Rapport de stage de Master 1
Sélection de variables bayésienne dans les modèles non linéaires à effets mixtes

Nils BAILLIE

M1 Mathématiques Appliquées - Université Paris-Saclay
INRAE, unité MaIAGE à Jouy-en-Josas
Encadré par Maud DELATTRE et Guillaume KON KAM KING
11 mai - 31 août 2023

Table des matières

1 Introduction

1.1 Contexte général

L'INRAE (Institut National de Recherche pour l'Agriculture, l'Alimentation et l'Environnement) est un institut public de recherche sous la double tutelle du ministère de l'enseignement supérieur et de la recherche et du ministère de l'agriculture. Les missions principales de l'institut sont de répondre à différents enjeux sociétaux actuels tels que la gestion des ressources naturelles, des écosystèmes et la transition des agricultures. Les disciplines de recherche sont diversifiées et les biologistes sont amenés à travailler avec des mathématiciens et des informaticiens.

L'INRAE compte 18 centres de recherche en France, 14 départements de recherche et plus de 10000 agents. Le département Mathématiques et Numérique (MathNum) rassemble 12 unités dont MaIAGE (Mathématiques et Informatique Appliquées du Génome à l'Environnement) localisée sur le site de Jouy-en-Josas et qui compte 22 chercheurs et enseignants-chercheurs et 20 doctorants.

Un enjeu important à l'heure actuelle est la compréhension de l'interaction entre la plante et son environnement. Comment savoir si une variété donnée de plante cultivée a la capacité de résister et de s'adapter au changement climatique ? C'est une des questions au coeur du travail de chercheurs et de doctorants de l'INRAE, notamment des membres de l'unité MaIAGE, qui développent de nouvelles méthodes statistiques selon différents axes de recherche (par exemple dans le cadre du projet Stat4Plant).

L'objet de ce stage est de réaliser une étude comparative de plusieurs méthodes de sélection de variables en grande dimension dans le contexte des modèles non linéaires à effets mixtes, et ce, par le biais de différentes évaluations sur des données simulées. Un des intérêts de ce travail est par exemple, de pouvoir identifier les marqueurs génétiques bénéfiques pour la croissance de plantes. Ces modèles sont pertinents ici car chaque individu est observé à plusieurs reprises au cours du temps. Il est important de disposer de méthodes efficaces étant donné que cela peut avoir des conséquences sur le travail des biologistes.

Les modèles non linéaires à effets mixtes sont intéressants dans ce contexte car ils permettent d'exprimer la variabilité entre les individus qui existe par les données observées et par des effets aléatoires, c'est cette combinaison que l'on appelle effets mixtes. De plus, étant donné que l'on effectue plusieurs observations pour un seul individu, on pourra suivre l'évolution au cours du temps du trait phénotypique souhaité, le modèle permet donc aussi d'exprimer une variabilité intra-individuelle. Ils peuvent être utilisés par exemple dans la modélisation de phénomènes de croissance logistique ou exponentielle comme en propagation d'épidémies ou en pharmacocinétique. Dans ce type de modèle, chaque variable est associée à un coefficient inconnu à estimer. La sélection de variables consiste à déterminer quelles variables ont une influence et cela se traduit par un coefficient éloigné de 0.

L'approche statistique choisie ici est purement bayésienne, c'est-à-dire que l'on va considérer le vecteur de coefficients à estimer comme une variable aléatoire, il faut donc lui associer une distribution de probabilité adéquate qui correspond à la connaissance a priori sur ce vecteur sans avoir pu observer les données au préalable. L'objectif clé de ce stage est de considérer plusieurs lois a priori (ou priors) pertinentes pour la sélection de variables, puis, de les comparer à l'aide de différents critères lors de simulations numériques.

Ce stage fait suite aux travaux de Marion Naveau [1], doctorante de l'unité MaIAGE, qui a étudié une méthode spécifique de sélection de variables sur des modèles non linéaires à effets mixtes dans un contexte de grande dimension. On peut mentionner cependant que, même si cette méthode donne de bons résultats et a été développée en détail, elle effectue souvent une sous-sélection, c'est-à-dire que parmi les variables qui sont véritablement significatives, certaines ne sont pas détectées, et ce phénomène apparaît quelque soit les réglages préliminaires. De plus, les données génomiques donnent souvent lieu à de fortes corrélations qui dégradent la qualité des résultats de cette méthode, c'est pourquoi on veut comparer différentes méthodes pour déterminer celles qui effectuent les meilleures sélections et qui sont le moins sensibles à la présence de corrélation. Cette approche diffère du travail de M. Naveau, qui a étudié en détail un seul type de prior très spécifique. On préfère cette approche bayésienne à l'approche fréquentiste car la non-linéarité du modèle rend l'expression de la vraisemblance difficile à utiliser, il n'est donc pas évident de trouver un estimateur commode pour cette étude comme l'estimateur du maximum de vraisemblance par exemple. On souhaite aussi utiliser le même algorithme pour tous les priors étudiés.

Pour l'algorithme d'inférence, on utilise le package Nimble de R [2] qui offre un cadre pour écrire des algorithmes de Monte-Carlo par chaîne de Markov (MCMC). Cela permet d'estimer les coefficients souhaités à partir d'échantillons de chaînes de Markov. Il sera cependant important de vérifier la convergence de ces chaînes pour être sûr d'échantillonner dans les distributions voulues. Cette méthode a pour avantage d'être très flexible car utilisable pour tous les priors étudiés. A contrario, l'article de M. Naveau [1] utilise un algorithme de type SAEM (Stochastic Approximation - Expectation - Maximisation) spécifiquement conçu pour le prior étudié.

Notre objectif est de dresser un récapitulatif des mérites comparés des différents priors testés en fonction des résultats et des différents scénarios.

1.2 Contributions

Lors de la phase préliminaire de bibliographie que j'ai effectué au début du stage, j'ai pu identifier différentes méthodes de sélection à l'aide des articles consultés. En parallèle de cela, je me suis familiarisé avec la programmation en Nimble sur des modèles non linéaires sans effets mixtes dans un premier temps pour pouvoir ensuite mener à bien les simulations plus complexes avec le modèle souhaité. J'ai également proposé un algorithme MCMC sur un autre modèle pour un collègue stagiaire.

Ma contribution majeure est d'avoir choisi les différents priors et critères de comparaison constituant le plan d'expérience. Après avoir procédé aux premiers calculs, j'ai cherché des propriétés d'ordre théorique sur les lois utilisées afin d'améliorer dans certains cas la convergence des chaînes de Markov.

En complément de cela, j'ai cherché à accélérer le premier code que j'avais produit, ainsi, j'ai pu proposer une implémentation en parallèle de mon code afin d'en améliorer conséquemment l'efficacité, notamment lorsqu'on souhaite utiliser un grand nombre de jeux de données. Cette fonctionnalité n'a pas été évidente à mettre en place dû au fonctionnement de Nimble, où les modèles probabilistes et les échantillonneurs MCMC doivent être compilés et configurés dans un premier temps. À la suite de cela, j'ai pu effectuer un nombre conséquent de simulations afin d'obtenir des résultats selon différents scénarios.

1.3 Plan du rapport

Dans un premier temps, on posera le cadre mathématique pour définir les modèles non linéaires à effets mixtes de cette étude, puis on détaillera les différents articles de la bibliographie et leurs apports respectifs.

Ensuite, on présentera dans la section 4 les différents priors puis on justifiera leur pertinence dans le contexte de la sélection de variables. Une fois les priors définis, on explicitera les critères qui permettront de comparer ces priors en fonction des informations qu'ils apportent. Enfin, on reviendra sur les notions qui interviennent dans les méthodes de Monte-Carlo par chaînes de Markov.

Par la suite, il sera question dans la section 5 de l'implémentation faite durant le stage, en présentant les spécificités du package Nimble et son fonctionnement. On reviendra aussi sur les premiers modèles qui ont été traités afin de se familiariser avec Nimble, puis, on se placera dans le cas des modèles non linéaires à effets mixtes. On mentionnera les difficultés rencontrées et les solutions qui ont été trouvées pour y remédier.

Une fois toutes les notions bien définies, on se penchera sur les différents résultats numériques de la section 6 obtenus à la suite des simulations et on procédera d'abord à la comparaison des priors dans les différents scénarios simulés, puis on effectuera une étude plus spécifique pour certains priors en observant comment le nombre d'erreurs évolue lorsque l'on est amené à faire varier les hyperparamètres utilisés.

L'étude précédente réalisée sur des données générées aura permis de choisir les priors les plus pertinents. On appliquera en section 7 l'algorithme utilisé sur des données réelles de sénescence de variétés de blé en reprenant ces priors. Enfin, on pourra conclure sur le travail effectué lors de ce stage et sur les différents résultats obtenus, et présenter les perspectives futures qui pourront être étudiées à la suite de ce travail.

2 Présentation détaillée du modèle

La situation est la suivante : on observe n individus de façon répétée au cours du temps, avec n_i observations pour l'individu i. Voici le modèle mathématique étudié :

Pour $1 \leq i \leq n$ et $1 \leq j \leq n_i$:

$$\begin{cases} y_{ij} = f(\varphi_i, t_{ij}) + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2) \quad (1) \\ \varphi_i = \mu + \beta^\top V_i + \xi_i, \quad \xi_i \sim \mathcal{N}(0, \Gamma) \quad (2) \end{cases}$$

Ici, y_{ij} désigne la grandeur réelle correspondant au trait phénotypique observé au temps t_{ij}, par exemple, si on s'intéresse à la croissance d'une variété de plantes, cette grandeur peut représenter la taille de l'individu.

On fait l'hypothèse que y_{ij} conditionnellement à φ_i se comporte comme une loi normale dont la moyenne s'exprime comme une fonction f non linéaire des instants t_{ij} et des variables φ_i, de plus, on suppose que cette fonction est toujours la même, quelque soit l'individu ou l'instant observé. La variance σ^2 correspond à l'erreur de mesure effectuée.

Les variables φ_i sont des variables latentes dans la mesure où on ne les observe pas directement, le modèle fait l'hypothèse que chaque φ_i s'exprime comme la somme d'un intercept μ, d'une combinaison linéaire de valeurs de covariables observées pour l'individu i désigné par le vecteur V_i, et enfin d'un bruit gaussien ξ_i de variance Γ qui exprime la variabilité aléatoire intrinsèque entre les individus. Pour simplifier la situation, on suppose ici que σ^2, μ et Γ sont connus.

