Régression bayésienne et sélection de variables

Notes de cours - ENSAI - 2024/2025

Contents

1 Objectifs du cours et rappels

Dans ce cours nous considérons les modèles de régression lorsque **le coefficient de régression est aléatoire**.
Une loi a priori est posée sur ce coefficient et plusieurs méthodes bayésiennes sont proposées pour construire des estimateurs a posteriori ou pour sélectionner les meilleurs coefficients.
Pas mal de références sont dans les livres de Robert et Casella (2005) ou Bishop (2006).
Nous verrons plusieurs modèles et nous étudierons les algorithmes associés. Nous les appliquerons sur des jeux de données via le logiciel R.
Les avantages de cette approche : elle permet de considérer des observations dépendantes et des échantillons de faible taille.

Les objectifs du cours sont :

- Savoir introduire un coefficient aléatoire dans un modèle de régression.
- Savoir manipuler les lois a priori et a posteriori.
- Savoir hiérarchiser un modèle bayésien (schéma hiérarchique).
- Savoir programmer un algorithme de type Gibbs, Metropolis Hasting ou ABC.

1.1 Rappel sur le modèle linéaire

On dispose d'une réalisation de $(Y_1, x_1), \cdots, (Y_n, x_n)$ avec Y_i v.a. indépendantes à valeurs dans $\mathbb{R}$, x_i vecteur de $\mathbb{R}^p$, tels que :

$$\begin{aligned} \mathbb{E}(Y_i) &= \beta_1 x_{i1} + \cdots + \beta_p x_{ip} \\ &= (X\beta)_i, \end{aligned}$$

avec

$$\begin{aligned} X &= \begin{pmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{pmatrix} \\ \beta &= (\beta_1, \cdots, \beta_p)'. \end{aligned}$$

Le vecteur β est le vecteur des coefficients de régression, X est la matrice de design de dimension $n \times p$.

Remarque On pourrait aussi prendre en compte une constante β_0 et ajouter une colonne à X qui serait alors de dimension $n \times (p+1)$.

On suppose en général que les Y_i suivent des loi normales de même variance, notée σ^2. Ou encore que Y est un vecteur gaussien de matrice de variance covariance $\sigma^2 \times I$, avec I matrice identité.

Exemple On étudiera par exemple le score qu'un individu attribue à un produit en fonction de différentes variables.

Le problème des covariables trop corrélées ou trop nombreuses Lorsque la matrice $X'X$ n'est pas inversible, ou bien lorsqu'elle est mal conditionnée, et donc d'inverse très instable, il n'y a pas de solution classique pour estimer β. On introduit alors une pénalisation.

On va considérer dans tout ce cours que les covariables sont normalisées. On peut également centrer Y pour simplifier.

1.2 Régression Ridge

Définition 1. *On définit l'estimateur ridge (Haerl et Kennard, 1970) par*

$$\widehat{\beta}_R = \underset{\beta \in \mathbb{R}^p}{\arg\min} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \; = \underset{\beta \in \mathbb{R}^p}{\arg\min} \|Y - X\beta\|_2^2 + \lambda \sum_{j=1}^{p} \beta_j^2, \tag{1}$$

où $\lambda > 0$ est un paramètre de shrinkage qui contrôle la force de la pénalité.

Vision bayésienne On peut montrer que l'approche ridge revient à mettre un a priori gaussien sur le coefficient de régression β. On peut voir alors l'estimateur comme le maximum (ou le mode) a posteriori. En effet, minimiser (1) en β revient à maximiser

$$-\|Y - X\beta\|_2^2 - \lambda \sum_{j=1}^{p} \beta_j^2$$

ou encore à maximiser (toujours en β)

$$\exp(-\|Y - X\beta\|_2^2) \exp(-\lambda \sum_{j=1}^{p} \beta_j^2) \quad \propto \quad f(Y|\beta) f(\beta),$$

où $Y \sim N(X\beta, 1/2)$ et $\beta_j \sim^{iid} N(0, 1/2\lambda)$.

1.3 Régression LASSO

Définition 2. *On définit l'estimateur LASSO (Tibshirani, 1996) par*

$$\widehat{\beta}_L \quad = \quad \underset{\beta\in\mathbb{R}^p}{\arg\min}\, \|Y - X\beta\|_2^2 + \lambda\|\beta\|_1 \quad = \underset{\beta\in\mathbb{R}^p}{\arg\min}\, \|Y - X\beta\|_2^2 + \lambda\sum_{j=1}^{p}|\beta_j|, \tag{2}$$

où $\lambda \geq 0$ est un paramètre de shrinkage qui contrôle la force de la pénalité.

Vision bayésienne On peut montrer que l'approche LASSO revient à mettre comme a priori une loi double exponentielle (ou Laplace) sur le coefficient de régression β et à étudier le maximum a posteriori. En effet, minimiser (2) en β revient à maximiser

$$-\|Y - X\beta\|_2^2 - \lambda\sum_{j=1}^{p}|\beta_j|$$

ou encore à maximiser (toujours en β)

$$\exp(-\|Y - X\beta\|_2^2)\exp(-\lambda\sum_{j=1}^{p}|\beta_j|) \quad \propto \quad f(Y|\beta)f(\beta),$$

où $Y \sim N(X\beta, 1/2)$ et $\beta_j \sim^{iid}$ *Laplace*.

1.4 Régression ELASTICNET

Définition 3. *On définit l'estimateur Elastic Net (Zou et Hastie, 2005) par*

$$\widehat{\beta}_{EN} \quad = \quad \underset{\beta\in\mathbb{R}^p}{\arg\min}\, \|Y - X\beta\|_2^2 + \lambda(\alpha\|\beta\|_1 + (1-\alpha)\|\beta\|_2^2), \tag{3}$$

où $\lambda \geq 0$ et $\alpha \in [0, 1]$.

Vision bayésienne L'approche Elastic Net peut être vue comme une approche bayésienne avec une loi a priori non classique sur le paramètre β.

2 Random Regression (RR)

On suppose que les variables sont normalisées (centrées réduites). Avec nos notations on a $X'X$ qui représente la covariance des variables et XX' celle des individus.

