Recherche de composante(s) explicative(s) par pénalisation

Mémoire de Master 1

Master *Sciences, Technologies et Santé*,
Mention *Mathématique*,
Parcours : Biostatistique

Auteur
Yonatan Carlos CARRANZA ALARCON

Superviseur
Xavier Bry

Jury
Benoîte de Saporta
Lionel Cucala

version 1 soutenu publiquement le 25 mai 2016

Table des matières

Table des figures

Remerciements

Je tiens à remercier mon encadrant Xavier Bry qui m'a enseigné, guidé, suivi et orienté et qui a répondu à toutes mes questions et autres problèmes rencontrés lors de ce modeste travail. Je tiens aussi à remercier à mes parents qui m'ont toujours soutenu, encouragé et aidé.

Introduction

La croissance evolution de la science statistique nous amenés à recherche de nouvelles méthodes sur le traitement des données complexes, en nous permettant d'analyser, de modéliser et aussi de prédire certains phénomènes aléaitoires difficiles à comprendre au premier regard. L'une d'entre elles, est la regression linéaire, un domaine de l'apprentissage automatique, et plus particulièrement l'apprentissage supervisé. L'apprentissage supervisé cherche à répondre une question posé par un des ses fondateurs *Tom M Mitchell* [3] :

> "How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes ?"

La regression linéaire est un example de cette dernière question, car cette méthode apprend au fur et mesure, à partir des données fournis (i.e. des experiences) en ajustant au maximum l'erreur de prevision (i.e. minimisant l'erreur de prévision).

La regression linéaire est confronté régulièrement à divers problématiques, et plus particulièrement, dans notre contexte d'étude : l'absence d'échantillon nécessaire[1] pour pouvoir appliquer des méthodes de regression classiques. Ce problème s'étend dans le domaine de la médecine, la biologie et la chimique, où le processus de prélèvement d'échantillon est très coûteux et parfois des longues périodes (e.g. des années). D'où, la motivation de ce travail pour tenter de *minimiser l'erreur de prevision* en utilisant de composantes explicatives[2] par pénalisation.

En résumé, le reste de ce rapport est organisé de la manière suivante :

Le chapitre 1 décrit brièvement l'état de l'art, c'est-à-dire les différents méthodes de regression linéaire existants dans la littérature scientifique.

Le chapitre 2 aborde la modélisation et l'optimisation de la nouvelle proposition de regression linéaire de composantes explicatives par pénalisation.

Enfin, le chapitre 3 montrera les résultats obtenus, les comparaisons avec les méthodes existantes, et la conclusion ainsi que les futurs travaux.

[1] C'est-a-dire, le nombre d'échantillon est inférieur au nombre de variables explicatives, cela veut dire que la matrice $X'X$ dans la regression classique ne sera plus inversible.

[2] Les composantes explicatives seront le nombre minimal des composantes principales qui expliquent le mieux la dispersion de nos données.

1

Regression

Dans ce chapitre, nous allons présenter les méthodes de regression existantes dans la littérature scientifique.

Tout d'abord, le mot régression a été inventé à parti d'un travail en 1885 par Sir Francis Glaton où il travaillais sur l'hérédité dont il cherchait à expliquer la taille des fils (i.e. variable à expliquer Y) en fonction de celle des pères (i.e variable explicative X) [1]. En general la regression est un méthode statistique utilisé pour analyser la relation d'une variable par rapport à une ou plusieurs autres, autrement dit, cette méthode essaie d'expliquer la variable aléatoire à expliquer Y conditionnement à la variable aléatoire explicative X.

L'un des modèles de régression le plus connu est le modèle de regression linéaire qui a été étudié depuis longtemps dans le domaine statistique et ailleurs. Ils essaient à travers d'une equation d'expliquer la variable aléatoire Y linéairement en fonction d'une variable explicative X et leurs erreurs de mesure ou/et bruit ε. Il existe plusieurs variantes de cette regression, mais le modèle le plus général est le suivant :

$$Y = X\beta + \varepsilon$$

Nous pouvons diviser la regression en deux grands groupes ; (1) la régression définie quand $rang(X) <<< n$, et (2) régression non définie quand $rang(X) >>> n$.

1.1 Regression linéaire avec $p << n$

Nous allons explorer les méthodes de regression le plus connue où le nombre de variables explicatives p est plus petit que le nombre d'échantillons n (i.e $p << n$), et par consequence, la matrice $X'X$ sera inversible.

1.1.1 Regression linéaire multiple gaussien

La regression linéaire multiple gaussien est un cas particulier de la regression linéaire. Nous soumettrons à cet modèle trois hypotheses :

- $\mathcal{H}_1 : rang(X) = p$
- $\mathcal{H}_2 : \mathbb{E}[\varepsilon] = 0, \Sigma_\varepsilon = \sigma^2\mathbb{I}$

– $\mathcal{H}_3 : \varepsilon \sim \mathcal{N}(0, \sigma^2\mathbb{I})$, ils sont i.i.d.

Le modèle de régression gaussien s'écrit donc : $y_i = x_i\beta_i + \epsilon_i$ où $\epsilon_i \sim \mathcal{N}(0, \sigma^2\mathbb{I})$, et par consequence, chaque y_i suit une loi normal (i.e. $y_i \sim \mathcal{N}(x_i\beta_i, \sigma^2\mathbb{I})$). Partant de ce dernier fait, nous pouvons donc calculer le maximum de vraisemblance (EMV) pour retrouver les valeurs optimales des coefficients $\boldsymbol{\beta}$.

$$\mathcal{L}(Y; \boldsymbol{\beta}, \sigma^2) = \prod_{i=1}^{n} f_Y(y_i) = \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left[-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{i,j}\beta_i)^2\right] \tag{1.1}$$

D'où, les estimateurs de maximum vraisemblance sont : $\hat{\boldsymbol{\beta}} = (X'X)^{-1}X'Y$ et $\sigma^2 = \left\|Y - X\hat{\boldsymbol{\beta}}\right\|^2 /n$, cet dernier étant un estimateur biaisé. Or, le résidus sont $\hat{\epsilon} = Y - \hat{Y}$, où $\hat{Y} = X\hat{\boldsymbol{\beta}}$. Une représentation géométrique se trouve dans la figure 1.1.

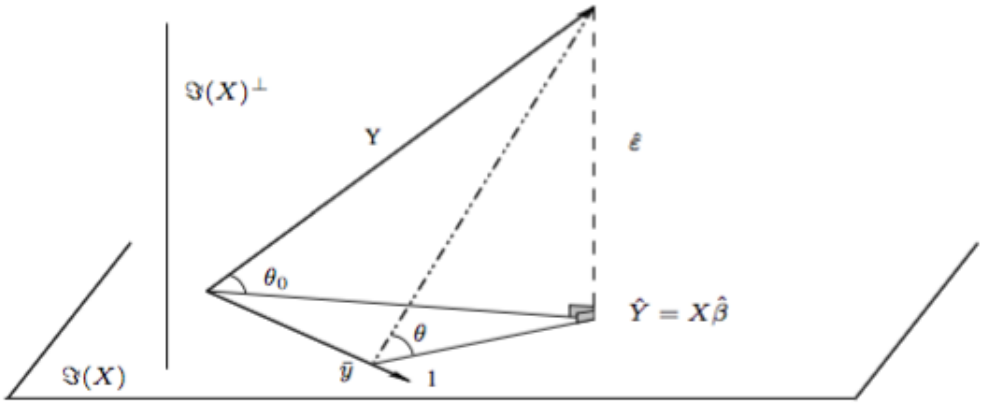

FIGURE 1.1 – *Représentation de variables et interprétation géométrique.* $\Im(X)$ est l'espace engendré par les colonnes de X, et la projection de Y sur $\Im(X)$ est la solution optimale $X\hat{\boldsymbol{\beta}}$.

1.1.2 Méthodes Forward-Stepwise et Backward-Stepwise

Il arrive régulièrement que l'estimation de moindre carrés (i.e. regression linéaire multiple) ne soit pas précise ; c'est-à-dire, (1) il peut exister des problèmes dans la précision de prediction à cause de leur faible biais et haute variance, cependant il peut s'améliore en réduisant les coefficients ou en mettant de zeros aux coefficients, (2) il peut aussi exister des problèmes de interprétations avec un grand nombre de régresseurs, en sachant qui nous pouvons trouver avec un sous-ensemble de régresseurs d'effets les plus forts, et (3) il y a parfois de cas où le nombre de variables explicatives est plus grand que l'échantillon.

A cause de ces derniers problèmes est qu'il existe des méthodes hybrides pour ajouter (i.e. Forward-Stepwise) ou supprimer (i.e. Backward-Stepwise) des régresseurs au modèle au fur et mesure qui améliorent le précision de la prevision.

Forward-Stepwise

Le méthode forward-stepwise est un modèle progressive, c'est-à-dire ; il commence avec la variable constante (ou interception en anglais) et il ajoute ensuite séquentiellement au modèle

des régresseurs qui améliorent encore plus la précision de la prevision en comparant le sous-modèle emboité avec le nouveau modèle emboité, et nous prenons le modèle emboité avec le régresseur ayant le plus grand F-statistic [2].

Backward-Stepwise

Contrairement au modèle précédente, Backward-Stepwise commence avec tous les régresseurs du modèle et il supprime séquentiellement chaque régresseur qui a le moins d'impact sur le modèle. Autrement dit, le régresseur supprimé a le plus petit F-statistic à partir du modèle courant [2].

La figure 1.2 a été cité dans le bouquin [4], elle montre une graphique comparative entres 4 méthodes dont deux ont été décrites auparavant.

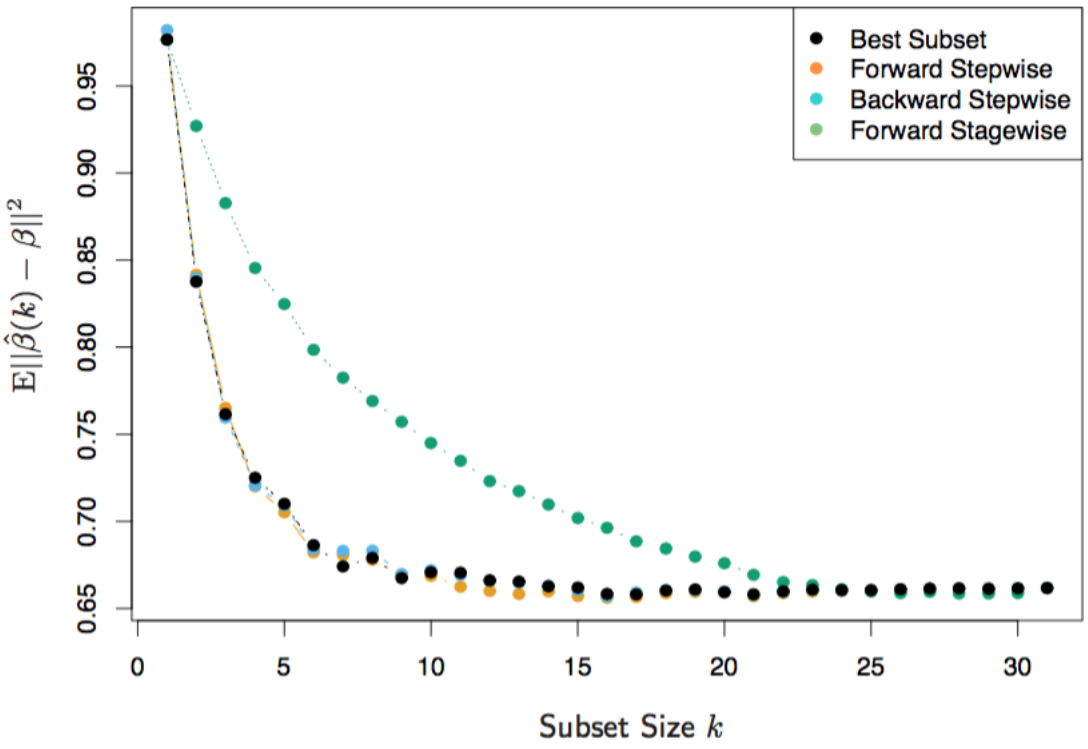

FIGURE 1.2 – *Comparatives de méthodes stepwise.* Comparaison de quatre techniques de sous-sélection des régresseurs soumis à un problème de regression linéaire : $Y = X\beta + \epsilon$. Il y a $n = 300$ individus avec $p = 31$ variables explicatives gaussiennes. Pour 10 variables, les coefficients sont tirés au sort à partir d'une loi $\mathcal{N}(0, 0.4)$, le reste est nul. Le bruit suite un loi normale $\epsilon \sim \mathcal{N}(0, 6.25)$. Les résultats sont moyennés sur 50 simulations. La figure montre l'erreur quadratique moyenne du coefficient estimé $\hat{\beta}(k)$ à chaque étape contre la vraie valuer β [2].