On cherche à faire la sélection sur ce que l'on appelle des covariables pour les distinguer des autres variables non sujettes à la sélection comme le temps par exemple, on désigne leur nombre par p. La matrice des covariables est désignée par $V \in \mathcal{M}_{n\times p}(\mathbb{R})$, elle peut contenir par exemple les marqueurs génétiques que l'on veut inclure dans la modélisation, dans ce cas les coefficients seront surtout binaires (présence ou absence du marqueur), mais on se placera dans le cas général où les valeurs observées sont réelles pour les simulations numériques.

L'équation (2) exprime la variabilité inter-individuelle, étant donné que chaque individu donne lieu à une valeur de φ différente, elle-même obtenue à la fois par un effet fixe donné par μ connu et β à estimer et par un effet aléatoire ξ_i qui dépend de l'individu. C'est la combinaison d'effets fixes et aléatoires qui justifie le nom "d'effets mixtes" du modèle.

L'équation (1) quant à elle exprime clairement la variabilité intra-individuelle, où on peut observer, pour un individu fixé, l'évolution de la grandeur y_{ij} au cours du temps.

L'objectif de la sélection de variables et de déterminer quelles covariables interviennent effectivement dans le phénomène observé, cela revient à savoir quels coefficients du vecteur $\beta \in \mathbb{R}^p$ sont considérés comme non-nuls après leur estimation. On considère que l'on se place dans un cadre de grande dimension, où $p \gg n$, et on supposera d'ailleurs que la grande majorité des coefficients de β sont négligeables.

3 Synthèse bibliographique

Comme cela a été mentionné dans l'introduction, le travail effectué lors de ce stage repose en grande partie sur l'article de M. Naveau [1]. Il est question ici du même type de modèle et des mêmes notions théoriques, la méthodologie et les objectifs de ce stage sont différents car on cherche à comparer plusieurs priors et non pas à étudier un seul prior en détail. Le prior en question est le Spike and Slab Continu Normal, il sera également utilisé ici. De plus, les simulations numériques de ce rapport reprennent des valeurs similaires pour les différents paramètres, enfin, le jeu de données réelles étudié à la fin a été fourni par M. Naveau elle-même qui a déjà pu appliquer sa méthode pour l'étudier. C'est également le seul article où un modèle à effets mixtes est étudié.

L'article de Porwal et Raftery [3] met en évidence 21 méthodes pour procéder à la sélection de modèles dans un contexte linéaire. Bien que plusieurs méthodes d'origine bayésienne soient utilisées, cet article ne se limite pas à

ce cadre, il fait mention du LASSO, de l'elastic net, mais également de la sélection de modèle par AIC par exemple. D'autres priors ont pu être utilisés comme le prior Zellner-Siow-Cauchy qui a servi de référence pour établir le classement, ce n'est toutefois pas le meilleur prior qui a été relevé car la première place appartient à une loi de type g-prior. Cet article est très succinct et les méthodes comparées ne sont pas clairement définies, les paramètres utilisés ne sont donc pas toujours précisés. Il est aussi intéressant car il présente différents critères repris ici afin d'effectuer des comparaisons.

Dans l'article de Malsiner-Walli et Wagner [4]. Il est spécifiquement question de sélection de variables pour la régression linéaire dans un contexte bayésien, où des priors de type Spike and Slab sont présentés et comparés, il présente notamment des versions continues et des versions utilisant une distribution de Dirac, il propose aussi des algorithmes MCMC spécifiques à chaque type de prior pour obtenir des échantillons selon les distribution a posteriori correspondantes. On reprendra certains d'entre eux pour cette étude. Pour l'article d'O'Hara et Sillanpää [5], l'approche est également bayésienne, mais n'est pas restreinte aux Spike and Slab.

La famille de priors de type Spike and Slab est très importante et adaptée à la sélection de variables, mais une autre famille de priors, celle de type Horseshoe semble également prometteuse. La première version de prior Horseshoe, qui est aussi la plus simple, est présentée par Carvahlo, Polson et Scott [6], mais il existe des versions plus récentes depuis, notamment le prior Horseshoe+, proposé par Bhadra et al. [7], cet article présente des graphes des vraisemblances de différents priors mais la comparaison faite a surtout lieu entre les priors Horseshoe et Horseshoe+, des propriétés théoriques sont également démontrées sur ce dernier. D'autres versions du Horseshoe sont proposées où des hyperpriors sont utilisés afin d'éviter de prendre des valeurs arbitraires pour les différents paramètres qui interviennent dans la définition du prior, comme dans l'article de Piironen et Vehtari [8]. Encore une fois, c'est dans un contexte linéaire que ces priors sont utilisés.

Un autre type de prior discuté ici sera le prior Laplace, qui est l'équivalent bayésien du LASSO, il est défini dans la thèse de Han [9] qui compare également plusieurs méthodes de sélection de variables en contexte bayésien, il utilise en particulier un prior Spike and Slab qui fait apparaître une loi de Laplace. Pour le LASSO bayésien, il pose un certain hyperprior dont les valeurs conseillées de paramètres sont données par Zhang [10].

Les modèles non linéaires à effets mixtes en grande dimension semblent donc être très intéressants à approfondir étant donné que ce type de modèle n'est mentionné dans aucun article, sauf dans celui de M. Naveau [1], c'est pourquoi il sert de base à notre étude. Les autres articles ne font mention que de modèles linéaires sans effets mixtes dans un contexte de petite dimension, que le cadre soit bayésien ou non.

Au-delà des considérations de modélisation, l'aspect algorithmique est également très important. Il est question ici d'utiliser des algorithmes MCMC par le biais du package Nimble [2], dôté d'un manuel d'utilisation [11] qui détaille comment définir et utiliser des modèles probabilistes dans Nimble, mais qui explique aussi les échantillonneurs MCMC utilisés. Ces derniers reposent fortement sur des algorithmes de type Metropolis-Hastings et échantillonneur de Gibbs, définis par exemple dans le livre de Marin et Robert [12].

4 Choix méthodologiques

Dans cette section, on détaillera les définitions des différents priors étudiés tout en justifiant pourquoi certains sont pertinents pour la sélection de variables, et quelles sont leurs limites éventuelles. Ensuite, les critères de comparaison seront présentés succinctement, et les notions importantes de MCMC seront présentées à la fin.

4.1 Priors étudiés

Dans le cadre bayésien, les estimations sont faites à partir de la distribution a posteriori de chacun des paramètres, elle-même obtenue par les données observées et par la loi a priori posée sur le paramètre selon la formule de Bayes :

$$p(\beta \mid y) \propto p(\beta) L(y \mid \beta)$$

Les lois a priori étudiées doivent posséder des propriétés pertinentes pour la sélection de variables et décrire l'information que l'on possède sur β avant d'avoir pu voir les données. Les informations a priori peuvent se résumer ainsi :

- La distribution doit être symétrique par rapport à l'axe des ordonnées, il n'y a pas plus de chances d'avoir un coefficient positif plutôt que négatif ou inversement.

- La majorité des coefficients seront négligeables, on veut donc une grande probabilité a priori d'avoir des coefficients nuls ou quasi-nuls.

- Les autres coefficients seront significatifs, il faut donc que la distribution qui leur soit associée soit assez peu informative, la forme de la distribution doit être assez plate avec une variance importante afin d'avoir de bonnes estimations car ces coefficients peuvent éventuellement prendre de grandes valeurs. Ils ne doivent pas se retrouver 'emprisonnés' dans un intervalle trop restreint.

On veut aussi que les priors proposés ne comptent pas trop de paramètres libres en général afin qu'ils soient analysables et de limiter les choix arbitraires. Voici les différents types de priors que l'on a été amené à étudier :

4.1.1 Spike and Slab Continu Normal

Pour $1 \leq l \leq p$:

$$\begin{cases} \alpha \sim Beta(a, b) \\ \delta_l \sim Bern(\alpha) \\ \beta_l \,|\, \delta_l \sim \mathcal{N}(0, (1-\delta_l)\nu_0 + \delta_l \nu_1) \quad \text{avec} \quad 0 < \nu 0 \ll \nu 1 \end{cases}$$

C'est le prior qui a été étudié en détail par M. Naveau [1], il a déjà démontré de bonnes qualités en sélection de variables. Le principe des priors de type Spike and Slab est de combiner deux distributions, ici toutes les deux gaussiennes, avec d'un côté, une distribution ayant une variance ν_0 très faible (Spike) qui est adapté pour les coefficients négligeables car ils sont alors fortement rapprochés de 0, et de l'autre côté une distribution plus aplatie à très forte variance ν_1 (Slab) qui convient pour les coefficients significatifs, étant donné que l'on veut qu'ils puissent varier sur une large plage de valeurs sans préférence particulière.

Le paramètre α designant ici la probabilité pour un coefficient donné d'être classé comme 'Slab', on lui associe une loi $Beta$ avec des hyperparamètres a et b de telle sorte à ce qu'un poids plus élevé soit mis du côté de 0 étant donné que peu de covariables sont supposées significatives. On distingue alors les deux classes avec $\delta_l = 0$ pour les covariables inactives et $\delta_l = 1$ pour celles qui sont actives.

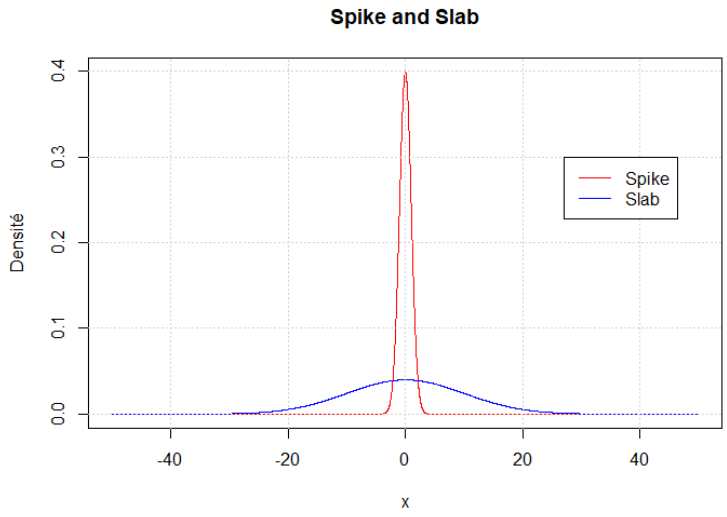

FIGURE 1 – Distributions Spike et Slab gaussiennes

4.1.2 Spike and Slab Continu Student

On rappelle la densité d'une loi de Student $\mathcal{T}_\nu(x_0, \sigma^2)$ non standard avec sa paramétrisation (ν étant le degré de liberté) :

$$x \longrightarrow \frac{\Gamma(\frac{\nu+1}{2})}{\Gamma(\nu/2)} \frac{1}{\sqrt{\sigma^2 \nu \pi}} \left(1 + \frac{(x - x_0)^2}{\sigma^2 \nu}\right)^{-\frac{\nu+1}{2}}$$

On définit alors le prior :

Pour $1 \leq l \leq p$:

$$\begin{cases} \alpha \sim Beta(a, b) \\ \delta_l \sim Bern(\alpha) \\ \beta_l \,|\, \delta_l \sim \mathcal{T}_\nu(0, (1 - \delta_l)\nu_0 + \delta_l \nu_1) \quad \text{avec} \quad 0 < \nu 0 \ll \nu 1 \end{cases}$$

Ce prior est très similaire au précédent, la paramétrisation est choisie de telle sorte à ce que quand ν devient très grand, alors on revient dans le cas du prior Spike and Slab Normal. Ce prior proposé par Malsiner-Walli et Wagner [4] a la propriété d'avoir des queues de distribution plus épaisses en utilisant une loi de Student plutôt qu'une gaussienne, cela induit une distribution moins informative pour les coefficients sélectionnés. Un autre intérêt est de considérer ce prior comme une généralisation du précédent, ainsi, le nombre de degrés de liberté ν apparaît comme un hyperparamètre à faire varier afin de voir comment évoluent les résultats.

