estimator achieves a Sparsity Inequality, i. e., a bound in terms of the number

of non-zero components of the 'true' regression coefficient. This bound is obtained under a similar weak Restricted Eigenvalue (RE) condition used for Lasso. Finally, a detailed experimental performance comparison of different Gril estimators is considered.

In Section 3.2, we focus on the Cnet method and sketch briefly other Gril estimators. A computational algorithm to approach their solutions is presented and we briefly summarize some of their statistical properties obtained in fixed dimensional setting. In Section 3.3, we define the Adaptive Gril estimator and begin by showing the property of grouping effect of AdaCnet in equal correlation case. Then, we establish the Statistical asymptotic theory of the AdaGril when the dimension diverges, including the oracle property. We end by showing that AdaGril achieves a Sparsity Inequality. Computational aspects of adaptive Gril are discussed in Section 3.4. A detailed simulation study is performed in Section 3.5, which illustrates the performance of particular three cases of Gril and AdaGril estimators in relation to AdaEnet estimator. A brief discussion is given in Section 3.6. All technical proofs are provided in appendix.

3.2 Different Gril estimators

In this section, we present a brief introduction of our alternative to Enet, called *the Elastic Corrnet (Cnet)*, which takes into account the correlation between predictors in the quadratic penalty. Two other competitor Gril estimators are presented and their statistical properties are summarized.

3.2.1 Doubly regularized techniques

Suppose that the predictors are $\mathbf{x}_i = (x_{i1}, \ldots, x_{ip})$ and response values y_i, for $i = 1, \ldots, n$.

Apart from lack of consistency, it is well known that Lasso has two limitations; for example a) Lasso does not encourage grouped selection in the presence of high correlated covariates and b) for $p > n$ case Lasso can select at most n covariates. To overcome these limitations, (Zou and Hastie, 2005) proposed elastic net which combines both ridge (ℓ_2) and Lasso (ℓ_1) penalties. So, Enet procedure corresponds to the Gril estimator with $\mathbf{Q} = \mathbf{I}_n$, where $\mathbf{I}_n$ is the nxn identity matrix.

Despite its popularity, Enet (an also AdaEnet) has been critiqued for being inadequate, notably in situations in which additional structural knowledge about predictors should be taken into account (**?**ElAnbari and Mkhadri, 2008; Daye and Jeng, 2009; Hebiri and van De Geer, 2010; Slawski et al., 2010; She, 2010). To this end, these authors complement ℓ_1 regularized with a second regularized based on the total variation or the quadratic penalty. The former aims at the explicit inclusion of structural knowledge about predictors, while the latter aims at taken into account some type of correlation between predictors. One example of the latter, is the Elastic Corr-Net (Cnet) (ElAnbari and Mkhadri, 2008) which is a modification of Enet in which the ridge penalty term is replaced by the correlation

based penalty term $P_c(\boldsymbol{\beta})$ defined by

$$P_c(\boldsymbol{\beta}) = \sum_{j=1}^{p-1} \sum_{j>i} \left\{ \frac{(\boldsymbol{\beta}_i - \boldsymbol{\beta}_j)^2}{1-\rho_{ij}} + \frac{(\boldsymbol{\beta}_i + \boldsymbol{\beta}_j)^2}{1+\rho_{ij}} \right\},$$

where $\rho_{ij} = \mathbf{x}_i^t \mathbf{x}_j$ denotes the (empirical) correlation between the ith and the jth predictors. The correlation based penalty $P_c(\boldsymbol{\beta})$, introduced by (Tutz and Ulbricht, 2009), will encourages grouping effect for highly correlated variables. This penalty can be written in a simple quadratic form

$$P_c(\boldsymbol{\beta}) = \boldsymbol{\beta}^t \mathbf{L} \boldsymbol{\beta},$$

where $\mathbf{L} = (\ell_{ij})_{1 \leq i,j \leq p}$ is a positive definite matrix with general term, assuming that $\rho_{ij}^2 \neq 1$ for $i \neq j$,

$$\ell_{ij} = \begin{cases} 2 \sum_{s \neq i} \frac{1}{1-\rho_{is}^2}, & i = j \\ -2 \frac{\rho_{ij}}{1-\rho_{ij}^2}, & i \neq j. \end{cases} \tag{3.4}$$

Hence, Cnet is a particular Gril estimator with the weighted matrix $\mathbf{Q}$ defined by (3.4). Cnet provided a good performance in simulations and real applications specially for highly correlated predictors.

We can mention also the Weighted Fusion (WFusion) (Daye and Jeng, 2009) and the Smooth-Lasso (S-Lasso) (Hebiri and van De Geer, 2010) as alternatives to Cnet. In the former, the correlation based penalty is replaced by a modified weighted penalty $\tilde{P}_c(\boldsymbol{\beta}) = \sum_{j>i} w_{ji}(\boldsymbol{\beta}_i - s_{ij}\boldsymbol{\beta}_j)^2$, where $w_{ji} = |\rho_{ij}|^\gamma/(1-|\rho_{ij}|)$, $s_{ij} = \text{sgn}(\rho_{ij})$ the sign of ρ_{ij} and $\gamma > 0$ is a tuning parameter. While, the latter is a modification of the Fused-Lasso procedure (Tibshirani et al., 2005), in which a second ℓ_1 Fused penalty is replaced by the smooth ℓ_2 norm penalty. This quadratic term helps to tackle situations where the regression vector is structured such that its coefficients vary slowly. Surprisingly, this simple modification leads to good performance, specially when the regression vector is 'smooth', i. e., when the variations between successive coefficients of the unknown parameter of the regression are small.

3.2.2 A computational algorithm

In this section we propose a modification of the Elastic-Net algorithm for finding a solution of the penalized least squares problem (3.3) of the Gril. The main idea is to transform the Gril problem into an equivalent Lasso problem on the augmented data (cf. (Zou and Hastie, 2005)). Let

$$\tilde{\mathbf{X}}_{(n+p)\times p} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{L}^t \end{pmatrix}, \quad \tilde{\mathbf{y}}_{(n+p)} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \quad \text{and} \quad \tilde{\varepsilon}_{(n+p)} = \begin{pmatrix} \varepsilon \\ -\sqrt{\lambda_2}\mathbf{L}^t\boldsymbol{\beta}^* \end{pmatrix}, \tag{3.5}$$

where $\mathbf{Q}$ is a real symmetric semi positive-definite square matrix with Choleski decomposition $\mathbf{Q} = \mathbf{L}\mathbf{L}^t$ and $\mathbf{L} = \mathbf{Q}^{\frac{1}{2}}$. The Gril estimator is defined as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1.$$

The latter result is a consequence of simple algebra, and it motivates the following comment on the Gril method.

Remark 1. The Gril estimates can be computed via the Lasso modification of the LARS algorithm. For a fixed λ_2, it constructs at each step, which corresponds to a value of λ_1, an estimator based on the correlation between covariates and the current residue. Then for a fixed λ_2, we obtain the evolution of the Gril estimator coefficient values when λ_1 varies. It provides the coefficient regularization paths of the Gril estimator which are piecewise linear (Efron et al., 2004). Consequently, the Gril algorithm requires the same order of magnitude of computational effort as the OLS estimate via the Lasso modification of the LARS algorithm.

Remark 2. If $p > n$, it is well known that LARS and its Lasso versions can select at most n variables before it puts all coefficients to nonzero. Now, applied LARS to augmented data $(\tilde{\mathbf{y}}, \tilde{\mathbf{X}})$, the lasso modification of the LARS algorithm is able to select all the p predictors in all situations. So the first limitation of the Lasso is easily surmounted. Moreover, the variable selection is performed in a fashion similar to the Lasso.

3.2.3 Statistical properties of Different Gril estimators

The model is assumed to be sparse, i. e. most the regression coefficients of $\boldsymbol{\beta}^*$ are exactly zero corresponding to predictors that are irrelevant to the response. Without loss of generality, we assume that the s first components of vector $\boldsymbol{\beta}^*$ are non-zero. We briefly summarizes in this section the classical properties of model selection consistency of particular Gril estimators.

(Yuan and Lin, 2007) are the first to give a necessary and sufficient condition on the generating covariance matrices for the Elastic net to select the true model when q and p are fixed. The latter is called the Elastic Irrepresentable Condition (EIC) which is an extension of the Irrepresentable Condition (IC), defined in (Zhao and Yu, 2006), for Lasso's model selection consistency. For the general scaling of q, p and n, (Jia and Yu, 2010) give conditions on the relationship between q, p and n such that EIC guarantees the Elastic net's model selection consistency. Moreover, they showed that EIC is weaker than IC. In the same spirit, consistency properties and asymptotic normality are established when $p \leq n$ for WFusion (Daye and Jeng, 2009). For high dimensional setting $p > n$, (Hebiri and van De Geer, 2010) established recently variable selection consistency results for their Quadratic estimator, which corresponds exactly to our Gril estimator. They showed that Gril estimator achieves a Sparsity Inequality, i. e., a bound in terms of the number non-zero components of the 'true' vector regression. The latter result for $n > p$ is extended to AdaGril estimator in the next Section and its oracle properties are detailed when p diverges.

3.3 The adaptive Gril estimator

Now a revised version of Gril estimator, called AdaGril, is proposed by incorporating the adaptive weights in the ℓ_1 penalty of equation (3.3). So, AdaGril is a combination of Gril and AdaLasso. We first assume that $\hat{\boldsymbol{\beta}}^0$ is an initial estimator of $\boldsymbol{\beta}^*$ which is a root n-consistent. For example, we can choose $\hat{\boldsymbol{\beta}}_{\text{ols}}$ or $\hat{\boldsymbol{\beta}}_{\text{Gril}}$, and we construct the weights by

$$\hat{\omega}_j = (|\hat{\boldsymbol{\beta}}_j(\text{Gril})|)^{-\gamma}, \quad j = 1, ..., p, \tag{3.6}$$

where γ is a positive constant. Let $(q_1, ..., q_p)$ be the diagonal elements of $\mathbf{Q}$ and let the $p \times p$ matrix $\mathbf{N}$ defined by

$$\mathbf{N} = \text{diag}\left(1 + \frac{\lambda_2 q_1}{n}, 1 + \frac{\lambda_2 q_2}{n}, \ldots, 1 + \frac{\lambda_2 q_p}{n}\right).$$

Then, the adaptive Gril estimates are defined by

$$\hat{\boldsymbol{\beta}}(\text{AdaGril}) = \mathbf{N}\left\{\arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_2 \boldsymbol{\beta}^t \mathbf{Q} \boldsymbol{\beta} + \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j |\boldsymbol{\beta}_j|\right\}. \tag{3.7}$$

To overcome dividing by zeros, we can choose $\hat{\omega}_j = (|\hat{\boldsymbol{\beta}}_j(\text{Gril}) + 1/n|)^{-\gamma}$ or $\hat{\omega}_j = \infty$.

Now, it is clear that AdaGril combines the strengths of Ridge regression and AdaLasso. So, AdaGril will avoid both the problem of collinearity and bias problem of Lasso in high dimensional setting. The tuning parameters λ_1^* and λ_1 are directly responsible of sparsity of the estimates and are allowed to be different. While the same value of λ_2 is used for Gril and AdaGril estimators, because the quadratic norm in the ℓ_2 penalty leads to the same kind of contribution in both estimators.

3.3.1 The grouping effect of AdaCnet

Grouping effect is expressed when the regression coefficients of a group of highly correlated variables tend to be equal (up to a change of sign if negatively correlated). Similar to Cnet estimator, the AdaCnet estimator has the natural tendency of grouping each pair of regression coefficients according to their correlations. We establish in the following lemma the grouping effect of AdaCnet in the case of equal correlations.

Lemma 3.3.1. *Given data* $(\mathbf{y}, \mathbf{X})$*, where* $\mathbf{X} = (\mathbf{x}_1|...|\mathbf{x}_p)$ *and parameters* (λ_1^*, λ_2)*, the response is centered and the predictors* $\mathbf{X}$ *standardized. Let* $\hat{\boldsymbol{\beta}}(\lambda_1^*, \lambda_2)$ *be the AdaCnet estimate.*

If $\hat{\boldsymbol{\beta}}_i(\lambda_1^*, \lambda_2)\hat{\boldsymbol{\beta}}_j(\lambda_1^*, \lambda_2) > 0$ *and* $\rho_{kl} = \rho$, *for all* (k, l) *, then*

$$\frac{1}{\|\mathbf{y}\|_2}\left|\hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\beta}}_i\right| \leq \frac{1 - \rho^2}{2(p + \rho - 1)\lambda_2}\left[\sqrt{2(1-\rho)} + \frac{\gamma\lambda_1^*}{\|\mathbf{y}\|_2 \min(|\overset{\circ}{\hat{\boldsymbol{\beta}}_i}|, |\overset{\circ}{\hat{\boldsymbol{\beta}}_j}|)^{\gamma+1}}|\overset{\circ}{\hat{\boldsymbol{\beta}}_i} - \overset{\circ}{\hat{\boldsymbol{\beta}}_j}|\right]$$

Remark 3. We note that $\gamma = 0$ leads the grouping effect of the Cnet as a special case. We also observe that the grouping effect has contributions not only from quadratic type penalty but also from L_1 type adaptive penalty. However if $\lambda_2 \to 0$, then it is not possible to capture any grouping effect from only the L_1 type adaptive penalty. Moreover, when considering $\hat{\boldsymbol{\beta}}_j$ as univariate OLS estimates with $\min(|\hat{\boldsymbol{\beta}}_i|, |\hat{\boldsymbol{\beta}}_j|) \geq 1$, the latter becomes

$$\frac{1}{\|\mathbf{y}\|_2} \left| \hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\beta}}_i \right| \leq \frac{1-\rho^2}{2(p+\rho-1)\lambda_2} (2 + \gamma\lambda_1^*) \sqrt{2(1-\rho)}.$$

3.3.2 Model selection consistency for AdaGril when p diverges

The oracle properties of the adaptive Elastic Net is provided in (Ghosh, 2011) for $p \leq n$. But, a detailed and much more elaborate discussion of the oracle properties of the adaptive elastic net is provided in (Zou and Zhang, 2009). In this section and as in (Zou and Zhang, 2009) we establish the oracle properties of the AdaGril estimator when p diverges (i. e. $p(n) = n^\nu, 0 \leq \nu < 1$). Moreover, we provide a bound on the mean squared sparsity inequality, that is a bound on the mean squared risk that takes into account the sparsity of the oracle regression vector $\boldsymbol{\beta}$.

3.3.2.1 Mean Sparsity Inequality

Now we establish the mean sparsity inequality achieved by the AdaGril estimator. For this purpose, we need the following assumption on the minimum and the maximum eigenvalues of the semi-positive definite matrices $\mathbf{X}^t\mathbf{X}$ and $\mathbf{Q}$, respectively.
(C1) Let $\lambda_{\min}(\mathbf{M})$ and $\lambda_{max}(\mathbf{M})$ denote the minimum and the maximum eigenvalues of a semi-positive definite matrix $\mathbf{M}$, respectively. Then we assume

$$b \leq \lambda_{\min}(\frac{1}{n}\mathbf{X}^t\mathbf{X}) \leq \lambda_{max}(\frac{1}{n}\mathbf{X}^t\mathbf{X}) \leq B$$

and

$$d \leq \lambda_{\min}(\mathbf{Q}) \leq \lambda_{max}(\mathbf{Q}) \leq D$$

where b, B, d and D are constants so that $b, B > 0$ and $d, D \geq 0$.

Now, given the data $(\mathbf{y}, \mathbf{X})$, let $\hat{\boldsymbol{\omega}} = (\hat{\omega}_1, ..., \hat{\omega}_p)$ be a vector whose components are all non-negative and can depend on $(\mathbf{y}, \mathbf{X})$. Define

$$\hat{\boldsymbol{\beta}}_{\hat{\boldsymbol{\omega}}}(\lambda_2, \lambda_1^*) = \left\{ \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_2 \boldsymbol{\beta}^t \mathbf{Q} \boldsymbol{\beta} + \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j |\boldsymbol{\beta}_j| \right\}$$

for non-negative parameters λ_1^* and λ_2. If $\hat{\omega}_j = 1$ for all j, we denote $\hat{\boldsymbol{\beta}}_{\hat{\boldsymbol{\omega}}}(\lambda_2, \lambda_1^*)$ by $\hat{\boldsymbol{\beta}}(\lambda_2, \lambda_1^*)$ for convenience. The assumption (C1) assume a reasonably good behavior of both the predictor and the weight matrices (Portnoy, 1984).

Theorem 3.3.1. *If we assume the model (3.1) and Assumption (C1), then*

$$I\!E\left(\|\hat{\boldsymbol{\beta}}_{\hat{\omega}}\left(\lambda_2,\lambda_1^*\right)-\boldsymbol{\beta}^*\|_2^2\right)\leq 4\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^{*2}I\!E\left(\sum_{j=1}^p\hat{\omega}_j^2\right)}{\left(bn+\lambda_2 d\right)^2}$$

In particular, when $\hat{\omega}_j=1$ for all j and if we note λ_1^* by λ_1, we obtain the mean sparsity inequality for the Gril estimator.

$$E\left(\|\hat{\boldsymbol{\beta}}\left(\lambda_2,\lambda_1\right)-\boldsymbol{\beta}^*\|_2^2\right)\leq 4\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^2p}{(bn+\lambda_2 d)^2}$$

The latter risk bounds in Theorem 3.3.1 are non-asymptotic. It implies that, under assumptions (C1)-(C6) defined below, $\hat{\boldsymbol{\beta}}(\lambda_1^*,\lambda_2)$ is a root-(n/p)-consistent estimator (cf. (Fan and Peng, 2004) for SCAD and (Zou and Zhang, 2009) for AdaEnet). So, the construction of the adaptive weight by using the Gril is appropriate.

3.3.2.2 Oracle properties

To establish the oracle properties, we need the same following assumptions used in (Zou and Zhang, 2009).
(C2) $\lim_{n\to\infty}\frac{\max_{i=1,2,\ldots,n}\sum_{j=1}^p x_{ij}^2}{n}=0$.
(C3) $I\!E[|\varepsilon|^{2+\delta}]<\infty$ for some $\delta>0$.
(C4) $\lim_{n\to\infty}\frac{\log(p)}{\log(n)}=\nu$ for some $0\leq\nu<1$.

To construct the adaptive weights $(\hat{\omega})$, we take a fixed $\gamma>\frac{2\nu}{1-\nu}$. In our numerical studies we let $\gamma=[\frac{2\nu}{1-\nu}]+1$ to avoid tuning on γ as in (Zou and Zhang, 2009). Once γ is chosen, we choose the regularization parameters according to the following conditions

(C5)

$$\lim_{n\to\infty}\frac{\lambda_2}{n}=0,\quad \lim_{n\to\infty}\frac{\lambda_2 q_i}{n}=0,\quad \text{for all}\quad i=1,...,p,\quad \lim_{n\to\infty}\frac{\lambda_1}{\sqrt{n}}=0,$$

and

$$\lim_{n\to\infty}\frac{\lambda_1^*}{\sqrt{n}}=0,\quad \lim_{n\to\infty}\frac{\lambda_1^*}{\sqrt{n}}n^{\frac{(1-\nu)(1+\gamma)-1}{2}}=\infty.$$

(C6)

$$\lim_{n\to\infty}\frac{\lambda_2}{\sqrt{n}}\sqrt{\sum_{j\in\mathcal{A}}\boldsymbol{\beta}_j^{*2}}=0,\quad \lim_{n\to\infty}\left(\frac{\lambda_1^*}{n}\right)^{\frac{2\gamma}{1+\gamma}}\frac{p\|\boldsymbol{\beta}^*\|_2^2}{\lambda_1^{*2}}=0,\quad \lim_{n\to\infty}\min\left(\frac{n}{\lambda_1\sqrt{p}},\left(\frac{\sqrt{n}}{\sqrt{p}\lambda_1^*}\right)^{\frac{1}{\gamma}}\right)\left(\min_{j\in\mathcal{A}}|\boldsymbol{\beta}_j^*|\right)\to\infty.$$

Theorem 3.3.2. *Let us write* $\boldsymbol{\beta}^* = (\boldsymbol{\beta}^*_{\mathcal{A}}, 0)$ *and define*

$$\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}} = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}\|_2^2 + \lambda_2 \boldsymbol{\beta}^t \mathbf{Q}_{\mathcal{A}} \boldsymbol{\beta} + \lambda_1^* \sum_{j \in \mathcal{A}} \hat{\boldsymbol{\omega}}_j |\boldsymbol{\beta}_j| \right\}. \tag{3.8}$$

Then, under the assumptions (C1)-(C6) and with probability tending to 1, $(\mathbf{N}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}, 0)$ *is solution to (3.7).*

Theorem 3.3.2 provides an asymptotic characterization of the solution to the adaptive Gril criterion. It demonstrates that the Adaptive Gril estimator is as efficient as an oracle one. Moreover, it is helpful in the proof of Theorem 3.3.3 below.

Theorem 3.3.3. *Under conditions (C1)-(C6), the adaptive Generalized Ridge Lasso (AdaGril) has the oracle property, that is, the estimator* $\hat{\boldsymbol{\beta}}(AdaGril)$ *must satisfy:*

1. *Consistency in selection :* $Pr\left(\{j : \hat{\boldsymbol{\beta}}(AdaGril)_j \neq 0\} = \mathcal{A}\right) \to 1,$
2. *Asymptotic normality :* $\alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \mathbf{N}_{\mathcal{A}}^{-1} \left(\hat{\boldsymbol{\beta}}(AdaGril)_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}\right) \to_d N(0, \sigma^2)$, *where* $\mathbf{X}_{\mathcal{A}}$, $\mathbf{Q}_{\mathcal{A}}$ *and* $\mathbf{N}_{\mathcal{A}}$ *are sub-matrices obtained by extracting the columns of* $\mathbf{X}$, $\mathbf{Q}$ *and* $\mathbf{N}$ *respectively according to the indices in* $\mathcal{A}$, $\Sigma_{\mathcal{A}} = \mathbf{X}^t_{\mathcal{A}}\mathbf{X}_{\mathcal{A}}$ *and* $\boldsymbol{\alpha}$ *is a vector of norm* 1.

Theorem 3.3.3 provides the selection consistency and asymptotic normality of AdaGril when the number of parameters diverges. So, AdaGril estimator enjoys the oracle property of SCAD in high dimensional setting. As a first special case and taking $\mathbf{Q} = \mathbf{I}$, we obtain the asymptotic normality of the Adaptive elastic net:

$$\alpha^t \frac{I + \lambda_2 \Sigma_{\mathcal{A}}^{-1}}{1 + \frac{\lambda_2}{n}} \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(\hat{\boldsymbol{\beta}}(\text{AdaEnet})_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}\right) \to_d \text{N}(0, \sigma^2).$$

Taking $\lambda_2 = 0$, we obtain the asymptotic normality of the Adaptive Lasso as a second special case:

$$\alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(\hat{\boldsymbol{\beta}}(\text{AdaLasso})_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}\right) \to_d \text{N}(0, \sigma^2).$$

3.3.3 Sparsity inequality for AdaGril estimator

Now we establish a sparsity inequality (SI) achieved by $\hat{\boldsymbol{\beta}}(\text{AdaGril})$, that is a bound on L_2 and L_1 error estimation, in terms of the number of non-zero components of the 'true' coefficient vector $\boldsymbol{\beta}^*$. Here the second parameter λ_2 is not free, but it depends on the parameter λ_1^* which is fixed as a function of (n, p, σ). Moreover, the Gril estimator (instead of OLS or Lasso estimator) is used as the initial estimator for the adaptive Gril method. Finally, our result of sparsity inequalities are obtained under a similar assumption on the Gram matrix used by the Lasso (cf. (Bickel et al., 2009)). Let us now establish the

assumptions needed.

Assumption RE. *There is a constant* $\psi > 0$ *such that, for any* $\mathbf{z} \in \mathbb{R}^p$ *that satisfies* $\sum_{j\in\mathcal{A}^c} |\mathbf{z}_j| \leq 4\max\left\{\frac{2}{\eta}, 1\right\} \sum_{j\in\mathcal{A}} |\mathbf{z}_j|$, *we have*

$$\mathbf{z}^t\mathbf{K}\mathbf{z} \geq \psi \sum_{j\in\mathcal{A}} \mathbf{z}_j^2,$$

where $\mathbf{K} = \tilde{\mathbf{X}}^t\tilde{\mathbf{X}}$ *and* $\eta = \min_{j\in\mathcal{A}}(|\boldsymbol{\beta}_j^*|)$.