1.2 Regression linéaire avec $p >> n$

Dans cette sous-section, nous allons explorer les cas où le nombre de variables explicatives p est plus grande que le nombre d'échantillons n ($p >>> n$), et par consequence, la matrice

$X'X$ ne sera plus inversible.

Pour valider la qualité de précision de prédiction des trois suivantes méthodes de regression qui s'ajustent grâce aux données, nous utiliserons la validation croisée qui sera explique dans la section suivante.

1.2.1 Regression Ridge

La méthode *Forward-Stepwise* est une exemple de reduction discrète de coefficients sur le modèle, lors que nous serons dans cas où $p >>> n$, car nous ajoutons incrémentalement des régresseurs optimisant le modèle jusqu'à $p < n$. Cependant, malheureusement cette méthode supprime certains régresseurs x_j qui puissent être importants pour le modèle. C'est pourquoi nous allons introduire le méthodes Ridge.

Cette méthode de regression *Ridge*, contrairement au méthode *Forward-Stepwise*, elle n'annule pas les régresseurs, elle réduit les coefficient des régresseurs en l'imposant une pénalité k à leur valeur, dans le but de pondérer les coefficients en le donnant plus d'importance à certains et pas d'autres. Il est aussi importante de remarque que cette penalisation nous aide à pouvoir inverser la matrice $X'X$ qu'au depart elle n'était pas, en augmentant toutes les valeurs propres et donc elles qui sont (quasi) nulles de $X'X$. Ainsi donc, la méthode de Ridge équivaut à résoudre le problème de minimisation suivant :

$$\hat{\boldsymbol{\beta}}_{ridge} = \arg\min \left[\sum_{i=1}^{n} (y_i - \boldsymbol{\beta}_0 - \sum_{j=1}^{p} x_{i,j}\boldsymbol{\beta}_j)^2 + k \sum_{j=1}^{p} \boldsymbol{\beta}_j^2 \right]$$

$$\hat{\boldsymbol{\beta}}_{ridge}(k) = (X'X + k\mathbb{I})^{-1} X'Y \tag{1.2}$$

D'où, $\hat{\boldsymbol{\beta}}_{ridge}(\tilde{k})$ sont les coefficient optimaux par rapport à un valeur "optimum" de $\tilde{k}$, ce dernier paramètre n'est pas être calculer analytiquement, c'est-à-dire on ne le connait pas apriori, ainsi donc le valeur $\tilde{k}$ sera choisie grâce aux données, et elle sera donc stochastique.

1.2.2 Regression Composantes Principales (PCR)

Cette méthode cherche à exprimer la variable à expliquer Y en fonctions de ces composantes principales de la variable explicative X. Le but de la regression sur composantes principales consiste à ne conserver qu'une partie des composantes principales, c'est-à-dire le modèle régression ne conservera que les k premieres composantes principales qui apportent plus d'information (i.e capture plus d'inertie ou plus de dispersion) pour ensuite supprimer l'information apporté pour les $(p-k)$ variables explicatives, car ils seront considéré comme négligeable (ou bruit).

La modélisation de cette méthode en conservant que les k premières composantes principales avec k < p est :

$$Y = X_1^*\boldsymbol{\beta}_1^* + ... + X_k^*\boldsymbol{\beta}_k^* + \epsilon$$

Où, X^* est la matrice composée des k premières composantes principales $X_1^*, ..., X_k^*$, et nous pouvons obtenir l'estimateur optimal ajusté au modèle qui est un vecteur à k coordonnées :

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{pcr}(k) &= (X^{*'}X^*)^{-1}X^{*'}Y \\ &= (X^{*'}_{[1:k}X^*_{[1:k]})^{-1}X^{*'}_{[1:k]}Y \end{aligned} \tag{1.3}$$

Comme dans le cas de la regression ridge, cette regression sera ajustée aux données car nous ne connaissons pas apriori le valeur optimale k minimisant l'erreur quadratiques de la prevision. Ainsi donc, cette valeur k sera considéré stochastique et elle sera choisie grâce aux données.

1.2.3 Regression Moindre Carré Partiels (PLS1)

Pareil que la regression sur composantes principales (PCR) décrit auparavant, nous somme à présent intéressés dans cette méthodes par de nouvelles variables explicatives $t^1, t^2, ..., t^k$, combinaisons linéaires des variables de départ $t^i = Xc_j$, qui soient orthogonales entre elles et classées par ordre d'importantes (comme les composantes principales X_j^* d'auparavant). Ces dernières variables ne doit pas être ajustées par la part de variabilité qui représente les variables explicatives originales (comme dans la regression sur composantes principales), sinon par leur lien avec la variable à expliquer.

La construction se fait itérativement afin de retrouver les k premier variables t^k, par synthèse nous allons montrer comme calculer la variable t^k, mais la processus complet se trouve dans [2].

Définition 1.1 *k^e étape : soit $Y^{(k)} = P_{t^{(k-1)^\perp}} Y^{(k-1)} = \hat{\epsilon}_{k-1}$ la partie non encore expliquée de Y. Soit $X^{(k)} = P_{t^{(k-1)^\perp}} X^{(k-1)}$ la partie de $X^{(k-1}$ n'ayant pas encore servie à expliquer. Le k^e composante PLS est choisie telle que :*

$$t^{(k)} = \underset{\substack{t=X^{(k-1)}w,\ w\in\mathbb{R}^p \\ w'w=1}}{\arg\max} < t, Y^{(k)} >$$

Ensuite nous effectuons la régression univariée de $Y^{(k)}$ sur $t^{(k)}$:

$$Y^{(k)} = r_k t^{(k)} + \hat{\epsilon}_k$$

où $r_k \in \mathbb{R}$ est le coefficient de la régression estimé par moindre carrés et $\hat{\epsilon}_k = P_{t^{(k)^\perp}} Y^{(k)}$

Théorème 1.1 *Nous pouvons donc écrire le modèle PLS comme :*

$$\begin{aligned} Y &= P_{t^{(1)}} Y^{(1)} + ... + P_{t^{(k)}} Y^{(k)} + \hat{\epsilon}_k \\ Y &= r_1 t^{(1)} + ... + r_k t^{(k)} + \hat{\epsilon}_k \end{aligned} \tag{1.4}$$

avec $\hat{\epsilon}_k = P_{t^{(k)^\perp}} Y^{(k)} = P_{\Im(t^{(1)},...,t^{(k)})^\perp} Y^{(k)}$ et $t^{(k)} = X\tilde{w}(j), 1 < j \leq k$,

$$\tilde{w}(j) = X \prod_{i=1}^{j} \left(\mathbb{I} - w^{(i)} (t^{(i)'} t^{(i)})^{-1} t^{(i)'} X \right) w^{(j)}$$

Théorème 1.2 *Le modèle PLS à k composantes peut donc s'écrire :*

$$Y^{(k)} = X\hat{\boldsymbol{\beta}}_{PLS}(k) + \hat{\epsilon}_k$$

où $\hat{\epsilon}_k$ est le résidu final $P_{t^{(k)^\perp}} Y^{(k)} = P_{\Im(t^{(1)},...,t^{(k)})^\perp} Y^{(k)}$ et $\hat{\boldsymbol{\beta}}_{PLS}(k) = r_1\tilde{w}^{(1)} + ... + r_k\tilde{w}^{(k)}$.

Ainsi comme dans les cas précédents, cette regression PLS1 sera ajustée aux données car nous ne connaissons pas apriori la valeur optimale des k-composantes minimisant l'erreur quadratique de la prevision. Ainsi donc, comme dans les autres cas, k sera considéré stochastique et elle sera choisie grâce aux données.

1.3 Techniques pour valider le modèle d'apprentissage

Il existe plusieurs techniques pour valider la qualité d'un modèle supérvisé ; par exemple : la classification supervisée utilise les *matrices de confusions* pour valider leur correcte classification. Cependant, ces techniques ont besoin d'une étape d'apprentissage où elles pourront bien calibrer la qualité du modèle. C'est pourquoi, nous allons introduire dans cette section une technique très connue dans l'étape d'apprentissage : *la validation croisée* (ou *Cross-Validation* en anglais).

1.3.1 Validation croisée

Probablement la méthode la plus simple et la plus largement utilisée pour estimer l'erreur de prediction est la *validation croisée*. Cette méthode estime directement l'espérance de l'erreur d'échantillon supplémentaire $Err = \mathbb{E}[L(Y, \hat{f}(X))]$ (où $L(Y, \hat{f}(X)) = (Y - \hat{f}(X))^2$), l'erreur moyenne généralisé lors que la méthode $\hat{f}(X)$ est appliqué à un échantillon d'essai indépendant de la distribution conjoint de X et Y [4]. En réalité, il y a au moins trois techniques de validation croisée : « testset validation » ou « holdout method », « k-fold cross-validation » et « leave-one-out cross-validation » (LOOCV), cependant nous allons aborder la deuxième méthode qui est adapté à nos besoins.

K-Fold Cross-Validation

Normalement, si nous avions suffisamment de données, nous aurions mis de côte un ensemble de validation et de l'utiliser pour évaluer la performance de notre modèle de prédiction. Cependant les cas où nous aurions suffisamment de données sont rares, nous allons donc utiliser cette technique K-Fold dont elle utilise une partie des nos données disponibles pour ajuster le modèle, et une autre partie pour le tester. C'est-à-dire, elle coupe les données en K parties d'égal taille, lesquels $K - 1$ parties seront utiliser pour ajuster le modèle et la partie k-ième sera utilisée pour calculer l'erreur de prediction du modèle ajusté. Nous allons faire cela pour $k = 1, 2, ...K$ et calculer l'erreur de prediction dans chaque cas.

En exemple de partitions du méthode K-Fold s'affiche dans la figure 1.3, avec $K = 5$, et la partie k-ième $= 3$ étant la partition de validation.