4.1.3 Spike and Slab Dirac Normal

Pour $1 \leq l \leq p$:

$$\begin{cases} \alpha \sim Beta(a, b) \\ \delta_l \sim Bern(\alpha) \\ \beta_l \,|\, \delta_l \sim (1 - \delta_l)\delta(0) + \delta_l \mathcal{N}(0, \nu_1) \quad \text{avec} \quad 1 \ll \nu 1 \end{cases}$$

Ce prior est également présenté dans l'article de Malsiner-Walli et al. [4], on pose ici une distribution de Dirac en 0 pour la composante Spike, tandis que la composante Slab demeure une gaussienne. On peut interpréter ce prior comme un cas limite du Spike and Slab Normal continu avec ν_0 très proche de 0. Un avantage de ce prior est de s'affranchir du paramètre ν_0.

4.1.4 g-prior et Spike and Slab g-slab

g-prior :

$$\beta \sim \mathcal{N}_p(0, g\Gamma(V^\top V)^{-1})$$

Ce prior a été testé au début du stage car il a donné de bons résultats d'après Porwal et Raftery [3] dans un modèle linéaire. Seulement, il se trouve que ce prior est très sensible à la présence de corrélation entre les covariables. De plus, il nécessite que la matrice $V^\top V$ soit inversible, on peut démontrer que cela est équivalent à ce que la famille des colonnes de V soit linéairement indépendante, or, cela implique aussi en particulier que $Ker(V) = \{0\}$, donc $rg(V) = p$ par le théorème du rang, et ainsi $p \leq n$. Le prior est donc inutilisable en grande dimension, bien qu'il existe une généralisation proposé par Maruyama et George [13], mais cette version n'a pas encore été implémentée car assez sophistiquée.

Spike and Slab Continu g-slab :

Pour $1 \leq l \leq p$:

$$\begin{cases} \alpha \sim Beta(a, b) \\ \delta_l \sim Bern(\alpha) \\ D_\delta = diag((1 - \delta)\nu_0 + \delta \nu_1) \\ \beta \,|\, \delta \sim \mathcal{N}_p(0, D_\delta^{1/2} g (V^\top V)^{-1} D_\delta^{1/2}) \end{cases}$$

Ce prior cherche à combiner les propriétés du Spike and Slab et du g-prior, le problème étant qu'il fait face aux mêmes écueils que le g-prior, son étude n'a pas été poursuivie sachant que l'on souhaite travailler en grande dimension.

On peut bien entendu imaginer une pléthore de possibilités pour un prior de type Spike and Slab (Spike gaussienne avec Slab de loi Laplace, etc) mais on se contentera d'étudier les variantes présentées ci-dessus.

4.1.5 Horseshoe

$C^+(x_0, \gamma)$ désigne une loi de Cauchy sur $\mathbb{R}^+$ qui a pour densité :

$$x \longrightarrow \frac{2}{\pi\gamma} \frac{1}{1 + \left(\frac{x - x_0}{\gamma}\right)^2}$$

Le prior est ainsi défini par :

Pour $1 \leq l \leq p$:

$$\begin{cases} \tau \sim C^+(0, 1) \\ \lambda_l \sim C^+(0, 1) \\ \beta_l \,|\, \lambda_l \sim \mathcal{N}(0, \lambda_l^2 \tau^2) \end{cases}$$

Les priors de type Horseshoe sont assez intéressants pour la sélection de variable car en définissant le coefficient de rétrécissement $\kappa_l = 1/(1 + \lambda_l^2)$, on a que :

$$\lambda_l \sim C^+(0, 1) \implies \kappa_l \sim Beta(1/2, 1/2)$$

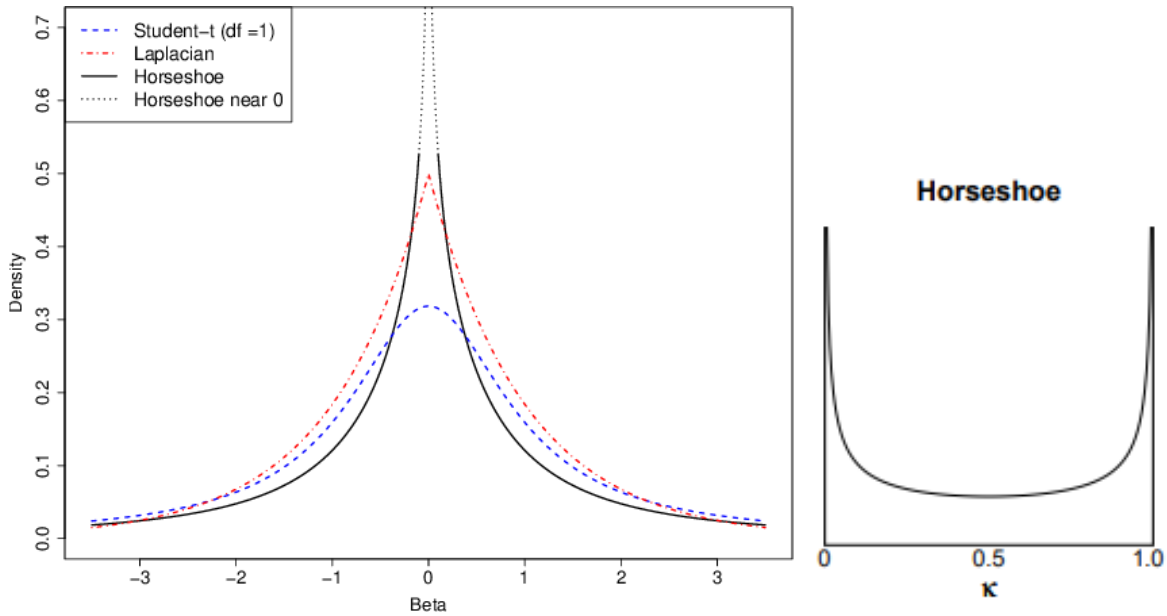

FIGURE 2 – Distribution Horseshoe sur β (à gauche) et distribution $Beta$ sur κ (à droite) (source images : [6])

Cette distribution de type $Beta$ est très commode car elle place un poids important sur les valeurs 0 et 1. Une valeur de 0 pour κ_l correspond à une absence de rétrécissement qui donne lieu à une variance λ_l^2 très importante, ce qui est souhaité pour des coefficients significatifs et a contrario, une valeur de 1 pour κ_l donne un rétrécissement vers 0 très appuyé, donc une variance faible, c'est ce que l'on attend pour des coefficients négligeables. De plus, sa forme rappelle un fer à cheval qui justifie le nom de ce prior.

Les propriétés évoquées ci-dessus se voient bien sur la distribution Horseshoe qui est très pointue et place beaucoup de masse en 0, mais tout en ayant des queues de distribution épaisses.

On peut noter la présence d'un paramètre τ qui correspond à un coefficient de rétrécissement global et affecte toutes les composantes de β. On souhaite que ce coefficient soit proche de 0 lorsque le nombre de covariables

négligeables est élevé, de telle sorte à ce que les variances des coefficients soient plus proches de 0 dans leur ensemble. La première version du prior Horseshoe proposée par Carvalho et al. [6] impose $\tau = 1$, mais des variantes que l'on retrouve dans l'article de Piironen et Vehtari [8] proposent de mettre une loi de Cauchy sur $\mathbb{R}^+$ pour τ, c'est cette dernière version que l'on a presenté et que l'on étudiera.

4.1.6 Horseshoe+

Pour $1 \leq l \leq p$:

$$\begin{cases} \tau \sim C^+(0,1) \\ \eta_l \sim C^+(0,1) \\ \lambda_l \sim C^+(0,\eta_l) \\ \beta_l \,|\, \lambda_l \sim \mathcal{N}(0, \lambda_l^2 \tau^2) \end{cases}$$

Cette version plus récente du Horseshoe proposée par Bhadra et al. [7] a, d'après cet article une masse plus forte en 0 et des queues de distributions plus épaisses que le Horseshoe précédent. On se propose cependant de le modifier légèrement, de telle sorte que le paramètre τ ait la même interprétation que pour le prior Horseshoe et que leur comparaison soit plus aisée. Une loi de Cauchy sur $\mathbb{R}^+$ est un hyperprior raisonnable pour τ.

Après un long calcul, on a les distributions a priori suivantes pour $1 \leq l \leq p$:

$$p(\lambda_l) = \frac{2}{\pi^2} \frac{\ln\left(\lambda_l^2\right)}{\lambda_l^2 - 1} \quad \text{et} \quad p(\kappa_l) = \frac{1}{\pi^2} \frac{\kappa_l^{-1/2}(1-\kappa_l)^{-1/2}}{1-2\kappa_l} \ln\left(\frac{1-\kappa_l}{\kappa_l}\right)$$

La distribution pour κ_l est symétrique selon l'axe $\kappa = 1/2$, comme le Horseshoe, mais sur le graphe de la distribution, on peut voir qu'un poids encore plus important est placé sur 0 et 1 ce qui est le but recherché et donc le Horseshoe+ est une amélioration du Horseshoe du point de vue théorique. On verra lors de la comparaison des résultats si cette amélioration est substantielle.