Le modèle RR (Henderson, 1975) est le premier modèle le plus simple, avec β aléatoire, mais sans hyperparamètre[1]. Il s'écrit

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \beta \sim N(0, G) & \text{souvent } G = \sigma^2_\beta I \\ \epsilon \sim N(0, R) & \text{souvent } R = \sigma^2_\epsilon I \end{cases}$$

Ici I désigne la matrice identité et $\mathbb{I}$ le vecteur $(1, \cdots, 1)$. On a

$$\mathbb{V}(Y) \quad = \quad XGX' + R.$$

Les cas particuliers diagonaux pour R et G reviennent à prendre les $\beta_j \sim^{iid} N(0, \sigma^2_\beta)$ et les $\epsilon_j \sim^{iid} N(0, \sigma^2_\epsilon)$. Dans ce cas on a

$$\mathbb{V}(Y) \quad = \quad \sigma^2_\beta XX' + \sigma^2_\epsilon I.$$

On voit que les Y_i sont dépendantes à travers la covariance XX' des individus.
$\hookrightarrow$ Illustration : modèle mixte de type ANOVA à deux facteurs. On choisit par exemple deux facteurs ($p = 2$) et 4 observations ($n = 4$) et on pose

$$\begin{aligned} X &= \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \\ \beta &= (\beta_1, \beta_2)'. \end{aligned}$$

On obtient alors

$$XX' \quad = \quad \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

ce qui montre la structure de dépendance des Y.

Définition 4. *Dans le cas où R et G sont diagonales on dit que σ^2_ϵ est la variance résiduelle et que σ^2_β est la variance commune (aux β).*

Schéma hiérarchique

[1] Un hyperparamètre est un paramètre supplémentaire qui s'ajoute au modèle et qui intervient dans la loi des paramètres initiaux du modèle devenus aléatoires

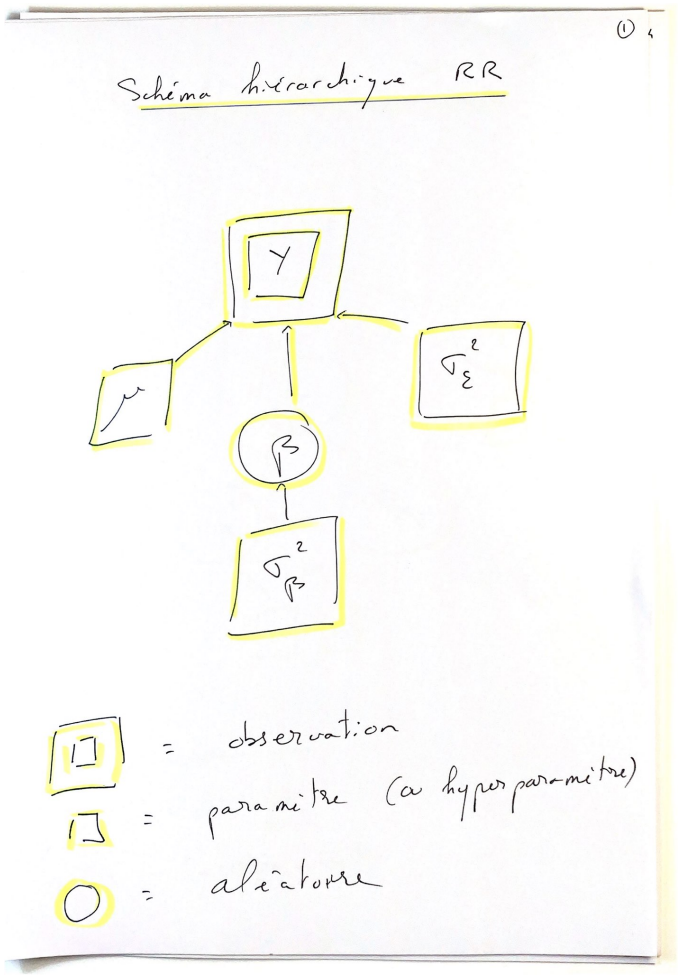

Figure 1: Schéma hiérarchique RR

Estimation de μ et β Les estimateurs sont obtenus par les équations de Henderson :

$$\begin{pmatrix} \mathbb{I}' & \mathbb{I} \\ X'\mathbb{I} & X'X + \frac{\sigma_\epsilon^2}{\sigma_\beta^2} I \end{pmatrix} \begin{pmatrix} \mu \\ \beta \end{pmatrix} = \begin{pmatrix} \mathbb{I}'Y \\ X'Y \end{pmatrix}$$

On obtient ainsi les estimateurs BLUP (Best Linear Unbiased Predictor).

Estimation de σ_ϵ^2 et σ_β^2 On utilise la méthode REML (Restricted ML) qui revient à maximiser la vraisemblance avec contrainte imposée sur la variance.

Lien entre BLUP et inférence bayésienne Le modèle RR peut être vu comme le niveau 0 du bayésien (pas d'hyperparamètres, variances constantes). Si on met une loi a priori sur μ on

obtient le modèle suivant :

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \beta \sim N(0, G) & \text{souvent } G = \sigma^2_\beta I \\ \epsilon \sim N(0, R) & \text{souvent } R = \sigma^2_\epsilon I \\ \mu \sim \text{ a priori vague} \end{cases}$$

où l'a priori vague (ou non informatif) signifie que la densité de μ est constante, ce qui revient à dire que μ est uniforme. Dans ce cas, si le support de μ est la droite réelle, on parle de loi impropre (non intégrable). Alors les espérances a posteriori de μ et β coïncident avec les estimateurs BLUP :

$$\begin{aligned} \mathbb{E}(\beta|Y) &= \widehat{\beta} \\ \mathbb{E}(\mu|Y) &= \widehat{\mu} \end{aligned}$$

Prédiction

$$\widehat{Y} = \widehat{\mu}\mathbb{I} + X\widehat{\beta}$$

Package R Package RRBLUP ou package mixed.solve
$\hookrightarrow$ calcule le BLUP et REML pour le modèle

$$Y = X\beta + Zu + \epsilon.$$

On a

- β effet fixe
- u random effect (effet aléatoire) avec $\mathbb{V}(u) = k\sigma^2_u$.
- $\mathbb{V}(\epsilon) = \sigma^2_\epsilon I$.

$\hookrightarrow mixed.solve(Y, Z =, k =, X =, method = "REML")$

3 Modèle de régression bayésienne : BAYES A

Motivation Avec le modèle RR, on a la même variance σ^2_β pour tous les β_j, ce qui n'est pas réaliste. Meuwissen et al. (2001) propose un modèle où chaque variable a une variance propre : ça ressemble au ridge bayésien.

Modèle Bayes A Si on met une loi a priori sur μ on obtient le modèle suivant :

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ \beta \sim N(0, diag(\sigma_{\beta_1}^2, \cdots, \sigma_{\beta_p}^2) \\ \sigma_{\beta_j}^2 \sim Inv.Gamma(a, b)\ j = 1, \cdots, p \\ \sigma_\epsilon^2 \sim Inv.Gamma(c, d) \\ \mu \sim uniform \end{cases}$$

L'a priori sur μ est vage. Les a priori inverse gamma sont conjuguées pour le modèle (les a posteriori sont des inverse gamma).