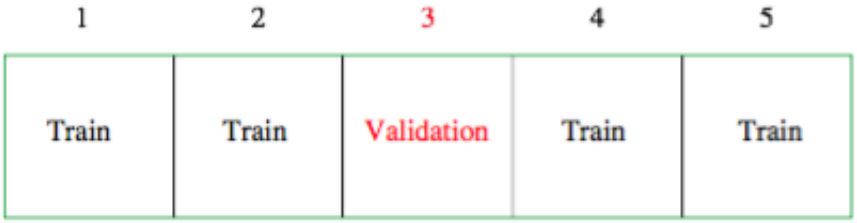

FIGURE 1.3 – *K-Fold Cross-Validation*

Après avoir calculé tous les erreurs de prediction, nous resterons avec le modèle ajusté ayant le minimum erreur de prediction.

D'un autre exemple de partitionnement se trouve dans la figure ci-dessous 1.4.

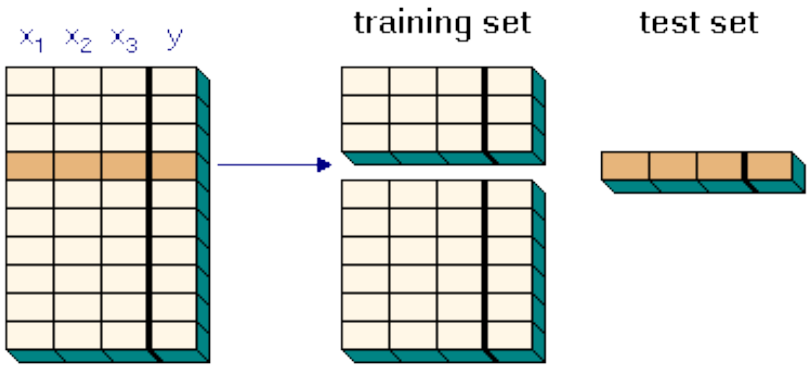

FIGURE 1.4 – *K-Fold Cross-Validation- Regression*

2

Nouvelle approche de regression par pénalisation

Après avoir examiné, dans le chapitre précédent, des méthodes existantes de regression abordant la problématique où le matrice $X'X$ n'est pas inversible, nous allons à présent proposer dans cette section une nouvelle méthode de regularisation fondée sur l'extraction de composantes par penalisation.

Tout d'abord, nous allons nous placer dans un contexte paramétrique où le but serait de trouver les vrais coefficients optimaux du modèle de regression par pénalisation proposé ci-dessous.

2.1 Definition du modèle de regression

Toutes les variables sont mesurées sur n unités statistiques indépendantes. La matrice diagonale des poids des unités est $W = \frac{1}{n}\mathbb{I}$ par défaut.

Les régresseurs (ou variables explicatives) sont partitionnés en deux groupes : $X = \{x^1, ..., x^p\}$, dans lequel nous cherchons à faire de la réduction dimensionnelles sous la forme d'une combinaison linéaire x_j, et $T = \{f^1, ..., f^K\}$, qui rassemblera les K composantes principales trouvées de X.

Les régresseurs $x^1, ..., x^p$ sont autant de variables numériques dont nous encodons les valeurs pour les n unités statistique dans une matrice $X_{(n,p)} = [x^1, ..., x^p]$. Pour chaque unité i, nous notons x_i le vecteur $x_i = \{x_i^1, ..., x_i^p\}$. Nous faisons de même pour T.

Les variables dépendantes (ou variables à expliquer) $y^1, ..., y^q$ sont de même codées dans une matrice $Y_{(n,q)} = [y^1, ..., y^q]$. Le vecteur correspondant à une unité i est : $y_i = \{y_i^1, ..., y_i^q\}$.

2.2 Modèle univarié

Nous allons d'abord analyser le modèle univarié que ne contient rien d'autre qu'une variable dépendante y^1 (i.e. $q = 1$ et $Y_{(n,1)}$), nous considérons donc le modèle gaussien suivant :

$$\begin{aligned} Y =& X\boldsymbol{\beta} + T\delta + \epsilon \; ; \quad \epsilon \sim N(0; \sigma^2\mathbb{I}) \\ & \text{où} \quad \boldsymbol{\beta} = \gamma u \quad \text{et} \quad u'u = 1 \\ & \text{donc} \quad \gamma = \|\boldsymbol{\beta}\| \quad \text{et} \; u = \frac{\boldsymbol{\beta}}{\|\boldsymbol{\beta}\|} \end{aligned}$$

L'espace paramétrique de ce modèle est $\Theta_{(u,\gamma,\delta,\sigma)} = \{\mathbb{R}^n, \mathbb{R}, \mathbb{R}^k, \mathbb{R}\}$

Nous posons $Y = f_Y(X; u, \delta, \gamma, \sigma^2)$ et $\ell = \ln(f)$ désignant la fonction paramétrique du modèle et la log-vraisemblance du modèle, où $f_Y \sim N(X\boldsymbol{\beta} + T\delta; \sigma^2\mathbb{I})$, ainsi que $s \in [0; 1]$ un paramètre de réglage. Nous cherchons donc de façon générale à résoudre le programme sous contraint suivant :

$$\mathcal{G} = \underset{\substack{\delta,\gamma,\sigma^2,u \\ u'u=1}}{\arg\max} \left[(1-s)\ell + sS(u)\right] \tag{2.1}$$

Où $S(u) > 0$ est la fonction de "pertinence structurelle", ce qui représente un bonus accordé à u dans la mesure où sa direction permet de capturer davantage d'information de X.

Nous allons dorénavant considerer le cas où $S(u) = u'Nu$, tel que $N = X'WX$.

Remarque 2.1 *Nous pouvons constater facilement que N est une matrice symétrique.*

Cette fonction de pertinence structurelle $S(u)$ correspond à donner un bonus à u égal à la variance de la composante $f = Xu$. Nous remarquerons que la maximisation isolée de $S(u)$ donne la première composante principale.

2.2.1 Cas où T est vide

Nous supposons à présent que T est vide et nous résolvons le programme $\mathcal{G}$ donné ci-dessus afin d'obtenir les estimateurs optimaux : $\hat{\gamma}, \hat{\sigma^2}, \hat{u}$ et $\hat{\lambda}$

D'abord, nous savons que les erreurs suivent une gaussienne multidimensionnelle, et par consequence, nous pouvons donc déduire facilement que la variable univarié Y suit aussi une gaussienne multidimensionnelle avec un moyenne $\mu_Y = X\boldsymbol{\beta}$ et une variance $\sigma_Y = \sigma^2\mathbb{I}$ (i.e $Y \sim N(X\boldsymbol{\beta}; \sigma^2\mathbb{I})$). Ainsi donc, nous pouvons récrire le vraisemblance de Y sans la variable T, comme suit :

$$\mathcal{L}(Y; u, \gamma, \sigma^2) = \prod_{i=1}^{n} f_Y(y_i) = \left(\frac{1}{2\pi\sigma^2}\right)^{n/2} \exp\left(-\frac{1}{2\sigma^2}\|Y - X\boldsymbol{\beta}\|^2\right) \tag{2.2}$$

Et nous pouvons aussi écrire leur log-vraisemblance :

$$\ell = -\frac{n\ln(2\pi)}{2} - \frac{n\ln(\sigma^2)}{2} - \frac{\|Y - X\boldsymbol{\beta}\|^2}{2\sigma^2} \tag{2.3}$$

Nous allons donc remplacer la log-vraisemblance et la fonction de pertinence structurelle $S(u) = u'Nu$ dans la fonction à maximiser $\mathcal{G}$:

$$\mathcal{G} = \left[(s-1)\frac{n\ln(2\pi)}{2} + (s-1)\frac{n\ln(\sigma^2)}{2} + (s-1)\frac{\|Y - X\boldsymbol{\beta}\|^2}{2\sigma^2} + su'Nu \right]$$

Nous allons ensuite enlever le premier terme car il est constant, et par consequence, il ne va rien nous apporter au moment de l'optimisation de $\mathcal{G}$, nous avons donc le programme suivante à maximiser :

$$\mathcal{G}' = \underset{\substack{\gamma,\sigma^2,u \\ u'u=1}}{\arg\max} \left[(s-1)\frac{n\ln(\sigma^2)}{2} + (s-1)\frac{\|Y - X\boldsymbol{\beta}\|^2}{2\sigma^2} + su'Nu \right] \tag{2.4}$$

En vue de maximiser la fonction $\mathcal{G}$ avec une contrainte, nous allons utiliser le *Multiplicateur de Lagrange* et interchanger le paramètre $\boldsymbol{\beta} = \gamma u$ afin de trouver les valeurs optimales des estimateurs, nous aurons donc la fonction L à dérive suivante :

$$L(u, \gamma, \sigma^2, \lambda) = (s-1)\frac{n\ln(\sigma^2)}{2} + (s-1)\frac{\|Y - \gamma Xu\|^2}{2\sigma^2} + su'Nu - \lambda(u'u - 1) \tag{2.5}$$

Dérivée par rapport à λ : Nous obtiendrons la contrainte :

$$\frac{\partial L}{\partial \lambda} = 0 \iff u'u = 1 \tag{2.6}$$

Dérivée par rapport à σ^2 : Nous aurons l'estimateur $\hat{\sigma}^2$ qui dépendra des paramètres γ et u.

$$\begin{aligned} \frac{\partial L}{\partial \sigma^2} = 0 &\iff \frac{n(s-1)}{2\sigma^2} - (s-1)\frac{\|Y - \gamma Xu\|^2}{2(\sigma^2)^2} = 0 \\ &\iff \hat{\sigma}^2 = \frac{\|Y - \gamma Xu\|^2}{n} \end{aligned} \tag{2.7}$$

Dérivée par rapport à γ : Nous aurons l'estimateur $\hat{\gamma}$ qui dépendra du paramètre u.

$$\begin{aligned} \frac{\partial L}{\partial \gamma} = 0 &\iff (1-s)\frac{u'X'(Y - \gamma Xu)}{\sigma^2} = 0 \\ &\iff \gamma u'X'Xu = u'X'Y \\ &\iff \hat{\gamma} = \frac{\langle\, Xu \mid Y \,\rangle}{\|Xu\|^2} \end{aligned} \tag{2.8}$$

Dérivée par rapport à u : Nous aurons l'estimateur $\hat{u}$ qui dépendra des paramètres γ, λ et σ^2.

$$\nabla_u L = 0 \iff (1-s)\frac{2\gamma X'(Y - \gamma Xu)}{2\sigma^2} + 2sNu - 2\lambda u = 0 \tag{2.9}$$

$$\begin{aligned} &\iff \left[(s-1)\gamma^2 X'X + 2\sigma^2(sN - \lambda\mathbb{I})\right] u = \gamma(s-1)X'Y \\ &\text{On pose : } \Omega = \left[(s-1)\gamma^2 X'X + 2\sigma^2(sN - \lambda\mathbb{I})\right] \\ &\iff \hat{u} = \gamma(s-1)\Omega^{-1}X'Y \end{aligned} \tag{2.10}$$

Nous calculerons le paramètre multiplicateur $\hat{\lambda}$ en multipliant par u' la équation (2.9) :

$$\times u' \iff (1-s)\frac{2\gamma u'X'(Y-\gamma Xu)}{2\sigma^2} + 2su'Nu - 2\lambda u'u = 0$$

Nous savons de l'équation (6) que : $u'u = 1$

$$\iff \hat{\lambda} = (1-s)\frac{\gamma u'X'\left(Y-\gamma Xu\right)}{2\sigma^2} + su'Nu$$

$$\iff \hat{\lambda} = \frac{(1-s)\hat{\gamma}\left[\langle Xu \mid Y\rangle - \hat{\gamma}\left\|Xu\right\|^2\right]}{2\sigma^2} + su'Nu \tag{2.11}$$

En replaçant $\hat{\gamma}$ sur la equation précédent, nous observerons une simplification de la partie gauche, obtenant ainsi le suivant $\hat{\lambda}$:

$$\hat{\gamma} = \frac{\langle\ Xu \mid Y\ \rangle}{\left\|Xu\right\|^2} \iff \hat{\lambda} = su'Nu \tag{2.12}$$

Nous savons apriori que le matrice $X'X$, et aussi $X'WX$, n'est pas inversible et que la matrice Ω ne peut pas non plus être inversible pour certains valeurs de (s, λ, γ) lors que nous cherchons les valeurs optimales des estimateurs : $\hat{\gamma}, \hat{s}$, et $\hat{\lambda}$, et il est donc incalculable.