Let $\hat{\boldsymbol{\beta}}$(Gril) and $\hat{\boldsymbol{\beta}}$(AdaGril) denote the Gril estimator and the adaptive Gril estimator, respectively. Here, the weights of the adaptive estimator are estimated from the Gril estimator $\hat{\boldsymbol{\beta}}$(Gril):

$$\hat{\omega}_j = \max(\frac{1}{|\hat{\boldsymbol{\beta}}(\text{Gril})_j|}, 1) \quad \text{for all} \quad j = 1, ..., p. \tag{3.9}$$

We note that (Zhou et al., 2009) have also considered the same weights in their analysis of the adaptive Lasso for high dimensional regression and Gaussian graphical models. These weights are easier to manipulate in our proof of the next Theorem than the classical weights (3.6).

Theorem 3.3.4. *Given data* $(\mathbf{y}, \mathbf{X})$. *Let* $s = |\mathcal{A}|, \eta = \min_{j\in\mathcal{A}}(|\boldsymbol{\beta}_j^*|)$ *and* $\varphi \in (0, 1)$. *We define our tuning parameter* λ_1^* *and* λ_2 *as follows*

$$\lambda_1^* = 8\sqrt{2}\sigma\sqrt{\frac{\log(p/\varphi)}{n}} \quad \textit{and} \quad \lambda_2 = \frac{\lambda_1^*}{8\|\mathbf{Q}\boldsymbol{\beta}^*\|_\infty}.$$

Now, let $\delta = 8\psi^{-1}\lambda_1^* s$. *and for* $0 < \theta \leq 1$, *consider the set* $\Gamma = \{\max_{j=1,...,p} 2|U_j| \leq \theta\lambda_1\}$ *with* $U_j = n^{-1}\sum_{i=1}^n x_{i,j}\varepsilon_i$. *Therefore, if assumption RE holds and in addition* $\eta \geq 2\delta$, *then with probability greater than* $1 - \varphi$ *on the* Γ *set, we have*

$$\|\mathbf{X}\boldsymbol{\beta}^* - \mathbf{X}\hat{\boldsymbol{\beta}}(\textit{AdaGril})\|_2^2 \;\leq\; 4\psi^{-1}\lambda_1^{*2}\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s,$$

$$\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\textit{AdaGril})\|_1 \;\leq\; 8\psi^{-1}\lambda_1^*\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s.$$

The Restricted Eigenvalue (RE) Assumption is widely used in the literature about the variable selection consistency of ℓ_1-penalized regression methods in high dimension ($p >> n$, see for instance (Bickel et al., 2009), (Zhou et al., 2009) and (Hebiri and van De Geer, 2010). On the one hand, the main difference of our RE assumption with that in (Bickel et al., 2009) are in the matrix $\tilde{K} = \mathbf{X}^t\mathbf{X} + \lambda_2\mathbf{Q}$ and the specified constant $4\max\left\{\frac{2}{\eta}, 1\right\}$ instead of $K_n = n^{-1}\mathbf{X}^t\mathbf{X}$ and an arbitrary constant *cte*, respectively. On the other hand, there is a minor difference with the assumption $B(\Theta)$ used in (Hebiri and van De Geer, 2010). Indeed, the latter authors only need to consider the vectors $\mathbf{z}$ such that

$\sum_{j\notin\Theta}|\mathbf{z}_j| \leq \rho_n\sqrt{\sum_{j\in\Theta}\mathbf{z}_j^2}$, where ρ_n is a scalar which depend of $(s, \lambda_1^*, \lambda_2, \boldsymbol{\beta}^*)$. Moreover, our choice of regularized parameters (λ_1^*, λ_2) are relatively similar to that used by (Hebiri and van De Geer, 2010) in Corollary 1 for the sparsity inequality of Gril estimator (called in that paper Quadratic estimator). We then refer the reader to the latter reference for more discussions about that choice.

3.4 Computation and tuning parameters selection

In this section we propose a modification of the Gril algorithm for finding a solution of the penalized least squares problem (3.7) of the AdaGril. The main idea is to transform the AdaGril problem into an equivalent Gril problem on the augmented data (cf. (Zou and Hastie, 2005)). The main steps of the AdaGril algorithm are as follows:

1. Input: Matrix $\mathbf{X}$ and $\hat{\boldsymbol{\omega}}$.

2. Put $\mathbf{x}_j^{**} = \mathbf{x}_j\hat{\omega}_j$ for $j = 1, \ldots, p$

3. Use the Gril algorithm described in the Section 3.2.2 for computing the AdaGril estimator $\hat{\boldsymbol{\beta}}$(AdaGril).

In practice, it is important to select appropriate tuning parameters $(\lambda_1, \lambda_2, \gamma)$ in order to obtain a good prediction precision. Choosing the tuning parameters can be done via minimizing an estimate of the out-of-sample prediction error. If a validation set is available, this can be estimated directly. Lacking a validation set one can use ten-fold cross validation. Note that we take a fixed $\gamma = [\frac{2\nu}{1-\nu}] + 1$ to avoid tuning on γ. So there are two tuning parameters in the AdaGril, so we need to cross-validate on a two dimensional surface. Typically we first pick a (relatively small) grid values for λ_2, say $(0, 0.01, 0.1, 1, 10, 100)$. Then, for each λ_2, LARS algorithm produces the entire solution path of the AdaGril. The other tuning parameter is selected by tenfold CV. The chosen λ_2 is the one giving the smallest CV error or generalized cross-validation (GCV). However, (Wang et al., 2007) showed that for the SCAD method (cf. (Fan and Li, 2001)), BIC criterion is a better tuning parameter selector than GCV and AIC. In our implementations the parameter γ is fixed for the three adaptive methods (AdaEnet, AdaLasso and AdaGril), while the couple of parameters (λ_1, λ_2) is selected using BIC criterion.

3.5 Numerical study

In this section we consider some simulation experiments to evaluate the finite sample performance of different AdaGril estimators. AdapCnet, AdapWfusion and AdapSlasso methods correspond to adaptive Cnet, adaptive WFusion and adaptive Smooth-Lasso methods, respectively. These three adaptive versions of AdaGril are compared with Lasso, AdaLasso and AdaEnet. We consider the first simulated example used in (Zou and Zhang, 2009). In this example we generate data from the model,

$$y = \mathbf{x}^t\boldsymbol{\beta}^* + \epsilon,$$

where $\boldsymbol{\beta}^*$ is a vector of length p and $\epsilon \sim \mathrm{N}(0, \sigma^2)$, $\sigma \in \{3, 6, 9\}$ and $\mathbf{x} \sim \mathrm{N}_p(\mathbf{0}, \mathbf{R})$, $\mathbf{R}$ is the correlation matrix whose (i, j)th element is $\mathbf{R}_{i,j} = \rho^{|i-j|}$. Results are given for $\rho = 0.5$ and $\rho = 0.75$. This example presents a situation in which the number of parameters depends on the sample size n as follows: $p = p_n = [4n^{1/2}] - 5$ for $n = 100, 200, 1000$. The true parameter is

$$\boldsymbol{\beta} = (1, 2, ..., q-1, q, \underbrace{0, ..., 0}_{p-3q}, \underbrace{3, ..., 3}_{q}, -1, -2, ..., -q+1, -q)^t,$$

where $q = [p_n/9]$. For this choice of n and p, we have $\nu = \frac{1}{2}$, so we used $\gamma = 3$ for calculating the adaptive weights for all adaptive methods.

Table 3.1, Table 3.2 and Table 3.3 summarize the performance of different adaptive and non adaptive methods in terms of prediction accuracy, estimation error and variable selection, respectively. Several observations can be made from these tables.

1. The adaptive methods outperform the non adaptive ones in terms of prediction and estimation accuracies, except in two small sample setting cases (i.e. $n = 100$ and 200 for $\sigma = 9$).
2. In small sample settings, AdaSlasso is the winner in term of prediction accuracy followed by AdaCnet or Adalasso (except in $[n = 100, \sigma = 9, \rho = 0.5]$ where Enet is the winner). However, for large sample, AdaCnet is the best in term of prediction accuracy followed by Adalasso (except in one case of $[\sigma = 9, \rho = 0.75]$).
3. The AdaSlasso (or Slasso for $n = 100$ and $\sigma = 6 - 9$, Table 2) seems to dominate its competitors in term of prediction error (i.e. $\mathrm{MSE}_{\boldsymbol{\beta}}$) in small sample settings ($n = 100-200$). It is followed by AdaCnet or Cnet. However, AdaCnet is by far better than all other method in large sample size $n = 1000$ (except the case $\sigma = 9$ and $\rho = 0.75$).
4. When increasing the noise level σ, the methods behave in the same way by increasing substantially their prediction and error accuracies, and regardless of the sample size n and the correlation coefficient ρ.
5. From Table 3, it can be seen that the performance in term of correct selection of all methods increases largely when the sample size increases, and whatever the value of noise level. While, their performance in term of incorrect selection increase slightly when the noise level increases, and especially in small sample settings. Moreover, the performance of the adaptive methods, in small settings, is relatively similar and is slightly better than those of the non adaptive ones (the difference between theme is about $3 - 5$ percent), and whatever the values of σ and ρ. However, in large sample setting, all methods behave in the same way by increasing largely their performance of correct selection of the relevant variables with a little advantage ($3 - 5$ percent) to AdaCnet and Cnet.

Finally, we can conclude that in this example, the adaptive methods perform better than the non-adaptive ones in terms of variable selection and prediction accuracies, and whatever the values of n, σ and ρ. Moreover, the AdaCnet and AdaSlasso outperform largely AdaEnet in quasi different situations. We have also considered a second example (Example 1 in (Zou and Zhang, 2009), results not reported here) where the structure of the parameter vector is smooth with small difference between successive coefficients. The

	$\sigma = 3$		$\sigma = 6$		$\sigma = 9$	
	$\rho = 0.5$	$\rho = 0.75$	$\rho = 0.5$	$\rho = 0.75$	$\rho = 0.5$	$\rho = 0.75$
n = 100						
Lasso	2.51	2.30	10.52	10.47	24.50	19.53
AdaLasso	1.74	1.73	8.38	9.61	22.95	19.26
Enet	2.32	2.14	9.52	8.88	19.89	17.52
AdaEnet	1.89	1.83	8.90	8.67	21.71	18.58
Slasso	2.42	2.21	9.70	9.21	20.14	16.75
AdaSlasso	1.60	1.44	7.67	8.30	20.25	16.76
Cnet	2.43	2.16	9.67	9.03	19.89	17.42
AdaCnet	1.81	1.72	8.64	8.57	21.24	17.92
Wfusion	2.35	2.25	9.52	8.67	19.93	17.55
AdaWfusion	1.89	1.94	8.80	8.77	21.29	18.30
n = 200						
Lasso	1.74	1.62	7.91	7.26	16.31	15.02
AdaLasso	1.04	0.99	5.06	5.54	13.04	14.80
Enet	1.62	1.54	7.23	6.67	14.63	13.92
AdaEnet	1.17	1.05	5.35	5.53	13.02	13.91
Slasso	1.89	1.77	7.70	6.63	14.82	13.67
AdaSlasso	0.99	0.94	4.71	4.77	11.40	11.94
Cnet	1.93	1.80	7.75	6.91	15.09	13.92
AdaCnet	1.08	0.98	5.12	5.41	12.47	13.54
Wfusion	1.63	1.54	7.23	6.68	14.63	13.92
AdaWfusion	1.10	1.01	5.25	5.43	12.76	13.74
n = 1000						
Lasso	0.84	0.72	3.50	2.79	7.62	6.55
AdaLasso	0.59	0.50	2.28	2.21	5.30	4.85
Enet	0.82	0.68	3.33	2.72	7.27	6.39
AdaEnet	1.03	1.29	2.80	3.08	6.12	5.82
Slasso	1.32	1.35	3.89	3.37	8.04	7.10
AdaSlasso	0.73	0.80	2.37	2.29	5.19	4.66
Cnet	3.32	2.71	6.83	5.54	12.0	9.81
AdaCnet	0.41	0.40	1.92	1.89	4.57	4.92
Wfusion	0.82	0.68	3.33	2.72	7.27	6.39
AdaWfusion	0.60	0.53	2.39	2.24	5.55	4.93

Table 3.1: Median mean-squared errors for $\rho \in \{0.5, 0.75\}$, $\sigma \in \{3, 6, 9\}$ and $n \in \{100, 200, 1000\}$ based on 100 replications.

	$\sigma = 3$		$\sigma = 6$		$\sigma = 9$	
	$\rho = 0.5$	$\rho = 0.75$	$\rho = 0.5$	$\rho = 0.75$	$\rho = 0.5$	$\rho = 0.75$
n = 100						
Lasso	3.30	6.51	12.66	22.70	26.41	32.97
AdaLasso	2.64	5.57	13.65	32.91	35.88	52.01
Enet	3.12	6.02	12.08	20.51	23.92	33.03
AdaEnet	2.75	5.44	14.35	28.79	34.10	52.08
Slasso	3.24	6.21	11.82	20.13	23.23	30.14
AdaSlasso	2.30	4.14	11.83	26.22	30.74	45.88
Cnet	3.20	6.03	12.10	20.62	23.61	31.72
AdaCnet	2.62	5.15	13.87	28.43	33.15	50.33
Wfusion	3.18	6.46	12.08	20.52	23.93	33.01
AdaWfusion	2.75	5.92	14.12	28.38	33.49	51.26
n = 200						
Lasso	2.27	4.53	9.40	19.00	20.27	35.36
AdaLasso	1.54	3.05	7.56	19.17	21.95	53.93
Enet	2.15	4.29	8.79	17.58	18.70	33.77
AdaEnet	1.65	3.06	7.67	17.97	20.29	47.83
Slasso	2.56	5.18	9.12	16.61	18.25	31.00
AdaSlasso	1.46	2.77	6.61	14.44	17.47	38.40
Cnet	2.45	4.89	9.14	17.57	18.81	32.66
AdaCnet	1.58	2.90	7.30	17.59	19.76	46.73
Wfusion	2.16	4.29	8.79	17.59	18.70	33.77
AdaWfusion	1.60	2.99	7.52	17.67	19.92	47.07
n = 1000						
Lasso	0.95	1.95	3.88	7.77	8.39	17.66
AdaLasso	0.83	1.37	3.25	6.05	7.34	13.63
Enet	0.94	1.88	3.77	7.62	8.16	17.35
AdaEnet	1.01	1.59	3.58	6.38	7.93	14.06
Slasso	1.81	4.94	4.69	10.35	9.10	19.32
AdaSlasso	1.23	3.03	3.46	6.52	7.09	12.86
Cnet	2.74	6.04	6.10	12.69	11.14	22.12
AdaCnet	0.64	1.29	2.91	5.69	6.64	16.02
Wfusion	0.94	1.88	3.77	7.62	8.16	17.36
AdaWfusion	0.84	1.40	3.37	6.09	7.62	13.65

Table 3.2: $\text{MSE}_\beta = \|\hat{\beta} - \beta^*\|_2^2$ errors for $\rho \in \{0.5, 0.75\}$, $\sigma \in \{3, 6, 9\}$ and $n \in \{100, 200, 1000\}$ based on 100 replications.

	$\sigma = 3$		$\sigma = 6$		$\sigma = 9$	
	C	IC	C	IC	C	IC
n = 100						
Lasso	22.23	0.11	23.10	1.16	24.35	3.38
AdaLasso	25.06	0.53	25.20	2.03	25.39	4.45
Enet	20.83	0.09	21.45	0.98	22.53	2.47
AdaEnet	24.37	0.31	24.47	1.72	24.67	3.65
Slasso	21.45	0.06	21.90	0.93	22.99	2.48
AdaSlasso	24.76	0.29	24.66	1.57	24.79	3.55
Cnet	21.04	0.08	21.51	0.96	22.67	2.47
AdaCnet	24.49	0.31	24.50	1.72	24.75	3.67
Wfusion	20.64	0.09	21.45	0.98	22.54	2.47
AdaWfusion	24.13	0.31	24.46	1.71	24.68	3.65
n = 200						
Lasso	31.64	0.02	32.40	0.43	32.47	1.12
AdaLasso	35.12	0.07	35.19	1.23	35.39	2.59
Enet	30.77	0.02	31.21	0.37	31.08	0.87
AdaEnet	34.67	0.05	34.58	1.00	34.68	1.98
Slasso	31.26	0.02	31.75	0.34	31.37	0.85
AdaSlasso	35.08	0.05	34.75	0.94	34.83	1.87
Cnet	31.67	0.02	31.66	0.37	31.27	0.86
AdaCnet	34.80	0.05	34.65	0.97	34.78	2.02
Wfusion	30.87	0.02	31.21	0.37	31.08	0.87
AdaWfusion	34.67	0.05	34.58	1.00	34.68	1.98
n = 1000						
Lasso	76.16	0.00	76.63	0.00	76.54	0.05
AdaLasso	76.16	0.00	76.63	0.00	76.54	0.05
Enet	75.76	0.00	75.67	0.00	75.49	0.04
AdaEnet	75.77	0.00	75.67	0.00	75.49	0.04
Slasso	77.13	0.00	76.09	0.00	76.15	0.05
AdaSlasso	77.13	0.00	76.09	0.00	78.76	0.07
Cnet	80.64	0.00	79.15	0.00	78.76	0.07
AdaCnet	80.64	0.00	79.15	0.00	78.76	0.07
Wfusion	75.76	0.00	75.67	0.00	75.49	0.04
AdaWfusion	75.76	0.00	75.67	0.00	75.49	0.04

Table 3.3: The median number of C and IC $\rho = 0.5$, $\sigma \in \{3, 6, 9\}$ and $n \in \{100, 200, 1000\}$ based on 100 replications.

results steal relatively similar to those obtained in example 1, but with some advantage to AdaSlasso in prediction accuracy and Slasso in prediction error.

So, when the structure of the parameter vector is smooth, Slasso and AdaSlasso will have a clear advantage than its competitors. When this structure is not smooth and the coefficients have different signs, then Cnet and AdaCnet seem to work well in this setting. On the other hand, Enet and AdaEnet will give good results in extreme correlation case ($\rho \approx 1$), while its competitors (Cnet and Wfusion) give good results when the correlation is moderate. When the correlation is small, Lasso or AdaLasso can do better.

3.6 Discussion

In this paper we propose AdaGril for variable selection with a diverging number of parameters in the presence of highly correlated variables. AdaGril is a generalization of AdaEnet by replacing the identity matrix in the L_2 norm penalty by any positive semi-definite matrix Q. Many possible choices of Q are in the literature. We show that under some conditions on the eigenvalues of Q we can extend results on variable selection consistency and asymptotic normality of the AdaEnet to the AdaGril. Moreover, we show that AdaGril estimator achieves a Sparsity Inequality, i. e., a bound in terms of the number of non-zero components of the 'true' regression coefficient. This bound is obtained under a similar weak Restricted Eigenvalue (RE) condition used for Lasso. Simulations studies show that some particular cases of AdaGril outperform its competitors. Simulated examples suggest that AdaGril methods improve both the AdaLasso and AdaEnet. The extension of the AdaGril to generalized linear models (McCullagh and Nelder, 1989) will be subject to future work.

3.A Proofs

3.A.1 Proof of Lemma 3.3.1

Proof. Let $\hat{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}(\lambda_1^*, \lambda_2) = \arg\min_{\boldsymbol{\beta}}\{L(\lambda_1^*, \lambda_2, \boldsymbol{\beta})\}$, where

$$L(\lambda_1^*, \lambda_2, \boldsymbol{\beta}) = \mathbf{N}\left\{\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_2\boldsymbol{\beta}^t\mathbf{Q}\boldsymbol{\beta} + \lambda_1^*\sum_{j=1}^{p}\hat{\omega}_j|\boldsymbol{\beta}_j|\right\},$$

and $\mathbf{Q}$ is defined by (3.4). If $\hat{\boldsymbol{\beta}}_i\hat{\boldsymbol{\beta}}_j > 0$, then both $\hat{\boldsymbol{\beta}}_i$ and $\hat{\boldsymbol{\beta}}_j$ are non-zero, and we have $\text{sign}(\hat{\boldsymbol{\beta}}_i) = \text{sign}(\hat{\boldsymbol{\beta}}_j)$. Then $\hat{\boldsymbol{\beta}}$ must satisfies

$$\frac{\partial L(\lambda_1^*, \lambda_2, \boldsymbol{\beta})}{\partial \boldsymbol{\beta}}|_{\boldsymbol{\beta}=\hat{\boldsymbol{\beta}}} = \mathbf{0}. \tag{3.10}$$

Hence we have

$$-2\mathbf{x}_i^t\{\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}\} + \lambda_1^*\hat{w}_i\text{sign}\{\hat{\boldsymbol{\beta}}_i\} + 2\lambda_2\sum_{k=1}^{p}q_{ik}\hat{\boldsymbol{\beta}}_k = 0, \tag{3.11}$$

and

$$-2\mathbf{x}_j^t\{\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}\}+\lambda_1^*\hat{w}_j\text{sign}\{\hat{\boldsymbol{\beta}}_j\}+2\lambda_2\sum_{k=1}^{p}q_{jk}\hat{\boldsymbol{\beta}}_k=0, \tag{3.12}$$

Subtracting equation (3.11) from (3.12) gives

$$2(\mathbf{x}_j^t-\mathbf{x}_i^t)\{\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}\}-\lambda_1^*(\hat{w}_j-\hat{w}_i)\text{sign}\{\hat{\boldsymbol{\beta}}_j\}-2\lambda_2\sum_{k=1}^{p}(q_{jk}-q_{ik})\hat{\boldsymbol{\beta}}_k=0,$$

which is equivalent to

$$\lambda_2\sum_{k=1}^{p}(q_{jk}-q_{ik})\hat{\boldsymbol{\beta}}_k=(\mathbf{x}_i^t-\mathbf{x}_j^t)\hat{\mathbf{r}}-\frac{\lambda_1^*}{2}(\hat{w}_j-\hat{w}_i)\text{sign}\{\hat{\boldsymbol{\beta}}_j\} \tag{3.13}$$

where $\hat{\mathbf{r}}=\mathbf{y}-\mathbf{X}\hat{\boldsymbol{\beta}}$ is the residual vector. Since $\mathbf{X}$ is standardized, then $\|\mathbf{x}_i-\mathbf{x}_j\|_2^2=2(1-\rho_{ij})$. Because $\hat{\boldsymbol{\beta}}$ is the minimizer we must have $L\{\lambda_1^*,\lambda_2,\hat{\boldsymbol{\beta}}(\lambda_1^*,\lambda_2)\}\leq L\{\lambda_1^*,\lambda_2,\boldsymbol{\beta}=\mathbf{0}\}$, i.e.

$$\|\hat{\mathbf{r}}\|^2+\lambda_2\hat{\boldsymbol{\beta}}^t\mathbf{Q}\hat{\boldsymbol{\beta}}+\lambda_1^*\sum_{k=1}^{p}\hat{w}_k|\hat{\boldsymbol{\beta}}_k|\leq\|\mathbf{y}\|_2^2. \tag{3.14}$$

So $\|\hat{\mathbf{r}}(\lambda_1^*,\lambda_2)\|_2\leq\|\mathbf{y}\|_2$.
Now, we apply the mean value Theorem, as in ((Ghosh, 2011), Proof of Theorem 3.3), to the function $g(x)=x^{-\gamma}$, we have $|g(x)-g(y)|=|g'(c)||x-y|$ for some $c\in[\min(x,y),\max(x,y)]$. Hence we obtain

$$\begin{aligned}
|\hat{w}_i-\hat{w}_j| &= ||\hat{\boldsymbol{\beta}}_i^{\circ}|^{-\gamma}-|\hat{\boldsymbol{\beta}}_j^{\circ}|^{-\gamma}| \\
&= \frac{\gamma}{c^{\gamma+1}}||\hat{\boldsymbol{\beta}}_i^{\circ}|-|\hat{\boldsymbol{\beta}}_j^{\circ}|| \quad \text{where} \quad c\in[\min(|\hat{\boldsymbol{\beta}}_i^{\circ}|,|\hat{\boldsymbol{\beta}}_j^{\circ}|),\max(|\hat{\boldsymbol{\beta}}_i^{\circ}|,|\hat{\boldsymbol{\beta}}_j^{\circ}|)] \\
&\leq \frac{\gamma}{\min(|\hat{\boldsymbol{\beta}}_i^{\circ}|,|\hat{\boldsymbol{\beta}}_j^{\circ}|)^{\gamma+1}}|\hat{\boldsymbol{\beta}}_i^{\circ}-\hat{\boldsymbol{\beta}}_j^{\circ}|.
\end{aligned}$$