Remarque 5. *Une variable aléatoire* $X \sim Inv.Gamma(a/2, ab/2) \Leftrightarrow X \sim \chi^{-2}(a, b)$.

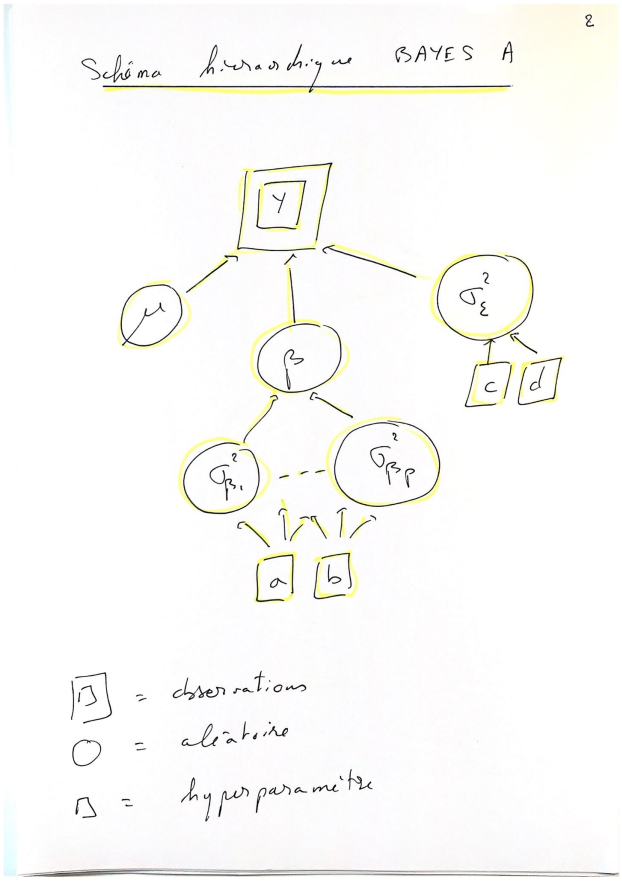

Figure 2: Schéma hiérarchique Bayes A

Schéma hiérarchique

Gibbs sampler Pour estimer les paramètres du modèle on utilise un Gibbs sampler. A chaque itération les étapes suivantes sont effectuées :

- Echantillonner chaque β_j :

$$\beta_j|\beta_{-j}, Y, \mu, \sigma^2_\epsilon, \sigma^2_{\beta_j} \quad \sim \quad N(\mu_j, \Sigma_{\beta_j})$$

  avec μ_j et Σ_{β_j} à calculer en exo. Ici β_{-j} désigne le vecteur β privé de β_j.

- Echantillonner μ :

$$\mu|Y, \beta, \sigma^2_\epsilon \quad \sim \quad N(\mathbb{I}'(Y - X\beta)/n, \sigma^2_\epsilon/n)$$

- Echantillonner $\sigma^2_{\beta_j}$

$$\sigma^2_{\beta_j}|\beta_j \quad \sim \quad Inv.Gamma(a + 1/2, b + 1/2\beta_j^2)$$

- Echantillonner σ^2_ϵ

$$\sigma^2_\epsilon|Y, \beta, \mu \quad \sim \quad Inv.Gamma(c + n/2, d + (Y - \mu\mathbb{I} - X\beta)'(Y - \mu\mathbb{I} - X\beta)/2)$$

Remarque 6. *L'astuce pour trouver toutes les loi a posteriori consiste à utiliser les conditionnements de Bayes. On peut ainsi exprimer les densités sous les formes suivantes :*

$$\begin{aligned} f(\beta, \mu, \sigma^2_\epsilon, \sigma^2_{\beta_1}, \cdots, \sigma^2_{\beta_p}|Y) &= f(Y|\beta, \mu, \sigma^2_\epsilon, \sigma^2_{\beta_1}, \cdots, \sigma^2_{\beta_p}) f(\beta, \mu, \sigma^2_\epsilon, \sigma^2_{\beta_1}, \cdots, \sigma^2_{\beta_p})/f(Y) \\ &\propto f(Y|\beta, \mu, \sigma^2_\epsilon) f(\beta|\sigma^2_{\beta_1}, \cdots, \sigma^2_{\beta_p}) f(\sigma^2_{\beta_1}, \cdots, \sigma^2_{\beta_p}) f(\sigma^2_\epsilon) f(\mu). \end{aligned}$$

Ici on a $f(\mu) = 1$.

Prédictions On récupère les prédictions suivantes :

$$\widehat{Y} \quad = \quad \widehat{\mu}\mathbb{I} + X\widehat{\beta}$$

avec $\widehat{\mu}$ moyenne des μ échantillonnés post-burn-in et $\widehat{\beta}$ moyenne des β échantillonnés post-burn-in.

Remarque 7. *Le choix des valeurs initiales pour l'algorithme peut être important pour bien converger. On peut prendre par exemple les estimateurs fréquentistes (pour* β *notamment).*

4 Modèle de régression bayésienne : BAYES B

Motivation Si on a beaucoup de variables on tombe sur le principe de parcimonie. Un grand nombre de β varient très peu, et comme ils sont centrés on pourrait leur assigner une valeur nulle. Ce n'est pas possible dans Bayes A. On a même $\mathbb{P}(\sigma^2_{\beta_j} = 0) = 0$.

Le modèle Bayes B

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ \beta \sim N(0, diag(\sigma_{\beta_1}^2, \cdots, \sigma_{\beta_p}^2) \\ \sigma_{\beta_j}^2 \begin{cases} = 0 & \text{avec proba } \pi \\ \sim Inv.Gamma(a,b) & \text{avec proba } 1-\pi \end{cases} \\ \sigma_\epsilon^2 \sim Inv.Gamma(c,d) \\ \mu \sim uniform \end{cases}$$

Ainsi la loi de $\sigma_{\beta_j}^2$ est un mélange d'une Dirac en zéro et d'une inverse gamma. Lorsqu'elle vaut zéro alors on pose aussi $\beta_j = 0$.

Remarque 8. *Bayes A est un cas particulier de Bayes B avec* $\pi = 0$

On voit que π est un paramètre de shrinkage qui va annuler les β_j faibles.

Gibbs sampler Pour estimer les paramètres du modèle on utilise un Gibbs sampler. L'algorithme diffère de celui de Bayes A car la loi de β_j dépend de la valeur de $\sigma_{\beta_j}^2$. On simule d'abord la loi de $\sigma_{\beta_j}^2 | Y, \mu, \beta_{-j}, \sigma_\epsilon^2$ par Metropolis Hasting car ce n'est pas une loi standard. Puis on simule $\beta_j | \sigma_{\beta_j}^2, Y, \mu, \beta_{-j}, \sigma_\epsilon^2$ comme dans Bayes A si $\sigma_{\beta_j}^2 > 0$, et sinon on pose zéro.