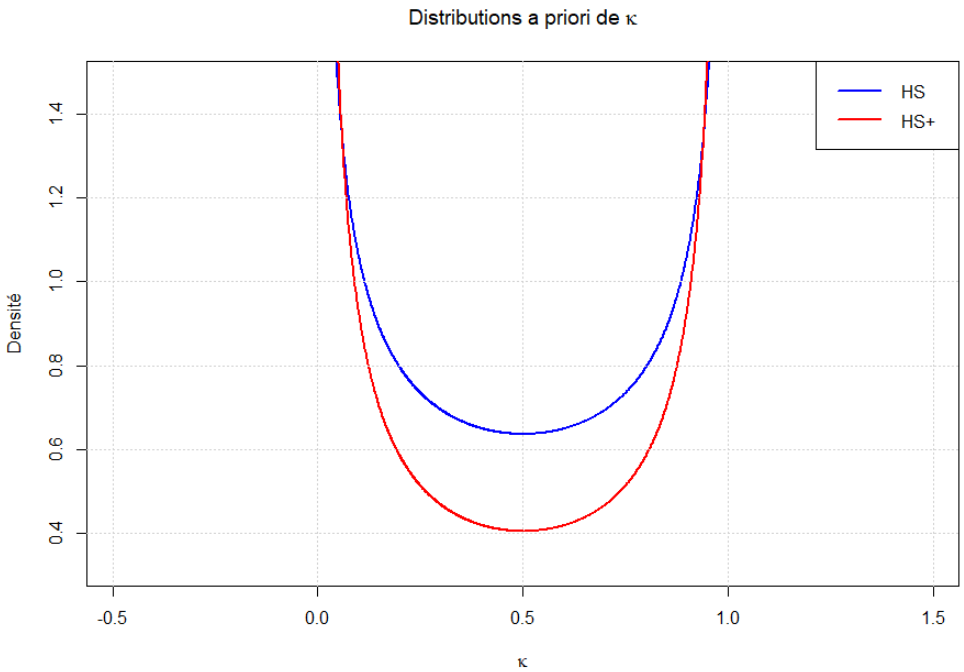

FIGURE 3 – Distributions a priori sur κ_l pour les priors Horseshoe et Horseshoe+

Une remarque importante est que lorsque l'on implémente les priors Horseshoe directement avec des lois de Cauchy, les chaînes de Markov convergent difficilement pour un grand nombre de coefficients. On utilise la propriété suivante pour introduire une variable latente et améliorer la convergence :

Propriété :

$$\begin{cases} U \,|\, \omega \sim InvGamma(1/2, 1/\omega) \\ \omega \sim InvGamma(1/2, 1/\gamma^2) \end{cases} \implies \sqrt{U} \sim C^+(0, \gamma)$$

(Démonstration dans l'annexe A)

Ainsi, pour détailler les structures respectives des priors Horseshoe et Horseshoe+ :

$$\text{HS} : \begin{cases} \omega_\tau \sim InvGamma(1/2, 1) \\ \tau^2 \sim InvGamma(1/2, 1/\omega_\tau) \\ \omega_{\lambda_l} \sim InvGamma(1/2, 1) \\ \lambda_l^2 \sim InvGamma(1/2, 1/\omega_{\lambda_l}) \\ \beta_l \,|\, \lambda_l \sim \mathcal{N}(0, \lambda_l^2 \tau^2) \end{cases} \qquad \text{HS+} : \begin{cases} \omega_\tau \sim InvGamma(1/2, 1) \\ \tau^2 \sim InvGamma(1/2, 1/\omega_\tau) \\ \omega_{\eta_l} \sim InvGamma(1/2, 1) \\ \eta_l^2 \sim InvGamma(1/2, 1/\omega_{\eta_l}) \\ \omega_{\lambda_l} \sim InvGamma(1/2, 1/\eta_l^2) \\ \lambda_l^2 \sim InvGamma(1/2, 1/\omega_{\lambda_l}) \\ \beta_l \,|\, \lambda_l \sim \mathcal{N}(0, \lambda_l^2 \tau^2) \end{cases}$$

4.1.7 Prior Laplace

On rappelle la densité d'une loi $Laplace(x_0, b)$:

$$x \longrightarrow \frac{1}{2b} \exp\left(-\frac{|x - x_0|}{b}\right)$$

On utilise la propriété suivante pour la simulation :

$$X \sim Rademacher\,,\, Y \sim Exp(b) \implies XY \sim Laplace(0, 1/b)$$

Le prior est donc le suivant :

Pour $1 \leq l \leq p$:

$$\begin{cases} \lambda^2 \sim Gamma(r, s) \\ \beta_l \,|\, \lambda^2 \sim Laplace(0, \sqrt{\Gamma/\lambda^2}) \end{cases}$$

Ce prior Laplace est l'équivalent bayésien du Lasso [9], une méthode fréquentiste qui permet d'effectuer de la sélection de variable même dans un cadre de grande dimension, tant que le nombre de coefficients significatifs n'est pas trop élevé, ce qui correspond à l'hypothèse posée.

On décide de placer un hyperprior de type $Gamma$ sur λ^2, selon les recommandations de Zhang [10] avec les valeurs $r = 0.1$ (shape) et $s = 1$ (rate) pour que cet hyperprior encourage les faibles régularisations.

4.2 Critères de comparaison

Avec les échantillons a posteriori obtenus par l'algorithme, on peut calculer toutes les informations que l'on souhaite, dont les différents critères qui sont présentés dans cette section. Ils mesurent différents aspects, à savoir, la qualité d'estimation, la qualité de sélection et l'efficacité de l'algorithme.

Critères d'estimation

— Erreur quadratique moyenne :

$$RMSE = \sqrt{\frac{1}{p} \sum_{l=1}^{p} (\beta_l^{true} - \hat{\beta}_l)^2}$$

On effectue ici la comparaison entre la vraie valeur de β utilisée pour générer les données et la moyenne a posteriori $\hat{\beta}$ qui sert d'estimation ponctuelle, l'intérêt de ce critère est réduit dans la mesure où seule la moyenne est utilisée, qui ne reflète pas l'incertitude d'estimation.

— Continuous Ranked Probability Score :

$$CRPS(\beta^{true}, F) = \int_{\mathbb{R}} (F(x) - \mathbb{1}(x \geq \beta^{true}))^2 dx$$

Ce critère permet de prendre en compte l'incertitude sur chaque coefficient car on utilise ici la fonction de répartition F obtenue à partir de l'échantillon a posteriori tout entier, on ne se restreint donc pas à la moyenne. On compare ici F à une fonction de type Heaviside qui correspondrait à une situation parfaite dans la mesure où la distribution a posteriori serait une distribution de Dirac en la vraie valeur, dans ce cas, on retrouverait toujours la valeur attendue avec une incertitude nulle.

— Mean Interval Score :

$$IS_\alpha(l, u, z) = (u - l) + \frac{2}{\alpha}(l - z)\mathbb{1}(z < l) + \frac{2}{\alpha}(z - u)\mathbb{1}(z > u)$$

Ce critère apparaît dans l'article de Porwal et Raftery [3], il permet de valoriser les posteriors qui donnent des intervalles de crédibilité de qualité, dans la mesure où ils sont à la fois étroits et contiennent la vraie valeur $z = \beta^{true}$. On choisit ici $\alpha = 0.05$. Cette mesure pourrait aussi s'utiliser pour une situation fréquentiste assortie d'un intervalle de confiance.

Critères de sélection

Le but principal étant de procéder à une sélection de variables, comment décide-t-on si une covariable est considérée comme significative à partir des résultats obtenus ? Deux conventions de sélection de variables ont été considérées. La première repose sur la moyenne a posteriori $\hat{\beta}_l$ et on considère que la covariable est sélectionnée lorsque $|\hat{\beta}_l| > seuil$, où $seuil > 0$. La deuxième utilise $IC(\beta_l)$, l'intervalle de crédibilité à 95 %, et décide que le coefficient est négligeable si $0 \in IC(\beta_l)$.

La première méthode a été conservée, car il arrive souvent que les intervalles de confiances soient assez grands pour chacun des priors, et ce, même en l'absence de corrélation. On risque donc d'avoir très souvent des faux négatifs, notamment pour des coefficients significatifs mais proches de 0. Le seuil est choisi de manière à être faible devant la valeur attendue du plus petit β non nul.

— Taux de mauvais classement :

$$Misc.\,Rate = \frac{FP + FN}{p}$$

On veut savoir combien d'erreurs ont été réalisées au total, on place aussi la même importance sur les faux positifs que sur les faux négatifs, on ne cherche pas à privilégier la sur-sélection ou la sous-sélection de covariables. Il faut cependant garder à l'esprit que les classes de covariables sont très déséquilibrées, les covariables négligeables sont en surnombre, ainsi, la règle de classification naïve qui assigne toutes les covariables comme étant négligeables aura automatiquement un taux de mauvais classement très faible.

— Sensibilité et Spécificité :

$$Sensitivity = \frac{TP}{TP + FN} \quad \text{et} \quad Specificity = \frac{TN}{TN + FP}$$

On calcule ces critères afin d'avoir des informations plus fines sur le type d'erreur réalisé par le prior, ils permettent notamment de savoir si un prior produit surtout des faux positifs ou des faux négatifs. Les critères de sensibilité et spécificité peuvent s'interpréter comme la proportion de covariables bien classées parmi les covariables effectivement significatives et négligeables respectivement.

— Courbes ROC et PR :

Étant donné que l'on ne veut pas se limiter à des comparaisons sur des seuils arbitraires, on cherchera à comparer les courbes ROC des différents priors, qui correspondent au graphe de la sensibilité en fonction de 1-spécificité obtenu en faisant varier le seuil sur $\mathbb{R}^+$. Seulement, cette approche n'est pas forcément la plus adaptée car la courbe ROC n'est pas sensible au fait que les covariables négligeables soient en surnombre par rapport aux covariables significatives. En plus des courbes ROC, on s'intéressera aussi aux courbes PR (Precision-Recall) où :

$$Precision = \frac{TP}{TP + FP} \quad \text{et} \quad Recall = Sensitivity$$

La précision exprime la proportion de covariables bien classées parmi celles qui sont estimées significatives, par conséquent, s'il y a une large majorité de covariables négligeables, la précision aura tendance à être moins élevée car le nombre de vrais positifs possibles sera alors considérablement réduit tandis que les faux positifs peuvent être très nombreux.

Critères d'efficacité algorithmique

— Temps d'exécution : C'est un critère important à prendre en compte car selon les échantillonneurs MCMC utilisés par un prior ou un autre, il peut y avoir des différences notables de temps d'exécution.

— Effective Sample Size (ESS) :

Ce critère correspond au nombre d'échantillons requis pour avoir la même variance que dans le cas où les tirages sont indépendants. Les tirages d'une chaîne de Markov sont très souvent autocorrélés à un degré variable, et plus cette autocorrélation est élevée, moins l'échantillon a posteriori apporte d'informations sur le coefficient estimé.