Then the equation (3.13) implies that

$$|\sum_{k=1}^{p}(q_{ik}-q_{jk})\hat{\boldsymbol{\beta}}_k| \leq \frac{1}{\lambda_2}\|\hat{\mathbf{r}}(\lambda_1^*,\lambda_2)\|\|\mathbf{x}_1-\mathbf{x}_2\|+\frac{\gamma\lambda_1^*}{\min(|\hat{\boldsymbol{\beta}}_i^{\circ}|,|\hat{\boldsymbol{\beta}}_j^{\circ}|)^{\gamma+1}}|\hat{\boldsymbol{\beta}}_i^{\circ}-\hat{\boldsymbol{\beta}}_j^{\circ}| \tag{3.15}$$

dividing by $\|\mathbf{y}\|_2$, we obtain

$$\frac{1}{\|\mathbf{y}\|_2}|\sum_{k=1}^{p}(q_{ik}-q_{jk})\hat{\boldsymbol{\beta}}_k| \leq \frac{\sqrt{2(1-\rho_{ij})}}{\lambda_2}+\frac{\gamma\lambda_1^*}{\|\mathbf{y}\|_2\min(|\hat{\boldsymbol{\beta}}_i^{\circ}|,|\hat{\boldsymbol{\beta}}_j^{\circ}|)^{\gamma+1}}|\hat{\boldsymbol{\beta}}_i^{\circ}-\hat{\boldsymbol{\beta}}_j^{\circ}| \tag{3.16}$$

On the other hand, we have:

$$q_{ii} = -\sum_{s\neq i} \frac{q_{is}}{\rho_{is}} \qquad \text{and} \qquad q_{jj} = -\sum_{s\neq j} \frac{q_{js}}{\rho_{js}}. \tag{3.17}$$

Then

$$\sum_{k=1}^{p}(q_{ik} - q_{jk})\hat{\boldsymbol{\beta}}_k = \frac{-2}{1-\rho_{ij}}[\hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\beta}}_i] + 2SN \tag{3.18}$$

where $SN = \sum_{k\neq i,j} \frac{1}{1-\rho_{ki}^2}[\hat{\boldsymbol{\beta}}_i - \rho_{ki}\hat{\boldsymbol{\beta}}_k] + \frac{1}{1-\rho_{kj}^2}[\rho_{kj}\hat{\boldsymbol{\beta}}_k - \hat{\boldsymbol{\beta}}_j]$. If $\rho_{ki} = \rho_{kj} = \rho, \quad \forall k = 1, ..., p$, then $SN = \frac{p-2}{1-\rho^2}(\hat{\boldsymbol{\beta}}_i - \hat{\boldsymbol{\beta}}_j)$. So using (3.19) we have:

$$\frac{1}{\|\mathbf{y}\|_2}\left|\hat{\boldsymbol{\beta}}_j - \hat{\boldsymbol{\beta}}_i\right| \;\leq\; \frac{1-\rho^2}{2(p+\rho-1)\lambda_2}\left[\sqrt{2(1-\rho)} + \frac{\gamma\lambda_1^*}{\|\mathbf{y}\|_2 \min(|\hat{\boldsymbol{\beta}}_i^{\circ}|, |\hat{\boldsymbol{\beta}}_j^{\circ}|)^{\gamma+1}}|\hat{\boldsymbol{\beta}}_i^{\circ} - \hat{\boldsymbol{\beta}}_j^{\circ}|\right]$$

This completes the proof. □

3.A.2 Proof of Theorem 3.3.1

Proof. The proof is similar to that of Theorem 3.1 in (Zou and Zhang, 2009). We must only take account of the second inequality in the assumption (C1) in different inequalities. We briefly skech here the steps of the proof. The generalized ridge solution is defined by

$$\hat{\beta}(\lambda_2, 0) = \arg\min_{\beta} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \lambda_2\beta^T\mathbf{Q}\beta,$$

and $\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)$ is the solution of

$$\hat{\boldsymbol{\beta}}_{\hat{\boldsymbol{\omega}}}(\lambda_2, \lambda_1) = \left\{\arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_2\boldsymbol{\beta}^t\mathbf{Q}\boldsymbol{\beta} + \lambda_1\sum_{j=1}^{p}\hat{\omega}_j|\boldsymbol{\beta}_j|\right\}.$$

Then, we have

$$\begin{aligned} & \|\mathbf{y} - \mathbf{X}\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)\|_2^2 + \lambda_2\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)^t\mathbf{Q}\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1) \\ \geq\; & \|\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_2, 0)\|_2^2 + \lambda_2\hat{\beta}(\lambda_2, 0)^t\mathbf{Q}\hat{\beta}(\lambda_2, 0), \end{aligned}$$

and

$$\begin{aligned} & \|\mathbf{y} - \mathbf{X}\hat{\beta}(\lambda_2, 0)\|_2^2 + \lambda_2\hat{\beta}(\lambda_2, 0)^t\mathbf{Q}\hat{\beta}(\lambda_2, 0) + \lambda_1\sum_{j=1}^{p}\hat{\omega}_j|\hat{\beta}(\lambda_2, 0)_j| \\ \geq\; & \|\mathbf{y} - \mathbf{X}\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)\|_2^2 + \lambda_2\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)^t\mathbf{Q}\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1) + \lambda_1\sum_{j=1}^{p}\hat{\omega}_j|\hat{\beta}_{\hat{\omega}}(\lambda_2, \lambda_1)_j|. \end{aligned}$$

Moreover, it is easy to see that

$$\begin{aligned} & \mathrm{E}\left(\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\beta^*\|_2^2\right) \\ \leq\ & 2\mathrm{E}\left(\|\hat{\beta}(\lambda_2,0)-\beta^*\|_2^2\right)+2\mathrm{E}\left(\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|_2^2\right). \end{aligned}$$

From the above first two inequalities and from simple algebra, we deduce

$$\lambda_1\sum_{j=1}^{p}\hat{\omega}_j(|\hat{\beta}(\lambda_2,0)_j|-|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)_j|) \ \geq\ (\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0))^t(\mathbf{X}^t\mathbf{X}+\lambda_2\mathbf{Q})(\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)), \tag{3.19}$$

and

$$\sum_{j=1}^{p}\hat{\omega}_j(|\hat{\beta}(\lambda_2,0)_j|-|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)_j|) \ \leq\ \sqrt{\sum_{j=1}^{p}\hat{\omega}_j^2}\|\hat{\beta}(\lambda_2,0)-\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)\|_2.$$

It is obvious that

$$\begin{aligned} bn+\lambda_2 d &\leq \lambda_{min}(\mathbf{X^T X})+\lambda_{\mathbf{2}}\lambda_{\mathbf{min}}(\mathbf{Q}) \\ &\leq \lambda_{min}(\mathbf{X^T X}+\lambda_{\mathbf{2}}\mathbf{Q}). \end{aligned}$$

Therefore, we have

$$\begin{aligned} (bn+\lambda_2 d)\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|_2^2 &\leq \lambda_{min}(\mathbf{X^T X}+\lambda_{\mathbf{2}}\mathbf{Q})\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|_2^2 \\ &\leq \|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|^2_{\mathbf{X^T X}+\lambda_{\mathbf{2}}\mathbf{Q}} \\ &\leq \lambda_1\sqrt{\sum_{j=1}^{p}\hat{\omega}_j^2}\|\hat{\beta}(\lambda_2,0)-\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)\|_2, \end{aligned}$$

which leads to the first bound

$$\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|_2 \ \leq\ \frac{\lambda_1\sqrt{\sum_{j=1}^{p}\hat{\omega}_j^2}}{bn+\lambda_2 d}. \tag{3.20}$$

On the other hand, note that

$$\hat{\beta}(\lambda_2,0)-\beta^*=-\lambda_2(\mathbf{X}^T\mathbf{X}+\lambda_2\mathbf{Q})^{-1}\mathbf{Q}\beta^*+(\mathbf{X}^T\mathbf{X}+\lambda_2\mathbf{Q})^{-1}\mathbf{X}^T\varepsilon.$$

Now, using the second inequality of the assumption (C1) and the fact

$$\lambda_{min}(\mathbf{X^T X})+\lambda_{\mathbf{2}}\lambda_{\mathbf{min}}(\mathbf{Q})\leq\lambda_{min}(\mathbf{X^T X}+\lambda_{\mathbf{2}}\mathbf{Q}),$$

we can show, following the same arguments used by (Zou and Zhang, 2009) in equation (6.4), that

$$\begin{aligned} & \mathrm{E}\left(\|\hat{\beta}(\lambda_2,0)-\beta^*\|_2^2\right) \\ \leq\ & 2\lambda_2^2\|(\mathbf{X}^t\mathbf{X}+\lambda_2\mathbf{Q})^{-1}\mathbf{Q}\beta^*\|_2^2+\mathrm{E}\left(\|(\mathbf{X}^T\mathbf{X}+\lambda_2\mathbf{Q})^{-1}\mathbf{X}^t\varepsilon\|_2^2\right) \\ \leq\ & 2(\lambda_{min}(\mathbf{X^tX})+\lambda_{\mathbf{2}}\lambda_{\mathbf{min}}(\mathbf{Q}))(\lambda_2^2D^2\|\beta^*\|_2^2+p\lambda_{max}(\mathbf{X}^t\mathbf{X})\sigma^2). \end{aligned} \tag{3.21}$$

Finally, from (3.20) and (3.21) and using assumption (C1), we obtain that

$$\begin{aligned} & \mathrm{E}\left(\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\beta^*\|_2^2\right) \\ \leq\ & 2\mathrm{E}\left(\|\hat{\beta}(\lambda_2,0)-\beta^*\|_2^2\right)+2\mathrm{E}\left(\|\hat{\beta}_{\hat{\omega}}(\lambda_2,\lambda_1)-\hat{\beta}(\lambda_2,0)\|_2^2\right) \\ \leq\ & \frac{4\lambda_2^2D^2\|\beta^*\|_2^2+4p\lambda_{max}(\mathbf{X}^T\mathbf{X})\sigma^2+\lambda_1^2\mathrm{E}\left[\sum_{j=1}^p\hat{\omega}_j^2\right]}{(\lambda_{min}(\mathbf{X^TX})+\lambda_{\mathbf{2}}\lambda_{\mathbf{min}}(\mathbf{Q}))^2} \\ \leq\ & 4\frac{\lambda_2^2D^2\|\beta^*\|_2^2+Bpn\sigma^2+\lambda_1^2\mathrm{E}\left[\sum_{j=1}^p\hat{\omega}_j^2\right]}{(bn+d\lambda_2)^2}. \end{aligned} \tag{3.22}$$

□

3.A.3 Proof of Theorem 3.3.2

Proof. To prove this Theorem, we must show, as in (Zou and Zhang, 2009), that $(\mathbf{N}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}},0)$ satisfies the Karush-Kuhn-Tucker condition of (3.7) with probability tending to 1. By the definition of $\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}$, it suffices to show

$$\Pr(\forall j\in\mathcal{A}^c,|-2\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})+2\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k|\leq\lambda_1^*\hat{\omega}_j)\to 1,$$

or equivalently

$$\Pr(\exists j\in\mathcal{A}^c,|-2\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})+2\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k|>\lambda_1^*\hat{\omega}_j)\to 0.$$

The term $2\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k$ does not appear in the proof of the same Theorem 4.2 for adaptive elastic net in (Zou and Zhang, 2009). So, taking into account this difference, some modifications of the proof of (Zou and Zhang, 2009)'s Theorem 4.2 are necessary. Let $\eta=\min_{k\in\mathcal{A}}(|\boldsymbol{\beta}^*_k|)$ and $\hat{\eta}=\min_{k\in\mathcal{A}}(|\hat{\boldsymbol{\beta}}(\mathrm{Gril})^*_k|)$. We note that

$$\begin{aligned} & \Pr(\exists j\in\mathcal{A}^c,|-2\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})+2\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k|>\lambda_1^*\hat{\omega}_j) \\ \leq\ & \sum_{j\in\mathcal{A}^c}\Pr(|-2\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})+2\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k|>\lambda_1^*\hat{\omega}_j,\hat{\eta}>\eta/2)+\Pr(\hat{\eta}\leq\eta/2). \end{aligned}$$

Since

$$|\hat{\boldsymbol{\beta}}(\text{Gril})_j| \;\geq\; \eta - \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_2 \;\text{ for all }\; j \in \mathcal{A}, \tag{3.23}$$

we have

$$\hat{\eta} \;\geq\; \eta - \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_2. \tag{3.24}$$

If $\hat{\eta} \leq \eta/2$, we have $\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_2 \geq \eta/2$ and so

$$\Pr(\hat{\eta} \leq \eta/2) \leq \Pr(\|\hat{\boldsymbol{\beta}}(\text{Gril}) - \boldsymbol{\beta}^*\|_2 \geq \eta/2) \leq \frac{I\!E(\|\hat{\boldsymbol{\beta}}(\text{Gril}) - \boldsymbol{\beta}^*\|_2^2)}{\eta^2/4}.$$

Then from Theorem 3.3.1 we have

$$\Pr(\hat{\eta} \leq \eta/2) \leq 16\frac{\lambda_2^2 D^2 \|\boldsymbol{\beta}^*\|_2^2 + Bpn\sigma^2 + p\lambda_1^2}{(bn + \lambda_2 d)^2 \eta^2}. \tag{3.25}$$

Moreover, let $M = (\lambda_1^*/n)^{1/(1+\gamma)}$ and using similar arguments as in the proof of equation (6.8) in (Zou and Zhang, 2009), we have

$$\begin{aligned}
&\sum_{j\in\mathcal{A}^c} \Pr(|-2\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*) + 2\sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*| > \lambda_1^*\hat{\omega}_j, \hat{\eta} > \eta/2) \\
\leq\; &\sum_{j\in\mathcal{A}^c} \Pr(|-2\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*) + 2\sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*| > \lambda_1^*\hat{\omega}_j, \hat{\eta} > \eta/2, |\hat{\boldsymbol{\beta}}(\text{Gril})_j| \leq M) \\
&+ \sum_{j\in\mathcal{A}^c} \Pr(|\hat{\boldsymbol{\beta}}(\text{Gril})_j| > M) \\
\leq\; &\frac{4M^{2\gamma}}{\lambda_1^{*2}} I\!E\left(\sum_{j\in\mathcal{A}^c} |-\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*) + \sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*|^2 I(\hat{\eta} > \eta/2)\right) \\
&+ \frac{I\!E\left(\|\hat{\boldsymbol{\beta}}(\text{Gril}) - \boldsymbol{\beta}^*\|_2^2\right)}{M^2} \\
\leq\; &\frac{4M^{2\gamma}}{\lambda_1^{*2}} I\!E\left(\sum_{j\in\mathcal{A}^c} |-\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*) + \sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*|^2 I(\hat{\eta} > \eta/2)\right) \\
&+ 4\frac{\lambda_2^2 D^2 \|\boldsymbol{\beta}^*\|_2^2 + Bpn\sigma^2 + p\lambda_1^2}{(bn + \lambda_2 d)^2 M^2}.
\end{aligned} \tag{3.26}$$

We have used the result of Theorem 3.3.1 for the second term of the last inequality. On the other hand, it is easy to show that

$$\sum_{j\in\mathcal{A}^c} |-\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*) + \sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*|^2 \;\leq\; 2\sum_{j\in\mathcal{A}^c} |\mathbf{X}_j^t(\mathbf{y} - \mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*)|^2 + 2\sum_{j\in\mathcal{A}^c}\left(\sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}_k^*\right)^2.$$

From (Zou and Zhang, 2009), in page 16, we have

$$\sum_{j\in\mathcal{A}^c}|\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})|^2 \;\leq\; 2Bn.Bn\|\boldsymbol{\beta}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2+2Bn\|\varepsilon\|_2^2,$$

which leads to the following inequality

$$\begin{aligned}&I\!E\left(\sum_{j\in\mathcal{A}^c}|\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})|^2 I(\hat{\eta}>\eta/2)\right)\\ \leq\;& 2B^2n^2I\!E\left(\|\boldsymbol{\beta}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2 I(\hat{\eta}>\eta/2)\right)+2Bnp\sigma^2.\end{aligned} \tag{3.27}$$

We now bound $I\!E\left(\|\boldsymbol{\beta}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2 I(\hat{\eta}>\eta/2)\right)$. Let

$$\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2,0)=\arg\min_{\boldsymbol{\beta}}\left\{\|\mathbf{y}-\mathbf{X}_{\mathcal{A}}\boldsymbol{\beta}\|_2^2+\lambda_2\boldsymbol{\beta}^t\mathbf{Q}_{\mathcal{A}}\boldsymbol{\beta}\right\}.$$

Then, as in (Zou and Zhang, 2009), by using the same arguments for deriving (3.20), we have

$$\|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2,0)\|_2\leq\frac{\lambda_1^*.\max_{j\in\mathcal{A}}\hat{\omega}_j\sqrt{|\mathcal{A}|}}{\lambda_{\min}(\mathbf{X}^t_{\mathcal{A}}\mathbf{X}_{\mathcal{A}})+\lambda_2 d}\leq\frac{\lambda_1^*\hat{\eta}^{-\gamma}\sqrt{p}}{bn+\lambda_2 d}. \tag{3.28}$$

Following the similar arguments used in the proof of Theorem 3.3.1, we obtain

$$\begin{aligned}&I\!E\left(\|\boldsymbol{\beta}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2 I(\hat{\eta}>\eta/2)\right)\\ \leq\;& 4\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^{*2}(\eta/2)^{-2\gamma}p}{(bn+\lambda_2 d)^2}.\end{aligned}$$

Now, we bound the second term $\sum_{j\in\mathcal{A}^c}\left(\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k\right)^2$. In fact, we have

$$\begin{aligned}\sum_{j\in\mathcal{A}^c}\left(\sum_{k\in\mathcal{A}}q_{jk}\tilde{\boldsymbol{\beta}}^*_k\right)^2 &\leq \sum_{j\in\mathcal{A}^c}\left[\left(\sum_{k\in\mathcal{A}}q_{jk}^2\right)\left(\sum_{k\in\mathcal{A}}\tilde{\boldsymbol{\beta}}^{*2}_k\right)\right]\text{ (Cauchy-Schwarz inequality)}\\ &\leq \sum_{j\in\mathcal{A}^c}\left[\left(\sum_{k\in\mathcal{A}}q_{jk}^2\right)\|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2\right]\\ &\leq \|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2\sum_{j=1}^{p}\sum_{k=1}^{p}q_{jk}^2\\ &= \|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2.\|\mathbf{Q}\|_F^2\quad(\|.\|_F\text{ is the Frobenius norm})\\ &\leq p\|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2.\|\mathbf{Q}\|_2^2\quad(\|.\|_2\text{ is the spectral norm})\\ &\leq pD^2\|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2\\ &\leq 2pD^2\|\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}-\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2+2pD^2\|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2.\end{aligned} \tag{3.29}$$

It leads to the following inequality

$$\begin{aligned} & I\!E\left(\sum_{j\in\mathcal{A}^c}\left(\sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}^*_k\right)^2 I(\hat{\eta}>\eta/2)\right) \\ \leq\ & 2pD^2 I\!E\left(\|\boldsymbol{\beta}^*_{\mathcal{A}}-\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}\|_2^2 I(\hat{\eta}>\eta/2)\right)+2pD^2\|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2. \end{aligned}$$

The combination of (3.25), (3.26), (3.27) and (3.30) yields

$$\begin{aligned} & \Pr(\exists j\in\mathcal{A}^c, |-2\mathbf{X}_j^t(\mathbf{y}-\mathbf{X}_{\mathcal{A}}\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}})+2\sum_{k\in\mathcal{A}} q_{jk}\tilde{\boldsymbol{\beta}}^*_k|>\lambda_1^*\hat{\omega}_j) \\ \leq\ & \frac{4M^{2\gamma}n}{\lambda_1^{*2}}\left(16(B^2n+D^2p)\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^{*2}(\eta/2)^{-2\gamma}p}{(bn+\lambda_2 d)^2}+4Bp\sigma^2\right) \\ & +16D^2\frac{M^{2\gamma}p}{\lambda_1^{*2}}\|\boldsymbol{\beta}^*\|_2^2 \\ & +\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^2p}{(bn+\lambda_2 d)^2}\frac{4}{M^2} \\ & +\frac{\lambda_2^2D^2\|\boldsymbol{\beta}^*\|_2^2+Bpn\sigma^2+\lambda_1^2p}{(bn+\lambda_2 d)^2}\frac{16}{\eta^2} \\ & \equiv L_1+L_2+L_3+L_4. \end{aligned}$$

We have chosen $\gamma>2\nu/(1-\nu)$, then under conditions (C1)-(C6) it follows that

$$\begin{aligned} L_1 &= O\left(\left(\frac{\lambda^*}{\sqrt{n}}n^{\frac{(1+\gamma)(1-\nu)-1}{2}}\right)^{-\frac{2}{1+\gamma}}\right)\to 0, \\ L_2 &= O\left(\left(\frac{\lambda_1^*}{n}\right)^{\frac{2\gamma}{1+\gamma}}\frac{p\|\boldsymbol{\beta}^*\|_2^2}{\lambda_1^{*2}}\right)\to 0, \\ L_3 &= O\left(\frac{p}{n}\left(\frac{n}{\lambda_1^*}\right)^{\frac{2}{1+\gamma}}\right)\to 0, \\ L_4 &= O\left(\frac{p}{n}\frac{1}{\eta^2}\right)O\left(\left(\lambda_1^*\sqrt{\frac{p}{n}}\eta^{-\gamma}\right)^{2/\gamma}\left(\frac{p}{n}\left(\frac{n}{\lambda_1^*}\right)^{2/(2+\gamma)}\right)^{(1+\gamma)/\gamma}p^{-2/\gamma}\right)\to 0. \end{aligned} \tag{3.30}$$

□

3.A.4 Proof of Theorem 3.3.3

Proof. The proof for selection consistency of the AdaGril is exactly similar to the proof of selection consistency of AdaEnet (cf. pages 1748-1749 in (Zou and Zhang, 2009).