Prédictions On a les prédictions suivantes :

$$\widehat{Y} \quad = \quad \widehat{\mu}\mathbb{I} + X\widehat{\beta}$$

avec $\widehat{\mu}$ moyenne des μ échantillonnés post-burn-in et $\widehat{\beta}$ moyenne des β échantillonnés post-burn-in.

5 LASSO bayésien

Park et Casella (2008)

Le modèle

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ \beta | \Lambda, \sigma_\epsilon^2 \sim N(0, \sigma_\epsilon^2 \Lambda) \quad \text{avec } \Lambda = diag(\tau_1, \cdots, \tau_p) \\ \tau_j | \lambda^2 \sim Exp(\lambda^2/2) \\ \lambda^2 \sim Gamma(e,f) \\ f(\sigma_\epsilon^2) = 1/\sigma_\epsilon^2 \quad (\text{ou } Inv.Gamma) \\ \mu \sim uniform \end{cases}$$

On peut remarquer que la loi de σ_ϵ^2 est impropre (ce n'est pas une densité de probabilité). C'est l'a priori de Jeffrey. L'idée est de pénaliser les grandes variances. Le paramètre λ joue le rôle du paramètre LASSO. On remarque que

$$\begin{aligned} \mathbb{E}(\tau_j | \lambda^2) &= 2/\lambda^2 \\ \mathbb{E}(\mathbb{V}(\beta_j | \sigma_\epsilon^2, \lambda^2, \Lambda) &= 2\sigma_\epsilon^2/\lambda^2. \end{aligned}$$

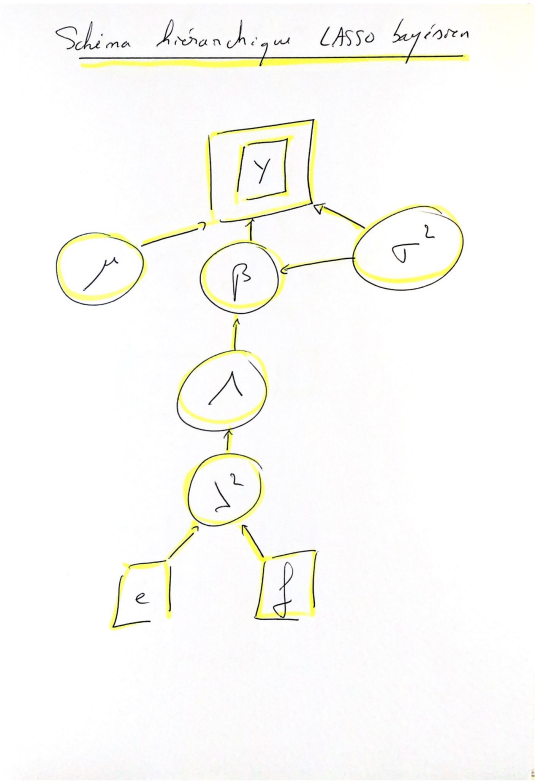

Figure 3: Schéma hiérarchique LASSO bayésien

Schéma hiérarchique

Gibbs Sampler Il y a une petite astuce dans le Gibbs, c'est d'intégrer sur μ pour simplifier les calculs. En effet, μ n'a pas beaucoup d'intérêt ici. Du coup on élimine ce paramètre en remplaçant $f(\mu, \theta)$ par $\int f(\mu, \theta) d\mu$. Ensuite l'algorithme est basé sur les mêmes calculs des probabilités conditionnelles (formules de Bayes) que celui détaillé pour Bayes A. Les étapes sont les suivantes : à chaque itération

- Echantillonner β

$$\beta | Y, \Lambda, \sigma_\epsilon^2 \quad \sim \quad N([XX' + n^{-1}]^{-1} X' \widetilde{Y}, V),$$

  avec $\widetilde{Y} = [I - \mathbb{I}\mathbb{I}'/n]Y$ le vecteur d'observations centré et $V = \sigma_\epsilon^2 [X'X + n^{-1}]^{-1}$.

- Echantillonner τ_j

$$1/\tau_j | \beta, \lambda^2, \sigma_\epsilon^2 \quad \sim \quad Inv.Gaussienne(\lambda \sigma_\epsilon / |\beta_j|, \lambda^2).$$

- Echantillonner λ^2

$$\lambda^2 | \Lambda \quad \sim \quad Gamma(p + e, (1/f + \sum \tau_j/2)^{-1}).$$

- Echantillonner σ^2_ϵ

$$\sigma^2_\epsilon|\widetilde{Y},\beta \quad \sim \quad Inv.Gamma(\frac{n-1+p}{2}, \frac{(\widetilde{Y}-X\beta)'(\widetilde{Y}-X\beta)+\beta'\Lambda^{-1}\beta}{2}).$$

Prédictions

$$\widehat{\widetilde{Y}} \quad = \quad X\widehat{\beta}.$$

Le paramètre λ Il contrôle le compromis entre bon-ajustement et complexité du modèle. Ansi

- Pour λ petit, on s'approche des moindres carrés.
- Lorsque λ augmente : comme λ^2 contrôle la forme de l'a priori pour β, la variance des β diminue. L'a priori est alors plus informatif et permet un shrinkage plus important.

Comparaison avec les autres méthodes Il y a un niveau hiérarchique de plus dans le LASSO bayésien par rapport au Bayes A.
Le Bayes A peut se voir comme un ridge bayésien.
On peut montrer (exercice) que la loi marginale de β_j est

- $\beta_j \sim$ *Student* dans le cas du Bayes A
- $\beta_j \sim$ *Laplace* (ou double-exponentielle) dans le cas du LASSO

Si on représente ces deux densités on voit le pic de la Laplace en zéro et une queue plus lourde ce qui indique un shrinkage plus fort autour de zéro et un shrinkage moins fort pour les grandes valeurs des coefficients (en valeur absolue).

Package R

- $BLR(Y, X_F, X_R, X_L, G_F, prior, nIter, burnIn)$
- $Y = \mu\mathbb{I} + X_F\beta_F + X_R\beta_R + X_L\beta_L + Zu + \epsilon.$

On peut ne considérer que β_L ici. Exercice : consulter l'aide de R sur BLR et reproduire le schéma hiérarchique proposé par ce package.