On s'intéressera surtout au rapport de ces deux dernières quantités (ESS/s) qui sera interprété comme un critère d'efficacité de l'algorithme MCMC associé au prior : on cherche à obtenir le plus d'informations en le moins de temps.

4.3 Principe du Monte-Carlo par chaînes de Markov

L'objectif des méthodes MCMC est de générer des variables aléatoires suivant une distribution arbitraire. On souhaite simuler la loi a posteriori de chaque paramètre, or en général, la distribution a posteriori ne correspond pas à une loi usuelle. Dans de rares cas, les lois a priori et a posteriori sont conjuguées, c'est-à-dire qu'elles appartiennent à la même famille de lois, mais avec des valeurs de paramètres différentes.

Pour pouvoir effectuer cette simulation, on utilise un résultat important des chaînes de Markov. Une chaîne de Markov $(X_n)_{n\in\mathbb{N}}$ est un processus aléatoire sur un espace d'états $(E, \mathcal{E})$ où pour tout $n \geq 0$ et $A \in \mathcal{E}$:

$$\mathbb{P}(X_{n+1} \in A \,|\, X_0, X_1, ..., X_n) = \mathbb{P}(X_{n+1} \in A \,|\, X_n)$$

Pour une chaîne de Markov homogène :

$$\forall\,(x, y) \in E \times E, \forall\, n \geq 0, \quad \mathbb{P}(X_{n+1} = y \,|\, X_n = x) = \mathbb{P}(X_1 = y \,|\, X_0 = x) = P(x, y)$$

où les $(P(x, \cdot))_{x\in E}$ sont les densités de transition. Une chaîne de Markov est déterminée par sa loi initiale π_0 et par ses densités de transition. À noter que Nimble utilise des méthodes MCMC adaptatives, les chaînes ne sont pas homogènes, ainsi, leurs densités de transition dépendent de n, mais elles tendent vers une chaîne homogène.

De façon informelle, c'est un processus qui permet de décrire par exemple la proportion d'individus se trouvant dans un état $x \in E$ donné au cours du temps, avec la particularité suivante : pour déterminer le système dans l'instant suivant, seule l'information du système à l'instant présent intervient, les instants antérieurs n'ont pas d'influence.

Dans l'étude d'une chaîne de Markov, on recherche notamment sa loi stationnaire π si elle existe, car alors, sous certaines hypothèses, la proportion d'individus dans l'état x va tendre vers $\pi(x)$, autrement dit, le système va converger vers la distribution π :

Théorème ergodique (cas homogène) : Si P est irréductible, apériodique et Harris-récurrente, alors, la loi stationnaire π existe et est unique, et pour toute loi initiale π_0 :

— Convergence de (X_n) en loi vers $X \sim \pi$

— Pour toute fonction f intégrable par rapport à π :

$$\frac{1}{n}\sum_{k=1}^{n} f(X_k) \xrightarrow[n\to+\infty]{} \mathbb{E}_\pi(f(X)) = \int f(x)\pi(x)dx$$

Ce théorème est valable pour une chaîne non homogène mais qui tend vers une chaîne homogène sous certaines conditions [14].

Pour cette étude, on se place dans le cas où $E = \mathbb{R}$, $\mathcal{E} = \mathcal{B}(\mathbb{R})$ est la tribu borélienne sur $\mathbb{R}$ avec la mesure de Lebesgue sur $\mathbb{R}$ pour les coefficients de β.

Ainsi, pour simuler la distribution a posteriori, on simule une chaîne de Markov dont la loi stationnaire est justement cette loi a posteriori. Il existe plusieurs algorithmes permettant de faire cela, les plus répandus étant les algorithmes de Gibbs et de Metropolis-Hastings utilisés par ailleurs dans Nimble, ils ont la qualité de pouvoir être utilisés dans des modèles très variés.

Par exemple, l'algorithme de Metropolis-Hastings permet de simuler la loi a posteriori d'un paramètre θ à partir du processus suivant :

— On simule θ_0 selon une loi initiale π_0.

— À l'itération t, il faut choisir une loi de proposition de densité q_t (une distribution gaussienne par exemple). On simule une valeur candidate θ_{cand} selon la loi q_t sachant θ_{t-1}. Il faut ensuite calculer le ratio de Metropolis-Hastings :

$$r_{MH} = \frac{p(\theta_{cand} \,|\, y) q_t(\theta_{t-1} \,|\, \theta_{cand})}{p(\theta_{t-1} \,|\, y) q_t(\theta_{cand} \,|\, \theta_{t-1})}$$

Puis on simule $U \sim Unif(0,1)$. Si $r_{MH} < U$, $\theta_t = \theta_{cand}$, sinon, $\theta_t = \theta_{t-1}$.

On remarque que l'expression de r_{MH} se simplifie si q_t est symétrique. De plus, on voit que pour simuler la loi a posteriori de θ, seule l'expression de sa densité p à une constante de normalisation près est nécessaire, c'est pourquoi cet algorithme est aussi flexible.

Une fois que l'on dispose d'un tel algorithme, il faut que les échantillons des chaînes de Markov soient suffisamment grands afin que les chaînes puissent converger vers la distribution souhaitée. Il faut également supprimer les premières itérations car l'état initial ne doit pas avoir d'influence sur le résultat final, c'est le temps de chauffe. Il faut tout de même vérifier la convergence des chaînes de Markov après coup, cela peut se faire visuellement, mais on peut utiliser un critère plus quantitatif, celui de Gelman-Rubin, qui est le suivant :

Soit N la taille de l'échantillon généré, W la variance intra-chaîne moyenne et B la variance inter-chaîne, on pose :

$$\hat{R}^2 = \frac{\frac{N-1}{N}W + \frac{1}{N}B}{W}$$

On considérera que la chaîne converge si elle est assez 'stable' et que $\hat{R} < 1.05$. Une solution pour que les chaînes de Markov convergent mieux est de choisir un nombre d'échantillons N plus grand, mais il faut cependant que le calcul puisse se faire en un temps raisonnable.

On dispose maintenant de tous les priors, critères et notions pour pouvoir procéder à l'étude numérique faite en section 6. On revient cependant d'abord sur les premiers modèles testés au début du stage dans la section suivante.

5 Premiers pas en inférence bayésienne

Dans cette section, on reviendra sur deux modèles introductifs qui ont été implémentés au début du stage. Puis, on détaillera plus la programmation en Nimble, notamment pour effectuer des calculs en parallèle.

5.1 Premiers modèles implémentés

5.1.1 Modèle de croissance de plante

Le premier modèle statistique qui a été étudié au début du stage en guise d'introduction et afin de se familiariser avec Nimble est le suivant :

$$y_t = y_\infty + \frac{2(y_0 - y_\infty)}{1 + \exp(t/\tau)} + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \sigma^2)$$

Avec : $0 \leq y_0 < y_\infty \leq y_{max}$, $\tau \in \mathbb{R}^+$, $t \in \{t_1, t_2, ..., t_T\}$.

Il s'agit d'un modèle non linéaire à effets fixes, il permet de modéliser par exemple, la vitesse de croissance d'une plante, en suivant l'évolution de sa taille au cours du temps. Tout d'abord, il a été question de simuler des données selon ce modèle pour voir l'influence des différents paramètres sur les courbes, on peut dire que :

— y_0 représente la taille initiale.
— y_∞ désigne la taille dans le régime stationnaire.
— τ correspond à un temps caractéristique.

Avec des priors peu informatifs tels que des lois uniformes, on retrouve les vraies valeurs efficacement. Il faut préciser que si σ^2 est très élevé, ou que si τ est grand par rapport à l'échelle de temps choisie, les résultats sont moins bons car soit la variance est très grande par rapport aux données, soit le régime stationnaire n'est pas visible et y_∞ en particulier est difficile à estimer.

5.1.2 Modèle de cinétique pour des données omiques

Par la suite, j'ai pu échanger avec Sylvain Procope-Mamert, un stagiaire en Master 2 à l'INRAE, également encadré par Guillaume Kon Kam King et Maud Delattre, au sujet d'un modèle décrivant l'évolution de la concentration en ARN messager au cours du temps. Il cherche à inférer les paramètres :

$$y_t \mid \varphi \sim \mathcal{N}(f(\varphi, t), \sigma_0), \quad t = (0, 1, 2, 4, 6, 8, 10)$$
$$\varphi = (\log(A), \log(B), lag_0, \log t_{\frac{1}{2}})$$
$$f(\varphi, t) = \log\left(B + A \exp_2\left(-\frac{\max(0, t - lag_0)}{t_{\frac{1}{2}}}\right)\right)$$

Avec les hypothèses suivantes :

$$\begin{cases} B \ll A \\ -30 \leq \log_2 \frac{B}{A+B} \leq -3 \\ 0 \leq lag_0 \leq 3 \end{cases}$$

L'algorithme de nature fréquentiste effectuant une estimation par maximum de vraisemblance utilisé par S. Procope-Mamert avait parfois tendance à donner des valeurs assez éloignées de ce qui était attendu. Je lui ait donc proposé un algorithme Nimble avec une approche bayésienne et les priors suivants :

$$\begin{cases} \sigma_0 \sim InvGamma(1/2, 1/2) \\ \log(A) \sim \mathcal{N}(9.5, 0.5^2) \\ D = \frac{A}{A+B} \sim Unif(2^{-30}, 2^{-3}) \\ \log(B) = \log(A) - \log(\frac{1-D}{D}) \\ \log \mathrm{t}_{\frac{1}{2}} \sim \mathcal{N}(0, 0.6^2) \\ lag_0 \sim Unif(0, 3) \end{cases}$$

Les hypothèses posées et les valeurs de paramètres dans les priors sont issues des notions biologiques intervenant dans le stage de S.Procope-Mamert. Ses retours par rapport au code produit ont été positifs, il l'utilise désormais au cours de son stage et de son début de thèse.

5.2 Programmation avec Nimble

La génération de chaînes de Markov avec Nimble [2] se fait de la façon suivante :

— On déclare le modèle probabiliste que l'on veut utiliser, en spécifiant les paramètres et leurs lois a priori.
— On déclare si certains paramètres sont des constantes, ou des données (amenés à changer)
— On compile le modèle, Nimble utilise du C++ pour accélérer les calculs.
— On configure l'échantillonneur MCMC pour déterminer quels sont les paramètres d'intérêts et pour imposer certains samplers sur certaines variables. On utilisera ici la configuration par défaut de Nimble. On compile ensuite aussi l'échantillonneur MCMC.
— On peut alors lancer les chaînes de Markov après avoir spécifié leur nombre, le nombre d'itérations et le temps de chauffe.