We now prove the asymptotic normality. For convenience we put

$$z_n = \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \mathbf{N}_{\mathcal{A}}^{-1} \left(\hat{\boldsymbol{\beta}}(\text{AdaGril})_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}\right).$$

Note that

$$\begin{aligned}
& \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}} - \mathbf{N}_{\mathcal{A}}^{-1} \boldsymbol{\beta}^*_{\mathcal{A}}\right) \\
= \; & \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}} - \mathbf{N}_{\mathcal{A}}^{-1} \boldsymbol{\beta}^*_{\mathcal{A}} + \tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0) - \tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0) + \boldsymbol{\beta}^*_{\mathcal{A}} - \boldsymbol{\beta}^*_{\mathcal{A}}\right) \\
= \; & \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\boldsymbol{\beta}^*_{\mathcal{A}} - \mathbf{N}_{\mathcal{A}}^{-1} \boldsymbol{\beta}^*_{\mathcal{A}}\right) + \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}} - \tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0)\right) \\
& + \; \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0) - \boldsymbol{\beta}^*_{\mathcal{A}}\right).
\end{aligned}$$

In addition, we have

$$\Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0) - \boldsymbol{\beta}^*_{\mathcal{A}}\right) = -\lambda_2 \Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{Q}_{\mathcal{A}} \boldsymbol{\beta}^*_{\mathcal{A}} + \Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{X}^t_{\mathcal{A}} \varepsilon.$$

Therefore, by Theorem 3.3.2 it follows that with probability tending to 1, $z_n = T_1 + T_2 + T_3$, where

$$T_1 = \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \mathbf{K}_{\mathcal{A}} \boldsymbol{\beta}^*_{\mathcal{A}} - \alpha^t \lambda_2 \Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{Q}_{\mathcal{A}} \boldsymbol{\beta}^*_{\mathcal{A}}, \text{ where}$$

$$\mathbf{K}_{\mathcal{A}} = \mathbf{I}_{\mathcal{A}} - \mathbf{N}_{\mathcal{A}}^{-1} = \text{diag}\left(\frac{\lambda_2 q_i}{n + \lambda_2 q_i}\right)_{i \in \mathcal{A}},$$

$$T_2 = \alpha^t \Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \left(\tilde{\boldsymbol{\beta}}^*_{\mathcal{A}} - \tilde{\boldsymbol{\beta}}^*_{\mathcal{A}}(\lambda_2, 0)\right),$$

$$T_3 = \alpha^t \Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{X}^t_{\mathcal{A}} \varepsilon.$$

We now show that $T_1 = o(1)$, $T_2 = o_P(1)$ and $T_3 \to_d N(0, \sigma^2)$. Then by the Slusky's theorem we know $z_n \to_d N(0, \sigma^2)$. From (C1) and the fact that $\alpha^t \alpha = 1$, we have

$$\begin{aligned}
T_1^2 &\leq 2\|\Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \mathbf{K}_{\mathcal{A}} \boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 + 2\|\lambda_2 \Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{Q}_{\mathcal{A}} \boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 \\
&\leq 2\|\mathbf{K}_{\mathcal{A}}\|_2^2 \|\Sigma_{\mathcal{A}}^{\frac{1}{2}} \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \|_2^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 + 2\lambda_2^2 \|\Sigma_{\mathcal{A}}^{-\frac{1}{2}} \mathbf{Q}_{\mathcal{A}}\|_2^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 \\
&\leq 2 \max_{i \in \mathcal{A}} \left(\frac{\lambda_2 q_i}{n + \lambda_2 q_i}\right)^2 \|\Sigma_{\mathcal{A}}^{\frac{1}{2}}\|_2^2 \| \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \|_2^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 + 2\lambda_2^2 \|\Sigma_{\mathcal{A}}^{-\frac{1}{2}}\|_2^2 \|\mathbf{Q}_{\mathcal{A}}\|_2^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 \\
&\leq 2 \max_{i \in \mathcal{A}} \left(\frac{\lambda_2 q_i}{n + \lambda_2 q_i}\right)^2 \|\Sigma_{\mathcal{A}}^{\frac{1}{2}}\|_2^2 \left(\|I\|_2 + \lambda_2 \|\Sigma_{\mathcal{A}}^{-1}\|_2 \|\mathbf{Q}_{\mathcal{A}}\|_2\right)^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 + 2\lambda_2^2 \|\Sigma_{\mathcal{A}}^{-\frac{1}{2}}\|_2^2 \|\mathbf{Q}_{\mathcal{A}}\|_2^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 \\
&\leq 2 \max_{i \in \mathcal{A}} \left(\frac{\lambda_2 q_i}{n + \lambda_2 q_i}\right)^2 Bn \left(1 + \frac{\lambda_2 D}{bn}\right)^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2 + 2\lambda_2^2 \frac{1}{bn} D^2 \|\boldsymbol{\beta}^*_{\mathcal{A}}\|_2^2.
\end{aligned}$$

To obtain previous inequalities we have used sub-multiplicativity and consistency of the $\|.\|_2$ matrix norms. Hence it follows from (C5) and (C6) that $T_1 = o(1)$. Similarly, we can bound T_2 as follows

$$\begin{aligned} T_2^2 &\leq \|\Sigma_{\mathcal{A}}^{\frac{1}{2}}\|_2^2 \| \left(I + \lambda_2 \Sigma_{\mathcal{A}}^{-1} \mathbf{Q}_{\mathcal{A}}\right) \|_2^2 \|\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^* - \tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*(\lambda_2, 0)\|_2^2 \\ &\leq Bn \left(1 + \frac{\lambda_2 D}{bn}\right)^2 \|\tilde{\boldsymbol{\beta}}_{\mathcal{A}}^* - \tilde{\boldsymbol{\beta}}_{\mathcal{A}}^*(\lambda_2, 0)\|_2^2 \\ &\leq Bn \left(1 + \frac{\lambda_2 D}{bn}\right)^2 \left(\frac{\lambda_1^* \hat{\eta}^{-\gamma} p}{bn + \lambda_2 d}\right)^2 \end{aligned}$$

where we have used (3.28) in the last step. Then $T_2^2 = O_{\mathrm{p}}(1)/n^2$. Next we consider T_3. Let $\mathbf{X}_{\mathcal{A}}[i,]$ denote the ith row of the matrix $\mathbf{X}_{\mathcal{A}}$. With such notation we can write $T_3 = \sum_{i=1}^n r_i \varepsilon_i$, where $r_i = \alpha^t (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}})^{-\frac{1}{2}} (\mathbf{X}_{\mathcal{A}}[i,])^t$. Then it is easy to see

$$\begin{aligned} \sum_{i=1}^n r_i^2 &= \sum_{i=1}^n \alpha^t (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}})^{-\frac{1}{2}} (\mathbf{X}_{\mathcal{A}}[i,])^t (\mathbf{X}_{\mathcal{A}}[i,]) (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}})^{-\frac{1}{2}} \alpha \\ &= \alpha^t (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}})^{-\frac{1}{2}} (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}}) (\mathbf{X}_{\mathcal{A}}^t \mathbf{X}_{\mathcal{A}})^{-\frac{1}{2}} \alpha \\ &= \alpha^t \alpha = 1. \end{aligned} \tag{3.31}$$

Furthermore, we have for $k = 2 + \delta$, $\delta > 0$

$$\sum_{i=1}^n \mathrm{E}[|\varepsilon_i|^{2+\delta}] |r_i^{2+\delta}| \leq \mathrm{E}[|\varepsilon_i|^{2+\delta}] (\sum_{i=1}^n |r_i^2| (\max_i |r_i^\delta|)) = \mathrm{E}[|\varepsilon|^{2+\delta}] (\max_i |r_i^2|)^{\frac{\delta}{2}}.$$

Note that $r_i^2 \leq \|\Sigma_{\mathcal{A}}^{-\frac{1}{2}} (\mathbf{X}_{\mathcal{A}}[i,])^t\|_2^2 \leq (\sum_{j \in \mathcal{A}} x_{ij}^2)(\lambda_{max}(\Sigma_{\mathcal{A}}^{-1})) \leq \frac{\sum_{j=1}^p x_{ij}^2}{bn}$.
Hence,

$$\sum_{i=1}^n \mathrm{E}[|\varepsilon_i|^{2+\delta}] |r_i^{2+\delta}| \leq \mathrm{E}[|\varepsilon|^{2+\delta}] \left(\frac{\max_i \sum_{j=1}^p x_{ij}^2}{bn}\right)^{\frac{\delta}{2}} \to 0. \tag{3.32}$$

From (3.31) and (3.32) Lyapunov conditions for the central limit theorem are established. Thus, $T_3 \to_d N(0, \sigma^2)$. This completes the proof. □

3.A.5 Proof of Theorem 3.3.4

Proof. Now, we consider the Adaptive Gril estimator, with Gril estimates as initial weights. The adaptive Gril estimates are defined by

$$\hat{\boldsymbol{\beta}}(\mathrm{AdaGril}) = \arg\min_{\boldsymbol{\beta}} \mathbf{N} \left\{ \|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2 + \lambda_2 \boldsymbol{\beta}^t \mathbf{Q} \boldsymbol{\beta} + \lambda_1^* \sum_{j=1}^p \hat{\omega}_j |\boldsymbol{\beta}_j| \right\}, \tag{3.33}$$

where

$$\hat{\omega}_j = \max(\frac{1}{|\hat{\boldsymbol{\beta}}(\text{Gril})_j|}, 1). \tag{3.34}$$

Then, the minimizer of (3.33) is also the minimizer of the Adaptive Lasso problem on augmented data $(\tilde{\mathbf{y}}, \tilde{\mathbf{X}})$ defined in (3.5). So, we have

$$\|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \;\leq\; \|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}^*\|_2^2 + \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j |\boldsymbol{\beta}_j^*|.$$

Since $tilde\mathbf{y} = \tilde{\mathbf{X}}\boldsymbol{\beta}^* + \tilde{\varepsilon}$, the latter is equivalent to

$$\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \;\leq\; \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) + 2\tilde{\varepsilon}^t \tilde{\mathbf{X}}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})).$$

Using the definition of $\tilde{\mathbf{X}}$ and $\tilde{\varepsilon}$ on the third term of the latter inequality, we have

$$\begin{aligned}\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \;\leq\; & \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) + 2\varepsilon^t \mathbf{X}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})) \\ & -2\lambda_2 \boldsymbol{\beta}^{*t} \mathbf{Q}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})).\end{aligned} \tag{3.35}$$

Obviously, we have

$$\sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) \;\leq\; \sum_{j=1}^{p} \hat{\omega}_j |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \tag{3.36}$$

and

$$-2\lambda_2 \boldsymbol{\beta}^{*t} \mathbf{Q}(\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})) \;\leq\; 2\lambda_2 \|\mathbf{Q}\boldsymbol{\beta}^*\|_\infty \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1. \tag{3.37}$$

For $0 < \theta \leq 1$, consider the set $\Gamma = \{\max_{j=1,\ldots,p} 2|U_j| \leq \theta\lambda_1\}$ with $U_j = n^{-1} \sum_{i=1}^{n} x_{i,j}\varepsilon_i$.

Therefore on the set Γ and using (3.35), (3.36) and (3.37), we obtain

$$\begin{aligned}\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \;\leq\; & \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) + \theta\lambda_1^* \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \\ & + \; 2\lambda_2 \|\mathbf{Q}^t \boldsymbol{\beta}^*\|_\infty \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1.\end{aligned} \tag{3.38}$$

Tacking $\theta = \frac{1}{4}$ and $\lambda_2 = \frac{\lambda_1^*}{8\|\mathbf{Q}\boldsymbol{\beta}^*\|_\infty}$ and adding $2^{-1}\lambda_1^* \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1$ to both sides of the previous inequality, we have

$$\begin{aligned}\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \frac{\lambda_1^*}{2}\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \;\leq\; & \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) \\ & + \; \lambda_1^* \sum_{j=1}^{p} |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j|\end{aligned} \tag{3.39}$$

Since $\hat{\omega}_j = \max\left(1/|\hat{\boldsymbol{\beta}}(\text{Gril})_j|, 1\right)$ for all $j = 1, ..., p$, we have

$$\begin{aligned}\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \frac{\lambda_1^*}{2}\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 &\leq \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) \\ &+ \lambda_1^* \sum_{j=1}^{p} \hat{\omega}_j |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j|. \quad (3.40)\end{aligned}$$

Let $\delta = 8\psi^{-1}\lambda_1^* s, \eta = \min_{j\in\mathcal{A}}(|\boldsymbol{\beta}_j^*|), \omega_{\max}(\mathcal{A}) = \max_{j\in\mathcal{A}} \hat{\omega}_j$. (Hebiri and van De Geer, 2010) show that on Γ

$$\delta \geq \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_\infty.$$

Suppose now that $\eta \geq 2\delta$, then we have $\eta \geq 2\delta \geq 2\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_\infty$, and

$$\begin{aligned}|\hat{\boldsymbol{\beta}}(\text{Gril})_j| &\geq \eta - \|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{Gril})\|_\infty \\ &\geq \frac{\eta}{2} \quad \text{for all} \quad j \in \mathcal{A}, \quad (3.41)\end{aligned}$$

Hence, we deduce that

$$\omega_{\max}(\mathcal{A}) \leq \max\left\{\frac{2}{\eta}, 1\right\}. \quad (3.42)$$

Using the fact that $|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j| + |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| = 0$, for all $j \in \mathcal{A}^c$, the triangular inequality and (3.42), we have

$$\begin{aligned}&\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \frac{\lambda_1^*}{2}\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \\ \leq\ & \lambda_1^* \sum_{j\in\mathcal{A}} \hat{\omega}_j (|\boldsymbol{\beta}_j^*| - |\hat{\boldsymbol{\beta}}(\text{AdaGril})_j| + |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \\ \leq\ & 2\lambda_1^* \sum_{j\in\mathcal{A}} \hat{\omega}_j |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \\ \leq\ & 2\lambda_1^* (\omega_{\max}(\mathcal{A})) \sum_{j\in\mathcal{A}} |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \\ \leq\ & 2\lambda_1^* \max\left\{\frac{2}{\eta}, 1\right\} \sum_{j\in\mathcal{A}} |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \quad (3.43)\end{aligned}$$

Since $\sum_{j\in\mathcal{A}} |\boldsymbol{\beta}_j^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j| \leq \sqrt{s}\|\boldsymbol{\beta}_{\mathcal{A}}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_{\mathcal{A}}\|_2$, we obtain that

$$\|\tilde{\mathbf{X}}\boldsymbol{\beta}^* - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \frac{\lambda_1^*}{2}\|\boldsymbol{\beta}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \leq 2\sqrt{s}\lambda_1^* \max\left\{\frac{2}{\eta}, 1\right\} \|\boldsymbol{\beta}_{\mathcal{A}}^* - \hat{\boldsymbol{\beta}}(\text{AdaGril})_{\mathcal{A}}\|_2. \quad (3.44)$$

According to inequality (3.43), we have

$$\|\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \leq 4\max\left\{\frac{2}{\eta}, 1\right\} \sum_{j\in\mathcal{A}} |\boldsymbol{\beta}^{*}_j - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j|. \tag{3.45}$$

So

$$\sum_{j\in\mathcal{A}^c} |\boldsymbol{\beta}^{*}_j - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j|) \leq 4\max\left\{\frac{2}{\eta}, 1\right\} \sum_{j\in\mathcal{A}} |\boldsymbol{\beta}^{*}_j - \hat{\boldsymbol{\beta}}(\text{AdaGril})_j|. \tag{3.46}$$

This last inequality shows that $\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})$ obeys to the assumption RE, and hence

$$\begin{aligned} \|\boldsymbol{\beta}^{*}_{\mathcal{A}} - \hat{\boldsymbol{\beta}}(\text{AdaGril})_{\mathcal{A}}\|_2^2 &\leq \|\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \\ &\leq \psi^{-1}\|\tilde{\mathbf{X}}\boldsymbol{\beta}^{*} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2. \end{aligned} \tag{3.47}$$

The combination of this last inequality with (3.44), give us

$$\begin{aligned} \|\tilde{\mathbf{X}}\boldsymbol{\beta}^{*} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 + \frac{\lambda_1^{*}}{2}\|\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 &\leq 2\sqrt{s}\lambda_1^{*}\sqrt{\psi^{-1}}\max\left\{\frac{2}{\eta}, 1\right\} \times \\ &\quad \|\tilde{\mathbf{X}}\boldsymbol{\beta}^{*} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2 \end{aligned} \tag{3.48}$$

So

$$\|\tilde{\mathbf{X}}\boldsymbol{\beta}^{*} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \leq 4\psi^{-1}\lambda_1^{*2}\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s. \tag{3.49}$$

Since

$$\begin{aligned} \|\tilde{\mathbf{X}}\boldsymbol{\beta}^{*} - \tilde{\mathbf{X}}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 &= \|\mathbf{X}\boldsymbol{\beta}^{*} - \mathbf{X}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \\ &+ \lambda_2(\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril}))^t\mathbf{Q}(\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})), \end{aligned} \tag{3.50}$$

we obtain

$$\|\mathbf{X}\boldsymbol{\beta}^{*} - \mathbf{X}\hat{\boldsymbol{\beta}}(\text{AdaGril})\|_2^2 \leq 4\psi^{-1}\lambda_1^{*2}\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s. \tag{3.51}$$

Using (3.44) and the fact that $\|\mathbf{v}\|_\infty \leq \|\mathbf{v}\|_1$ for all $\mathbf{v} \in \mathbb{R}^p$, we have

$$\|\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_1 \leq 8\psi^{-1}\lambda_1^{*}\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s, \tag{3.52}$$

and

$$\|\boldsymbol{\beta}^{*} - \hat{\boldsymbol{\beta}}(\text{AdaGril})\|_\infty \leq 8\psi^{-1}\lambda_1^{*}\left(\max\left\{\frac{2}{\eta}, 1\right\}\right)^2 s. \tag{3.53}$$

□

Chapter 4

Regularization in regression: comparing Bayesian and frequentist methods in a poorly informative situation

Résumé : Nous proposons une approche globale de la sélection bayésienne de variables en régression construite sur les lois a priori g de Zellner dans une approche similaire mais non identique à celle de (Liang et al., 2008). Notre choix ne nécessite aucune calibration. Nous comparons les approches de régularisation bayésienne et fréquentielle dans un contexte peu informatif où le nombre de variables est presque égal au nombre d'observations.

4.1 Introduction

Given a response variable, y and a collection of p associated potential predictor variables $x_1, \ldots, x_p$, the classical linear regression model imposes a linear dependence on the conditional expectation (Rao, 1973)

$$\mathbb{E}[y|x_1, \ldots, x_p] = \beta_0 + \beta_1 x_1 + \ldots \beta_p x_p \,.$$

A fundamental inferential direction for those models relates to the variable selection problem, namely that only variables of relevance should be kept within the regression while the others should be removed. While we cannot discuss at length the potential applications of this perspective, variable selection is particularly relevant when the number p of regressors is larger than the number n of observations (as in microarray and other genetic data analyzes).

To deal with poorly or ill-posed regression problems, many regularization methods have been proposed, like ridge regression (Hoerl and Kennard, 1970) and Lasso (Tibshirani, 1996). Recently the interest for frequentist regularization methods has increased and this

has produced a flurry of methods (see, among others, Candes and Tao, 2007; Zou and Hastie, 2005; Zou, 2006; Yuan and Lin, 2007).

However, a natural approach for regularization is to follow the Bayesian paradigm as demonstrated recently by the Bayesian Lasso of Park and Casella (2008). The amount of literature on Bayesian variable selection is quite enormous (a small subset of which is, for instance, Mitchell and Beauchamp, 1988; George and Foster, 1993; Chipman, 1996; Smith and Kohn, 1996; George and McCulloch, 1997; Dupuis and Robert, 2003; Brown and Vannucci, 1998; Philips and Guttman, 1998; George, 2000; Kohn et al., 2001; Nott and Green, 2004; Schneider and Corcoran, 2004; Casella and Moreno, 2006; Cui and George, 2008; Liang et al., 2008; Bottolo and Richardson, 2010). The number of approaches and scenarii that have been advanced to undertake the selection of the most relevant variables given a set of observations is quite large, presumably due to the vague decisional setting induced by the question *Which variables do matter?* Such a variety of resolutions signals a lack of agreement between the actors in the field.

Most of the solutions, including Liang et al. (2008) and Bottolo and Richardson (2010), focus on the use of the g-prior, introduced by Zellner (1986). While this prior has a long history and while it reduces the prior input to a single integer, g, the influence of this remaining prior factor is long-lasting and large values of g are no guarantee of negligible effects, in connection with the Bartlett or Lindley–Jeffreys paradoxes (Bartlett, 1957; Lindley, 1957; Robert, 1993), as illustrated for instance in Celeux and Robert (2006) or Marin and Robert (2007). In order to alleviate this influence, some empirical Bayes [Cui and George (2008)] and hierarchical Bayes [Zellner and Siow (1980), Celeux and Robert (2006), Marin and Robert (2007), Liang et al. (2008) and Bottolo and Richardson (2010)] solutions have been proposed. In this paper, we pay special attention to two calibration-free hierarchical Zellner g-priors. The first one is the Jeffreys prior which is not location invariant. A second one avoids this problem by only considering models with at least one variable in the model.

The purpose of our paper is to compare the frequentist and the Bayesian points of views in regularization when n remains (slightly) greater than p. As a matter of fact, the use of g-prior implies that the sample size n has to be greater than p. This comparison is considered from both the predictive and the explicative point of views. The outcome of this study is that Bayesian methods are quite similar while dominating their frequentist counterpart.

The plan of the paper is as follows: we recall the details of Zellner's (1986) original g-prior in Section 4.2, and discuss therein the potential choices of g. We present hierarchical noninformative alternatives in Section 4.3. Section 4.4 compares the results of Bayesian and frequentist methods on simulated and real datasets. Section 4.5 concludes the paper.

4.2 Zellner's g-priors

Following standard notations, we introduce a variable $\boldsymbol{\gamma} \in \boldsymbol{\Gamma} = \{0,1\}^{\otimes p}$ that indicates which variables are active in the regression, excluding the constant vector corresponding to the intercept that is assumed to be always present in the linear regression model.

We observe $\mathbf{y}, \mathbf{x}_1, \ldots, \mathbf{x}_p \in \mathbb{R}^n$, the model $\mathcal{M}_{\boldsymbol{\gamma}}$ is defined as the conditional distribution

$$\mathbf{y}|\mathbf{X}, \boldsymbol{\gamma}, \boldsymbol{\beta}^{\boldsymbol{\gamma}}, \sigma^2 \sim \mathcal{N}_n\left(\mathbf{X}^{\boldsymbol{\gamma}}\boldsymbol{\beta}^{\boldsymbol{\gamma}}, \sigma^2 I_n\right) , \tag{4.1}$$

where

- $p_{\boldsymbol{\gamma}} = \sum_{i=1}^{p} \gamma_i$,
- $\mathbf{X}^{\boldsymbol{\gamma}}$ is the $(n, p_{\boldsymbol{\gamma}} + 1)$ matrix which columns are made of the vector $\mathbf{1}_n$ and of the variables $\mathbf{x}_i$ for which $\gamma_i = 1$,
- $\boldsymbol{\beta}^{\boldsymbol{\gamma}} \in \mathbb{R}^{p_{\boldsymbol{\gamma}}+1}$ and $\sigma^2 \in \mathbb{R}_+^*$ are unknown parameters.

The same symbol for the parameter σ^2 is used across all models. For model $\mathcal{M}_{\boldsymbol{\gamma}}$, Zellner's g-prior is given by

$$\boldsymbol{\beta}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}+1}(\tilde{\boldsymbol{\beta}}^{\boldsymbol{\gamma}}, g_{\boldsymbol{\gamma}}\sigma^2((\mathbf{X}^{\boldsymbol{\gamma}})'\mathbf{X}^{\boldsymbol{\gamma}})^{-1}) ,$$

$$\pi(\sigma^2|\mathbf{X}, \boldsymbol{\gamma}) \propto \sigma^{-2} .$$

The experimenter chooses the prior expectation $\tilde{\boldsymbol{\beta}}^{\boldsymbol{\gamma}}$ and $g_{\boldsymbol{\gamma}}$. For such a prior, we obtain the classical average between prior and observed regressors,

$$\mathbb{E}(\boldsymbol{\beta}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \mathbf{y}) = \frac{g_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}^{\boldsymbol{\gamma}} + \tilde{\boldsymbol{\beta}}^{\boldsymbol{\gamma}}}{g_{\boldsymbol{\gamma}} + 1} .$$

This prior is traditionally called Zellner's g-prior in the Bayesian folklore because of the use of the constant $g_{\boldsymbol{\gamma}}$ by Zellner (1986) in front of Fisher's information matrix $((\mathbf{X}^{\boldsymbol{\gamma}})'\mathbf{X}^{\boldsymbol{\gamma}})^{-1}$. Its appeal is that, by using the information matrix as a global scale,

- it avoids the specification of a whole prior covariance matrix, which would be a tremendous task;
- it allows for a specification of the constant $g_{\boldsymbol{\gamma}}$ in terms of observational units, or virtual prior pseudo-observations in the sense of de Finetti (1972).

However, fundamental feature of the g-prior is that this prior is improper, due to the use of an infinite mass on σ^2. From a theoretical point of view, this should jeopardize the use of posterior model probabilities since these probabilities are not uniquely scaled under improper priors, because there is no way of eliminating the residual constant factor in those priors (DeGroot, 1973; Kass and Raftery, 1995; Robert, 2001). However, under the assumption that σ^2 is a parameter that has a meaning common to all models $\mathcal{M}_{\boldsymbol{\gamma}}$, Berger et al. (1998) develop a framework that allows to work with a single improper prior that is common to all models (see also Marin and Robert, 2007). A fundamental appeal of Zellner's g-prior in model comparison and in particular in variable selection is its simplicity, since it reduces the prior input to the sole specification of a scale parameter g.

At this stage, we need to point out that an alternative g-prior is often used (Berger et al., 1998; Fernandez and Steel, 2001; Liang et al., 2008; Bottolo and Richardson, 2010), by singling out the intercept parameter in the linear regression. By first assuming a centering of the covariates, i.e. $\mathbf{1}_n'\mathbf{x}_i = 0$ for all i's, the intercept α is given a flat prior while the other parameters of $\boldsymbol{\beta}^{\boldsymbol{\gamma}}$ are associated with a corresponding g-prior. Thus, this is an alternative to model $\mathcal{M}_{\boldsymbol{\gamma}}$, which we denote by model $\mathcal{M}_{\boldsymbol{\gamma}}^{\text{inv}}$ to stress the distinctions between both representations and which is such that

$$\mathbf{y}|\mathbf{X}, \boldsymbol{\gamma}, \alpha, \boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}}, \sigma^2 \sim \mathcal{N}_n\left(\alpha\mathbf{1}_n + \mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}}\boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}}, \sigma^2 I_n\right) , \tag{4.2}$$

where

- $\mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}}$ the $(n, p_{\boldsymbol{\gamma}})$ matrix which columns are made of the variables $\mathbf{x}_i$ for which $\gamma_i = 1$,
- $\alpha \in \mathbb{R}$, $\boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}} \in \mathbb{R}^{p_{\boldsymbol{\gamma}}}$ and $\sigma^2 \in \mathbb{R}_+^*$ are unknown parameters.