6 Sélection de variables

Dans les modèles précédents de régression (RR, Bayes A et Bayes B, LASSO bayésien), après la période de burn-in, on récupère des valeurs simulées des paramètres. On peut ainsi estimer les β_j, et en général ils ne sont pas nuls. Pour sélectionner les meilleurs coefficients on peut

- Garder les plus grands $|\widehat{\beta}_j|$ (les variables doivent être normalisées).

- On peut aussi garder les β_j associées au plus grandes variances (sauf dans le RR car c'est la même). Dans ce cas en Bayes A ou B on regarde les plus grands $\widehat{\sigma^2}_{\beta_j}$. En LASSO on regarde les plus grands $\widehat{\tau}_j$.
- On peut aussi construire des intervalles de confiance (sur la distribution empirique simulée) et choisir les IC qui ne contiennent pas zéro.

7 Stochastic Search Variable Selection

La méthode SSVS a été introduite par George et McCulloch (1993). L'idée est d'introduire un paramètre γ (*spike and slab parameter*) qui va indiquer si un coefficient de β est sélectionné ou non. Ainsi γ est un vecteur de dimension p qui contient des 1 indiquant que les β_j associé sont non nuls, et des zéros indiquant que les β_j associés sont nuls. On note β_γ le vecteur des coefficients non nuls. On posera

$$d_\gamma \;=\; \sum_{j=1}^{p} \gamma_j,$$

le nombre de coefficients sélectionnés. On note X_γ la matrice design associée composée uniquement des colonnes associées aux coefficients sélectionnés.

Cette approche Spike and Slab peut être utilisée dès que l'on doit sélectionner des variables parmi un nombre important de candidats. Par exemple, pour modéliser la mortalité en prenant en compte deux dimensions : la cohorte et l'âge des individus. On peut contruire une série temporelle (un champ aléatoire ici) qui dépend d'un passé représenté par une grille en dimension 2. Choisir les meilleurs candidats (au sens d'une vraisemblance) dans cette grille peut être trop lourd en calcul. Un problème similaire est rencontré en actuariat avec les triangles de provisionnement.

Le modèle

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ \begin{cases} \gamma_k = 1 \text{ si } \beta_k \neq 0 & \text{sélection} \\ \gamma_k = 0 \text{ si } \beta_k = 0 & \text{non sélection} \end{cases} \\ \beta_\gamma | \gamma, \sigma_\epsilon^2 \sim N(0, \sigma_\epsilon^2 c\, (X_\gamma' X_\gamma)^{-1}) \\ \mathbb{P}(\gamma_j = 1) = \pi \\ f(\sigma_\epsilon^2) = 1/\sigma_\epsilon^2 \\ \mu \sim uniform \end{cases}$$

Le paramètre $c > 0$ est un facteur d'échelle, encore appelé *selection coefficient* dans Bottolo et Ridchardson (2010). Il est en général choisi entre 10 et 100. Le paramètre $0 < \pi < 1$ contrôle le nombre de β_j sélectionnés. C'est un a priori qui permet de limiter le nombre de β sélectionnés (il faut que l'a posteriori soit "fort" pour retenir un β si π est petit). Attention : un π trop grand sélectionnera trop de coefficients. On peut aussi faire varier π dans l'algorithme si on veut

privilégier au fur et à mesure les β_j qui ont été retenus le plus souvent. L'a priori pour β est appelé l'a priori de Zellner (voir Celeux et al. 2006). La structure de covariance est automatique, gérée par $\sigma_\epsilon^2 c\,(X'_\gamma X_\gamma)^{-1}$. Elle reflète ainsi la structure des données (et donc la corrélation des variables). Le paramètre c permet de prendre plus ou moins en compte cette structure. Si jamais la matrice $X'_\gamma X_\gamma$ est mal conditionnée on peut faire appel à du Ridge-Zellner, c'est à dire que l'on rajoute un paramètre Ridge $\lambda > 0$ de la manière suivante :

$$\beta_\gamma|\gamma, \sigma_\epsilon^2 \sim N(0, \sigma_\epsilon^2 c(X'_\gamma X_\gamma + \lambda I)^{-1}).$$

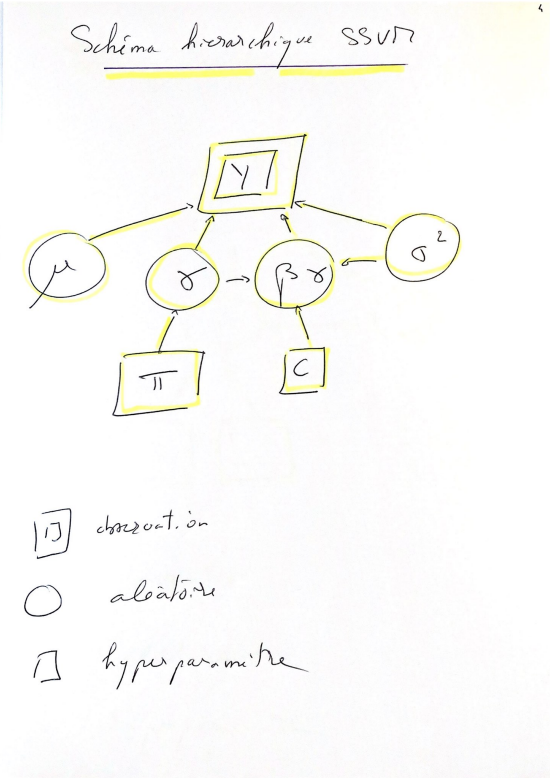

Figure 4: Schéma hiérarchique SSVM

Schéma hiérarchique

Algorithme de Metropolis-Hasting On arrive à intégrer tous les paramètres et à ne garder que la distribution des γ. C'est justement ce paramètre qui nous intéresse. En effet, les β ne sont pas tous sélectionnés et ils sont difficilement comparables. On simule donc $\gamma|Y$ avec un

Metropolis-Hasting (détails en TP). On peut montrer que la densité s'écrit :

$$f(\gamma|Y) \propto (1+c)^{d_\gamma/2} \prod_{j=1}^{p} \pi^{\gamma_j}(1-\pi)^{1-\gamma_j}\Gamma((n-1)/2) \left(\frac{1}{2}\widetilde{Y}'(I - \frac{c}{1+c}X_\gamma(X_\gamma'X_\gamma)^{-1}X_\gamma'\widetilde{Y}\right)^{-(n-1)/2}.$$

On prend alors un noyau symétrique $q(.|.)$ pour l'algorithme de Metropolis-Hasting : on change aléatoirement r éléments de γ (1 devient 0 et inversement).
A l'étape $t+1$ on a ainsi :

- $\gamma^* \sim q(\gamma^*|\gamma^{(t)})$
- $$\gamma^{(t+1)} = \begin{cases} \gamma^* & \text{avec prob. } \rho(\gamma^*, \gamma^{(t)}) \\ \gamma^{(t)} & \text{avec prob. } 1-\rho(\gamma^*, \gamma^{(t)}) \end{cases}$$

  avec

  $$\rho(\gamma^*, \gamma^{(t)}) = \min\left\{1, \frac{f(\gamma^*|Y)q(\gamma^{(t)}|\gamma^*)}{f(\gamma^{(t)}|Y)q(\gamma^*|\gamma^{(t)})}\right\}$$

  qui se simplifie car q est symétrique puisque l'on choisit au hasard les candidats pour "une naissance (1) ou une mort (0)". Si d_γ est constant on a une deuxième simplification (à vérifier en exercice).