Ainsi donc, nous allons introduire un outil extrêmement utile dans l'analyse de nombreuses méthodes statistiques, nommé *Décomposition en valeurs singulières*(SVD), recommandé par [4] dans la regression ridge (section 3.4.1 page. 64), afin de pouvoir calculer leur pseudo-inverse de Ω.

Théorème 2.1 *Soit X une matrice $m \times p$ dont les coefficients appartiennent au corps K, où $K = \mathbb{R}$ ou $K = \mathbb{C}$. Alors il existe une factorisation, nommé* Décomposition en Valeurs Singulières *de X, de la forme :*

$$X = U\Sigma V'$$

Où si $K = \mathbb{R}$, les matrices unitaires sont de matrices orthogonales et :

- *U une matrice unitaire $m \times m$ sur K.*
- *Σ une matrice diagonale rectangulaire $m \times n$ dont les coefficients diagonaux sont des réels positifs ou nuls et tous les autres sont nuls.*
- *V est la matrice unitaire $n \times n$ sur K.*

En applicant le théorème 2.1 dans notre estimateur u, nous avons donc :

$$X = U\Sigma V' \iff X'X = V\Sigma^2 V' \text{ et } N = V\Sigma W^*\Sigma V',\ \text{où } W^* = U'WU$$

$$\iff \Omega^* = \left[(s-1)\hat{\gamma}^2\Sigma^2 + 2\hat{\sigma}^2(s\Sigma W^*\Sigma - \hat{\lambda}\mathbb{I})\right]$$

$$\iff \hat{u} = \hat{\gamma}(s-1)V{\Omega^*}^{-1}\Sigma U'Y \tag{2.13}$$

D'où, la matrice Ω^* est une matrice carrée diagonale et elle est donc pseudo-inversible.

Remarque 2.2 *Si nous poussons le valeur de réglage à qui tend à 0 (i.e. $s \to 0$), nous obtiendrons l'estimateur de moindre carrés, car $\lambda = su'Nu$ et la partie droite de la matrice Ω, et aussi Ω^*, s'annule, et au fur et mesure, que le valeur de réglage s increment la matrice Ω est pénalisé par la fonction de pertinence structurelle et λ, en retrayant le diagonale de la matrice non inversible X'X.*

Propriétés de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$

1. **Biais de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$**
   Revenons aux estimateur de moindre carré (MC) et notre estimateur, nous avons :

$$\hat{\boldsymbol{\beta}}_{MC} = (X'X)^{-1}X'Y$$
$$\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}^2(s-1)\Omega^{-1}X'Y$$

   En multipliant l'estimateur MC à gauche par $X'X$, nous avons $(X'X)^{-1}\hat{\boldsymbol{\beta}}_{MC} = X'Y$, cela nous donne alors :

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{xbry} =& \hat{\gamma}^2(s-1)\left[(s-1)\hat{\gamma}^2X'X + 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I})\right]^{-1}X'X\hat{\boldsymbol{\beta}}_{MC} \\
&\text{Nous posons : } \Psi = 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I}) \\
\hat{\boldsymbol{\beta}}_{xbry} =& \left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\left[(s-1)\hat{\gamma}^2X'X + \Psi - \Psi\right]\hat{\boldsymbol{\beta}}_{MC} \\
\hat{\boldsymbol{\beta}}_{xbry} =& \hat{\boldsymbol{\beta}}_{MC} - \left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC} \qquad (2.14)
\end{aligned}$$

   En appliquant l'espérance à l'equation (2.14) :

$$\mathbb{E}\left[\hat{\boldsymbol{\beta}}_{xbry}\right] = \boldsymbol{\beta}_{MC} - \mathbb{E}\left[\left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC}\right] \qquad (2.15)$$

   Nous avons donc un biais de : $\mathbb{E}\left[\left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC}\right]$, et nous pouvons conclure que notre estimateur $\hat{\boldsymbol{\beta}}_{xbry}$ est baisse, sauf dans les valeur extremes de la valeur de réglage $s \in \{0, 1\}$

2. **Variance de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$**
   Calculer la variance de cet estimateur devient un peu complique, car il y a trop de variables aléatoires, à savoir : $\gamma, \sigma^2, \lambda$.

$$\mathbb{V}\left[\hat{\boldsymbol{\beta}}_{xbry}\right] = (s-1)^2\mathbb{V}\left[\hat{\gamma}^2\left[(s-1)\hat{\gamma}^2X'X + 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I})\right]^{-1}X'Y\right] \qquad (2.16)$$

   Nous avons donc laissé par la suite du travail, afin de pouvoir de trouver une méthode qui nous aide à comparer la variance de notre estimateur avec celui de moindre carrées, et de confirmer que elle est plus faible, et ainsi, il vise mieux l'estimation.

Algorithme itératif pour le modèle

En vue de que nous ne pouvons pas obtenir les estimateurs analytiquement (i.e. des formules qui font juste participer X et Y), nous allons donc écrire un algorithme itératif afin de trouver le maximum de la fonction $\mathcal{G}'$, et les valeurs optimales, à savoir : $\hat{u}, \hat{\gamma}$ et $\hat{\sigma}^2$.

Cet algorithme se fera en deux étapes : (1) nous aurons l'estimateur fixé $\hat{u}_{[t]}$ et nous calculerons les autres estimateur avec celui-ci, (2) avec les autres estimateurs calculés $\hat{\lambda}, \hat{\gamma}$ et $\hat{\sigma}^2$, nous calculerons le nouveau valeur de l'estimateur $\hat{u}_{[t+1]}$, jusqu'à que le norme carrée de la soustraction de estimateur auparavant $\hat{u}_{[t]}$ et le nouveau converge $\hat{u}_{[t+1]}$: $\left\|\hat{u}_{[t]} - \hat{u}_{[t+1]}\right\|^2 > 10^{-6}$ ou $\langle\, \hat{u}_{[t]} \mid \hat{u}_{[t+1]} \,\rangle^2 < 1 - 10^{-6}$.

Algorithm 1 Regression par pénalisation et T vide

Entrée: Échantillon (Y, X), valeur de réglage s, matrice de points W
Sortie: Valuers d'estimateurs $\hat{u}, \hat{\gamma}$ et $\hat{\sigma}^2$

$\hat{u}_{[1:n]} := 10^{-20}$ et $t := 1$
while $\langle \hat{u}_{[t]} | \hat{u}_{[t-1]} \rangle^2 < 1 - 10^{-6}$ **do**
  $\hat{\gamma}_{[t]} := \langle\ X\hat{u}_{[t]} \mid Y\ \rangle * \left\| X\hat{u}_{[t]} \right\|^{-2}$
  $\hat{\sigma}^2_{[t]} := \left\| Y - \hat{\gamma}_{[t]} X\hat{u}_{[t]} \right\|^2 * n^{-1}$
  $\hat{\lambda}_{[t]} := s * \hat{u}'_{[t]} X'WX\hat{u}_{[t]}$
  $\Omega^*_{[t]} := \left[(s-1)\hat{\gamma}^2_{[t]}\Sigma^2 + 2\hat{\sigma}^2_{[t]}(s\Sigma W^*\Sigma - \hat{\lambda}_{[t]}\mathbb{I}) \right]$
  $\hat{u}_{[t+1]} := \hat{\gamma}_{[t]}(s-1)V\Omega^{*-1}_{[t]}\Sigma U'Y$
  $t := t + 1$
end while
return $\hat{u}, \hat{\gamma}$ et $\hat{\sigma}^2$

Nous obtiendrons avec cet algorithme les valeurs optimales et nous pourrons régresser afin de prédire les nouvelles valeurs y_{n+1} et mesurer la qualité de notre estimateur $\boldsymbol{\beta}xbry$ dans la suite du travail.

Remarque 2.3 *Nous pouvons nous rendre compte facilement que le estimateur $\hat{u_1}$ devient le vecteur propre de la premier composante principale $f^1 = X\hat{u_1}$, grâce à la fonction de pertinence structurelle choisi $S(u) = su'Nu$.*

Dans la suite du travail, nous allons considérer le cas où notre matrice T qui rassemblera les composantes, ne sera pas vide.

2.2.2 Cas où T n'est pas vide

Après avoir trouvée la première composante principal et avoir effectué la prevision avec cette dernière. Nous allons à présent nous poser, dans cette section, le cas où la matrice T n'est pas vide.

Partant de ce dernier fait, T sera défini comme la matrice qui rassemblera toutes les autres composantes principales trouvée par chaque itération, en sachant que chaque composante principales est défini comme : $f^i = X\hat{u}_i\ , \forall i \in \{1, ..., K\}$ où K est el nombre de composante principales permise par la variable X et u_i le vecteur propre de la composante principale i. Ainsi donc, la matrice T sera fixé comme suit : $T_{n\times k} = \{f^1, f^2, ..., f^K\}$, où n est le nombre d'échantillons.

Définition 2.1 *Nous soumettrons à notre programme à optimiser la contrainte d'orthogonalité aux composantes principales précédents. C'est-à-dire : $\forall k \in \{1, ..., K\}$, $f_k = Xu_k$ et $F^{k-1} = [f^1, ..., f^{k-1}]$;*

$$\forall k : f^1, ..., f^{k-1} \perp f^k \iff F^{k-1}Wf^k = 0 \iff F^{k-1}WXu_k = 0$$

D'où, la covariance des composantes principales est :$cov(f_i, f_j) = 0, \forall i, j \in \{1, ..., K\}$ et $j \neq i$.