The parameters σ^2 and α are denoted the same way across all models and rely on the same prior. Namely, for model $\mathcal{M}_{\boldsymbol{\gamma}}^{\text{inv}}$, the corresponding Zellner's g-prior is given by

$$\boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}}(\tilde{\boldsymbol{\beta}}_{\text{inv}}^{\boldsymbol{\gamma}}, g_{\boldsymbol{\gamma}}\sigma^2((\mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}})'\mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}})^{-1}) ,$$

$$\pi(\alpha, \sigma^2|\mathbf{X}, \boldsymbol{\gamma}) \propto \sigma^{-2} .$$

In that case, we obtain

$$\mathbb{E}(\boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \mathbf{y}) = \frac{g_{\boldsymbol{\gamma}}\hat{\boldsymbol{\beta}}_{\text{inv}}^{\boldsymbol{\gamma}} + \tilde{\boldsymbol{\beta}}_{\text{inv}}^{\boldsymbol{\gamma}}}{g_{\boldsymbol{\gamma}} + 1} ,$$

and

$$\mathbb{E}(\alpha|\mathbf{X}, \boldsymbol{\gamma}, \mathbf{y}) = \bar{\mathbf{y}} = \frac{1}{n}\sum_{i=1}^{n} y_i .$$

For models $\mathcal{M}_{\boldsymbol{\gamma}}$ and $\mathcal{M}_{\boldsymbol{\gamma}}^{\text{inv}}$, in a noninformative setting, we can for instance choose $\tilde{\boldsymbol{\beta}}^{\boldsymbol{\gamma}} = 0_{p_{\boldsymbol{\gamma}}+1}$ or $\tilde{\boldsymbol{\beta}}_{\text{inv}}^{\boldsymbol{\gamma}} = 0_{p_{\boldsymbol{\gamma}}}$ and $g_{\boldsymbol{\gamma}}$ large. However, as pointed out in Marin and Robert (2007, Chapter 3) among others, there is a lasting influence of $g_{\boldsymbol{\gamma}}$ over the resulting inference and it is impossible to "let $g_{\boldsymbol{\gamma}}$ go to infinity" to eliminate this influence, because of the

Bartlett and Lindley-Jeffreys (Bartlett, 1957; Lindley, 1957; Robert, 1993) paradoxes that an infinite value of $g_{\boldsymbol{\gamma}}$ ends up selecting the null model, regardless of the information brought by the data. For this reason, data-dependent versions of $g_{\boldsymbol{\gamma}}$ have been proposed with various degrees of justification:

- Kass and Wasserman (1995) use $g_{\boldsymbol{\gamma}} = n$ so that the amount of information about the parameters contained in the prior equals the amount of information brought by one observation. As shown by Foster and George (1994), for n large enough this perspective is very close to using the Schwarz (Kass and Wasserman, 1995) or BIC criterion in that the log-posterior corresponding to $g = n$ is equal to the penalized log-likelihood of this criterion.
- Foster and George (1994) and George and Foster (2000) propose $g_{\boldsymbol{\gamma}} = p_{\boldsymbol{\gamma}}^2$, in connection with the Risk Inflation Criterion (RIC) that penalizes the regression sum of squares.
- Fernandez and Steel (2001) gather both perspectives in $g_{\boldsymbol{\gamma}} = \max(n, p_{\boldsymbol{\gamma}}^2)$ as a conservative bridge between BIC and RIC, a choice that they christened "benchmark prior".
- George and Foster (2000) and Cui and George (2008) resort to empirical Bayes techniques.

These solutions, while commendable since based on asymptotic properties (see in particular Fernandez and Steel, 2001 for consistency results), are nonetheless unsatisfactory in that they depend on the sample size and involve a degree of arbitrariness.

4.3 Mixtures of g-priors

The most natural Bayesian approach to solving the uncertainty on the parameter $g_{\boldsymbol{\gamma}} = g$ is to put a hyperprior on this parameter:

- This was implicitely proposed by Zellner and Siow (1980) since those authors introduced Cauchy priors on the $\boldsymbol{\beta}^{\boldsymbol{\gamma}}$'s since this corresponds to a g-prior augmented by a Gamma $\mathcal{G}a(1/2, n/2)$ prior on g^{-1}.
- For model $\mathcal{M}_{\boldsymbol{\gamma}}^{\text{inv}}$, Liang et al. (2008), Cui and George (2008) and Bottolo and Richardson (2010) use
$$\boldsymbol{\beta}_{\text{inv}}^{\boldsymbol{\gamma}} | \mathbf{X}, \boldsymbol{\gamma}, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}}(0_{p_{\boldsymbol{\gamma}}}, g\sigma^2((\mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}})'\mathbf{X}_{\text{inv}}^{\boldsymbol{\gamma}})^{-1})$$
and an hyperprior of the form
$$\pi(\alpha, \sigma^2, g | \mathbf{X}, \boldsymbol{\gamma}) \propto (1+g)^{-a/2}\sigma^{-2}\,,$$
with $a > 2$. This constraint on a is due to the fact that the hyperprior must be proper, in connection with the separate processing of the intercept α and the use of

a Lebesgue measure as a prior on α. We note that a needs to be specified, $a = 3$ and $a = 4$ being the solutions favored by Liang et al. (2008).

- For model $\mathcal{M}_{\boldsymbol{\gamma}}$, Celeux and Robert (2006) and Marin and Robert (2007) used

$$\boldsymbol{\beta}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}+1}(0_{p_{\boldsymbol{\gamma}}+1}, g\sigma^2((\mathbf{X}^{\boldsymbol{\gamma}})'\mathbf{X}^{\boldsymbol{\gamma}})^{-1})$$

and a hyperprior of the form

$$\pi(\sigma^2, g|\mathbf{X}) \propto \sigma^{-2} g^{-1} \mathbb{I}_{\mathbb{N}^*}(g)\,,$$

as a simple way of circumventing computational difficulties.

For model $\mathcal{M}_{\boldsymbol{\gamma}}$ a more convincing modelling is possible since the Jeffreys prior is available. Indeed, if

$$\boldsymbol{\beta}^{\boldsymbol{\gamma}}|\mathbf{X}, \boldsymbol{\gamma}, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}+1}(0_{p_{\boldsymbol{\gamma}}+1}, g\sigma^2((\mathbf{X}^{\boldsymbol{\gamma}})'\mathbf{X}^{\boldsymbol{\gamma}})^{-1})\,,$$

then

$$\mathbf{y}|\mathbf{X}, \boldsymbol{\gamma}, g, \sigma^2 \sim \mathcal{N}_{p_{\boldsymbol{\gamma}}+1}\left(0_n, \sigma^2\left[\mathbf{I}_n - \frac{g}{g+1}\mathbf{P}_{\boldsymbol{\gamma}}\right]^{-1}\right)\,,$$

where $\mathbf{P}_{\boldsymbol{\gamma}}$ is the orthogonal projector on the linear subspace spanned by the columns of $\mathbf{X}^{\boldsymbol{\gamma}}$. Since, the Fisher information matrix is

$$\mathfrak{I}(\sigma^2, g) = \left(\frac{1}{2}\right)\begin{bmatrix} n/\sigma^4 & (p_{\boldsymbol{\gamma}}+1)/(\sigma^2(g+1)) \\ (p_{\boldsymbol{\gamma}}+1)/(\sigma^2(g+1)) & (p_{\boldsymbol{\gamma}}+1)/(g+1)^2 \end{bmatrix},$$

the corresponding Jeffreys prior on (σ^2, g) is

$$\pi(\sigma^2, g|\mathbf{X}) \propto \sigma^{-2}(g+1)^{-1}\,.$$

For such a prior modelling, there exists a closed-form representation for posterior quantities in that

$$\pi(\boldsymbol{\gamma}, g|\mathbf{X}, \mathbf{y}) \propto (g+1)^{n/2-(p_{\boldsymbol{\gamma}}+1)/2-1}(1+g(1-\mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y}))^{-n/2}$$

and

$$\pi(\boldsymbol{\gamma}|\mathbf{X}, \mathbf{y}) \propto \frac{{}_2F_1(n/2, 1; (p_{\boldsymbol{\gamma}}+3)/2; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})}{p_{\boldsymbol{\gamma}}+1}\,, \tag{4.3}$$

where ${}_2F_1$ is the Gaussian hypergeometric function (Butler and Wood, 2002). We can thus proceed to undertake Bayesian variable selection without resorting at all to numerical methods (Marin and Robert, 2007). Moreover, the shrinkage factor due to the Bayesian modelling can also be expressed in closed form as

$$\begin{aligned}\mathbb{E}(g/(g+1)|\mathbf{X}, \boldsymbol{\gamma}, \mathbf{y}) &= \frac{\displaystyle\int_0^\infty g(g+1)^{n/2-(p_{\boldsymbol{\gamma}}+1)/2-2}(1+g(1-\mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y}))^{-n/2}\mathrm{d}g}{\displaystyle\int_0^\infty (g+1)^{n/2-(p_{\boldsymbol{\gamma}}+1)/2-1}(1+g(1-\mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y}))^{-n/2}\mathrm{d}g} \\ &= \frac{2\,{}_2F_1(n/2, 2; (p_{\boldsymbol{\gamma}}+3)/2+1; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})}{(p_{\boldsymbol{\gamma}}+3)\,{}_2F_1(n/2, 1; (p_{\boldsymbol{\gamma}}+3)/2; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})}.\end{aligned}$$

This obviously leads to straightforward representations for Bayes estimates. If $\mathbf{X}_{\text{new}}$ is a $q \times p$ matrix containing q *new* values of the explanatory variables for which we would like to predict the corresponding response $\mathbf{y}_{\text{new}}$, the Bayesian predictor of $\mathbf{y}_{\text{new}}$ is given by

$$\begin{aligned}\hat{\mathbf{y}}^{\boldsymbol{\gamma}}_{\text{new}} &= \mathbb{E}\left[\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}, \mathbf{X}, \boldsymbol{\gamma}, \mathbf{y}\right] \\ &= 2\,\frac{{}_2F_1(n/2, 2; (p_{\boldsymbol{\gamma}}+3)/2+1; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})}{(p_{\boldsymbol{\gamma}}+3)\,{}_2F_1(n/2, 1; (p_{\boldsymbol{\gamma}}+3)/2; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})}\,\mathbf{X}_{\text{new}}\hat{\boldsymbol{\beta}}^{\boldsymbol{\gamma}}\,.\end{aligned}$$

Similarly, the Bayesian model averaging predictor of $\mathbf{y}_{\text{new}}$ is given by

$$\begin{aligned}\hat{\mathbf{y}}_{\text{new}} &= \mathbb{E}\left[\mathbf{y}_{\text{new}}|\mathbf{X}_{\text{new}}, \mathbf{X}, \mathbf{y}\right] \\ &= 2\,\frac{\sum_{\boldsymbol{\gamma}\in\Gamma}\,{}_2F_1(n/2, 2; (p_{\boldsymbol{\gamma}}+3)/2+1; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})/\left[(p_{\boldsymbol{\gamma}}+1)(p_{\boldsymbol{\gamma}}+3)\right]}{\sum_{\boldsymbol{\gamma}\in\Gamma}\,{}_2F_1(n/2, 1; (p_{\boldsymbol{\gamma}}+3)/2; \mathbf{y}'\mathbf{P}_{\boldsymbol{\gamma}}\mathbf{y}/\mathbf{y}'\mathbf{y})/(p_{\boldsymbol{\gamma}}+1)}\,\mathbf{X}_{\text{new}}\hat{\boldsymbol{\beta}}^{\boldsymbol{\gamma}}\,.\end{aligned} \tag{4.4}$$

This numerical simplification in the derivation of Bayesian estimates and predictors is found in Liang et al. (2008) and exploited further in Bottolo and Richardson (2010). Note also that Guo and Speckman (2009) have furthermore established the consistency of the Bayes factors based on such priors.

In contrast with this proposal, the prior of Liang et al. (2008) depends on a tuning parameter a. Despite that, there also exist arguments to support this prior modelling, including the important issue of invariance under location-scale transforms. As seen in the above formulae, the Jeffreys prior associated to model $\mathcal{M}_{\boldsymbol{\gamma}}$ ensure scale invariance but not location invariance. In order to ensure location invariance for model $\mathcal{M}_{\boldsymbol{\gamma}}$, it would be necessary to center the observation variable y as well as the dependent variables X. Obviously, this centering of the data is completely unjustified from a Bayesian perspective and further it creates artificial correlations between observations. However it could be argued that the lack of location invariance only pertains to quite specific and somehow artificial situations and that it is negligible in most situations. We will return to this point in the comparison section.

A location scale alternative consists in using the prior of Liang et al. (2008) with $a = 2$ and excluding the null model from the competitors. This prior leads to the model posterior probability

$$\pi(\boldsymbol{\gamma}|\mathbf{X}, \mathbf{y}) \propto \frac{{}_2F_1((n-1)/2, 1; (p_{\boldsymbol{\gamma}}+2)/2; (\mathbf{y}-\bar{\mathbf{y}})'\mathbf{P}_{\boldsymbol{\gamma}}(\mathbf{y}-\bar{\mathbf{y}})/(\mathbf{y}-\bar{\mathbf{y}})'(\mathbf{y}-\bar{\mathbf{y}}))}{p_{\boldsymbol{\gamma}}}\,. \tag{4.5}$$

Equations (4.3) and (4.5) are similar. However, in the last part of (4.5), $\mathbf{y}$ is centered, ensuring the location invariance of the selection procedure.

4.4 Numerical comparisons

We present here the results of numerical experiments aiming at comparing the behavior of Bayesian variable selection and of some (non-Bayesian) popular regularization methods

in regression, when considered from a variable selection point of view: The regularization methods that we consider are the Lasso, the Dantizg selector, and elastic net, described in Section 4.4.1. The Bayesian variable selection procedures we consider oppose strategies for selecting the hyperparameter g in Zellner's g-priors: We include in this comparison the intrinsic prior (Casella and Moreno, 2006) which is another default objective prior for the non informative setting that does not require any tuning parameters and is also invariant under location and scale changes. All procedure under comparison are described in Table 4.1. We have also included in this comparison the highly standard AIC and BIC penalized likelihood criteria. Moreover, we will refer to the performances of an ORACLE procedure that assumes the true model is known and that estimate the regression coefficients with the least squares method.

4.4.1 Regularization methods

***1) The Lasso*:** Introduced by Tibshirani (1996), the Lasso is a shrinkage method for linear regression. It is defined as the solution to the following ℓ_1 penalized least squares optimization problem

$$\hat{\boldsymbol{\beta}}_{\text{Lasso}} = \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - X\boldsymbol{\beta}||_2^2 + \lambda \sum_{j=1}^{p} |\beta_j|,$$

where λ is a positive tuning parameter.

***2) The Dantzig Selector*:** Candes and Tao (2007) introduced the Dantzig Selector as an alternative to the Lasso. The Dantzig Selector is the solution to the optimization problem

$$\min_{\beta \in \mathbb{R}^p} \|\beta\|_1 \quad \text{subject to} \quad \|\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)\|_\infty \leq \lambda,$$

where λ is a positive tuning parameter. The constraint $\|\mathbf{X}^t(\mathbf{y} - \mathbf{X}\beta)\|_\infty \leq \lambda$ can be viewed as a relaxation of the normal equation in the classical linear regression.

***3) The Elastic Net (Enet)*:** The Lasso has at least two limitations: a) Lasso does not encourage grouped selection in the presence of high correlated covariates and b) for the $p > n$ case Lasso can select at most n covariates. To overcome these limitations, Zou and Hastie (2005) proposed an elastic net that combines both ridge ℓ_2 and Lasso ℓ_1 penalties, i.e.

$$\hat{\boldsymbol{\beta}}_{\text{Enet}} = \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - X\boldsymbol{\beta}||_2^2 + \lambda \sum_{j=1}^{p} |\beta_j| + \mu \sum_{j=1}^{p} \beta_j^2,$$

where λ and μ are two positive tuning parameters.

4.4.2 Numerical experiments on simulated datasets

We have designed six different simulated datasets as benchmarks chosen as follows:

1. Example 1 (**sparse uncorrelated design**) corresponds to an uncorrelated covariate setting ($\rho = 0$), with $p = 10$ predictors and where the components of $\mathbf{x}_i$ ($i = 1, \ldots, 10$) are iid $\mathcal{N}_1(0, 1)$ realizations. The response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n(2 + \mathbf{x}_2 + 2\mathbf{x}_3 - 2\mathbf{x}_6 - 1.5\mathbf{x}_7, I_n) .$$

2. Example 2 (**sparse correlated design**) corresponds to a correlated case ($\rho = 0.9$), with $p = 10$ predictors and $\mathbf{x}_i = (\mathbf{z}_i + 3\mathbf{z}_{11})/\sqrt{10}$, for $i = 1, 2$, $\mathbf{x}_i = (\mathbf{z}_i + 3\mathbf{z}_{12})/\sqrt{10}$, for $i = 3, 4, 5$, and $\mathbf{x}_i = (\mathbf{z}_i + 3\mathbf{z}_{13})/\sqrt{10}$ for $i = 6, \ldots, 10$, the components of $\mathbf{z}_i$ ($i = 1, \ldots, 13$) being iid $\mathcal{N}_1(0, 1)$ realizations. The use of common terms in the $\mathbf{x}_i$'s obviously induces a correlation among those $\mathbf{x}_i$'s: the correlation between variables $\mathbf{x}_1$ and $\mathbf{x}_2$ is 0.9, as for the variables ($\mathbf{x}_3$, $\mathbf{x}_4$ and $\mathbf{x}_5$), and for the variables ($\mathbf{x}_6$, $\mathbf{x}_7$, $\mathbf{x}_8$, $\mathbf{x}_9$ and $\mathbf{x}_{10}$). There is no correlation between those three groups of variables. The response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n(2 + \mathbf{x}_2 + 2\mathbf{x}_3 - 2\mathbf{x}_6 - 1.5\mathbf{x}_7, I_n) .$$

3. Example 3 (**sparse noisy correlated design**) involves $p = 8$ predictors. Those variables are generated using a multivariate Gaussian distribution with correlations

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = 0.5^{|i-j|} .$$

   The response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n(3\mathbf{x}_1 + 1.5\mathbf{x}_2 + 2\mathbf{x}_5, 9I_n) .$$

4. Example 4 (**saturated correlated design**) is the same as Example 4, except that the response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n\left(0.85 \sum_{i=1}^{8} \mathbf{x}_i, I_n\right) .$$

5. Example 5 involves $p = 9$ predictors. Those variables are generated using a multivariate Gaussian distribution with correlations

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = 0.7^{|i-j|} .$$

   The response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n(2\mathbf{x}_2 - 3\mathbf{x}_4, I_n) .$$

6. Example 6 (**null model**) involves $p = 8$ predictors. Those variables are generated using a multivariate Gaussian distribution with correlations

$$\rho(\mathbf{x}_i, \mathbf{x}_j) = 0.5^{|i-j|}\,.$$

The response is simulated as

$$\mathbf{y} \sim \mathcal{N}_n(2, 4I_n)\,.$$

Each dataset consists of a training set of size $n = 15$, on which the regression model has been fitted and a test set T of size $n_T = 200$ for assessing performances. Tuning parameters in the Lasso, the Dantzig selector (DZ), and the elastic net (ENET) have been selected by minimizing the cross-validation prediction error through leave-one-out. For each example, 100 independent datasets have been simulated. We use three measures of performances:

1. The root mean squared error (MSE)

$$\mathrm{MSE}_y = \sqrt{\textstyle\sum_{i=1}^{n_T}(y_i - \hat{y}_i)^2/n_T}\,,$$

$\hat{y}_i$ being the prediction of y_i in the test set;

2. HITS: the number of correctly identified influential variables;

3. FP (False Positives): the number of non-influential variables declared as influential.

Using those six different datasets as benchmarks, we compare the variable selection methods listed in Table 4.1. The performances of the above selection methods are summarized in Tables 4.2–4.13. In the Bayesian approaches, the set of variables is naturally selected according to the maximum posterior probability $\pi(\boldsymbol{\gamma}|\mathbf{X}, \mathbf{y})$ and the predictive is obtained via the Bayesian model averaging predictors.

In this numerical experiment, the Bayesian procedures are clearly much more parsimonious than the regularization procedures in that they almost always avoid overfitting. In all examples, the false positive rate FP is smaller for the Bayesian solutions than for the regularization methods. Except for the ZS-F and OVS scenarios which behave slightly worse than the others, all the Bayesian procedures tested here produce the same selection of predictors. It seems that ZS-F has a slight tendency to select too many variables. The performances of OVS are somewhat disappointing and this procedure seems to have a tendency to be too parsimonious. From a predictive viewpoint, computing the MSE by model averaging, Bayesian approaches also perform better than regularization approaches except for the saturated correlated example (Example 4). We further note that the classical selection procedures based on AIC and BIC do not easily reject variables and are thus slightly worse than Bayesian and regularization procedures (a fact not surprising for AIC). In all examples, the NIMS and HG-2 approaches lead to optimal performances in that they select the right covariates and only the right covariates, while achieving close to the minimal root mean squared error compared with all the other Bayesian solutions we considered. They also do almost systematically better than BIC and AIC.

AIC	Akaike Information Criterion
BIC	Bayesian Information Criterion
BRIC	g prior with $g = \max(n, p^2)$ (Fernandez and Steel, 2001)
EB-L	Local EB estimate of g in g-prior (Cui and George, 2008)
EB-G	Global EB estimate of g in g-prior (Cui and George, 2008)
ZS-N	Base model in Bayes factor taken as the null model (Liang et al., 2008)
ZS-F	Base model in Bayes factor taken as the full model (Liang et al., 2008)
OVS	Objective variable selection using the intrinsic prior (Casella and Moreno, 2006)
HG-3	Hyper-g prior with $a = 3$ (Liang et al., 2008)
HG-4	Hyper-g prior with $a = 4$ (Liang et al., 2008)
HG-2	Hyper-g prior with $a = 2$ (Liang et al., 2008), null model excluded
NIMS	Jeffreys prior on the non-invariant model
LASSO	Lasso (Tibshirani, 1996)
DZ	The Dantzig Selector (Candes and Tao, 2007)
ENET	The elastic-net (Zou and Hastie, 2005)

Table 4.1: Accronyms and description for the variable selection methods compared in the numerical experiment. (The blocks separate the methods by their natures.

A global remark about this coparison is that all Bayesian procedures have a very similar MSE and thus that they all correspond to the same regularization effect, except for OVS which does systematically worse. However it is important to notice that the MSE for OVS has not been computed by model averaging, but by using the best model. Otherwise, it would be hazardous to recommend one of the priors from those simulations since there is no sensitive difference between them from both selection and prediction points of view.

Translating the data Since NIMS is not location invariant, it is important to measure the impact of adding a constant to all observations. As stressed by a reviewer, when this constant goes to infinity, keeping n fixed, the last argument of ${}_2F_1$ in (4.3) goes to one for all models. Thus if the empirical mean is large relative to the regression sum of squares, the data end up having little input in distinguishing between models. In order to measure this possible negative impact of adding a large constant, we replace in Example 1 $\mathbf{y}$ by $\mathbf{y} = \mathbf{y} + 10^k$ RSS (Regression Sum of Squares) for k $\in \{1, 2, 3\}$. The results derived from NIMS criterion are summarized in Tables 4.14 and 4.15: as predicted, the NIMS criterion tends to choose the null model as kk increases and the null model with no variable is always selected when $k = 3$. Therefore some prior assumption must be made about the magnitude of the intercept when using NIMS. Otherwise, the criterion is over-parsimonious. If this is a possible case, we suggest using instead the HG-2 approach.