Sélection des variables Les variables les plus pertinentes correspondent aux éléments de γ valant le plus souvent 1. Ils peuvent être facilement identifier à partir des simulations post-burn-in des γ. On regarde tout simplement leurs fréquences $\Rightarrow$ on sélectionne ainsi les variables associées aux plus grandes fréquences.

Le grand avantage de cette méthode c'est sa rapidité car on manipule seulement X_γ au lieu de X. Elle permet de traiter des données de grande dimension (des données génomiques par exemple).

8 ABC (Approximate Bayesian Computation)

Le principe de l'ABC On va donner le principe dans le cadre de la régression. Mais il se généralise à n'importe quel modèle bayésien. Supposons que l'on s'intéresse au modèle suivant :

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ (\mu, \beta, \sigma_\epsilon^2) \sim \pi \end{cases}$$

avec π une loi a priori. Notons $\theta = (\mu, \beta, \sigma_\epsilon^2)$. On veut simuler la loi a posteriori de $\theta|Y$. Pour cela on va simuler un θ^* suivant sa loi π. Comme on sait que $Y|\theta \sim N(\mu\mathbb{I} + X\beta, \sigma_\epsilon^2)$, on peut simuler des $Y^*|\theta^*$. On regarde si ces $Y^*|\theta^*$ tombe proche des Y observés (qui sont des $Y|\theta$). Si oui, on en déduit que notre θ^* peut convenir, et on le garde. Et on itère l'algorithme.

Algorithm A l'origine l'algorithme ABC (encore appelé Exact ABC) pouvait s'écrire comme suit (avec les notations précédentes), en supposant que l'on a n observations de $Y\theta$:

Exact ABC algorithm

---

for $j = 1$ to N do
  repeat
    Generate θ^* from π
    Generate n values of Y^* from $f(.|\theta^*)$
  until $(Y_1^*, \cdots, Y_n^*) = (Y_1, \cdots, Y_n)$
  set $\theta^*(j) = \theta^*$
end for

Ici π est la loi a priori jointe des paramètres $(\mu, \beta, \sigma_\epsilon^2)$ qu'il faut savoir simuler. La loi de $Y|\theta$ est immédiate d'après le modèle. On a tout simplement

$$Y|\mu, \beta, \sigma_\epsilon^2 \quad \sim \quad N(\mu\mathbb{I} + X\beta, \sigma_\epsilon^2 I).$$

Cet algorithme permet de simuler N valeurs de la distribution a posteriori. On montre que la loi des $\theta^*(j)$ est exactement la loi a posteriori de $\theta|Y$. En effet, on peut écrire la densité de θ^* comme suit :

$$\begin{aligned}
\pi^*(\theta^*(j)) &= \int f(y^*, \theta^*(j)) dy^* \\
&= \int f(y^*, \theta^*(j)) \delta_y(y^*) dy^* \\
&= \int f(y^*|\theta^*(j)) \pi(\theta^*(j)) \delta_y(y^*) dy^* \\
&= f(y|\theta^*(j)) \pi(\theta^*(j)) \\
&\propto \pi(\theta^*|y).
\end{aligned}$$

Mais il est connu (Pritchard et al., 1999) que la règle d'acceptation est trop restrictive ici. C'est pour cela que l'on ajoute une *summary statistic* S et une distance ρ telles qu'on accepte la valeur de θ^* si la distance $\rho(S(Y^*) - S(Y))$ est petite. L'algorithme devient :

Smooth ABC algorithm

---

for $j = 1$ to N do
  repeat
    Generate θ^* from π
    Generate Y^* from $f(.|\theta^*)$
  until $\rho(S(Y^*) - S(Y)) \leq \alpha$
  set $\theta_j^* = \theta^*$

end for

Ici α est un seuil qui permet d'être plus ou moins stricte sur l'acceptation. Un α élevé fera accepter beaucoup de valeurs de θ, mais trop souvent, et donc trop éloignées. Un α trop petit fera rejeter trop souvent et l'algorithme sera très long, surtout si l'espace des paramètres est grand. Mais les valeurs seront meilleures.

On peut imaginer mettre un poids sur les β retenus en fonction de la "distance" entre les Y observés et les Y^* simulés. Certains package proposent des pondérations en se basant sur un lissage des donnés observées ou sur du machine learning (régression, random forest) pour relier les β et les summary statistic.

Package R *abc, easyabc, abctools, abccrf*

9 Compléments

ElasticNet Bayésien

$$\begin{cases} Y = \mu\mathbb{I} + X\beta + \epsilon \\ \epsilon \sim N(0, \sigma_\epsilon^2 I) \\ \beta|\lambda_1, \lambda_2, \sigma_\epsilon^2 \sim \exp(-\dfrac{1}{2\sigma_\epsilon^2}(\lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2) \\ \\ f(\sigma_\epsilon^2) = 1/\sigma_\epsilon^2 \\ \mu \sim Uniform \end{cases}$$

En fait, même en prenant des λ_1 et λ_2 déterministes (pas d'a priori) il est difficile de simuler la loi de β. Un résultat de Li et Lin (2010) réécrit cette densité comme un mélange pour pouvoir la simuler (avec conditions).