Il est cependant possible d'effectuer toutes ces étapes avec une seule commande, ce qui est pratique au début, seulement, cela implique de refaire la compilation du modèle à chaque itération si on veut répéter le calcul. Cette étape de compilation peut être longue, notamment si on souhaite faire des calculs sur plusieurs modèles différents. Pour optimiser cela, il faut réaliser la compilation une seule fois au maximum, ce qui oblige à déclarer chaque étape mentionnée ci-dessus.

Pour être sûr d'effectuer une étude précise, il faut faire le calcul sur un grand nombre de jeux de données. Seulement, faire le calcul séquentiellement peut prendre un temps considérable, on se propose d'implémenter un code qui effectue ce calcul en parallèle sur plusieurs clusters. On combine alors l'utilisation de Nimble avec le package R `doParallel`. Les étapes sont les suivantes :

— On déclare toutes les fonctions et paramètres utiles pour le calcul.
— On déclare une session parallèle avec le nombre de clusters souhaités.
— On exporte l'environnement des variables enregistrées dans tous les clusters (avec `ClusterExport`).
— On initialise le générateur aléatoire dans chaque cluster afin de rendre le calcul reproductible (avec `ClusterApply`).
— On génère toutes les données (y et V) à l'avance que l'on partitionne en plusieurs parties qui sont chacune réparties dans un des clusters.
— Dans chaque cluster, on effectue la compilation du modèle Nimble et de l'échantillonneur MCMC une seule fois pour gagner du temps. On lance ensuite les chaînes de Markov et on effectue les calculs souhaités (avec `ClusterEvalQ`).
— Une fois le calcul terminé, on rassemble les résultats.

Dans la documentation Nimble, un exemple de calcul en parallèle est présenté, cependant, il y a une recompilation à chaque itération. Après avoir essayé plusieurs approches infructueuses, je suis parvenu à proposer un programme qui permet de s'affranchir de cela. Le temps d'exécution est donc plus court et il se compose d'une première phase où toutes les compilations sont faites, puis les chaînes de Markov sont lancées en parallèle et les différents jeux de données sont traités au fur et à mesure.

Dans Nimble, de multiples échantillonneurs MCMC ou samplers sont disponibles. Dans les priors que l'on étudie, on retrouve notamment le 'RW sampler' qui correspond à un algorithme de Metropolis-Hastings avec une loi de proposition gaussienne adaptative et le 'conjugate sampler' qui utilise un algorithme de Gibbs, il intervient notamment quand une relation de conjugaison est détectée entre les lois a priori et a posteriori d'un paramètre donné. On peut configurer l'échantillonneur MCMC de telle sorte à n'utiliser que des 'RW samplers', mais on peut se rendre compte que lorsqu'on exploite les relations de conjugaisons, l'ESS des chaînes de Markov se voit être amélioré de façon conséquente (au prix d'un temps de calcul légérèment plus élevé). C'est pourquoi l'utilisation de

loi Gamma inverses pour simuler des lois de Cauchy dans les priors Horseshoe et Horseshoe+ donne lieu à une meilleure convergence des chaînes de Markov.

6 Étude par simulations numériques

6.1 Plan expérimental

Dans les différentes simulations qui seront faites, on se place dans un contexte de croissance logistique en choisissant :

$$f(\varphi_i, t_{ij}) = \frac{\psi_1}{1 + \exp\left(-\frac{t_{ij} - \varphi_i}{\psi_2}\right)}$$

On utilise les valeurs suivantes : $n = 50$, $p = 100$, $\sigma^2 = 30$, $\Gamma = 200$, $\psi_1 = 200$, $\psi_2 = 300$, $\mu = 1200$.

Pour tout $i \in [\![1, n]\!]$, $n_i = J = 10$.
Pas de temps : Pour tout $j \in [\![1, J]\!]$, $t_{ij} = t_j = 150 + (j-1)\frac{3000-150}{J-1}$.

Les covariables sont simulées selon une loi gaussienne multivariée : $\forall i \in [\![1, n]\!]$, $V_i \sim \mathcal{N}_p(0, \Sigma)$
On choisit la matrice de covariance $\Sigma = (\rho_\Sigma^{|i-j|})_{i,j}$.

Pour chaque valeur de $\rho_\Sigma \in \{0, 0.3, 0.9\}$, on simule 100 jeux de données, on applique chaque prior sur chacun d'entre eux pour récupérer les valeurs des critères. Ainsi, pour chaque prior, on récupère 100 valeurs différentes pour un critère donné, on rassemble ces valeurs en faisant leur moyenne que l'on représente dans un tableau à double entrée dans la sous-section suivante. On cherche à observer ces différentes situations pour savoir si certains priors sont plus sensibles à la présence de corrélation que d'autres. Les valeurs 0, 0.3 et 0.9 pour ρ_Σ correspondent respectivement aux cas de covariables indépendantes, de covariables moyennement corrélées et de covariables très fortement corrélées. À noter que ce dernier cas n'est pas forcément réaliste, car en pratique, on effectuerait un pré-traitement des covariables pour en supprimer certaines si de telles corrélation étaient observées. Ce pré-tri des covariables pose par contre la difficulté de choisir de bons représentants.

On utilise comme vraie valeur des coefficients : $\beta^{true} = (100, 50, 20, 0, \dots, 0)$

Voici un exemple de données générées dans le cas de covariables indépendantes :

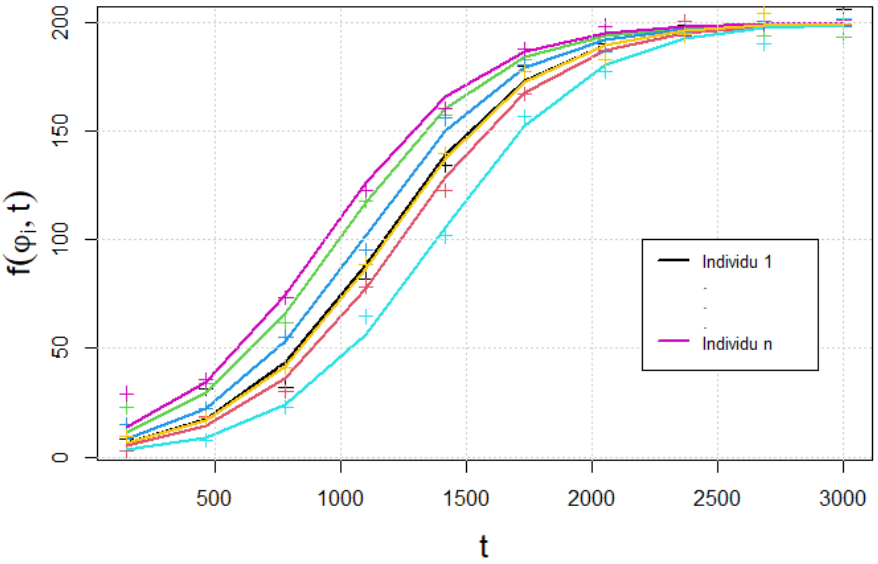

FIGURE 4 – Exemple de données simulées pour y (croix) et de fonctions différentes selon φ_i (traits pleins)

On observe bien sur ce graphe l'évolution de y en fonction du temps (variabilité intra-individuelle), et les différences entre individus avec les courbes de différentes couleurs (variabilité inter-individuelle).

Pour les priors de type Spike and Slab, on choisira $a = 1$ et $b = p$ dans la loi $Beta$ car on sait qu'il y a très peu de covariables significatives. On fixe $\nu_0 = 1$, $\nu_1 = 10000$ (variances des composantes Spike et Slab) et $\nu = 10$ (degré de liberté pour les lois de Student).

Maintenant que les différents paramètres sont fixés, il faut faire la calibration de l'algorithme MCMC. On doit vérifier que les différentes chaînes de Markov convergent correctement. Pour chaque coefficient de β, on lance 2 chaînes initialisées respectivement à 0 et 10 car on s'attend à avoir des valeurs proches de 0 pour un grand nombre de coefficients, mais il faut tout de même vérifier que le fait de prendre une initialisation plus éloignée de 0 n'influe pas trop sur le comportement de la chaîne excepté au début.

Une fois avoir effectué ces calculs, on estime que 15000 itérations sont suffisantes pour voir la convergence des chaînes de Markov, on place également un temps de chauffe de 5000 itérations pour supprimer l'effet de l'initialisation. On note cependant que les chaînes convergent mieux pour certains priors que pour d'autres, pour le Horseshoe+, la convergence est moins bien établie et le critère de Gelman-Rubin n'est pas respecté pour une trentaine de coefficients sur les exemples traités, ce qui est tout de même bien mieux que pour la première version avec lois de Cauchy où la grande majorité des coefficients ne vérifiaient pas le critère. Pour les autres priors, il y a au maximum une dizaine de coefficients qui peuvent poser problème. La convergence est parfois détériorée en présence de corrélation extrême ($\rho_\Sigma = 0.9$), mais pour garder un temps de calcul raisonnable, on conserve 15000 itérations.

6.2 Résultats de comparaison

Après un premier calcul rapide, on observe bien la répartition des coefficients selon deux groupes :

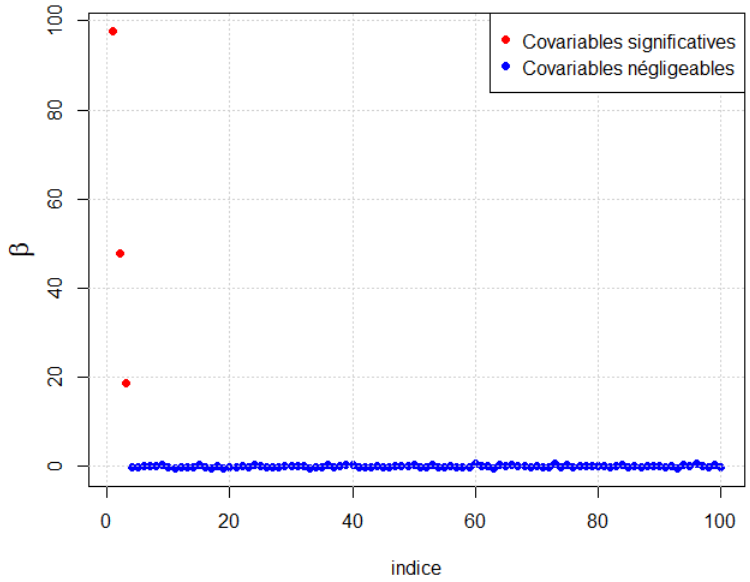

FIGURE 5 – Estimation a posteriori de β par le Spike and Slab continu normal sur 1 dataset, $\rho_\Sigma = 0$

On se propose alors d'effectuer une première comparaison des priors en prenant $seuil = 1$. Cette valeur de seuil est suffisamment faible devant $\beta_3 = 20$ et assez élevée pour ne pas faire de faux positifs dans l'estimation ci-dessus. On fera varier le seuil par la suite.