	MSE_y	HITS	FP
ORACLE	1.24(0.02)	4.00(0.00)	0.00(0.00)
AIC	1.75(0.08)	3.94(0.02)	2.78(0.17)
BIC	1.69(0.08)	3.90(0.03)	2.29(0.17)
BRIC	1.43(0.04)	3.75(0.05)	0.65(0.09)
EB-L	1.46(0.04)	3.80(0.04)	0.66(0.09)
EB-G	1.45(0.04)	3.78(0.04)	0.65(0.09)
ZS-N	1.44(0.03)	3.78(0.04)	0.65(0.09)
ZS-F	1.49(0.03)	3.90(0.03)	1.73(0.14)
OVS	1.52(0.06)	3.63(0.06)	0.54(0.09)
HG-3	1.49(0.04)	3.75(0.05)	0.55(0.09)
HG-4	1.57(0.04)	3.65(0.05)	0.54(0.08)
HG-2	1.50(0.04)	3.75(0.05)	0.59(0.09)
NIMS	1.45(0.03)	3.75(0.05)	0.57(0.08)
LASSO	1.67(0.05)	3.89(0.03)	2.68(0.20)
DZ	1.66(0.06)	3.72(0.07)	2.41(0.15)
ENET	1.72(0.05)	3.89(0.04)	2.79(0.29)

Table 4.2: Example 1: Mean of MSE, HITS and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8	9	10
AIC	0.47	0.95	1.00	0.45	0.44	0.99	1.00	0.46	0.52	0.44
BIC	0.41	0.91	1.00	0.38	0.40	0.99	1.00	0.32	0.44	0.34
BRIC	0.18	0.77	1.00	0.10	0.11	0.99	0.99	0.07	0.10	0.09
EB-L	0.17	0.81	1.00	0.11	0.11	0.99	1.00	0.07	0.11	0.09
EB-G	0.17	0.79	1.00	0.11	0.11	0.99	1.00	0.07	0.10	0.09
ZS-N	0.17	0.79	1.00	0.11	0.11	0.99	1.00	0.07	0.10	0.09
ZS-F	0.34	0.90	1.00	0.29	0.33	1.00	1.00	0.20	0.33	0.24
OVS	0.14	0.72	0.98	0.07	0.08	0.97	0.96	0.08	0.10	0.07
HG-3	0.17	0.77	1.00	0.11	0.10	0.99	0.99	0.07	0.09	0.08
HG-4	0.15	0.77	1.00	0.10	0.08	0.99	0.99	0.07	0.08	0.07
HG-2	0.10	0.83	0.99	0.07	0.16	0.98	0.95	0.13	0.06	0.07
NIMS	0.15	0.77	1.00	0.09	0.09	0.99	0.99	0.06	0.10	0.08
LASSO	0.49	0.91	1.00	0.41	0.45	0.98	1.00	0.49	0.47	0.37
DZ	0.42	0.84	0.96	0.41	0.47	0.97	0.95	0.38	0.37	0.36
ENET	0.45	0.93	1.00	0.45	0.43	0.99	0.97	0.52	0.44	0.50

Table 4.3: Example 1: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	HITS	FP
ORACLE	1.19(0.01)	4.00(0.00)	0.00(0.00)
AIC	1.81(0.06)	3.12(0.08)	2.75(0.16)
BIC	1.76(0.05)	2.97(0.09)	2.39(0.16)
BRIC	1.46(0.02)	2.44(0.10)	0.99(0.10)
EB-L	1.45(0.02)	2.43(0.10)	1.03(0.10)
EB-G	1.45(0.02)	2.42(0.10)	0.95(0.10)
ZS-N	1.45(0.02)	2.43(0.10)	1.03(0.10)
ZS-F	1.42(0.02)	2.97(0.08)	2.18(0.10)
OVS	1.71(0.04)	2.16(0.11)	1.09(0.09)
HG-3	1.45(0.02)	2.32(0.11)	0.96(0.10)
HG-4	1.45(0.02)	2.35(0.10)	0.86(0.09)
HG-2	1.52(0.04)	2.35(0.10)	0.81(0.09)
NIMS	1.45(0.02)	2.42(0.10)	0.96(0.09)
LASSO	1.66(0.05)	3.35(0.09)	2.95(0.15)
DZ	1.59(0.03)	2.83(0.09)	2.23(0.10)
ENET	1.50(0.03)	3.70(0.07)	4.36(0.17)

Table 4.4: Example 2: Mean of MSE, HITS and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8	9	10
AIC	0.46	0.79	0.88	0.44	0.46	0.78	0.67	0.52	0.48	0.39
BIC	0.41	0.71	0.86	0.43	0.33	0.77	0.63	0.42	0.45	0.35
BRIC	0.21	0.60	0.80	0.17	0.13	0.65	0.39	0.18	0.18	0.12
EB-L	0.22	0.59	0.80	0.17	0.14	0.66	0.38	0.19	0.19	0.12
EB-G	0.21	0.59	0.81	0.16	0.13	0.65	0.37	0.19	0.16	0.10
ZS-N	0.22	0.59	0.80	0.17	0.14	0.66	0.38	0.19	0.19	0.12
ZS-F	0.40	0.72	0.84	0.37	0.31	0.79	0.62	0.38	0.41	0.31
OVS	0.23	0.44	0.74	0.17	0.23	0.62	0.36	0.19	0.18	0.09
HG-3	0.21	0.54	0.80	0.16	0.13	0.63	0.35	0.18	0.18	0.10
HG-4	0.18	0.56	0.81	0.15	0.11	0.63	0.35	0.17	0.17	0.08
HG-2	0.22	0.60	0.78	0.16	0.13	0.59	0.42	0.10	0.15	0.11
NIMS	0.19	0.59	0.80	0.16	0.14	0.66	0.37	0.19	0.18	0.10
LASSO	0.47	0.77	0.90	0.53	0.40	0.89	0.79	0.57	0.55	0.43
DZ	0.40	0.65	0.79	0.46	0.37	0.76	0.63	0.32	0.36	0.32
ENET	0.68	0.85	0.97	0.74	0.74	0.96	0.92	0.76	0.75	0.69

Table 4.5: Example 2: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	HITS	FP
ORACLE	3.31(0.03)	3.00(0.00)	0.00(0.00)
AIC	4.32(0.09)	2.11(0.07)	2.06(0.14)
BIC	4.24(0.08)	1.97(0.07)	1.68(0.14)
BRIC	4.07(0.07)	1.66(0.07)	0.53(0.08)
EB-L	4.06(0.06)	1.84(0.07)	0.79(0.09)
EB-G	4.07(0.07)	1.88(0.07)	0.83(0.09)
ZS-N	4.01(0.06)	1.81(0.07)	0.76(0.09)
ZS-F	4.04(0.07)	2.10(0.07)	1.26(0.11)
OVS	4.27(0.09)	1.78(0.07)	0.64(0.09)
HG-3	4.05(0.06)	1.81(0.07)	0.77(0.09)
HG-4	4.08(0.06)	1.84(0.07)	0.78(0.09)
HG-2	3.98(0.05)	1.80(0.08)	0.73(0.10)
NIMS	3.99(0.06)	1.83(0.07)	0.77(0.09)
LASSO	4.03(0.06)	2.33(0.07)	1.61(0.16)
DZ	4.32(0.10)	2.20(0.11)	2.06(0.16)
ENET	4.13(0.06)	2.38(0.06)	2.04(0.16)

Table 4.6: Example 3: Mean of MSE, HITS and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8
AIC	0.89	0.52	0.45	0.43	0.70	0.36	0.42	0.40
BIC	0.89	0.44	0.39	0.36	0.64	0.30	0.33	0.30
BRIC	0.82	0.35	0.09	0.13	0.49	0.12	0.08	0.11
EB-L	0.87	0.38	0.13	0.19	0.59	0.18	0.14	0.15
EB-G	0.89	0.39	0.15	0.20	0.60	0.18	0.14	0.16
ZS-N	0.87	0.37	0.13	0.19	0.57	0.16	0.13	0.15
ZS-F	0.92	0.51	0.23	0.34	0.67	0.22	0.24	0.23
OVS	0.86	0.37	0.12	0.14	0.55	0.16	0.08	0.14
HG-3	0.87	0.38	0.13	0.19	0.56	0.16	0.14	0.15
HG-4	0.88	0.38	0.13	0.19	0.58	0.17	0.14	0.15
HG-2	0.80	0.46	0.19	0.17	0.60	0.21	0.12	0.15
NIMS	0.87	0.38	0.12	0.19	0.58	0.17	0.14	0.15
LASSO	0.96	0.70	0.32	0.40	0.67	0.29	0.23	0.37
DZ	0.82	0.71	0.42	0.47	0.67	0.47	0.31	0.39
ENET	0.97	0.71	0.49	0.50	0.70	0.40	0.30	0.35

Table 4.7: Example 3: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	HITS	FP
ORACLE	1.43(0.03)	8.00(0.00)	0.00(0.00)
AIC	1.60(0.03)	6.32(0.11)	0.00(0.00)
BIC	1.64(0.03)	5.99(0.12)	0.00(0.00)
BRIC	1.79(0.04)	4.35(0.11)	0.00(0.00)
EB-L	1.75(0.04)	4.39(0.10)	0.00(0.00)
EB-G	1.76(0.04)	4.34(0.10)	0.00(0.00)
ZS-N	1.74(0.04)	4.38(0.10)	0.00(0.00)
ZS-F	1.62(0.04)	5.37(0.10)	0.00(0.00)
OVS	2.22(0.04)	3.82(0.10)	0.00(0.00)
HG-3	1.76(0.04)	4.32(0.10)	0.00(0.00)
HG-4	1.78(0.03)	4.19(0.09)	0.00(0.00)
HG-2	1.77(0.04)	4.18(0.11)	0.00(0.00)
NIMS	1.75(0.04)	4.39(0.10)	0.00(0.00)
LASSO	1.59(0.04)	7.13(0.12)	0.00(0.00)
DZ	1.56(0.03)	6.82(0.11)	0.00(0.00)
ENET	1.54(0.03)	7.53(0.08)	0.00(0.00)

Table 4.8: Example 4: Mean of MSE, HITS and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8
AIC	0.80	0.81	0.78	0.75	0.76	0.86	0.77	0.79
BIC	0.76	0.76	0.75	0.72	0.68	0.83	0.71	0.78
BRIC	0.45	0.58	0.50	0.65	0.54	0.55	0.48	0.60
EB-L	0.46	0.57	0.52	0.67	0.54	0.54	0.50	0.59
EB-G	0.45	0.57	0.52	0.66	0.54	0.53	0.48	0.59
ZS-N	0.46	0.57	0.52	0.67	0.54	0.54	0.49	0.59
ZS-F	0.62	0.69	0.60	0.78	0.65	0.67	0.62	0.74
OVS	0.38	0.57	0.45	0.64	0.40	0.49	0.44	0.45
HG-3	0.45	0.57	0.51	0.67	0.54	0.53	0.48	0.57
HG-4	0.44	0.57	0.48	0.66	0.51	0.52	0.45	0.56
HG-2	0.53	0.56	0.50	0.50	0.54	0.55	0.53	0.47
NIMS	0.46	0.58	0.51	0.67	0.54	0.54	0.50	0.59
LASSO	0.82	0.90	0.96	0.92	0.85	0.91	0.87	0.90
DZ	0.84	0.85	0.84	0.82	0.83	0.91	0.89	0.84
ENET	0.89	0.93	0.96	0.97	0.96	0.93	0.96	0.93

Table 4.9: Example 4: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	HITS	FP
ORACLE	1.07(0.09)	2.00(0.00)	0.00(0.00)
AIC	1.48(0.05)	1.93(0.02)	2.88(0.19)
BIC	1.39(0.04)	1.94(0.02)	2.04(0.18)
BRIC	1.24(0.02)	1.93(0.02)	0.50(0.09)
EB-L	1.27(0.02)	1.93(0.02)	0.58(0.10)
EB-G	1.27(0.02)	1.93(0.02)	0.60(0.10)
ZS-N	1.26(0.02)	1.93(0.02)	0.57(0.10)
ZS-F	1.33(0.03)	1.94(0.02)	1.84(0.14)
OVS	1.32(0.04)	1.89(0.03)	0.76(0.08)
HG-3	1.28(0.02)	1.93(0.02)	0.53(0.09)
HG-4	1.30(0.02)	1.93(0.02)	0.54(0.09)
HG-2	1.25(0.02)	1.93(0.02)	0.36(0.09)
NIMS	1.22(0.02)	1.93(0.02)	0.57(0.10)
LASSO	1.39(0.03)	1.99(0.01)	2.93(0.21)
DZ	1.36(0.04)	1.91(0.03)	2.70(0.18)
ENET	1.43(0.03)	1.96(0.02)	3.25(0.20)

Table 4.10: Example 5: Mean of MSE, HITS and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8	9
AIC	0.36	0.94	0.47	0.99	0.35	0.36	0.34	0.53	0.47
BIC	0.30	0.94	0.38	1.00	0.26	0.24	0.22	0.35	0.29
BRIC	0.10	0.94	0.09	1.00	0.10	0.03	0.05	0.08	0.05
EB-L	0.10	0.93	0.14	1.00	0.11	0.04	0.05	0.08	0.06
EB-G	0.11	0.93	0.14	1.00	0.11	0.04	0.05	0.08	0.07
ZS-N	0.10	0.93	0.13	1.00	0.11	0.04	0.05	0.08	0.06
ZS-F	0.29	0.94	0.32	1.00	0.23	0.22	0.19	0.31	0.28
OVS	0.16	0.92	0.10	0.97	0.15	0.07	0.09	0.11	0.08
HG-3	0.10	0.93	0.11	1.00	0.11	0.03	0.04	0.08	0.06
HG-4	0.10	0.93	0.12	1.00	0.11	0.03	0.04	0.08	0.06
HG-2	0.08	0.95	0.07	1.00	0.04	0.03	0.02	0.06	0.06
NIMS	0.06	0.97	0.10	1.00	0.11	0.08	0.05	0.08	0.08
LASSO	0.51	0.99	0.35	1.00	0.47	0.38	0.37	0.41	0.44
DZ	0.50	0.93	0.32	0.98	0.42	0.45	0.26	0.32	0.43
ENET	0.52	0.96	0.37	1.00	0.55	0.44	0.43	0.50	0.44

Table 4.11: Example 5: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	FP
ORACLE	1.99(0.01)	0.00(0.00)
AIC	2.80(0.07)	3.16(0.21)
BIC	2.62(0.06)	2.24(0.19)
BRIC	2.19(0.02)	0.59(0.11)
EB-L	2.12(0.02)	2.87(0.15)
EB-G	2.11(0.02)	1.54(0.19)
ZS-N	2.26(0.02)	1.02(0.17)
ZS-F	2.31(0.03)	2.51(0.17)
OVS	2.57(0.06)	2.10(0.17)
HG-3	2.13(0.02)	2.18(0.18)
HG-4	2.10(0.01)	2.54(0.17)
HG-2	2.16(0.02)	2.17(0.15)
NIMS	2.24(0.02)	0.99(0.13)
LASSO	2.19(0.04)	1.79(0.22)
DZ	2.57(0.05)	2.49(0.20)
ENET	2.20(0.04)	2.23(0.23)

Table 4.12: Example 6: Mean of MSE and FP. The numbers between parentheses are the corresponding standard errors.

Variables	1	2	3	4	5	6	7	8
AIC	0.38	0.36	0.31	0.37	0.49	0.42	0.41	0.42
BIC	0.26	0.22	0.23	0.26	0.31	0.36	0.33	0.27
BRIC	0.09	0.04	0.07	0.08	0.08	0.09	0.09	0.05
EB-L	0.37	0.27	0.28	0.30	0.43	0.43	0.38	0.41
EB-G	0.19	0.12	0.16	0.16	0.21	0.27	0.25	0.18
ZS-N	0.14	0.07	0.11	0.10	0.16	0.16	0.18	0.10
ZS-F	0.29	0.27	0.23	0.28	0.41	0.38	0.34	0.31
OVS	0.26	0.26	0.36	0.23	0.28	0.26	0.28	0.17
HG-3	0.27	0.21	0.20	0.26	0.32	0.35	0.30	0.27
HG-4	0.32	0.25	0.23	0.29	0.40	0.38	0.35	0.32
HG-2	0.25	0.19	0.23	0.25	0.31	0.35	0.32	0.27
NIMS	0.12	0.06	0.10	0.11	0.14	0.17	0.18	0.11
LASSO	0.22	0.17	0.23	0.22	0.24	0.25	0.29	0.17
DZ	0.23	0.30	0.17	0.20	0.30	0.27	0.25	0.23
ENET	0.30	0.26	0.27	0.25	0.28	0.28	0.33	0.26

Table 4.13: Example 6: Relative frequencies of the selected variables for methods under comparison.

	MSE_y	HITS	FP
$\mathbf{y} = \mathbf{y} + 10 \times RSS$	3.41(0.03)	0.15(0.04)	0.00(0.00)
$\mathbf{y} = \mathbf{y} + 10^2 \times RSS$	3.59(0.03)	0.01(0.01)	0.00(0.00)
$\mathbf{y} = \mathbf{y} + 10^3 \times RSS$	3.59(0.02)	0.00(0.00)	0.00(0.00)

Table 4.14: Example 1: Mean of MSE, HITS and FP after replacing $\mathbf{y}$ by $\mathbf{y} = \mathbf{y} + 10^k RSS$ for $k \in \{1, 2, 3\}$. The numbers between parentheses are the corresponding standard errors for the NIMS selection procedure.

Variables	1	2	3	4	5	6	7	8	9	10
$\mathbf{y} = \mathbf{y} + 10 \times RSS$	0.00	0.00	0.09	0.00	0.00	0.05	0.01	0.00	0.00	0.00
$\mathbf{y} = \mathbf{y} + 10^2 \times RSS$	0.00	0.00	0.01	0.00	0.00	0.01	0.00	0.00	0.00	0.00
$\mathbf{y} = \mathbf{y} + 10^3 \times RSS$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Table 4.15: Example 1: Relative frequencies of the selected variables after replacing $\mathbf{y}$ by $\mathbf{y} = \mathbf{y} + 10^k RSS$ for $k \in \{1, 2, 3\}$.

4.4.3 Real datasets

Two datasets considered in this section are associated with a moderate number of variables against the number of observations.

Body fat dataset The body fat dataset has been first used by Penrose and Fisher (1985). The corresponding study aims at estimating the percentage of body fat from various body circumference measurements observed on 252 men. The thirteen regressor variables are:

1. age,
2. weight (lbs),
3. height (inches),
4. neck circumference,
5. chest circumference,
6. abdomen 2 circumference,
7. hip circumference,
8. thigh circumference,
9. knee circumference,
10. ankle circumference,
11. biceps (extended) circumference,

12. forearm circumference,

13. wrist circumference.

In order to investigate the performances of the different methods, a dataset from Penrose and Fisher (1985) has been split 25 times into a training set of 151 observations and a test set of 101 observations. Tuning parameters for the frequentist regularization methods have been chosen by minimizing the (ten fold) cross-validated prediction error.

For this dataset, the Bayesian procedures we investigated are much more parsimonious than the standard regularization procedures, as shown in Table 4.16. There is no variability in the prediction MSE. (We stress that MSEs are computed by model averaging for the Bayesian procedures.) As in the simulation experiment, all Bayesian approaches are highly similar, except for ZS-F which remains more open to incorporating the last two covariates.

Ozone data This second benchmark dataset is taken from Breiman and Friedman (1985) and consists in daily measurements of the maximum ozone concentration and of eight meteorological variables near Los Angeles. Those variables are:

1. the daily ozone concentration (maximum one hour average, parts per million) at Upland, CA which is the response variable;

2. the Vandenburg 500 millibar pressure height (m);

3. the wind speed (mph) at Los Angeles International Airport (LAX);

4. the humidity (percent) at LAX;

5. the Sandburg Air Force Base temperature (F^o);

6. the inversion base height at LAX;

7. the inversion base temperature at LAX;

8. the Daggett Pressure gradient (mm Hg) from LAX to Daggett, CA;

9. the visibility (miles) at LAX.

The original Ozone database contains 366 observations, of which 203 are complete. Our study is made just on the complete observations. We split this dataset 25 times into a training set of 101 observations and a test set of 102 observations.

For this dataset, as shown by Table 4.19, all Bayesian approaches, as well as AIC and BIC, select about three variables, while the regularization methods opt for five. The MSE differences between all procedures are negligible.

4.5 Conclusion

In this numerical study, we have compared Bayesian variable selection methods with regularisation methods in a poorly informative setting. From a variable selection point of view, it appears that the Bayesian methods are more parsimonious and more relevant than the regularisation methods. From a predictive point of view, there is no significant difference between both approaches. Regularisation methods could however be expected to perform better from this latter point of view since they minimize a cross-validated prediction error. But, owing to model averaging, efficiency, Bayesian methods provide competitive MSE's.

An additional appeal of this study is to single-out and to assess two calibration-free prior models (NIMS and HG-2). They both appear as valuable competitors when compared with earlier Bayesian approaches. However, both methods have a clear drawback (NIMIS is not location invariant and HG-2 excludes the null model). Nonetheless our series of examples shows that they provide an acceptable objective Bayesian solution for Bayesian variable selection and regularization in linear models.

A limitation of this study on our objective Bayesian approach is that we do not consider large dimensions as in Bottolo and Richardson (2010), which require different computational tools to face the enormous number of potential models. This difficulty is obviously faced by all Bayesian solutions considered in this paper and is not an issue in terms of the validity of the prior modelling.

	MSE_y	Mean of selected variables
AIC	4.58(0.05)	5.56(0.20)
BIC	4.60(0.05)	4.20(0.18)
BRIC	4.51(0.05)	2.84(0.15)
EB-L	4.52(0.05)	3.00(0.18)
EB-G	4.52(0.05)	3.28(0.17)
ZS-N	4.52(0.05)	2.96(0.18)
ZS-F	4.49(0.05)	4.28(0.20)
OVS	4.65(0.07)	2.96(0.18)
HG-3	4.54(0.05)	3.00(0.18)
HG-4	4.56(0.05)	3.24(0.17)
HG-2	4.50(0.05)	2.48(0.14)
NIMS	4.50(0.05)	2.44(0.14)
LASSO	4.54(0.05)	8.17(0.52)
DZ	4.51(0.06)	11.03(0.11)
ENET	4.54(0.05)	9.04(0.56)

Table 4.16: Body fat dataset: Mean of the MSE_y and of the selected variables.

Variables	1	2	3	4	5	6	7	8	9	10	11	12	13
AIC	0.44	0.84	0.16	0.64	0.04	1.00	0.20	0.16	0.08	0.16	0.44	0.80	0.88
BIC	0.08	0.84	0.08	0.32	0.00	1.00	0.12	0.08	0.04	0.00	0.16	0.28	0.40
BRIC	0.08	0.84	0.08	0.32	0.00	1.00	0.12	0.08	0.04	0.00	0.16	0.24	0.40
EB-L	0.08	0.84	0.08	0.32	0.00	1.00	0.12	0.08	0.04	0.00	0.16	0.28	0.40
EB-G	0.08	0.88	0.08	0.36	0.00	1.00	0.08	0.08	0.04	0.00	0.20	0.36	0.40
ZS-N	0.08	0.84	0.08	0.32	0.00	1.00	0.12	0.08	0.04	0.00	0.16	0.24	0.40
ZS-F	0.20	0.84	0.12	0.40	0.00	1.00	0.12	0.12	0.08	0.04	0.24	0.60	0.68
OVS	0.12	0.68	0.08	0.16	0.04	1.00	0.08	0.00	0.00	0.00	0.04	0.24	0.52
HG-3	0.08	0.84	0.08	0.32	0.00	1.00	0.12	0.08	0.04	0.00	0.16	0.28	0.40
HG-4	0.08	0.88	0.08	0.32	0.00	1.00	0.08	0.08	0.04	0.00	0.16	0.36	0.40
HG-2	0.04	0.88	0.00	0.08	0.00	1.00	0.08	0.04	0.00	0.04	0.16	0.28	0.60
NIMS	0.04	0.88	0.04	0.08	0.00	1.00	0.04	0.08	0.04	0.00	0.04	0.04	0.12
LASSO	1.00	0.28	1.00	0.88	0.24	1.00	0.44	0.52	0.28	0.56	0.68	0.84	1.00
DZ	1.00	0.80	1.00	0.88	0.60	1.00	0.80	0.72	0.40	0.88	0.92	0.88	0.96
ENET	1.00	0.40	1.00	0.80	0.28	1.00	0.40	0.64	0.44	0.64	0.68	0.84	1.00

Table 4.17: Body fat dataset: relative frequencies of selections of the variables over the 25 random splits+.

	MSE_y	Mean number of selected variables
AIC	4.79(0.05)	3.52(0.14)
BIC	4.77(0.05)	2.88(0.07)
BRIC	4.78(0.05)	2.88(0.07)
EB-L	4.78(0.05)	2.88(0.07)
EB-G	4.78(0.05)	2.92(0.05)
ZS-N	4.78(0.05)	2.88(0.07)
ZS-F	4.77(0.05)	3.12(0.07)
OVS	4.81(0.05)	2.88(0.10)
HG-3	4.78(0.05)	2.88(0.07)
HG-4	4.78(0.05)	2.92(0.05)
HG-2	4.80(0.05)	2.68(0.10)
NIMS	4.79(0.05)	2.68(0.10)
LASSO	4.78(0.05)	5.24(0.21)
DZ	4.80(0.05)	5.12(0.13)
ENET	4.79(0.05)	5.32(0.16)

Table 4.18: Ozone dataset: Mean of the MSE_y and of the selected variables.