Schéma hiérarchique

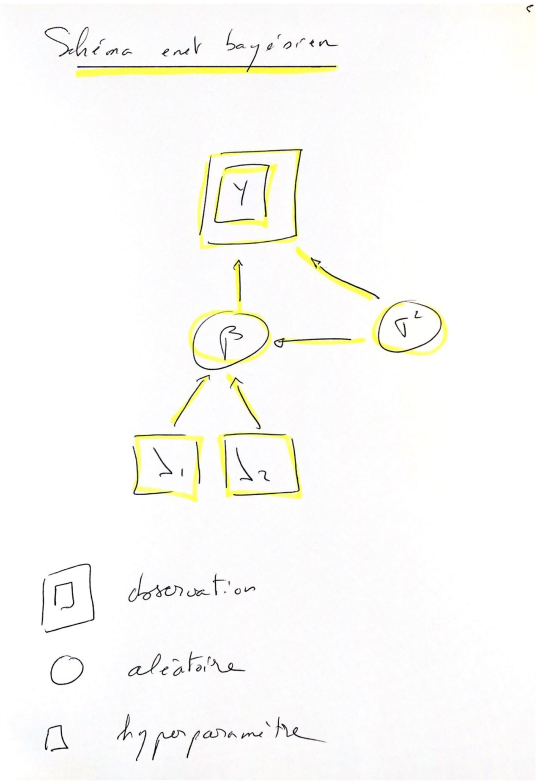

Figure 5: Schéma hiérarchique Enet Bayésien

Avec R Package EBglmnet : LASSO, LASSO bayésien et Enet bayésien.

10 Ajout de variables d'intérêt

On peut ajouter d'autres covariables Z sur lesquelles on ne souhaite pas faire de sélection. Il suffit de distinguer les paramètres :

$$Y = \mu\mathbb{I} + X\beta + Z\alpha + \epsilon$$

avec les variables de Z centrées et $\alpha \sim N(0, \sigma_\alpha^2 I)$.

11 Extension aux modèles GLM

Des extensions du modèle linéaire bayésien au modèle linéaire généralisé existent. On peut ainsi considérer un modèle Probit bayésien. L'avantage du probit (par raport au logit) est qu'il utilise une loi normale plus facile à manipuler. Chaque extension nécessite des calculs nouveaux pour obtenir les lois conditionnelles.

On peur préférer au modèile logit un modèle probit, car on a une loi normale plus facile à manipuler. On peut introduire des variables latentes telles que

$$Y_i = \begin{cases} 1 & \text{si } L_i > 0 \\ 0 & \text{si} L_i < 0 \end{cases}$$

avec $L|\beta \sim N(X\beta, I)$. On a alors un niveau hiérarchique supplémentaire avec les $L_i|\beta$, juste en dessous des Y_i. On a

$$\mathbb{P}(Y_i = 1|\beta) = \mathbb{P}(L_i > 0|\beta) = 1 - \Phi(X_i'\beta).$$

12 Prise en compte des mélanges

Un mélange de K modèles est de la forme :

$$\mathcal{M} = \sum_{j=1}^{K} \alpha_j \mathcal{M}_j,$$

avec $\sum_{j=1}^{K} \alpha_j = 1$. On prend comme prior pour le vecteur $\alpha = (\alpha_1, \cdots, \alpha_K)$ une loi de Dirichlet de paramètres $\lambda = (\lambda_1, \cdots, \lambda_K)$ de densité :

$$\frac{1}{B(\lambda)} \prod_{j=1}^{K} \alpha_j^{\lambda_j - 1},$$

où B est la fonction Beta définie par

$$B(\lambda) = = \frac{\prod_{j=1}^{K} \Gamma(\lambda_j)}{\Gamma(\sum_{j=1}^{K} \lambda_j)}.$$

Les modèles $\mathcal{M}_j$ peuvent être des modèles de régression. On peut aussi rendre aléatoire le paramètre K.

13 Gibbs et Metropolis-Hasting

Gibbs sampler On veut générer aléatoirement $(\theta_1, \cdots, \theta_d)$ suivant une loi jointe π de densité connue. Dans nos exemples π est la loi des paramètres (θ) sachant les observations, donc a loi a posteriori. A la $i+1$e étape on génère :

$$\begin{aligned} \theta_1^{i+1} &\sim \pi(\theta_1|\theta_2^i, \cdots, \theta_d^i) \\ \theta_2^{i+1} &\sim \pi(\theta_2|\theta_1^{i+1}, \theta_3^i, \cdots, \theta_d^i) \\ \ldots \theta_d^{i+1} &\sim \pi(\theta_d|\theta_1^{i+1}, \cdots, \theta_{d-1}^i) \end{aligned}$$

On peut même simuler simultanément plusieurs variables si on connaît leur loi jointe conditionnellement aux autres (*grouping technic*). Par exemple :

$$\begin{aligned} \theta_1^{i+1} &\sim \pi(\theta_1|\theta_2^i, \cdots, \theta_d^i) \\ (\theta_2^{i+1}, \theta_3^{i+1}) &\sim \pi(\theta_2, \theta_3|\theta_1^{i+1}, \theta_4^i, \cdots, \theta_d^i) \\ \ldots \theta_d^{i+1} &\sim \pi(\theta_d|\theta_1^{i+1}, \cdots, \theta_{d-1}^i) \end{aligned}$$

Il faut initialiser les valeurs des paramètres. On voit que l'on bouge vers des zones de densité plus élevée (on fait le rapport). On peut parfois tomber sur un maximum local et alors il faut que la proposal q nous permette de bouger suffisamment pour pouvoir continuer à explorer la densité.

Remarque 9. *Les simulations s'obtiennent après une période de burn-in pour que la chaîne converge. Les simulations sont de loi π mais elles sont dépendantes. On peut les espacer pour diminuer cette dépendance.*

- Avantage du Gibbs : il est rapide et converge facilement.
- Inconvénient du Gibbs : il faut connaître (savoir simuler) les lois conditionnelles.

Metropolis-Hasting On a le même objectif que pour le Gibbs, on veut simuler un vecteur (ou scalaire) aléatoire θ, mais on ne sait pas simuler toutes les lois conditionnelles. On utilise alors la connaissance de la densité π de θ. A l'étape $i+1$ on dispose de la valeur du vecteur θ_i simulé. On simule alors un vecteur $\theta_{i+1}^* \sim q(.|\theta_i)$ (q est appelée *proposal*). On décide alors

$$\theta_{i+1} = \begin{cases} \theta_{i+1}^* & \text{avec proba } p = \min\{\frac{\pi(\theta_{i+1}^*)}{\pi(\theta_i)} \frac{q(\theta_i|\theta_{i+1}^*)}{q(\theta_{i+1}^*|\theta_i)}, 1\} \\ \theta_i & \text{avec proba } 1-p \end{cases}$$

Souvent on prend une proposal symétrique ce qui simplifie $\frac{q(\theta_i|\theta_{i+1}^*)}{q(\theta_{i+1}^*|\theta_i)} = 1$.

Par ex.

- Avantage du Metropolis-Hasting : il ne nécessite pas de connaître les lois, ni de les simuler directement. Il s'applique donc à toutes les densités.
- Inconvénient du Metropolis-Hasting : il faut une bonne "proposal" pour que la convergence soit rapide. Il ne faut pas une trop grande dimension pour les paramètres (100 max ?).