Model	RMSE	CRPS	Misc. rate	Sensitivity	Specificity	MIS	ESS/s
S&S Continuous Normal	0.8794	1.635	0.005	1	0.995	4.409	109.9
S&S Continuous Student	0.9265	1.720	0.006	1	0.993	4.939	78.8
S&S Dirac Normal	0.7519	0.390	0.003	1	0.997	0.705	101.8
Horseshoe	0.9607	1.085	0.043	1	0.956	6.327	33.1
Horseshoe+	0.9649	0.972	0.046	1	0.953	6.150	25.7
Laplace	2.8764	6.991	0.574	1	0.408	20.335	9.1

FIGURE 6 – Comparaison sans corrélation des différents priors pour 100 datasets, $\rho_\Sigma = 0$

Model	RMSE	CRPS	Misc. rate	Sensitivity	Specificity	MIS	ESS/s
S&S Continuous Normal	0.9639	1.697	0.004	0.997	0.996	4.627	111.9
S&S Continuous Student	1.0059	1.779	0.005	0.993	0.995	5.162	82.1
S&S Dirac Normal	0.8691	0.470	0.003	0.993	0.997	0.915	102.5
Horseshoe	0.9875	1.099	0.042	1.000	0.957	6.425	33.4
Horseshoe+	0.9719	0.969	0.044	1.000	0.955	6.211	26.3
Laplace	2.4806	6.611	0.560	1.000	0.422	20.016	8.9

FIGURE 7 – Comparaison avec corrélation des différents priors pour 100 datasets, $\rho_\Sigma = 0.3$

Model	RMSE	CRPS	Misc. rate	Sensitivity	Specificity	MIS	ESS/s
S&S Continuous Normal	2.9935	3.191	0.008	0.827	0.997	10.888	127.7
S&S Continuous Student	2.7228	3.088	0.009	0.807	0.997	11.169	85.2
S&S Dirac Normal	2.9107	1.971	0.010	0.803	0.996	7.116	108.0
Horseshoe	2.6352	2.218	0.034	0.893	0.968	9.739	28.0
Horseshoe+	2.6235	2.100	0.039	0.900	0.963	9.679	22.9
Laplace	3.0020	8.613	0.634	1.000	0.346	28.775	5.4

FIGURE 8 – Comparaison avec corrélation des différents priors pour 100 datasets, $\rho_\Sigma = 0.9$

Pour le cas des covariables indépendantes, on peut faire plusieurs remarques :

— Les priors Spike and Slab sont ceux qui présentent les meilleurs taux d'erreurs (moins de 10 %), ils sont tous les trois très performants dans l'ensemble des critères, en particulier, le Spike and Slab Dirac Normal apparaît comme le meilleur prior d'entre tous, il est premier dans tous les critères excepté en ESS/s. Sur ces trois priors, la version Student semble être un peu moins bonne.

— Les priors Horseshoe et Horseshoe+ proposent de bonnes estimations des coefficients, notamment en CRPS, mais ils sont légèrement moins bons en termes de sélection. Par contre, leur efficacité est bien deçà des Spike and Slab. On note aussi que la différence entre Horseshoe et Horseshoe+ est très peu visible sur ces résultats.

— Le prior Laplace est nettement moins bon que les autres en termes d'estimation, de sélection et d'efficacité. Il se trompe sur plus de la moitié des variables.

Lorsqu'on effectue le même calcul avec des covariables corrélées, on voit que dans l'ensemble, les valeurs des critères a tendance à se dégrader. En termes de performance de sélection, les conclusions sont les mêmes que dans

le cas précédent. Les Spike et Slab donnent des résultats de sélection très similaires et apparaîssent comme étant les meilleurs priors dans cette catégorie. La version continue a un meilleur ESS/s mais elle estime un peu moins bien les coefficients que le Spike de Dirac.

Avec les résultats de sensibilité et de spécificité, notamment en présence de corrélation, on peut voir que les priors Horseshoe, Horseshoe+ et Laplace ont tendance à effectuer une sous-sélection moins forte que les Spike and Slab, mais ils font une sur-sélection en contrepartie. Étant donné que les covariables négligeables sont en surnombre, ce sont les priors qui donnent le moins de faux positifs qui sont les plus valorisés par le taux de mauvais classement, ici, ce sont les Spike and Slab.

Le défaut de la méthode de sélection choisie est de poser une valeur arbitraire pour $seuil$, c'est pourquoi on veut observer comment évolue le taux d'erreur en fonction de cette valeur pour chacun des priors. On pourra retrouver ces graphes dans l'annexe B1.

On retrouve bien les résultats précédents dans le cas indépendant pour $seuil = 1$: les trois priors Spike and Slab ont un taux d'erreur très proche de 0 et sont difficilement discernables. Seulement, pour des valeurs de seuil plus petites que 1, on voit bien que le prior de Dirac est bien plus performant que le continu Normal en 2ème et le Student en 3ème.

Les deux priors Horseshoe et Horseshoe+ sont presque confondus pour tout seuil, ils apparaissent comme moins bons que les Spike and Slab pour des petites valeurs de seuil comprises entre 0.2 et 2, au-delà de celles-ci, ils sont confondus avec les Spike and Slab continus. Le prior Laplace donne un taux d'erreur tout à fait supérieur aux autres et cela pour toutes les valeurs de seuil observées ici.

Dans le cas des variables moyennement ou fortement corrélées, la configuration est très similaire au cas précédent, les taux d'erreurs sont très légèrement plus élevés ici.

Les différentes courbes ROC et PR se trouvent aussi dans l'annexe B1. Dans le cas des courbes ROC, on observe que pour des covariables indépendantes ou faiblement corrélées, les priors sont tous très proches de la règle parfaite, de plus, si on calcule l'aire sous chacune des courbes (ROC-AUC) on obtient à chaque fois des valeurs supérieures à 0.99. D'un point de vue compromis spécificité-sensibilité, les priors apparaissent tous comme étant très bons, bien meilleurs que la règle aléatoire. Par contre, les courbes sont très difficilement comparables entre elles.

Dans le cas de la très forte corrélation, les différences entre priors sont plus visibles, ils demeurent tous bons avec des valeurs de ROC-AUC ≥ 0.94, mais le prior Laplace apparaît comme étant celui qui réalise le meilleur compromis dans cette situation.

Pour les courbes PR, les remarques précédentes sont également valables dans l'ensemble, si ce n'est que les différences entre priors sont un peu plus marquées. Le prior Laplace apparaît comme étant un peu moins bon que les autres priors dans le cas de covariables aucunement ou faiblement corrélées, mais il domine les autres en situation de corrélation extrême.

En conclusion de cette étude sur simulations numériques, on peut dire que l'ensemble des priors testés donnent de bons résultats en termes de sélection, notamment quand les covariables sont simulées indépendantes. Dans les cas d'absence où de faible corrélation, les courbes ROC et PR ne permettent pas de distinguer un prior au-delà des autres, c'est seulement dans le cas de la très forte corrélation qu'une préférence se dégage en faveur du prior Laplace. Bien que ce prior semble être plus résistant à la corrélation que les autres pour la sélection, il faut tout de même mentionner que ce prior donne lieu à d'assez mauvaises valeurs d'estimation des coefficients, surtout en CRPS et qu'il est très peu efficace avec un ESS/s très mauvais.

De plus, la situation de corrélation extrême n'est pas vraiment réaliste, car on préfère en pratique réduire le plus possible les corrélations dans les données avant de procéder à la sélection de variables avec l'un ou l'autre prior, le prior Laplace n'apparaît donc pas comme étant préférable de manière générale. Ces figures à seuil variable incitent en tout cas à noter que le choix du seuil peut être une question importante et doit être fait soigneusement.

Les priors Horseshoe et Horseshoe+ ont de bons résultats de sélection et d'estimation quelque soit le niveau de corrélation, mais ils demeurent mauvais en termes d'efficacité, leur ESS/s étant plutôt faible. Comme cela a déjà été mentionné, l'amélioration théorique proposée par le Horseshoe+ n'est pas substantielle par rapport aux résultats obtenus, il n'y a donc pas de raison particulière à le préférer au prior Horseshoe classique dans cette situation.

Les priors restants sont ceux de type Spike and Slab, pour les valeurs d'hyperparamètres testées, le prior Student semble donner des résultats d'estimation un peu moins précis que les autres, mais c'est surtout en termes d'ESS/s qu'il apparaît comme moins préférable, en effet, il ne dispose pas des relations de conjugaison pour β qui apparaissent quand on utilise des lois normales. Enfin, le choix final se fait entre les deux versions du Spike and Slab gaussien, la version continue semble être un peu plus efficace que la version Dirac mais elle donne des estimations des coefficients un peu moins bonnes, que ce soit une estimation ponctuelle ou une estimation de la distribution. Par conséquent, on utilisera la version que l'on préfère en fonction de l'objectif imposé à savoir, soit avoir les meilleures estimations, soit avoir la plus grande efficacité.

6.3 Variations des hyperparamètres

On a imposé dans la section précédente des valeurs arbitraires pour les Spike and Slab continus normal et Student, à savoir $\nu_0 = 1$ (variance de la loi Spike) et $\nu = 10$ (degré de liberté), on s'intéresse dans cette partie aux différences de performances observées lorsqu'on fait varier ces quantités. Les courbes des résultats sont dans l'annexe B2.

Pour ce qui est de faire varier ν_0 dans le Spike and Slab gaussien, on observe que, pour des covariables indépendantes, les valeurs entre 0.1 et 1.5 sont les meilleures et donnent des résultats similaires à partir des courbes de taux d'erreurs, ROC et PR. Dans le cas de covariables fortement corrélées ($\rho_\Sigma = 0.9$), toutes les courbes ROC sont confondues, mais les graphes de taux d'erreurs et les courbes PR mettent aussi en valeur l'intervalle $[0.1, 1.5]$ sans pour autant dégager une préférence particulière pour l'une des valeurs. Le choix de $\nu_0 = 1$ dans la partie précédente est donc pertinent par rapport à ces résultats, et on ne note pas d'amélioration significative lorsque ν_0 est légèrement modifié à partir de cette valeur.

Par rapport au Spike and Slab continu Student, on fixe $\nu_0 = 1$, et on fait varier ν. Pour les covariables indépendantes, on voit qu'il y a un nombre plus élevé d'erreurs pour de faibles valeurs de seuil quand $\nu = 1$ (qui correspond à une loi de Cauchy sur $\mathbb{R}$), mais au niveau des courbes ROC et PR, aucune différence claire n'apparaît. Le prior est quasiment assimilable à la règle parfaite pour toutes les valeurs de ν testées.