Variables	1	2	3	4	5	6	7	8
AIC	0.20	0.12	0.96	1.00	0.56	0.08	0.44	0.16
BIC	0.04	0.00	0.96	1.00	0.60	0.00	0.36	0.04
BRIC	0.04	0.00	0.96	1.00	0.60	0.00	0.40	0.04
EB-L	0.04	0.00	0.96	1.00	0.60	0.40	0.36	0.04
EB-G	0.04	0.00	0.96	1.00	0.60	0.00	0.36	0.04
ZS-N	0.04	0.00	0.96	1.00	0.60	0.00	0.36	0.04
ZS-F	0.04	0.08	0.92	1.00	0.60	0.00	0.40	0.08
OVS	0.00	0.00	1.00	0.92	0.00	0.00	0.80	0.08
HG-3	0.04	0.00	0.96	1.00	0.60	0.00	0.36	0.04
HG-4	0.04	0.00	0.96	1.00	0.60	0.00	0.36	0.04
HG-2	0.04	0.00	0.96	1.00	0.60	0.00	0.32	0.04
NIMS	0.04	0.00	0.96	1.00	0.60	0.00	0.32	0.04
LASSO	0.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00
DZ	0.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00
ENET	0.00	0.00	1.00	1.00	1.00	0.00	1.00	1.00

Table 4.19: Ozone dataset: relative frequencies of selections of the variables over the 25 random splits.

Chapter 5

Regularization and variable selection using the SF-MCP

Résumé : Dans ce chapitre, nous avons proposé la méthode Fused-MCP pour résoudre le problème du biais du Lasso, surtout dans des situations où nous avons un certain ordre sur les variables successives. Nous rencontrons ce genre de données en spectroscopie de masse par exemple. La méthode est fondée sur la combinaison de deux pénalités : la première est la pénalité MCP (Zhang, 2010) qui a l'avantage de pénaliser différemment les petits et les grands coefficients de régression, tandis que la deuxième pénalité est une norme l_2 de la différence entre les coefficients successifs du vecteur de régression.

5.1 Introduction

Consider a linear regression model

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta}^* + \boldsymbol{\varepsilon}, \tag{5.1}$$

where $\mathbf{y}$ is the variable response, $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_p)$ is a nxp matrix of n subjects described by p predictors, $\boldsymbol{\varepsilon}$ is a random error vector with $\mathbb{E}(\boldsymbol{\varepsilon}) = 0$ and $\boldsymbol{\beta}^*$ is a vector of unknown parameters which are to be estimated. Using suitable location and scale transformations we can assume without loss of generality that the response is centered and the predictors are centered and standardized. However, all numerical results are presented in the original scale of the data.

During the latest few years a great deal of attention has been focused on the penalized least squares methods which perform estimation and variable selection in a continuous fashion by shrinking regression coefficients towards zero and in addition of setting some coefficients exactly equal to zero. The most popular regularization approach is the least absolute shrinkage and selection operator Lasso Tibshirani (1996) which is based on the penalized least squares by the ℓ_1 penalty on the vector parameter. This method was made particularly appealing by the advent of the Lars algorithm Efron et al. (2004) which

provided a highly efficient means to simultaneously produce the set of Lasso fits for all values of the tuning parameter. Despite its good properties, the shrinkage introduced by the Lasso results in significant bias towards 0 for large regression coefficients.

Some recent alternative nonconvex penalties are SCAD Fan and Li (2001) and MCP Zhang (2010), which have been proposed to overcome the Lasso bias. SCAD and MCP enjoy the oracle property, that is, the SCAD and MCP estimators can perform as well as the oracle if the penalization parameter is appropriately chosen. But, these two nonconvex penalties present numerical difficulties in the computation procedure for the estimation of the regression coefficients and particularly in high dimension settings. However, recent coordinate descent algorithms Mazumder et al. (2011), Wu and Lang (2007), Friedman et al. (2007) for fitting Lasso-type penalized regressions have been shown to be competitive to Lars algorithm. Moreover, Breheny and Huang (2011) provided Code R of these algorithms to nonconvex penalty SCAD and MCP and established its theoretical convergence properties.

In addition, some kinds of data can have structure in the predictor variables. For example in microarray data analysis, the genes have a grouping structure and it is thus desirable to simultaneously select and group genes. On the other hand in proteomics setting, the measurements based on intensity at successive wavelengths of light, may be ordered in some meaningful way. Many methods has been proposed to deal with the first situation of grouping effect. The most popular among is the elastic net (Enet), proposed by Zou and Hastie (2005), which uses both the Ridge and Lasso penalties. For the second problem of the structure in coefficients, Hebiri and van De Geer (2010) considered the Smooth-Lasso procedure (SLasso), a modification of the Fused-Lasso procedure Tibshirani et al. (2005), in which a second ℓ_1 Fused penalty is replaced by the smooth ℓ_2 norm penalty. We have found in our empirical studies that SLasso provided good results even in situations where it is not favorable, so confirming the excellent results obtained by Hebiri and van De Geer (2010). However, all these methods inherit the first drawback of Lasso, because of the presence of the ℓ_1 penalty. So, they cannot achieve selection consistency and estimation efficiency simultaneously. Recently, Huang et al. (2010) proposed a new method, called Mnet, which combines the MCP and Ridge penalties. So it is applicable to $p \gg n$ regression problems with highly correlated predictors and have the oracle property.

Inspired by SLasso and Mnet approaches, we propose a new penalized method that uses the combination of the MCP and Smooth-Fused penalties. We call it the Smooth-Fused Mcp (SF-Mcp). So, SF-Mcp will conserve the advantages of both Mnet and SLasso: it is adapted to high dimension setting $p \gg n$, overcomes the grouping effect of Lasso, takes into account some structure the regression coefficients, the estimation of the coefficients can be computed efficiently via a coordinate descent algorithm and will certainly achieve selection consistency and estimation efficiency simultaneously. However, we limit our attention in this note to present the SF-Mcp with its basic characteristics and to describe a coordinate descent algorithm for computing the regression coefficients estimates.

In Section 5.2 we give a brief description of the Slasso and Mnet methods. Section 5.3 presents our SF-Mcp method and describes the computation of the solutions of the corresponding algorithm. A detailed simulation study and application to real data set are

performed in Section 5.4.

5.2 SLasso and MNet methods

In this section, we give a brief description of the two recent penalized least squares methods which are based on the modification of the penalty of the Elastic net approach.

Despite its popularity, Enet has been critiqued for being inadequate, notably in situations in which additional structural knowledge about predictors should be taken into account (cf. El Anbari and Mkhadri 2008, Daye and Jeng (2009), Hebiri and van De Geer (2010). To this end, these authors complement ℓ_1-regularized with a second regularized based on the total variation or a quadratic penalty. For example, Hebiri and van De Geer (2010) proposed the Smooth Lasso approach (SLasso) based on a modification of Enet in which the ℓ_2 penalty on successive differences between coefficients replace the ℓ_2 penalty on regression coefficients, it is defined by

$$\hat{\boldsymbol{\beta}}_{SLasso}(\lambda_1,\lambda_2) = \arg\min_{\boldsymbol{\beta}} ||\mathbf{y} - X\boldsymbol{\beta}||_2^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=2}^{p} (\beta_j - \beta_{j-1})^2,$$

where λ_1 and λ_2 are two positive tuning parameters. This simple modification seems to work better than Enet in different experiments and even in situations in which it is not favorable (cf. Hebiri and van De Geer 2010).

On the other hand, Enet inherits the Lasso bias because of the presence of the ℓ_1 penalty. To overcome this problem, Huang et al, (2010) proposed recently the Mnet method which combines the MCP and Ridge penalties, it is defined by

$$\hat{\boldsymbol{\beta}}_{Mnet}(\lambda_1,\lambda_2,\gamma) = \arg\min_{\boldsymbol{\beta}} \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \sum_{j=1}^{p} p_{\lambda_1,\gamma}(|\beta_j|) + \lambda_2 \sum_{j=1}^{p} \beta_j^2,$$

where, $p_{\lambda_1,\gamma}$ is the MCP penalty defined by

$$p_{\lambda_1,\gamma}(z) = \begin{cases} \lambda_1 z - \frac{z^2}{2\gamma}, & \text{if } z \leq \gamma\lambda_1 \\ \frac{1}{2}\gamma\lambda_1^2, & \text{if } z > \gamma\lambda_1, \end{cases} \tag{5.2}$$

λ_1, λ_2 is a pair of penalty parameter and γ is a regularization parameter. The MCP penalty can be easily understood by considering its derivative $\dot{p}_{\lambda_1,\gamma}(z) = \lambda_1(1 - |z|/(\gamma\lambda_1))_+ \text{sgn}(z)$ where $\text{sgn}(z) = -1, 0$ or 1 if $z < 0, = 0$ or > 0. It begins by applying the same rate of penalization as Lasso, but continuously relaxes that penalization, until $|z| > \gamma\lambda_1$, until the rate of penalization drops to zeo. In the literature, the penalty $p_{\lambda_1,\gamma}$ produces estimates with basic properties such that: unbiasedness, sparsity and continuity (Fan and Li, 2001). On the other hand, Huang et al. (2010) showed that the Mnet is selection consistent and equat to the oracle ridge estimator with high probabiliy.

5.3 The SF-Mcp method

Despite its good empirical results, SLasso inherits the first drawback of Lasso, because of the presence of the ℓ_1 penalty. On the other hand, the Mnet can be critiqued for being inadequate, notably in situations in which additional structural knowledge about predictors should be taken into account: explicit inclusion of structural knowledge about predictors and some type of correlation between predictors, for example. Moreover, a close inspection to the Mnet'th experimental results revels that Enet and Mnet are comparable in term of prediction accuracy, while Mnet has better performance in variable selection in the presence of highly correlated predictors.

Therefore, there is a need to develop methods that take into account of additional structural information of predictors and have the oracle property. In the same spirit of Mnet, we propose the SF-Mcp procedure which is based on the penalized least squares with the penalty which is a combination of the MCP and the Smooth Fused penalties. The SF-Mcp estimator is defined as solution to the following optimization problem.

$$\hat{\boldsymbol{\beta}}(\lambda_1, \lambda_2, \gamma) = \arg\min_{\boldsymbol{\beta}} \left\{ \|\mathbf{y} - \mathbf{X}\beta\|_2^2 + \sum_{j=1}^{p} p_{\lambda_1,\gamma}(|\beta_j|) + \lambda_2 \sum_{j=2}^{p-1} (\beta_j - \beta_{j-1})^2 \right\}, \tag{5.3}$$

where, $p_{\lambda_1,\gamma}$ is given in (5.2), λ_1 and λ_2 are tuning parameters and γ is a positive regularization parameter. Thus, the SF-Mcp is a doubly regularized technique combining the MCP penalty and and ℓ_2 penalty on successive differences between coefficients. Consequently, it achieves both goals of handling the problem of collinearity in high dimension, takes into account of structural information of predictors and enjoys the oracle property.

To find a solution of the penalized least squares problem (5.3), we transform this problem into an equivalent MCP problem on an augmented data

$$\tilde{\mathbf{X}}_{(n+p)\times p} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{L}^t \end{pmatrix}, \quad \tilde{\mathbf{y}}_{(n+p)} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix} \tag{5.4}$$

where,

$$\mathbf{L} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 1 & -1 & 0 & \ddots & \vdots \\ 0 & 1 & -1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 1 & -1 \end{pmatrix} \tag{5.5}$$

Using simple algebra we can see that

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}\|_2^2 + \sum_{j=1}^{p} p_{\lambda_1,\gamma}(|\beta_j|).$$

The latter result shows that the SF-Mcp can be seen as a MCP problem on the augmented data (5.4), so the SF-Mcp estimates can be computed easily via the coordinate descent algorithm. The computation details are given in Algorithm 3:

Algorithme 3. (The Coordinate Descent Algorithm for SF-Mcp).

1. Inputs: The response vector $\mathbf{y}$ and the covariates matrix $\mathbf{X}$.
2. Define the augmented data
$$\tilde{\mathbf{X}}_{(n+p)\times p} = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda_2}\mathbf{L}^t \end{pmatrix}, \quad \tilde{\mathbf{y}}_{(n+p)} = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}.$$
3. Output
$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \|\tilde{\mathbf{y}} - \tilde{\mathbf{X}}\boldsymbol{\beta}\|_2^2 + \sum_{j=1}^{p} p_{\lambda_1,\gamma}(|\beta_j|).$$

5.4 Numerical experiments

We examine the performance of SF-Mcp in comparaison with Lasso, Enet, Slasso and Mnet through four simulation examples and an application to real data set.

5.4.1 Simulation study

A simulation study was run to examine the performance of the SF-Mcp in four situations, with the Lasso, the elastic net (Enet), the Smooth Lasso (Slasso) and the Mnet. For each example, 100 data sets were simulated from the regression model (5.1) consisting of a training set to fit the model, an independent validation set to select tuning parameters and an independent testing set to evaluate prediction. The notation ././. denotes the number of observations in the training, validation and test data sets respectively. For each method and each simulation we compute the prediction accuracy $\text{MSE}_{\mathbf{y}}$ (Mean square error of the response), Hits and FP (respectively the number of correctly identified influential variables and the number of non-influential variables declared as influential) over 100 data sets.

1. In this example which is considered in Zou and Hastie (2005), we simulate 50/50/400 observations with 40 predictors. The true regression parameter is such that $\beta = (\underbrace{3,...,3}_{15},\underbrace{0,...,0}_{25})^t$. Let $\sigma = 6$, the predictors were generated as follows: $\mathbf{x}_i = Z_1 + \varepsilon_i^x, Z_1 \sim N(0,1)$ for $i = 1,...,5$, $\mathbf{x}_i = Z_2 + \varepsilon_i^x, Z_2 \sim N(0,1)$ for $i = 6,...,10$, $\mathbf{x}_i = Z_3 + \varepsilon_i^x, Z_3 \sim N(0,1)$,for $i = 11,...,15$, ε_i^x are (i.i.d.) $N(0,0.16)$ for $i = 1,...,15$ and $\mathbf{x}_i \sim N(0,1)$ for $i = 16,...,40$. In this model of three equally important groups, the correlation ($\rho \approx 0.85$) between relevant predictors is high.
2. In this example taken from Wang et al. (2010), we simulate 50/50/400 observations with 40 predictors. The true regression parameter is such that:
$$\beta = (\underbrace{3,...,3}_{5},\underbrace{-2,...,-2}_{5},\underbrace{0,...,0}_{30})^t.$$

Let $\sigma = 3$, the predictors were generated as follows: $\mathbf{x}_i = 3Z + \varepsilon_i^x$ and $Z \sim N(0,1)$ for $i = 1, ..., 10$, $\mathbf{x}_i \sim N(0,1)$ for $i = 11, ..., 40$, where ε_i^x are independent identically distributed (i.i.d.) $N(0,1), i = 1, ..., 10$. The correlation between each pair of the first 10 variables is ($\rho \approx 0.9$). The remaining 30 variables are independent with each other, and also independent with the first 10 variables. So, we are in sparse situation with two grouped predictors with different signs.

3. This example is also considered by Wang et a. (2010) where we simulate 50/50/400 observations with 40 predictors. The true regression parameter is such that

$$\beta = (3, 3, -2, 3, 3, -2, \underbrace{0, ..., 0}_{34})^t.$$

Let $\sigma = 6$, the predictors were generated as follows: $\mathbf{x}_i = 3Z + \varepsilon_i^x$ and $Z \sim N(0,1)$ for $i = 1, ..., 6$, $\mathbf{x}_i \sim N(0,1)$ for $i = 7, ..., 40$ and ε_i^x are (i.i.d.) $N(0,1), i = 1, ..., 10$. Moreover, for $i = 7, \ldots, 40$, the random variables X_i's are i.i.d $N(0,1)$. The correlation between each pair of the first 6 variables is ($\rho \approx 0.9$). The remaining 34 variables are independent with each other, and also independent with the first 6 variables. So, we are in sparse situations of equally grouped predictors with different signs.

4. The last example, considered by Wu et al. (2009), corresponds to high dimensional simulation scenarios correlated groups with large $p = 200$ and small n. The X_i are simulated from $N(0, \Sigma)$ where the jk-th element of Σ is $0.5^{|j-k|}$ and $\boldsymbol{\beta} = (3\mathbf{1}_5, -1.5\mathbf{1}_5, \mathbf{1}_5, 2\mathbf{1}_5, \mathbf{0}_{180})^t$. So there are only 20 grouped relevant predictors and 180 noisy predictors. The notation $3\mathbf{1}_5$ means the value 3 is repeated 5 times.

As can be seen from Table 5.1, SF-Mcp outperforms all methods in term of prediction accuracy (median mean-squared errors). It is followed by SLasso in the first three examples and Mnet in the last example. In term of variable selection, SF-Mcp performs better in all situations than all other methods. It has a tendency to include all relevant predictors with large Hits value and seems to often exclude more noised predictors than all other methods (except in Example 4 where Mnet has slightly less FP followed by SF-Mcp).

5.4.2 Real dataset

Body fat data This dataset has been first used by the Nelson and Fisher (1985). The corresponding study aims at estimating the percentage of body fat from various body circumference measurements observed on 252 men described by thirteen regressor variables. The thirteen regressors are age (1), weight (lbs) (2), height (inches) (3), neck circumference (4), chest circumference (5), abdomen 2 circumference (6), hip circumference (7), thigh circumference (8), knee circumference (9), ankle circumference (10), biceps (extended) circum- ference (11), forearm circumference (12), and wrist circumference (13).

In order to investigate the performances of the different methods, the dataset has been split 25 times into a training set of 101 observations and a test set of 151 observations.

Example		MSE_y	HITS	FP
1	Lasso	12.58	09	2
	Enet	4.05	15	1
	Slasso	4.06	15	2
	Mnet	8.05	12	2
	SF-Mcp	2.20	15	0
2	Lasso	20.99	10	15
	Enet	20.79	10	15
	Slasso	5.87	10	08
	Mnet	10.72	10	07
	SF-Mcp	2.59	10	01
3	Lasso	16.39	4	02
	Enet	13.10	4	00
	Slasso	9.20	4	0.5
	Mnet	12.51	4	01
	SF-Mcp	9.11	4	00
4	Lasso	27.11	14	9
	Enet	25.04	14	8
	Slasso	13.53	17	6
	Mnet	6.97	17	1
	SF-Mcp	6.65	18	2

Table 5.1: Median mean-squared errors for the simulated examples of five methods based on 100 replications. The median number of HITS and false positives are also reported.

Tuning parameter have been chosen by minimizing the (ten fold) cross-validated prediction error. As shown in Table 5.2, for this dataset, the SF-Mcp is much more parsimonious than all competitors with a median of 4 variables in the final model. In terms of MSE, SF-Mcp gives the best performance followed by the Slasso.

Pollution data This data set of 15 independent variables and a measure of mortality on 60 US metropolitan areas in 1959-1961. The response variable is the total age adjusted mortality rate (Y). The fifteen regressors are mean annual precipitation in inches (x1), mean January temperature in degrees Fahrenheit (x2), mean July temperature in degrees Fahrenheit (x3), percent of 1960 SMSA population that is 65 years of age or over (x4) population per household, 1960 SMSA (x5), median school years completed for those over 25 in 1960 SMSA (x6), percent of housing units that are found with facilities (x7), population per square mile in urbanized area in 1960 (x8), percent of 1960 urbanized area population that is non-white (x9), percent employment in white-collar occupations in 1960 urbanized area (x10), percent of families with income under 3; 000 in 1960 urbanized area (x11), relative population potential of hydrocarbons, HC, (x12), relative pollution potential of

Method	median $MSE_{\mathbf{y}}$	median no. of selected variables
Lasso	23.40	8
Enet	22.78	9
Slasso	21.98	7
Mnet	22.07	12
SF-Mcp	21.78	4

Table 5.2: Body fat data - median test mean squared error over 25 random splits for different methods.

oxides of nitrogen, NOx (x13), relative pollution potential of sulfur dioxide, SO2 (x14), and percent relative humidity, annual average at 1 p.m (x15)

As for the Body fat data, we randomly split 25 times the Pollution data set into a training set of 20 observations and a test set of 40 observations. Results are summarized in Table 5.3. In terms of prediction accuracy, the best performance is given by the Mnet method, followed by de SF-Mcp. The worst performance is given by the Slasso. In terms of variable selection, the most parsimonious model is given by the Lasso, followed by the Enet. The SF-Mcp produces more parsimonious model than Mnet and Slasso.

Method	median $MSE_{\mathbf{y}}$	median no. of selected variables
Lasso	2742.22	4
Enet	2841.39	6
Slasso	3879.21	11
Mnet	2230.20	9
SF-Mcp	2381.05	7

Table 5.3: Pollution data - median test mean squared error over 25 random splits for different methods.

5.5 conclusion

In this paper, we have proposed the SF-Mcp, a penalized least squares method for regularization and variable selection in the context of linear regression models. We have seen that the proposed estimates can be calculated efficiently for a grid of values of the tuning parameters, using coordinate decent algorithm. It has been demonstrated through simulated and real world data sets, that the proposed method gives best performances in both prediction and variable selection viewpoints. The method have tendency to give best prediction combined with parsimonious models. The SF-Mcp is very useful in situations such

that, the number of regressors is much larger than sample size and/or where successive regression coefficients are known to vary slowly.

The exploration of theoretical properties of the proposed estimator, such as sparsity inequalities and variable selection consistency, is a work in progress.

Chapter 6

Conclusions et Perspectives

6.1 Bilan

Les travaux de cette thèse s'inscrivent dans le cadre de la sélection de variables en régression linéaire. Ce thème était très en vogue depuis la fin des années 90 et il a connu un essor considérable depuis l'année 2000 après le succès du *Lasso* (Tibshirani, 1996) et l'*Elastic net* (Zou and Hastie, 2005), via l'utilisation de l'algorithme LARS (Efron et al., 2004), permettant simultanément l'estimation et la sélection des coefficients de régression. Ces méthodes sont fondées sur la pénalisation de la fonction de vraisemblance par une pénalité de type norme l_1 du vecteur des coefficients de régression ou une combinaison de ses normes l_1 et l_2. L'intérêt de ces méthodes réside, en général, dans une bonne qualité de prédiction combinée avec une représentation parcimonieuse des données, surtout si la dimension des données dépasse largement la taille de l'échantillon.

Notre objectif dans cette thèse a été de proposer des méthodes de sélection de variables dans le cadre de la régression linéaire. Ce problème a été traité selon deux points de vues : fréquentiel et bayésien. D'un point de vue fréquentiel, des extensions type *Elastic net* ont été proposées via l'introduction du coefficient de corrélation ou d'une structure d'ordre entre les coefficients de régression dans la seconde pénalité. D'un point de vue bayésien, une nouvelle approche globale de sélection de variables en régression a été proposée en utilisant de manière judicieuse des lois a priori sur les paramètres du modèle.

Dans le chapitre 2, nous avons proposé une nouvelle méthode de régularisation et de sélection de variables appelée L1CP. Nous avons montré à travers une série d'exemples simulés et à travers des bases de données réelles, que la méthode L1CP améliore les performances de prédiction du *Lasso* et de l'*Elastic net* tout en jouissant de l'avantage de donner des modèles parcimonieux. En outre, la méthode proposée possède un effet de groupement : les variables fortement corrélées ont tendance à être toutes incluses ou toutes exclues du modèle final. La méthode L1CP est aussi très utile, pour des situations dans lesquelles le nombre de variables p est beaucoup plus grand que la taille de l'échantillon n.

Dans le chapitre 3, nous avons proposé l'ensemble d'estimateurs *Adaptive Generalized Ridge-Lasso* (AdaGril), permettant de corriger les défauts théoriques (absence de propriété

Oracle) des estimateurs L1CP, Elastic net, smooth Lasso et d'autres, tout en conservant la propriété d'effet de groupement. La méthode est fondée sur l'introduction d'un poids adaptatif dans la norme l_1 et en considérant une norme l_2 générale. Ainsi l'estimateur L1CP est un cas particulier d'estimateurs AdaGril obtenu en fixant les poids égaux à 1 et en considérant la norme l_2 non ordinaire CP. Une étude théorique détaillée des propriétés statistiques de l'ensemble d'estimateurs AdaGril a été présentée dans le cas où le nombre de variables p est une fonction divergente de la taille d'échantillon n ($p = p(n)$ et $p \to \infty$ lorsque $n \to \infty$).

Le problème de sélection de variables en régression selon le point de vue bayésien a été considéré dans le chapitre 4. Nous avons proposé une approche fondée sur un modèle gaussien avec des lois a priori g de Zellner qui fournissent des formes explicites de la loi a posteriori des modèles candidats. C'est un avantage très important en pratique, car ça évite le recours à des algorithmes de simulation de chaînes de Markov qui peuvent coûter chers en temps de calcul. De plus et contrairement aux méthodes bayésiennes concurrentes, en particulier celle de Liang et al. (2008), la nouvelle approche est indépendante de tout paramètre de contrôle qu'il faut fixer de manière subjective. Une étude numérique très détaillée a été effectuée sur des donneés simulées et des exemples réels dans un cadre peu informatif où le nombre de variables est presque égal à la taille de l'échantillon.