Critères de convergence de la chaîne Pour vérifier si la chaîne de Markov a convergé on peut utiliser les critères suivants :

- La trajectoire de la chaîne ne doit pas montrer de tendance, ni de changement de variance (variance homogène le long de la chaîne), ni de palier (pas d'échange).
- Si on lance plusieurs chaîne simultanément, les trajectoires doivent se mélanger (croisements des traces).

- La moyenne est constante le long de la chaîne.
- Les autocorrélations de la chaîne dimminuent avec le "lag".
- Deux critères sont souvent utilisés :
  - Le ESS (Effective Sample Size)
  - Le Gelman-Rubin (GR ou Rhat) diagnostic

14 Vers des algorithmes de Machine Learning en régression

Les algorithmes de type Data Augmentation permettent, à partir d'un jeu de données observé (des vecteurs en général), de générer des variables synthétiques ayant des caratéristiques proches des jeux observés. Les objectifs sont : soit de reproduire des données "anonymisées", fidèles aux données d'origine ; soit de rééquilibrer des échantillons en générant des données rares ; soit de créer des modalités non observées. Plusieurs types d'algorithmes existent, comme par exemple :

- SMOTE qui créer des données synthétiques en interpolant les variables continues ou en choisissant des modalités de variables qualitatives les plus représentées dans l'échantillon.
- CTGAN qui est un Generative Adversial Network qui prend les données observées en entrée, propose de nouvelles données modifiées, et les accepte si elles ressemblent aux données initiales. Il apprend ainsi à créer des données synthétiques proches de celles observées.
- MIAMI qui suppose un modèle de type réseau de neurone paramétrique pour les vecteurs d'observations et qui, une fois estimés les paramètres, permet de générer autant de variables vectorielles suivant cette loi.
- SynthPop qui créer des régression CART conditionnelles et propose de reconstruire un vecteur synthétique en proposant au fur et à mesure les valeurs les plus probables sachant les autres valeurs déjà simulées.
- Gaussian Noise (GN) qui génère des bruits gaussiens autour dez vairbales quantitatives, et qui, combiné à SMOTE, choisit alétatoirement les variables qualitatives.

L'avantage des derniers algorithmes proposés comme CTGANSynthetizer (Python) ou SynthPop (R) : ils fonctionnent avec les données mixtes (vecteurs comportant une partie continue et une partie discrète ou qualitative).

$\hookrightarrow$ Tous ces algorithmes permettent de créer des données synthétiques proches de la celles observées.

$\hookrightarrow$ Comment utiliser ces algorithmes en régression ?

Supposons que l'on veuille régresser Y sur X (X et un vecteur pouvant contenir des quanti et

des quali, Y peut être multivariée). On met alors dans un même vecteur $Z = (Y, X)$. Ce vecteur est ensuite passé dans un algorithme de Data Augmentation. On récupère ainsi des nouvelles observations (synthétiques) de (Y, X).

$\Rightarrow$ on peut utiliser l'idée d'acceptation-rejet de l'ABC pour estimer par exemple $\mathbb{E}(Y|X = X_0)$. En effet, on simule le vecteur (Y, X) et on ne retient que les valeurs asociées à $d(X, X_0) < \epsilon$, pour une certaine distance d et pour un $\epsilon > 0$ fixé. Si $\epsilon = 0$ on obtient exactement $X = X_0$ mais avec un coût de simulation lourd si la dimension de X est grande. Une fois récoltée nos "bonnes" simulations on peut évaluer notre régression par la moyenne empirique des Y synthétiques. On peut même estimer les lois conditionnelles et marginales.

Exercice : utiliser CTGANsynthetizer ou SynthPop pour créer un grand nombre d'observations synthétiques. Ces générations sont basées sur des modèles non bayésiens (régression avec poids). On récupère ces jeux de données augmentés et on peut alors faire un modèle de régression bayésienne. Avantage : on utilise deux méthodes différentes pour construire les données et le modèle.

15 Annexes

Lois utilisées

- Densité de la loi inverse gamma $Inv.Gamma(a, b)$, $a = shape$,$b = scale$.

$$\frac{b^a}{\Gamma(a)} x^{-(a+1)} \exp(-\frac{b}{x})$$

- Densité de la loi inverse gamma khi-deux : $\chi^{-2}(a, b)$, $a = ddl$,$b = scale$.

$$\frac{(ab/2)^{a/2}}{\Gamma(b/2)} x^{-(a/2+1)} \exp(-\frac{ab}{2x})$$

- Densité de la loi inverse Gaussienne $Inv.Gaussienne(m, s)$, $s = moyenne$,$s = shape$.

$$(\frac{s}{2\pi x^3})^{1/2} \exp(s\frac{s(x-m)^2}{2m^2 x})$$

Lois conjuguées Une famille de lois a priori $F = \{P_\theta; \theta \in \Theta\}$ est conjuguée (par rapport à Y) si la famille des loi a posteriori $F|Y = \{P_\theta|Y; \theta \in \Theta\}$ est égale à F. En bayésien les lois a priori sont en général choisies car elles sont conjuguées.

16 Références

- Bishop, C.M. (2006). Pattern Recognition and Machine Learning, Springer,

- Bottolo, L and Richardson, S. (2010) Evolutionary stochastic search for bayesian model exploration. Bayesian Analysis. 5, 583–618.
- Celeux, G. Marin, J.M. Robert, C. (2006) Sélection bayésienne de variables en régression linéaire. Journal de la Société Française de Statistique. 147, 59–79
- Park T, Casella G. (2008) The bayesian lasso. Journal of the American Statistical Association. 103, 681–686
- George, E. I. and McCulloch, R. E. (1993). Variable Selection Via Gibbs Sampling. Journal of the American Statistical Association. 88, 881–889.
- Henderson, C.R. (1975). Best linear unbiased estimation and prediction under a selection model. Biometrics. 31, 423–447
- Li, Q. Lin, N. (2010) The Bayesian elastic net. Bayesian Analysis. 5, 151–170
- Meuwissen T. H. E., Hayes B. J., Goddard M. E., (2001). Prediction of total genetic value using genome-wide dense marker maps. Genetics. 157, 1819–1829
- Pritchard, J.K., Seielstad, M.T., Perez-Lezaun, A. Feldman, M.W. (1999) Population growth of human Y chromosomes: a study of Y chromosome microsatellites. Molecular Biology and Evolution. 16, 1791–1798
- Robert, C. Casella, G. (2005) Monte Carlo Statistical Methods. Springer Texts in Statistics.
- R. Tibshirani (1996) Regression shrinkage and selection via the Lasso. Journal of the Royal Statistical Society, Series B, Volume 58, Issue 1, pp. 267–288.