Propriété 2.1 *Pour tout : $i, j \in \{1, ..., K\}$ et $i \neq j$ tel que $f_i \perp f_j$ et $cov(f_i, f_j) = 0$, alors f_i et f_j ne sont pas corrélés, et de plus, la matrice $F^k F'^k$ est inversible.*

Nous allons donc essayer de maximiser le programme $\mathcal{G}$ avec la nouvelle matrice $T_{n\times k}$, nous pouvons donc réécrire la vraisemblance comme suit :

$$\mathcal{L}_Y(X,T;u,\delta,\gamma,\sigma^2)=\prod_{i=1}^{n} f_Y(y_i)=\left(\frac{1}{2\pi\sigma^2}\right)^{n/2}\exp\left(-\frac{1}{2\sigma^2}\left\|Y-X\boldsymbol{\beta}-T\delta\right\|^2\right) \tag{2.17}$$

Et nous pouvons aussi réécrire leur log-vraisemblance :

$$\ell_Y=-\frac{n\ln(2\pi)}{2}-\frac{n\ln(\sigma^2)}{2}-\frac{\left\|Y-X\boldsymbol{\beta}-T\delta\right\|^2}{2\sigma^2} \tag{2.18}$$

En remplaçant la fonction de pertinence structurelle $S(u)=u'Nu$, la variable $\boldsymbol{\beta}=\gamma u$, et en enlevant les termes qui sont constants du programme $\mathcal{G}$, nous aurons donc le nouveau programme à maximiser $\mathcal{Q}$ avec leurs contraintes :

$$\mathcal{Q}=\underset{\substack{\delta,\gamma,\sigma^2,u\\ u'u=1\\ u_k'X'WXu=0}}{\arg\max}\left[(s-1)\frac{n\ln(\sigma^2)}{2}+(s-1)\frac{\left\|Y-\gamma Xu-T\delta\right\|^2}{2\sigma^2}+su'Nu\right] \tag{2.19}$$

D'où, u_k est le vector propre de la composante principale k, et d'après la propriété 2.1, ils sont orthogonales deux à deux (i.e. $u_i\perp u_j,\forall i\neq j$). Nous allons donc poser les suivantes équivalente à la contrainte $u_k'X'WXu=0$ pour simplifier le calculs des estimateurs :

$$\text{On pose donc : } A_k'=u_k'X'WX \iff A_k'u=0,\forall k\in\{1,2,...,K\}$$

A première vue, la maximisation du programme $\mathcal{Q}$ n'est pas évidante, à cause de leur complexité en derivation et la dépendance circulaire entre les estimateurs, autrement dit, le estimateur γ dépend de δ (i.e. $\hat{\gamma}=\phi(\delta)$), et au même temps, δ dépend de γ (i.e. $\hat{\delta}=\phi(\gamma)$).

Par conséquent, afin de maximiser le programme $\mathcal{Q}$ avec leurs contraintes, nous allons proposer un algorithme itératif à deux étapes, avec un nombre maximum d'iterations égal à K, ce dernier étant le nombre de composantes principales.

Alors, nous commence algorithme avec $t=1$ et $t\in\{1,...,K\}$

Étape 1 : En fixant d'abord le vecteur $\hat{u}_{[t]}$, le programme $\mathcal{Q}$ devient donc :

$$\mathcal{Q}'=\underset{\delta,\gamma,\sigma^2}{\arg\max}\left[n\ln(\sigma^2)+\frac{\left\|Y-\gamma X\hat{u}_{[t]}-T\delta\right\|^2}{\sigma^2}\right] \tag{2.20}$$

Car la fonction de pertinence structurelle devient constante (i.e. $S(\hat{u}_{[t]})=\hat{u}'_{[t]}N\hat{u}_{[t]}$) et nous avons aussi supprimées les autres variables constantes. Nous pouvons constater de ce dernier programme $\mathcal{Q}'$ adopte la forme d'un problème de moindre carrée ordinaire (i.e. regression linéaire multiple), puisque $X\hat{u}_{[t]}$ et T sont constantes, nous pouvons donc poser le modèle suivant :

$$\mathcal{Q}''=\underset{\delta,\gamma,\sigma^2}{\arg\max}\left[n\ln(\sigma^2)+\frac{\left\|Y-Z\Lambda\right\|^2}{\sigma^2}\right] \tag{2.21}$$

D'où, $Z=(Xu,T)$ et $\Lambda=(\gamma,\delta)'$, et la solution qui maximise ce programme est par définition $\hat{\Lambda}=(Z'Z)^{-1}Z'Y$. Cette solution et la propriété 2.1 nous assurent que la

matrice $Z'Z$ est inversible car toutes les colonnes sont des composantes principales indépendantes. Cependant, lors de l'optimisation computationnelle, cette dernière matrice n'est pas inversible car nous ne connaissons pas le valeur optimal de u, nous allons donc utiliser la *Décomposition en valeurs singulières* pour calculer leur pseudo-inverse, nous obtenons donc : $Z = U\Sigma V' \iff \hat{\Lambda} = V\Sigma^{-1}U'Y$. Enfin, la solution optimale obtenue est : $\hat{\Lambda} = (\hat{\gamma}, \hat{\delta})'$.

En outre, l'estimateur baissé obtenu de $\hat{\sigma}^2$ est le suivant :

$$\hat{\sigma}^2 = \frac{\left\|Y - Z\hat{\Lambda}\right\|^2}{n} \tag{2.22}$$

La figure 2.1 ci-dessous fait une représentation géométrique de l'étape 1, où nous devons trouver le meilleur projection de Y sur le plan Z représenté par $Xu_{[t]}$ et T.

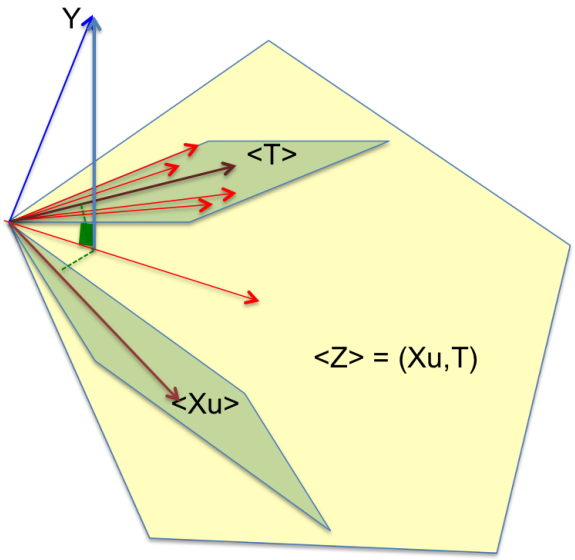

FIGURE 2.1 – *Représentation géométrique de l'étape 1*

Étape 2 : En fixant les estimateurs calculés à l'étape précédente : $\hat{\delta}_{[t]}, \hat{\gamma}_{[t]}$, $\hat{\sigma^2}_{[t]}$ et le vecteur $\hat{u}_{[t]}$ déjà fixé, nous allons à présent calculer le vecteur optimal $\hat{u} = \hat{u}_{[t+1]}$ inconnu soumis au programme $\mathcal{Q}$ qui devient donc :

$$\mathcal{Q}''' = \underset{\substack{u,\ u'u=1 \\ A_k'u=0 \\ \text{où } A_k'=u_kX'WX}}{\arg\max} \left[(s-1)\frac{\left\|Y - \hat{\gamma}_{[t]}Xu - T\hat{\delta}_{[t]}\right\|^2}{2\hat{\sigma}^2_{[t]}} + su'Nu \right] \tag{2.23}$$

En vue de maximiser le programme $\mathcal{Q}'''$, nous allons utiliser la méthode des *Multiplicateurs de Lagrange* afin de trouver les valeurs optimales qui maximisent le programme avec contraintes. Nous allons poser par lisibilité : $\hat{\delta} = \hat{\delta}_{[t]}, \hat{\gamma} = \hat{\gamma}_{[t]}$ et $\hat{\sigma^2} = \hat{\sigma^2}_{[t]}$, et nous aurons donc le fonction L à dériver.

$$L(u, \lambda, \tau) = (s-1)\frac{\left\|Y - \hat{\gamma}Xu - T\hat{\delta}\right\|^2}{2\hat{\sigma}^2} + su'Nu - \lambda(u'u - 1) - \tau' A_k'u \tag{2.24}$$

Dérivée par rapport à λ et τ :

$$\frac{\partial L}{\partial \lambda} = 0 \iff u'u = 1 \quad \text{et} \quad \underset{\tau}{\nabla} L = 0 \iff A_k' u = 0 \tag{2.25}$$

Dérivée par rapport à u : Nous aurons donc l'estimateur $\hat{u}_{[t+1]}$ qui dépendra des paramètres : $\hat{u}_{[t]}, \hat{\gamma}, \hat{\sigma}^2, \hat{\delta}, \hat{\lambda}$ et $\hat{\tau}$.

$$\begin{aligned}
\underset{u}{\nabla} L = 0 &\iff (1-s)\frac{\hat{\gamma}X'(Y - \hat{\gamma}Xu - T\hat{\delta})}{\hat{\sigma}^2} + 2sNu - 2\lambda u - A_k\tau = 0 \qquad (2.26)\\
&\iff \left[(s-1)\hat{\gamma}^2 X'X + 2\hat{\sigma}^2(sN - \lambda\mathbb{I})\right] u = \hat{\gamma}(s-1)X'\left(Y - T\hat{\delta}\right) + \hat{\sigma}^2 A_k\tau \\
&\text{Nous posons : } \Omega = \left[(s-1)\hat{\gamma}^2 X'X + 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I})\right] \\
&\iff \hat{u}_{[t+1]} = \Omega^{-1}\left[\hat{\gamma}(s-1)X'\left(Y - T\hat{\delta}\right) + \hat{\sigma}^2 A_k\hat{\tau}\right] \qquad (2.27)
\end{aligned}$$

Comme dans le premier cas, où T était vide, la matrice Ω n'est pas inversible lors que nous cherchons le valeur optimal du vecteur $\hat{u}_{[t+1]}$. Nous allons donc utiliser encore un fois la *Décomposition en valeurs singulières* afin de calculer la pseudo-inverse de Ω, nous obtenons donc :

$$\begin{aligned}
&\text{En posant : } \Omega^* = \left[(s-1)\hat{\gamma}^2\Sigma^2 + 2\hat{\sigma}^2(s\Sigma W^*\Sigma - \hat{\lambda}\mathbb{I})\right], \text{ où : } W^* = U'WU \\
X = U\Sigma V' &\iff \hat{u}_{[t+1]} = V\Omega^{*-1}\left[\hat{\gamma}(s-1)\Sigma U'\left(Y - T\hat{\delta}\right) + \hat{\sigma}^2\Sigma W^*\Sigma V' u_k\hat{\tau}\right] \qquad (2.28)
\end{aligned}$$

D'où, la matrice Ω^* est une matrice carrée diagonale et elle est donc pseudo-inversible.

Nous calculerons à présent le paramètre multiplicateur $\hat{\lambda}$ en multipliant par u' à l'équation (2.26) :

Nous posons d'abord : $\Delta = \frac{(1-s)\hat{\gamma}}{2\hat{\sigma}^2}\left[\langle\, Xu \mid Y \,\rangle - \hat{\gamma}\,\|Xu\|^2 - \langle\, Xu \mid T\hat{\delta} \,\rangle\right]$

$$\begin{aligned}
&u' \times (2.26) \iff \Delta + 2su'Nu - 2\lambda u'u - (A_k'u)'\tau = 0 \\
&\text{D'après les équations (2.25) : } u'u = 1 \text{ et } A_k'u = 0 \\
&\iff 2\Delta + 2su'Nu - 2\lambda = 0 \\
&\iff \hat{\lambda} = \Delta_{[t]} + s\hat{u}_{[t]}'N\hat{u}_{[t]} \qquad (2.29)
\end{aligned}$$

Nous calculerons enfin le vecteur multiplicateur $\hat{\tau}$ en multipliant par A_k' à l'équation (2.26) :

$$\begin{aligned}
A_k' \times (2.26) &\iff (A_k'A_k)\,\hat{\tau} = A_k'\left[\frac{(1-s)\hat{\gamma}}{\hat{\sigma}^2}X'(Y - \hat{\gamma}Xu - T\hat{\delta}) + 2sNu\right] - 2\lambda\underbrace{A_k'u}_{0} \\
&\iff \hat{\tau} = \left[A_k'A_k\right]^{-1}A_k'\left[\frac{(1-s)\hat{\gamma}}{\hat{\sigma}^2}X'(Y - \hat{\gamma}Xu - T\hat{\delta}) + 2sNu\right] \qquad (2.30)
\end{aligned}$$

Comme la matrice $A_k'A_k$ est composé des composante principales indépendantes, alors cette dernière est fortement inversible, nous n'avons pas besoin de calculer leur pseudo-inverse avec SVD.