Dans le dernier chapitre, nous avons proposé la méthode Fused-MCP pour résoudre le problème du biais du Lasso, surtout dans des situations où nous avons un certain ordre sur les variables successives. Nous rencontrons ce genre de données en spectroscopie de masse par exemple. La méthode est fondée sur la combinaison de deux pénalités : la première est la pénalité MCP (Zhang, 2010) qui a l'avantage de pénaliser différemment les petits et les grands coefficients de régression, tandis que la deuxième pénalité est une norme l_2 de la différence entre les coefficients successifs du vecteur de régression. Ainsi le problème du biais du Lasso est corrigé d'une façon différente de celle des estimateurs AdaGril sans avoir recours à des poids dépendants d'un estimateur initial convergent.

Nous pouvons donc faire les conclusions suivantes à partir l'ensemble des travaux présentés dans cette thèse :

1. Nous avons confirmé que les techniques de régularisation par la vraisemblance pénalisée étaient un outil statistique très puissant. En effet, cette approche permet d'estimer les coefficients et sélectionner les variables simultanément. Elle permet d'améliorer la qualité de prédiction tout en produisant des modèles parcimonieux et donc facilement interprétables. Nous avons aussi vu, qu'à travers une double pénalisation, cette approche permet de mieux se comporter avec des structures supplémentaires entre les variables (présence de corrélation, ordre entre les variables). Par ailleurs, cette approche est particulièrement utile pour des situations où le nombre de variables est beaucoup plus grand grand que la taille d'échantillon, ce qui est une caractéristique de plusieurs bases de données modernes.

2. Nous avons montré que les méthodes proposées dans le cadre fréquentiel, se ramènent à un problème du type *Lasso* ou un problème du type *MCP* appliqués à des données augmentées. Donc, nous pouvons bénéficier des algorithmes très efficaces développés

pour ces techniques tels que les algorithmes *LARS* ou *Coordinate Descent*, pour retrouver les chemins de régularisation des coefficients de régression pour une grille de valeurs des paramètres de contrôle. Ces deux algorithmes sont rapides et librement disponibles, ce qui rend facile la mise en oeuvre des techniques nouvellement proposées.

3. Nous avons également vu, que nous pouvons faire de la régularisation et de la sélection de variables en régression, en suivant une approche bayésienne. D'après une étude numérique détaillée, nous avons pu constater qu'au niveau sélection, l'approche bayésienne fournit des modèles plus parcimonieux, que l'approche fréquentielle fondée sur la pénalisation de la fonction de vraisemblance et qu'au niveau prédiction, les méthodes bayésiennes fournissent de bonnes performances en faisant du MA (*model averaging*) utilisant tous les modèles.

4. Sur le plan théorique, en introduisant une norme l_1 pondérée par des poids adaptatifs dans le critère de la vraisemblance pénalisée et sous certaines conditions de régularité, nous avons montré que l'ensemble d'estimateurs AdaGril possède la propriété Oracle (au sens de Donoho and Johnstone 1994 et Fan and Li 2001) et des inégalités de parcimonie (Sparsity inequalities) assurant par conséquent une performance optimale en grande dimension.

6.2 Perspectives

Nous terminons ce manuscrit par donner quelques perspectives et prolongements naturels de ce travail de thèse.

1. Un travail en cours, qui est une suite logique des travaux de cette thèse, est l'exploration des propriétés théoriques de l'estimateur Fused-MCP présenté dans le chapitre 5. Nous sommes intéressés à démontrer des inégalités type Oracle (au sens de Donoho and Johnstone 1994 et Fan and Li 2001), la consistance en sélection de variables et la consistance en prédiction de cet estimateur.

2. Dans les techniques de régularisation et de sélection de variables via la vraisemblance pénalisée, la complexité du modèle et le taux de rétrécissement appliqué aux coefficients de régression sont fortement liés au choix des paramètres de régularisation. Il est remarqué que l'utilisation de la validation croisée pour sélectionner les paramètres de régularisation optimaux donne des modèles qui contiennent beaucoup de variables parasites. Nous allons nous intéresser à ce problème dans nos travaux futurs. Nous allons chercher sous quelles conditions les critères d'information AIC ou BIC peuvent être de bonnes alternatives pour choisir les paramètres de régularisation optimaux et par conséquent les variables et seulement les variables pertinentes.

3. Les modèles linéaires généralisés constituent un cadre unifié contenant de nombreuses extensions du modèle linéaire classique : la régression logistique pour les

réponses binaires, la régression de Poisson pour les données de comptage ou les modèles log-linéaires pour les tableaux de contingence en sont des exemples importants. Pour les modèles linéaires généralisés, la vraisemblance pénalisée peut être utilisée pour sélectionner les variables importantes. Supposons que nous disposions d'un échantillon indépendant $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$, où, pour tout $i = 1, ..., n$, y_i est une réalisation d'une variable aléatoire Y_i et $\mathbf{x}_i$ est un vecteur dans $\mathbb{R}^p$. Supposons que, conditionnellement à $\mathbf{x}_i$, la densité de $\mathbf{Y}_i$ est $f_i(g(\mathbf{x}_i^t\boldsymbol{\beta}), y_i)$, où g est une fonction de lien connue. Soit $l_i = \log f_i$, la log-vraisemblance conditionnelle de Y_i. Une forme de vraisemblance pénalisée est :

$$\sum_{i=1}^{n} l_i(g(\mathbf{x}_i^t\boldsymbol{\beta}), y_i) - P(\boldsymbol{\beta}), \tag{6.1}$$

où $P(.)$ est une pénalité connue. En prenant $P(\boldsymbol{\beta}) = \lambda\|\boldsymbol{\beta}\|_1 = \lambda\sum_{j=1}^{p}|\beta_j|$, Park and Hastie (2007) ont fait une extension du *Lasso* au cas des modèles linéaires généralisés. Fan and Li (2001) ont fait un autre travail intéressant dans le cadre (6.1), où les auteurs ont discuté différents choix de la pénalité. Dans un axe futur de recherche, nous allons s'intéresser au problème (6.1). Nous allons faire une étude théorique et pratique en utilisant des pénalités plus adaptées à des structures supplémentaires sur les données (présence de groupes, ordres sur les variables), ce qui est une extension naturelle de ce que nous avons fait dans le cadre des modèles linéaires.

4. Pour améliorer les performances des méthodes proposées au niveau sélection de variables qu'au niveau prédiction, un des axes futurs de recherche est l'utilisation de techniques de calculs intensifs telle que la technique *bootstrap*. Le principe général du fonctionnement de cette technique est le suivant : supposons que nous disposions d'un échantillon d'apprentissage $\mathbf{Z} = (z_1, ..., z_n)$, où $z_i = (x_i, y_i)$. L'idée de base est de générer d'une manière aléatoire B échantillons $\mathbf{Z}^{*b}$, $b = 1, ..., B$, en faisant des tirages avec remise à partir de l'échantillon d'apprentissage. Chaque échantillon généré a la même taille que l' échantillon d'apprentissage initial. Nous estimons les paramètres en utilisant chaque échantillon *bootstrap* $\mathbf{Z}^{*b}$, $b = 1, ..., B$. L'estimateur final dépendra des estimations obtenues à chaque réplication.

5. Dans cette thèse, nous avons proposé une méthode bayésienne de sélection de modèles, basée sur des lois a priori de Zellner. Nous avons aussi comparé les approches bayésiennes et fréquentielles dans un cadre peu informatif, où le nombre de variables est presque égal à la taille de l'échantillon. Cette comparaison nous encourage à mieux explorer la voie bayésienne de la sélection de variables en régression. L'approche bayésienne présentée dans cette thèse est adaptée seulement aux situations pour lesquelles le nombre de variables p est inférieur à la taille de l'échantillon n. Pour rendre les techniques de régularisation bayésiennes applicables à des problèmes de la statistique moderne, il faut développer des méthodes capables de se comporter avec des bases de données dans lesquelles le nombre de variables est beaucoup plus grand que la taille de l'échantillon, tel que le travail qui a été fait par Bottolo and

Richardson (2010). Ceci sera un autre axe futur de recherche auquel nous allons donner plus d'intérêt.

Bibliographie

Bartlett, M. (1957). A comment on D.V. Lindley's statistical paradox. *Biometrika*, 44 :533–534.

Berger, J., Pericchi, L., and Varshavsky, I. (1998). Bayes factors and marginal distributions in invariant situations. *Sankhya A*, 60 :307–321.

Bickel, P., Ritov, Y., and Tsybakov, A. (2009). Simultaneous analysis of lasso and dantzig selector. *Annals of Statistics*, 37(4) :1705–1732.

Bondell, H. D. and Reich, B. J. (2008). Simultaneous regression shrinkage, variable selection and clustering of predictors with oscar. *Biometrics*, 64 :115–123.

Bottolo, L. and Richardson, S. (2010). Evolutionary stochastic search for Bayesian model exploration. *Bayesian Analysis*, 5 :583–618.

Breheny, P. and Huang, J. (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Ann. Appl. Stat*, 5 :232–253.

Breiman, L. (1995). Better subset regression using the nonnegative garrote. *Technometrics*, 37(4) :373–384.

Breiman, L. and Friedman, J. (1985). Estimating optimal transformations for multiple regression and correlation. *J. American Statist. Assoc*, 85 :580–598.

Brown, J. and Vannucci, M. (1998). Multivariate Bayesian variable selection and prediction. *J. Royal Statist. Soc. Series B*, 60 :627–641.

Bunea, F., Tsybakov, and Wegkamp, M. (2007). Sparsity oracle inequalities for the lasso. *Electronic Journal of Statistics*, 1 :169–194.

Butler, R. and Wood, A. (2002). newblock laplace approximations for hypergeometric functions with matrix arguments. *Ann. Statist*, 30 :1155–1177.

Candes, E. and Tao, T. (2007). The Dantzig Selector : statistical estimation when p is much larger than n. *Ann. Statist*, 35(6) :2313–2351.

Casella, G. and Moreno, E. (2006). Objective Bayesian variable selection. *J. American Statist. Assoc*, 101(473) :157–167.

Celeux, G., M. J. and Robert, C. (2006). Sélection bayésienne de variables en régression linéaire. *Journal de la Société Française de Statistique*, 147(1) :59–79.

Chen, S., Donoho, D., and Saunders, M. (1998). Atomic decomposition by basis pursuit. *SIAM J. on Sci. Comp.*, 20(1) :33–61.

Chipman, H. (1996). Bayesian variable selection with related predictors. *Canadian Journal of Statistics*, 1 :17–36.

Clemmensen, L., Hastie, T., Witten, D., and Ersboll, B. (2011). Sparse discriminant analysis. *Technometrics*, To appear.

Cui, W. and George, E. (2008). Empirical Bayes vs. fully Bayes variable selection. *Journal of Statistical Planning and Inference*, 138 :888–900.

Daye, Z. J. and Jeng, X. J. (2009). Shrinkage and model selection with correlated variables via weighted fusion. *Computational Statistics and Data Analysis*, 53 :1284–1298.

de Finetti, B. (1972). *Probability, Induction and Statistics*. John Wiley, New York.

DeGroot, M. (1973). Doing what comes naturally : Interpreting a tail area as a posterior probability or as a likelihood ratio. *J. American Statist. Assoc.*, 68 :966–969.

Donoho, D. and Johnstone, J. (1994). Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3) :425–455.

Dupuis, J. and Robert, C. (2003). Bayesian variable selection in qualitative models by Kullback-Leibler projections. *J. Statist. Plann. Inference*, pages 77–94.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*, 32 :407–499.

ElAnbari, M. and Mkhadri, A. (2008). Penalized regression with a combination of the l1 norm and the correlation based penalty. Technical Report RR-6746, INRIA.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96 :1348–1360.

Fan, J. and Lv, J. (2008). Sure independence sceening for ultra-high dimensional feature space. *Journal of Royal Statistical Association*, 70(5) :849–911.

Fan, J. and Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. *The Annals of Statistics*, 32 :928–961.

Fang, K. and Wang, Y. (1994). *Number-Theoretic Methods in Statistics*. Chapman and Hall, London.

Fernandez, C., L. E. and Steel, M. (2001). Benchmark priors for Bayesian model averaging. *J. Econometrics*, 100 :381–427.

Foster, D. and George, E. (1994). The risk inflation criterion for multiple regression. *Ann. Statist.*, 22 :1947–1975.

Frank, I. and Friedman, J. (1993). A statistical view of some chemometrics regression tools. *Technometrics*, 35(2) :109–135.

Friedman, J., Hastie, T., Hoefling, and Tibshirani, R. (2007). Pathwise coordinate optimization. *Ann. Appl. Statist*, 1 :302–332.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Biometrika*, 33(1) :1–22.

Fu, W. (1998). Penalized regressions : the bridge vs the lasso. *Journal of Computational and Graphical Statistics*, 7(3) :397–416.

Genkin, A., Lewis, D., and Madigan, D. (2007). Large-scale bayesian logistic regression for text categorization. *Technometrics*, 49 :291–304.

George, E. (2000). The variable selection problem. *J. American Statist. Assoc.*, 95 :1304–1308.

George, E. and Foster, D. (1993). Variable selection via Gibbbs sampling. *J. American Statist. Assoc.*, 88 :881–889.

George, E. and Foster, D. (2000). Calibration and empirical Bayes variable selection. *Biometrika*, 87(4) :731–747.

George, E. and McCulloch, R. (1997). Approaches to Bayesian variable selection. *Statistica Sinica*, 7 :339–373.

Ghosh, S. (2011). On the grouped variable selection and model complexity of the adaptive elastic net. *Statistics and Computing*, 21(1) :451–462.

Guo, R. and Speckman, P. (2009). Bayes factor consitency in linear models. *The 2009 International Workshop on Objective Bayes Methodology, Philadelphia, June 5-9, 2009*.

Hastie, T., Rosset, S., Tibshirani, R., and Zhu, J. (2004). The entire regularization path for the support vector machine. *The Journal of Machine Learning Research*, 5 :1391–1415.

Hebiri, M. and van De Geer, S. (2010). The smooth-lasso and other $l_1 + l_2$ penalized methods. Technical report, ArXiv.

Hoerl, A. and Kennard, R. (1970). Ridge regression : biased estimation for non orthogonal problems. *Technometrics*, 12 :55–67.

Huang, J., Breheny, P., Ma, S., and Zhang, C.-H. (2010). The mnet method for variable selection. Technical Report 402, Depart. Stat., Iowa Univ.

James, G., Radchenko, P., and Lv, J. (2009). Dasso : Connections between the dantzig selector and lasso. *Journal of the Royal Statistical Society, Series B*, 71 :127–142.

Jia, J. and Yu, B. (2010). On model consistency of the elastic net when $p >> n$. *Statistica Sinica*, 20 :595–612.

Kass, R. and Raftery, A. (1995). Bayes factor and model uncertainty. *J. American Statist. Assoc.*, 90 :773–795.

Kass, R. and Wasserman, L. (1995). A reference Bayesian test for nested hypotheses and its,relationship to the Schwarz criterion. *J. American Statist. Assoc.*, 90 :928–934.

Knight, K. and Fu, W. (2000). Asymptotics for lasso-type estimators. *The Annals of Statistics*, 28 :1356–1378.

Kohn, R., Smith, M., and Chan, D. (2001). Nonparametric regression using linear combinations of basis functions. *Statistics and Computing*, 11 :313–322.

Krishnapuram, B., Figueiredo, M., Carin, L., and Hartemink, A. (2005). Sparse multinomial logistic regression : Fast algorithms and generalization bounds. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27 :957–968.

Li, Y. and Zhu, J. (2008). L1-norm quantile regression. *Journal of Computational and Graphical Statistics*, 7(1) :163–185.

Liang, F., Paulo, R., Molina, G., Clyde, M., and Berger, J. (2008). Mixtures of g-priors for Bayesian variable selection. *J. American Statist. Assoc.*, 103(481) :410–423.

Lindley, D. (1957). A statistical paradox. *Biometrika*, 44 :187–192.

Marin, J. and Robert, C. (2007). *Bayesian Core : A Practical Approach to Computational Bayesian Statistics*. Springer-Verlag, New York.

Mazumder, R., Friedman, J., T., and Hastie (2011). Sparsenet : Coordinate descent with non-convex. *Journal of the American Statistical Association*, 58.

McCullagh, P. and Nelder, J. (1989). *Generalized Linear Models*. Chapman & Hall.

Meier, L., van de Geer, S., and Buhlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society : Series B*, 70 :53–71.

Mitchell, T. and Beauchamp, J. (1988). Bayesian variable selection in linear regression. *J. American Statist. Assoc.*, 83 :1023–1032.

Nelson, K. W. and Fisher, A. G. (1985). Generalized body composition prediction equation for men using simple measurement techniques. *Medicine and Science in Sports and Exercise*, 17.

Nott, D. J. and Green, P. J. (2004). Bayesian variable selection and the Swendsen-Wang algorithm. *J. Comput. Graph. Statist.*, 13 :1–17.

Osborne, M. R., Presnell, B., and Turlach, B. A. (2000). On the lasso and its dual. *Journal of Computational and Graphical Statistics*, 9(2) :309–337.

Park, M. and Hastie, T. (2007). L1-regularization path algorithm for generalized linear models. *Journal of the Royal Statistical Society : Series B*, 69(4) :659–677.

Park, T. and Casella, G. (2008). The Bayesian lasso. *J. American Statist. Assoc.*, 103(473) :681–686.

Penrose, K., N. A. and Fisher, A. (1985). Generalized body composition prediction equation for men using simple measurement techniques. *Medicine and Science in Sports and Exercise*, 17(2) :189.

Philips, R. and Guttman, I. (1998). new criterion for variable selection. *Statist. Prob. Letters*, 38 :11–19.

Portnoy, S. (1984). Asymptotic behavior of m-estimators of p regression parameters when p^2/n is large. i. consistency. *The Annals of Statistics*, 12 :1298–1309.

Rao, C. (1973). *Linear Statistical Inference and its Applications*. John Wiley, New York.

Robert, C. (1993). A note on the Jeffreys-Lindley paradox. *Statistica Sinica*, 3 :601–608.

Robert, C. (2001). *The Bayesian Choice*. Springer-Verlag, New York.

Schneider, U. and Corcoran, J. (2004). Perfect sampling for Bayesian variable selection in a linear regression model. *J. Statist. Plann. Inference*, 126 :153–171.

She, Y. (2010). Sparse regression with exact clustering. *Electronic J. Statistics*, 4 :1055–1096.

Shevade, S. and Keerthi, S. (2003). A simple and efficient algorithm for gene selection using sparse logistic regression. *Bioinformatics*, 19(17) :2246–2253.

Slawski, M., zu Castell, W., and Tutz, G. (2010). Feature selection guided by structural information. *Annals of Applied Statistics*, 4 :1056–1080.

Smith, M. and Kohn, R. (1996). Nonparametric regression using Bayesian variable selection. *Journal of Econometrics*, 75 :317–343.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. Roy. Statist. Soc. Ser. B*, 58(1) :267–288.

Tibshirani, R., Sunders, M., Rosset, S., Zhu, J., and K. Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal statistical Society, B*, 67 :91–108.

Tutz, G. and Ulbricht, J. (2009). Penalized regression with correlation based penalty. *Statistics & Computing*, 19 :239–253.

van der Kooij, A. (2007). Prediction accuracy and stability of regresssion with optimal scaling transformations. Technical report, Dept. Data Theory, Leiden University.

Varmuza, K. and Filzmoser, P. (2009). *Introduction to Multivariate Statistical Analysis in Chemometrics*. CRC Press Taylor and Francis group.

Wang, H., Li, R., and Tsai, C.-L. (2007). Tuning parameter selectors for the smoothly clipped absolute deviation. *Biometrika*, 94 :553–568.

Witten, D. and Tibshirani, R. (2009). Covariance-regularized regression and classification for high-dimensional problems. *Journal of the Royal Statistical Society, Series B*, 71(3) :615–636.

Wu, S., Shen, X., and Geyer, C. J. (2009). Adaptive regularization using the entire solution surface. *Biometrika*, 96(3) :513–527.

Wu, T. and Lang, K. (2007). Coordinate descent procedures for lasso penalized regression. *Ann. Appl. Statist*, 2 :224–244.

Wu, T. and Lange, K. (2008). Coordinate descent algorithms for lasso penalized regression. *Annals of Applied Statistics*, 2(1) :224–244.

Yuan, M. and Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 68(1) :49–67.

Yuan, M. and Lin, Y. (2007). On the no-negative garrote estimator. *J. R. Statist. Soc*, 69 :143–161.

Zellner, A. (1986). On assessing prior distributions and Bayesian regression analysis with g-prior distribution regression using Bayesian variable selection. *Bayesian inference and decision techniques : Essays in Honor of Bruno De Finetti*, pages 233–243.

Zellner, A. and Siow, A. (1980). Posterior odds ratios for selected regression hypotheses. *Bayesian Statistics*, pages 585–603.

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *Ann. Statist*, 38 :894–942.

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. *Journal of Machine Learning Research*, 7 :2541–2563.

Zhou, S., Van de Geer, S., and Buhlmann, P. (2009). Adaptive lasso for high dimensional regression and gaussian graphical modeling. Technical report, ArXiv.

Zou, H. (2006). The adaptive lasso and its oracle properties. *J. American Statist. Assoc.*, 101 :1418–1429.

Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, 67(2) :301–320.

Zou, H. and Zhang, H. (2009). On the adaptive elastic net with a diverging number of parameters. *The Annals of Statistics*, 37(4) :1733–1751.

TITRE : Régularisation et sélection de variables par le biais de la vraisemblance pénalisée.

Résumé : Dans cette thèse nous nous intéressons au problème de la sélection de variables en régression linéaire. Ces travaux sont en particulier motivés par les développements récents en génomique, protéomique, imagerie biomédicale, traitement de signal, traitement d'image, en marketing, etc. Nous regardons ce problème selon les deux points de vue fréquentielle et bayésienne.

Dans un cadre fréquentiel, nous proposons des méthodes pour faire face au problème de la sélection de variables, dans des situations pour lesquelles le nombre de variables peut être beaucoup plus grand que la taille de l'échantillon, avec présence possible d'une structure supplémentaire entre les variables, telle qu'une forte corrélation ou un certain ordre entre les variables successives. Les performances théoriques sont explorées ; nous montrons que sous certaines conditions de régularité, les méthodes proposées possèdent de bonnes propriétés statistiques, telles que des inégalités de parcimonie, la consistance au niveau de la sélection de variables et la normalité asymptotique.

Dans un cadre bayésien, nous proposons une approche globale de la sélection de variables en régression construite sur les lois a priori g de Zellner dans une approche similaire mais non identique à celle de Liang et al. (2008). Notre choix ne nécessite aucune calibration. Nous comparons les approches de régularisation bayésienne et fréquentielle dans un contexte peu informatif où le nombre de variables est presque égal à la taille de l'échantillon.

Mots clés : Réduction de la dimension, Grandes dimensions, Lasso, Scad, Elastic-net, Sélection de modèles, Propriétés d'Oracle, Les moindres carrées pénalisées, La vraisemblance pénalisée, Lois a priori non informatives, Zellner's g–prior, calibration.

TITLE : Regularization and variable selection using penalized likelihood.

Abstract : We are interested in variable selection in linear regression models. This research is motivated by recent development in microarrays, proteomics, brain images, among others. We study this problem in both frequentist and bayesian viewpoints.

In a frequentist framework, we propose methods to deal with the problem of variable selection, when the number of variables is much larger than the sample size with a possibly presence of additional structure in the predictor variables, such as high correlations or order between successive variables. The performance of the proposed methods is theoretically investigated ; we prove that, under regularity conditions, the proposed estimators possess statistical good properties, such as *Sparsity Oracle Inequalities*, *variable selection consistency* and *asymptotic normality*.

In a Bayesian framework, we propose a global noninformative approach for Bayesian variable selection. In this thesis, we pay special attention to two calibration-free hierarchical Zellner's g-priors. The first one is the Jeffreys prior which is not location invariant. A second one avoids this problem by only considering models with at least one variable in the model. The practical performance of the proposed methods is illustrated through numerical experiments on simulated and real world datasets, with a comparison between Bayesian and frequentist approaches under a low informative constraint when the number of variables is almost equal to the number of observations.

Keywords : Dimensionality reduction, high dimensionality, LASSO, SCAD, ELASTIC-NET, model selection, oracle property, penalized least squares, penalized likelihood, variable selection, noninformative priors, Zellner's g–prior, calibration.