Condition : Enfin, nous évaluons la convergence de la soustraction de la norme carrée du vecteur $\hat{u}_{[t]}$ fixé dans la étape 1 et du vecteur $\hat{u}_{[t+1]}$ de l'étape 2 : $\left\|\hat{u}_{[t]} - \hat{u}_{[t+1]}\right\|^2 > 10^{-6}$ ou $\langle\, \hat{u}_{[t]} \mid \hat{u}_{[t+1]} \,\rangle^2 < 1 - 10^{-6}$.

Propriétés de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$

1. **Biais de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$**
   Revenons aux estimateur de moindre carré (MC) et notre estimateur, nous avons :

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{MC} &= (X'X)^{-1}X'Y \\
\hat{\boldsymbol{\beta}}_{xbry} &= \Omega^{-1}\left[\hat{\gamma}^2(s-1)X'\left(Y - T\hat{\delta}\right) + \hat{\gamma}\hat{\sigma}^2 A_k\hat{\tau}\right]
\end{aligned}$$

   Nous allons poser : $\Xi = \left[\hat{\gamma}^2(s-1)T\hat{\delta} + \hat{\gamma}\hat{\sigma}^2 A_k\hat{\tau}\right]$ et $\Psi = 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I})$, et de plus, nous savons que $(X'X)^{-1}\hat{\boldsymbol{\beta}}_{MC} = X'Y$, ces equivalences nous donne alors :

$$\begin{aligned}
\hat{\boldsymbol{\beta}}_{xbry} =& \hat{\gamma}^2(s-1)\Omega^{-1}X'Y - \Omega^{-1}\Xi \\
\hat{\boldsymbol{\beta}}_{xbry} =& \hat{\gamma}^2(s-1)\left[\underbrace{(s-1)\hat{\gamma}^2X'X + \Psi}_{\Omega}\right]^{-1} X'X\hat{\boldsymbol{\beta}}_{MC} - \Omega^{-1}\Xi \\
\hat{\boldsymbol{\beta}}_{xbry} =& \left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\left[(s-1)\hat{\gamma}^2X'X + \Psi - \Psi\right]\hat{\boldsymbol{\beta}}_{MC} - \Omega^{-1}\Xi \\
\hat{\boldsymbol{\beta}}_{xbry} =& \hat{\boldsymbol{\beta}}_{MC} - \left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC} - \Omega^{-1}\Xi
\end{aligned} \tag{2.31}$$

   En appliquant l'espérance à l'equation (2.31) :

$$\mathbb{E}\left[\hat{\boldsymbol{\beta}}_{xbry}\right] = \boldsymbol{\beta}_{MC} - \mathbb{E}\left[\left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC} - \Omega^{-1}\Xi\right] \tag{2.32}$$

   Nous avons donc un biais de : $\mathbb{E}\left[\left[(s-1)\hat{\gamma}^2X'X + \Psi\right]^{-1}\Psi\hat{\boldsymbol{\beta}}_{MC} - \Omega^{-1}\Xi\right]$, et nous pouvons conclure que notre estimateur $\hat{\boldsymbol{\beta}}_{xbry}$ est baisse.

2. **Variance de l'estimateur $\hat{\boldsymbol{\beta}}_{xbry} = \hat{\gamma}\hat{u}$**
   Calculer la variance de cet estimateur devient un peu complique, car il y a trop de variables aléatoires, à savoir : $\gamma, \sigma^2, \lambda, \delta$ et τ.

$$\mathbb{V}\left[\hat{\boldsymbol{\beta}}_{xbry}\right] = \mathbb{V}\left[\Omega^{-1}\left[\hat{\gamma}^2(s-1)X'\left(Y - T\hat{\delta}\right) + \hat{\gamma}\hat{\sigma}^2A_k\hat{\tau}\right]\right] \tag{2.33}$$

   Nous avons donc laissé par la suite du travail, afin de pouvoir de trouver une méthode qui nous aide à comparer la variance de notre estimateur avec celui de moindre carrées, et de confirmer que elle est plus faible, et ainsi, il vise mieux l'estimation.

Algorithme itératif pour le modèle à deux étapes

En vue de que nous avons modéliser l'optimisation à deux étapes, nous allons écrire deux algorithmes afin de maximiser le programme $\mathcal{Q}$ et de trouver les valeur optimales des estimateurs, à savoir : $\gamma, \sigma^2, \lambda, \delta$ et τ.

L'algorithme 2 retrouve les estimateur optimaux en fonction de la matrice d'entrée T ayant $K-1$ composantes principales retrouvées auparavant et un ensemble de paramètres de réglages s ($s \in [0,1]$). Nous aurons comme sortie les coefficients optimaux $\hat{\boldsymbol{\beta}}^k = \hat{\lambda}\hat{u}^k$ et le paramètre de réglage optimal $\hat{s}^k$ (où k représente la k-ième régression effectuée en utilisant comme pénalisation les $K-1$ premières composantes principales retrouvée auparavant).

Algorithm 2 Regression par penalisation

Entrée: Échantillon (Y, X), matrice de points W et la matrice de composantes principales $T = \{Xu^1, Xu^2, ..., Xu^{k-1}\}$.
Sortie: Valuers d'estimateurs $\hat{u}, \hat{\gamma}, \hat{\delta}$ et $\hat{\sigma}^2$

1: Initialisation du paramètre de réglage $s = \{0.001, 0.002, ..., 0.999\}$
2: Initialisation des variables $\hat{u}_{[1:n]} = 10^{-20}$ et $t = 1$
3: **while** $\langle \hat{u}_{[t]} | \hat{u}_{[t-1]} \rangle^2 < 1 - 10^{-6}$ **do**
4: $\quad Z := (Xu_{[t]}, T)$;
5: $\quad (\hat{\gamma}_{[t]}, \hat{\delta}_{[t]}) := (Z'Z)^{-1} Z'Y$
6: $\quad \hat{\sigma}^2_{[t]} := \left\| Y - \hat{\gamma}_{[t]} X \hat{u}_{[t]} \right\|^2 * n^{-1}$
7: $\quad \hat{\lambda}_{[t]} := \Delta_{[t]} + s\hat{u}'_{[t]} N \hat{u}_{[t]}$
8: $\quad \hat{\tau}_{[t]} := [A'_k A_k]^{-1} A'_k \left[\frac{(1-s)\hat{\gamma}_{[t]}}{\hat{\sigma}^2} X'(Y - \hat{\gamma}_{[t]} X \hat{u}_{[t]} - T\hat{\delta}_{[t]}) + 2sN\hat{u}_{[t]} \right]$
9: $\quad \Omega^*_{[t]} := \left[(s-1)\hat{\gamma}^2_{[t]} \Sigma^2 + 2\hat{\sigma}^2_{[t]} (s\Sigma W^* \Sigma - \hat{\lambda}_{[t]} \mathbb{I}) \right]$
10: $\quad \hat{u}_{[t+1]} := V \Omega^{*-1}_{[t]} \left[\hat{\gamma}_{[t]} (s-1) \Sigma U' \left(Y - T\hat{\delta}_{[t]} \right) + \hat{\sigma}^2_{[t]} \Sigma W^* \Sigma V' u_k \hat{\tau}_{[t]} \right]$
11: $\quad t := t + 1$
12: **end while**
13: **return** $\hat{u}, \hat{\gamma}, \hat{\delta}$ et $\hat{\sigma}^2$

L'algorithme 3 execute l'algorithme 2 (i.e. regression par penalisation), en le passant comme argument, incrémentalement, les $K-1$ premières composantes principales retrouvées T, c'est-à-dire, dans chaque itération la matrice T rassemblera les $K-1$ premières composantes afin de l'y utiliser pour pénaliser la regression suivante. Enfin, il retournera la matrice avec les K composantes principales retrouvée T et la matrice de coefficients $\boldsymbol{\beta}$ pour chaque regression éffectuée.

Algorithm 3 Recherche de composantes principales par pénalisation

Entrée: Échantillon (Y, X) et K nb. de composantes.
Sortie: Estimateurs $\hat{u}, \hat{\gamma}, \hat{\delta}$

1: Centrée et réduite chaque variable x^j de $X = \{x^1, x^2, .., x^p\}$, où $x^j \in \mathbb{R}^n$
2: Centrée la variable univarié $Y = \{y^1\}$, où $y^1 \in \mathbb{R}^n$
3: Initialisation des variables $T = \{\}$ et $W = diag\,(1/n)$ et $\boldsymbol{\beta} = \{\}$
4: **for all** $k = 1, 2, ..., K$ **do**
5: $\quad regression := regression\_par\_penalisation(X, Y, W, T)$
6: $\quad T := T \cup \{X * regression.\hat{u}\}$
7: $\quad \boldsymbol{\beta} := \boldsymbol{\beta} \cup \{regression.\hat{\lambda} * regression.\hat{u}\}$
8: **end for**
9: **return** T et $\boldsymbol{\beta}$

Dans la suite du travail, nous allons donner des outils pour comparer la qualité de notre modèle avec les autres modèles existantes dans la littérature (i.e. Ridge, PCR et PLS1).

2.2.3 Validation du modèle de regression

La validation du modèle est une partie importante dans la regression linéaire, car celle-ci nous aide à comprendre si le modèle de regression proposé est mieux performance (i.e. de meilleur

qualité) que les autres. Dans cette sous-section, nous allons décrire les outils nécessaires pour valider notre méthode de regression avec les autres méthodes de regression déjà mentionnées auparavant.

Centrage et réduction

En général, las variables explicatives sont parfois à différentes échelles de mesures et par conséquence les coefficients $\boldsymbol{\beta}$ vont dépendre de ces échelles de mesures qui donnent parfois plus d'importance à certains coefficients qu'autres (ou plus de variabilité).

Ainsi, à la différence de la régression classique, où les variables sont en général conservées telles que mesurées, il est d'usages de centrer et réduire les variables explicatives[1]. Nous allons donc dorénavant considérer, dans le modèle de regression, la variable centrée et réduite suivante :

$$\tilde{X} = (X_j - \bar{x}_j \mathbb{I})/\hat{\sigma}_{X_j} \tag{2.34}$$

D'où, $\bar{x}_j$ est la moyenne empirique de la colonne j de la matrice X (i.e. $\bar{x}_j = \sum_{i=1}^{n} x_{ij}/n$) et $\hat{\sigma}_{X_j}$ est la variance empirique de la même colonne (i.e. $\hat{\sigma}_{X_j} = \sum_{i=1}^{n}(x_{ij} - \bar{x}_j)^2/n$).

Nous allons aussi centrée la variable Y afin d'enlever le coefficient associé à la variable $\mathbb{1}$ du modèle de regression (aussi appelé *coefficient constant* et *intercept* en anglais), nous allons donc dorénavant considérer, la variable centrée :

$$\tilde{Y} = Y - \bar{y} \tag{2.35}$$

Pour un valeur fixe de $\tilde{s}$, notre estimateur à ajuster sera :

$$\begin{aligned} \hat{u} &= \Omega^{-1}\left[\hat{\gamma}(s-1)\tilde{X}'\left(\tilde{Y} - T\hat{\delta}\right) + \hat{\sigma}^2 A_k \hat{\tau}\right] \\ \text{où } \Omega &= \left[(s-1)\hat{\gamma}^2\tilde{X}'\tilde{X} + 2\hat{\sigma}^2(sN - \hat{\lambda}\mathbb{I})\right] \text{ et } N = \tilde{X}W\tilde{X} \end{aligned} \tag{2.36}$$

Afin de prédire une nouvelle valeur $x'_{n+1} = (x_{n+1,1}, ..., x_{n+1,p})$, en tenant en compte que les variables de la matrice X ont été centrée et réduites, et que Y a aussi été centrée, y_{n+1} s'écrira comme suit :

$$\hat{y}^p_{xbry,n+1} = \left[\sum_{j=1}^{p}\left(\frac{x_{n+1,j} - \bar{x}_j}{\hat{\sigma}_{X_j}}\left[\hat{\beta}_{xbry}(\tilde{s})\right]_j\right)\right] + \left[\sum_{j=1}^{p}\left(\frac{x_{n+1,j} - \bar{x}_j}{\hat{\sigma}_{X_j}}\left[\hat{u}_k(\tilde{s})\hat{\delta}(\tilde{s})\right]_j\right)\right] + \bar{y} \tag{2.37}$$

où $\hat{\boldsymbol{\beta}}_{xbry}(\tilde{s}) = \hat{\gamma}\hat{u}(\tilde{s})$ et $\hat{u}_k(\tilde{s})$: sont les k premiers vecteurs propres de k composantes.

Le $\hat{\boldsymbol{\beta}}_{xbry}(\tilde{s})$ est calculé à partie de l'algorithme interactif (1) ou (2). Une version vectorielle la plus intuitive de l'equation précédent serait :

$$\hat{y}^p_{xbry,n+1} = \tilde{X}_{n+1}\hat{\beta}_{xbry}(s) + \tilde{T}_{n+1}\hat{\delta} + \bar{y}, \text{ d'où, } \tilde{T}_{n+1} = \tilde{X}_{n+1}\hat{u}_k \tag{2.38}$$

Choix de $\hat{s}$

Afin de parvenir à choisir une optimale valeur de s, désormais nommé $\hat{s}$, nous allons utiliser une technique déjà expliquée dans le chapitre 2, appelé validation croisée (ou *Cross Validation* en anglais), car il est pratiquement impossible de l'y calculer avec une méthode analytique. Cette valeur $\hat{s}$ s'ajustera à nos données et elle sera donc stochastique.

Apprentissage et Validation

La procédure de validation consiste à séparer de manière aléatoire les données en deux parties distinctes $(\tilde{X}_a, \tilde{Y}_a)$ et $(\tilde{X}_v, \tilde{Y}_v)$ (où a : apprentissage et v : validation)[1]. Le modèle de regression utilisera le jeu des données d'apprentissage $(\tilde{X}_a, \tilde{Y}_a)$ pour calculer les coefficients $\hat{\boldsymbol{\beta}}_{xbry}(s)$ pour chaque valeur d'une grille de valeurs pour s, comprises entre 0 et une valeur maximale (dans notre cas 1). Ensuite, en utilisant tous ces coefficients et l'échantillon de validation $(\tilde{X}_v, \tilde{Y}_v)$, nous allons calculer les valeurs prédites $\hat{Y}^p_{xbry,v}(s)$ pour chaque valeur s en utilisant l'equation 2.37.

Afin de mesurer la qualité du modèle, nous allons utiliser la mesure de la distance entre les vraies observations et les valeurs prédites, par un critère. Le critère utilisé par chaque s sera le PRESS[1] :

$$PRESS(s) = \left\| \hat{Y}^p_{xbry,v}(s) - Y_v \right\|^2 \tag{2.39}$$

Validation croisée

En vue de trouver la valeur optimale de s, nous allons donc utiliser deux outils, le premier sera le PRESS mesurant la qualité du modèle par rapport une grille de valeurs pour s et le second sera la *validation croisée* en coupant l'échantillon à différentes ensembles de deux parties distinctes $(\tilde{X}_{a_1,\ldots a_n}, \tilde{Y}_{a_1,\ldots a_n})$ et $(\tilde{X}_{v_1,\ldots,v_n}, \tilde{Y}_{v_1,\ldots,v_n})$. Ensuite, nous allons choisir un grille de valeurs possibles pour s et après nous choisirons le valeur $\hat{s}$ qui minimise le critère PRESS.

$$\hat{s} = \underset{s \in \,]0,1[}{\arg\min} \left\| \hat{Y}^p_{xbry,v}(s) - Y_v \right\|^2 \tag{2.40}$$

[1] Predicted residual error sum of squares

3

Résultats et Conclusion

Dans ce chapitre, nous allons présenter les résultats expérimentaux de la comparaison de notre modèle de regression proposé contre les modèles de regression décrits dans le chapitre 2 (i.e. Ridge, PCR et PLS1).

Nous allons découper ce chapitre en trois sections : description du jeu de données, évaluation de la regression avec la premier composante trouvée, évaluation de la regression avec K composantes permises, et finalement, conclusion et travaux futurs.

3.1 Jeu de données

Ce jeu de données a été cité par Brown et al. (2001)[5]. Nous sommes en présence de biscuits non cuits pour lesquels nous souhaitons connaître rapidement et à moindre coût, la composition en quatre ingrédients : les lipides, les sucres, la farine et l'eau [1]. Il existe actuellement des méthodes de chimie analytique efficaces pour retrouver et mesurer la composition des biscuit, cependant elles sont assez longues et coûteuses et ne peuvent pas être mise en ligne sur un chaine de production.

Il est nécessaire de remplacer ce processus par un autre moins coûteux, ce processus est la mesure d'un spectre d'absorbance dans le domaine proche infrarouge (ou spectre proche infrarouge), et pour y savoir si le nouveau processus est possible d'y le mettre en ligne, nous allons tenter d'expliquer avec les méthodes de regression la composition des biscuit par le spectre.

Ce spectre mesure l'absorbance à une longueur d'onde donnée, pour tous les longueurs d'ondes entre 1100 et 2498 manomètres et régulièrement espacées de 2 manomètres. Nous aurons donc un peu près 700 variables potentiellement explicatives.

Comme nous allons travailler avec la validation croisée, nous allons couper notre échantillons en deux parties, chaque échantillon seront mesurés par les spectres proches infrarouges et ensuite ils seront mesuré par les méthodes classiques de chimie analytique pour connaître leurs vraies compositions, nous aurons donc ; l'échantillons pour l'apprentissage et l'échantillons pour la validation. L'échantillon d'apprentissage sera composé de $n_a = 40$ biscuit non cuits. L'échantillon de validation comportera de $n_v = 32$ biscuit non cuits et ne sera jamais utilisé pour estimer les coefficients du modèle quel qu'il soit, car il va nous servir pour com-

parer une méthode de regression avec une autre et à connaître leur capacité de prévision. Or, nous ne nous poserons donc pas de question de cette répartition de l'échatillon.

Nous avons pu constater que cet échantillons a quatre variables à expliquer (modèle multivariée), à savoir ; les lipides, les sucres, la farine et l'eau, toutefois nous allons nous intéresser uniquement au pourcentage de sucres (i.e. modèle univarié).

Nous sommes à présent dans le cas où l'estimateur de moindre carrés classiques $(X'X)^{-1}X'Y$ n'est pas défini, car nous avons $p = 700$ variables explicatives et $n_a = 40$ individus, le rang de X'X est donc 40 ici.

Dans la suite, nous appliquerons les modèles de regression adaptés à ce dernière contrainte où $(X'X)^{-1}$ n'est pas inversible et comparer leur capacité de prédiction de chacun.

Remarque 3.1 *Les variables explicatives seront dorénavant centrées et réduites, et les variables à expliquer seront juste centrées.*

3.2 Première Composante Principale

Nous d'abord comparer notre nouvelle approche de regression avec la première composantes trouvée, c'est-à-dire, nous allons régresser avec T vide.

La figure ci-dessous, nous montre l'evolution des paramètre de réglage, dans le cas de la regression ridge et notre regression proposée, et l'erreur moyenne quadratique.

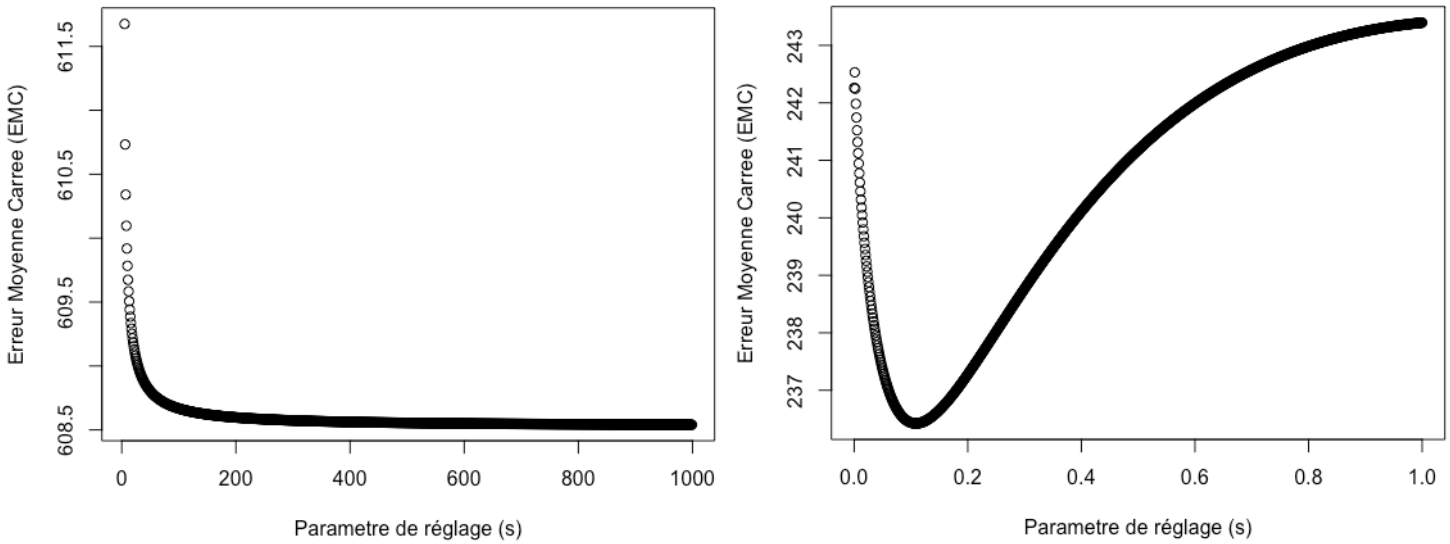

FIGURE 3.1 – *Relation entre le paramètre de réglage et EMQ.* Ce deux graphiques montre l'evolution des erreurs moyenne quadratique par rapport à l'intervalle $s, k \in [0, 1]$. A gauche s'affiche l'evolution du paramètre s de la nouvelle approche et à droite s'affiche l'evolution du paramètre k de la regression ridge.

Remarque 3.2 *Chaque fois que nous cherchons à optimiser la méthode de régression proposée, nous chercherons le valeur optimale du paramètre de réglage (s) qui minimise l'erreur quadratique (EQM).*

Nous allons nous servir des outils décrits dans la sous-section 2.2.3 pour pouvoir comparer la qualité de prediction de notre méthode de regression avec T vide contre toutes les autres.

Ci-dessous se trouve le tableau comparatif des modèles de régression. Cette comparaison se fait par rapport à notre échantillon de validation et non l'échantillon d'apprentissage lequel nous avons utilisé pour trouver les coefficients.

Modèle de Regression	**Paramètre de réglage/nb. Composantes**	**EMQ***
Moindre carrés multiple	—	4304
Ridge	0.1081081	4.95
Composantes principales (PCR)	3 Comps.	1.03
Moindres carrés partiels (PLS)	5 Comps.	0.78
Nouvelle approche	T vide et 0.001	14.79

Nous pouvons remarque que le modèle régression proposé bat considérablement à la regression de moindre carré multiple et qu'il s'approche à avoir une meilleur precision de prediction que la regression ridge et les autres, nous allons regarde dans la suite comme notre approche se comporte si nous utilisons les composantes principales trouvées comme pénalisation.

3.3 K Composante principale permises

Nous allons choisir d'abord le nombre de composantes principales par penalisation que nous utiliserons, dans ce cas expérimental, nous avons choisi $K = 10$ car au-delà de cela, nous considérons que nous trouverons de bruit sur nous données.

La figure ci-dessous 3.2 nous affiche l'evolution de l'erreur quadratique par rapport aux K premières composantes principales utilisée par penalisation, c'est-à-dire lors que T est de dimension $(n, 1)$ jusqu'à $(n, K-1)$. A chaque étape que nous trouvons une composante principale, celle-ci pénalisera notre suivante régression afin de mieux fixé notre prediction. Ainsi donc dans la figure ci-dessous 3.2 nous montre qu'en pénalisant avec deux composantes principales, nous pouvons donc trouver un erreur quadratique minimum.

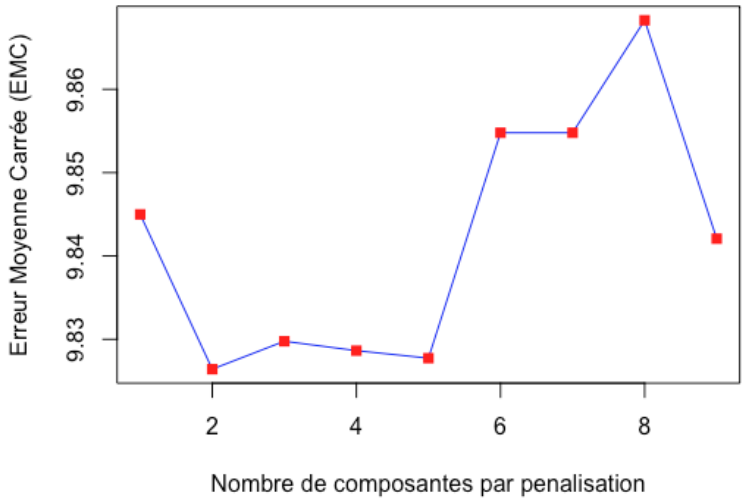

FIGURE 3.2 – *Nb. composantes par penalisation et EMQ.* Cette error quadratique correspond à l'échantillon de validation.

La figure ci-dessous, nous affiche la evolution du paramètre de réglage dans chaque régression effectuée depuis que T est vide jusqu'à T est de dimension $(n, 9)$.

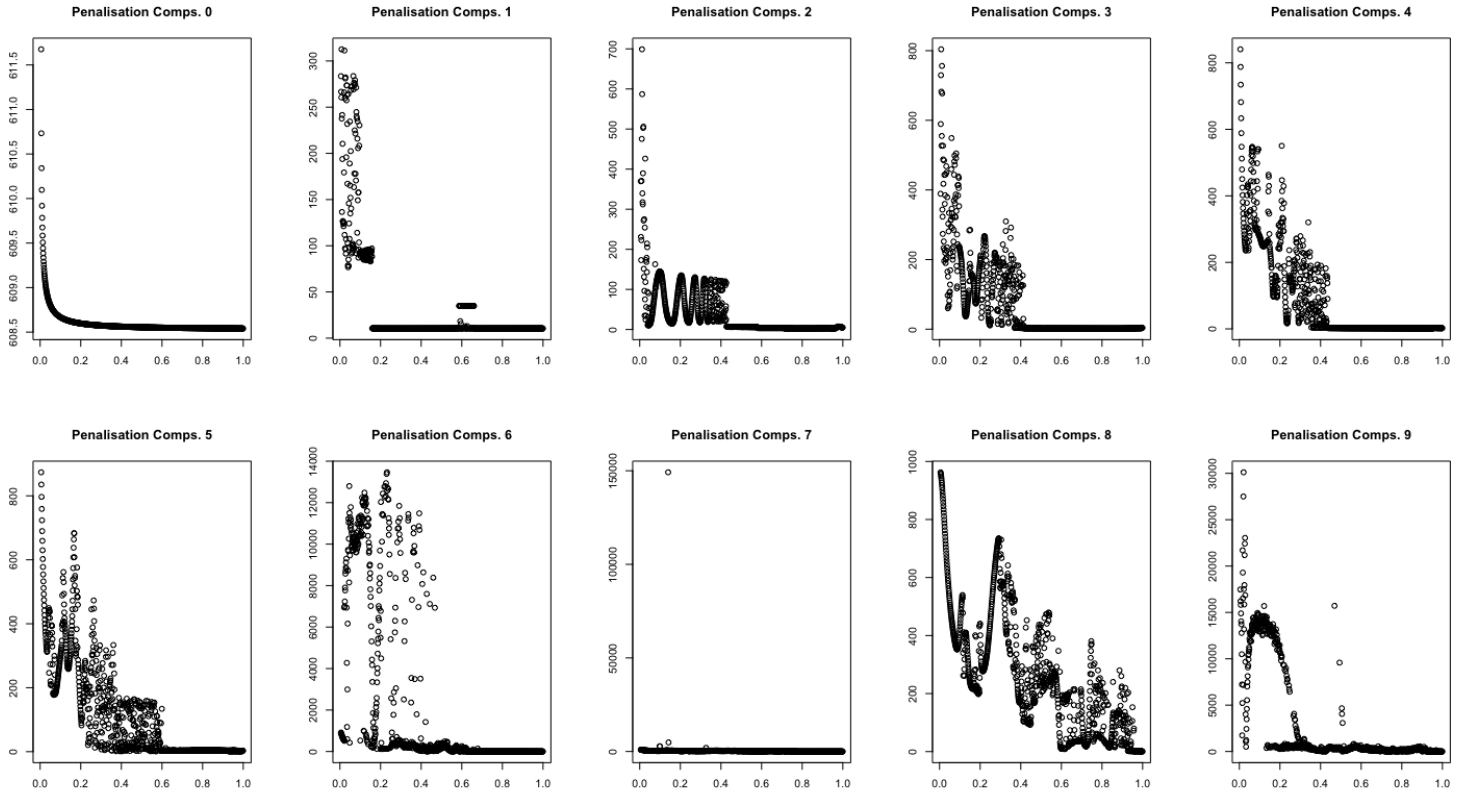

FIGURE 3.3 – *Relation entre le paramètre de réglage de chaque régression et EMQ.* Les abscisses dans la figure sont l'intervalle du paramètre de réglage, et l'ordonnées sont les erreurs quadratiques trouvées avec l'échantillon d'apprendisage.

Ci-dessous se trouve le tableau comparatif des modèles de régression. Cette comparaison se fait par rapport à notre échantillon de validation et non l'échantillon d'apprentissage lequel nous avons utilisé pour trouver les coefficients.

Modèle de Regression	**Paramètre de réglage/nb. Composantes**	**EMQ***
Moindre carrés multiple	—	4304
Ridge	0.1081081	4.95
Composantes principales (PCR)	3 Comps.	1.03
Moindres carrés partiels (PLS)	5 Comps.	0.78
Nouvelle approche	T avec 2 comps. et 0.9391	9.8264
Nouvelle approche	T avec 5 comps. et 0.9451	9.8277

Nous pouvons remarquer que l'erreur quadratique à baisser avec 2 composantes, et qu'avec 5 composantes il est aussi proche au même erreur quadratique, il faut aussi constater que même si nous n'avons pas pu dépasser les autres méthodes, celle-ci est une approche prometteuse à faire évolué.

3.4 Conclusion et Ouvertures

3.4.1 Bilan du travail

Ainsi, ce travail a été plutôt difficile car les différentes techniques et concepts que j'ai appliqués, n'ont pas été évidents à assimiler au premier abord. Il m'a fallu en certain temps de recherche pour bien m'imprégner du sujet. Ainsi donc, j'ai réussi à respecter au maximum les consignes demandées pour ce projet malgré la contrainte du temps.

Après cette première étape importante, je suis parvenu à modéliser la nouvelle approche et d'obtenir les estimateurs optimisant le modèle, pour que je puisse ensuite le comparer avec les autres méthodes de régression existantes dans la littérature.

Ensuite, j'ai pris du temps à analyser les méthodes possibles existantes pour inverser une matrice non définie, car sans cette dernière méthode aurait pas mal de problèmes au moment de l'y passer à l'ordinateur.

Enfin, j'ai pu comparer la nouvelle approche proposée avec ceux existantes dans la littérature (i.e. PLS1, PCR, Ridge).

3.4.2 Bilan personnel

Ce travail m'a permis d'approfondir encore plus mes connaissances en statistique, ou précisément en apprentissage supervisé au niveau de la régression (i.e. prédiction) avec un ensemble de données (i.e. échantillon de validation et d'apprentissage). J'ai aussi pu apprendre quelques concepts sur le domaine de la manipulation de matrices, comme par exemple, la Décomposition en valeur singulières (SVD) afin de calculer la pseudo-inverse d'une matrice.

3.4.3 Ouvertures

La généralisation du modèle de régression univarié à un modele multivariée (i.e. la variable explicative a plusieurs colonnes à expliquer), aurait été intéressante à l'analyser et le comparer avec les méthodes existantes.

Enfin et le plus intéressant aurait été de modéliser autres variantes de la fonction de pertinence structurelle $S(u)$, par exemple : $S(u) = \left(\sum_{j=1}^{J}(u'N_j u)^l\right)^l$.

Bibliographie

[1] Cornillon P., Matzner-Løber E., Régression avec R, Springer Paris, 2011

[2] N.H Bingham, John M.Fry Regression Linear Models in Statistics Springer London, 2010

[3] Manuel LOTH Algorithmes d'Ensemble Actif pour le LASSO Thèse Docteur, Université de Lille

[4] T. Hastie, R. Tibshirani, J. Tibshirani, The Elements of Statistical Learning Springer, 2009, Second Edition

[5] Brown P., Fearn T., Vannucci M. Bayesian wavelet regression on curves with application to a spectroscopic calibration problem. 2001, J. Am. Stat. Assoc., 96(398–408).