En présence de fortes corrélations, bien qu'elle donne toujours plus d'erreurs, la version $\nu = 1$ semble être meilleure que les autres à en croire les courbes ROC et PR. En général, comme on a tendance à vouloir réduire les corrélations dans le jeu de données avant de procéder à l'inférence, cette situation ne devrait pas se présenter. En tout cas, les résultats sont assez similaires dès que $\nu \geq 5$, de plus, pour $\nu = 100$, on peut quasiment assimiler la loi de Student à une gaussienne, on reviendrait alors dans le cas du Spike and Slab continu normal qui est un prior plus efficace grâce à la conjugaison.

En conclusion de cette partie, on peut recommander pour le Spike and Slab normal une valeur $\nu_0 \in [0.1, 1.5]$, tandis que pour le Spike and Slab Student, le paramètre ν semble avoir peu d'influence dans l'ensemble pour $\nu_0 = 1$ fixé. Pour faire une étude plus exhaustive, il faudrait refaire les mêmes calculs pour une gamme de valeurs de ν_0, mais cela prendrait un certain temps, sachant que le Spike and Slab normal est de toute façon plus efficace.

7 Étude sur des données réelles

On reprend ici les données étudiées par M. Naveau [1], il est question ici de $n = 220$ variétés de blé chacune observées $J = 18$ fois au cours du temps. On observe ici la sénescence de ces variétés à partir de la proportion de feuilles desséchées, elle peut être causée par exemple si la plante manque d'azote, ce qui affecte sa productivité. La sénescence est donc exprimée par un pourcentage allant de 0 à 100 en suivant une croissance logistique. La sélection de variables se fait donc sur les marqueurs moléculaires, 34000 ont été mesurés sur le génome au total, mais on se restreindra ici seulement à un seul chromosome appelé 6A, il y a donc 1124 covariables à traiter.

Les données utilisées ici sont déjà pré-traitées, la corrélation est donc réduite par rapport aux données initiales, de plus, 5 variables seront d'office considérées comme significatives, elles correspondent aux 5 premières composantes principales dans une ACP réalisée lors du pré-traitement. Ainsi, $p = 1119$.

Le modèle probabiliste étudié ici est donc :

Pour $1 \leq i \leq n$ et $1 \leq j \leq J$:

$$\begin{cases} y_{ij} = \dfrac{100}{1 + \exp\left(-\dfrac{t_{ij} - \varphi_i}{\psi}\right)} + \varepsilon_{ij}, \quad \varepsilon_{ij} \sim \mathcal{N}(0, \sigma^2) \\ \varphi_i = \lambda^\top v_{1:6} + \beta^\top V_i + \xi_i, \quad \xi_i \sim \mathcal{N}(0, \Gamma) \end{cases}$$

On veut estimer $\lambda \in \mathbb{R}^6$ et $\beta \in \mathbb{R}^p$. Les valeurs utilisées pour les autres paramètres sont celles qui ont été estimées par M. Naveau : $\psi = 2.8$, $\sigma^2 = 15.2$, $\Gamma = 1.1$.

D'après les premières estimations de M. Naveau [1], il semblerait que la plupart des coefficients de β soient assez proches de zéro, c'est pourquoi on choisit $\nu_0 = 10^{-4}$ et $\nu_1 = 1$ comme paramètres Spike and Slab. De plus, les deux chaînes de Markov sont initialisées respectivement à 0 et 0.5 pour chaque coefficient. On cherche à appliquer l'algorithme utilisé précédemment pour les priors Spike and Slab normal continu et Dirac. On associe une distribution Slab aux covariables dont on sait qu'elles seront sélectionnées.

Les biologistes ont identifié 3 marqueurs responsables de la floraison sur le chromosome 6A, la méthode de M. Naveau a sélectionné un seul marqueur nommé 'cfn2905337' qui se situe sur le génome à moins d'une mégabase de l'un des 3 marqueurs de floraison. On considère alors que ce marqueur sélectionné est assimilable à l'un d'entre eux.

Malheureusement, avec ces données-là, on se rend compte après un temps de calcul très long, (plus de 10 heures pour le prior Dirac) que les chaînes de Markov ne convergent pas du tout dans cette configuration, les résultats obtenus ne sont donc pas exploitables car on ne simule pas les lois a posteriori attendues. Étant donné que le temps de calcul est déjà très long, il ne paraît pas raisonnable de prendre un nombre d'itérations plus important. Bien que l'algorithme utilisé ait la qualité d'être très flexible pour être appliqué à différents priors, il semble inadapté pour les ordres de grandeur proposés pour n et p avec ce cas de figure.

On cherche désormais à effectuer la sélection sur un sous-échantillon de $p = 300$ covariables. On place dans ce jeu de covariables celle qui a été sélectionnée par M. Naveau. Dans ce cas, la convergence des chaînes de Markov est meilleure, mais elle reste discutable pour un certain nombre de coefficients. En choisissant un seuil de 0.01, le prior continu a sélectionné 17 covariables et la version Dirac en a retenu 16. Treize des covariables choisies sont communes aux deux priors. La variante continue a su retrouver le marqueur 'cfn2905337' mais pas le prior Dirac. Si on compare les positions des marqueurs sélectionnés sur le génome à celles des marqueurs de floraison, on se rend compte que seul 'cfn2905377' se situe à moins d'une mégabase de l'un d'entre eux. Ainsi, les deux priors ont sélectionné des covariables supplémentaires par rapport à la méthode de M. Naveau, mais elles ne correspondent pas aux marqueurs relevés par les biologistes.

Pour un jeu de données tel que celui-ci, il vaut mieux utiliser des algorithmes spécifiques et pensés pour chaque prior comme cela a été développé par M. Naveau dans son article [1] pour le Spike and Slab normal continu.

8 Conclusion et remerciements

Pour conclure le travail réalisé lors de ce stage, on peut revenir sur les différents résultats obtenus. Les priors de type Spike and Slab gaussiens apparaissent comme étant les meilleurs bien que tous les priors testés donnent de bons résultats de sélection. Si on cherche à obtenir une meilleure estimation des coefficients de β, on choisira la version Dirac, si on met plus d'importance sur l'efficacité de l'algorithme, la version continue sera préférable.

L'algorithme n'est pas adapté aux jeux de données trop fournis, comme cela a été constaté dans la section 7, son intérêt est donc limité car il n'est pas optimisé pour accueillir un trop grand nombre de covariables. Une

perspective intéressante serait de proposer un algorithme spécifique utilisant le Spike and Slab Dirac normal pour traiter ce type de situations, comme cela a déjà été fait pour la version continue [1]. Cet algorithme devrait alors se dispenser de générer des chaînes de Markov pour chaque coefficient car cela est très coûteux en temps de calcul. La convergence n'est pas toujours assurée, c'est l'écueil qui a été rencontré ici. Cependant, on retiendra que pour des jeux de données de taille plus raisonnable comme ceux qui ont été générés en section 6, les résultats de sélection sont tout à fait corrects même en présence de corrélation entre les covariables.

Je tiens à remercier chaleureusement Guillaume Kon Kam King et Maud Delattre de m'avoir accompagné et guidé tout au long de ce stage qui m'a permis de prendre du recul sur la statistique bayésienne, sur les différentes notions théoriques tout comme pratiques qui ont pu intervenir au cours de ce travail, mais aussi pour m'avoir offert cette belle occasion de travailler de façon concrète dans le domaine de la recherche en mathématiques et statistiques. Ils ont été à l'écoute de mes questions et m'ont aidé à comprendre certains résultats. Je remercie Marion Naveau pour ses explications claires quant à son travail qui a servi de base à ce sujet.

Je veux également remercier l'ensemble de l'équipe MaIAGE, chercheurs, doctorants et stagiaires pour leurs conseils et le cadre de travail très agréable lors de ce stage.

9 Annexes

9.1 Annexe A : Démonstration de propriété

Propriété :

$$\begin{cases} U \,|\, \omega \sim InvGamma(1/2, 1/\omega) \\ \omega \sim InvGamma(1/2, 1/\gamma^2) \end{cases} \implies \sqrt{U} \sim C^+(0, \gamma)$$

Démonstration :
On rappelle la densité d'une loi $InvGamma(a, b)$ sur $\mathbb{R}^+$ où $a > 0$ et $b > 0$:

$$x \longrightarrow \frac{b^a}{\Gamma(a)} x^{-a-1} e^{-b/x}$$

Ainsi, la distribution de la loi jointe est :

$$\begin{aligned} p(U, \omega) = p(U \,|\, \omega) p(\omega) &= \frac{1}{\pi\sqrt{\omega}} u^{-3/2} e^{-1/(\omega u)} \frac{1}{\gamma} \omega^{-3/2} e^{-1/(\gamma^2 \omega)} \\ &= \frac{1}{\pi\gamma} u^{-3/2} \omega^{-2} \exp\left(-\frac{1}{\omega}(u^{-1} + \gamma^{-2}) \right) \end{aligned}$$

On en déduit la loi marginale de U :

$$p(U) = \int_0^{+\infty} p(U, \omega) d\omega = \frac{1}{\pi\gamma} u^{-3/2} \int_0^{+\infty} \omega^{-2} \exp\left(-\frac{1}{\omega}(u^{-1} + \gamma^{-2}) \right) d\omega$$

On reconnaît dans l'intégrale la densité (à une constante près) d'une loi $InvGamma(1, u^{-1} + \gamma^{-2})$, ainsi :

$$p(U) = \frac{1}{\pi\gamma} u^{-3/2} \frac{1}{u^{-1} + \gamma^{-2}} = \frac{1}{\pi\gamma} u^{-1/2} \left(1 + \frac{u}{\gamma^2} \right)^{-1}$$

Déterminons la loi de $Y = \sqrt{U}$ par la méthode de la variable muette :
Soit ψ une fonction réelle définie sur $\mathbb{R}^+$ continue et bornée, on a que :

$$\mathbb{E}(\psi(Y)) = \mathbb{E}(\psi(\sqrt{U})) = \int_0^{+\infty} \psi(\sqrt{u}) \frac{1}{\pi\gamma} u^{-1/2} \left(1 + \frac{u}{\gamma^2} \right)^{-1} du = \int_0^{+\infty} \psi(y) \frac{2}{\pi\gamma} \left(1 + \frac{y^2}{\gamma^2} \right)^{-1} dy$$

Après le changement de variable, on retrouve exactement la densité attendue, d'où le résultat.

□

9.2 Annexe B : Liste des figures

9.2.1 Annexe B1 : Résultats de comparaison

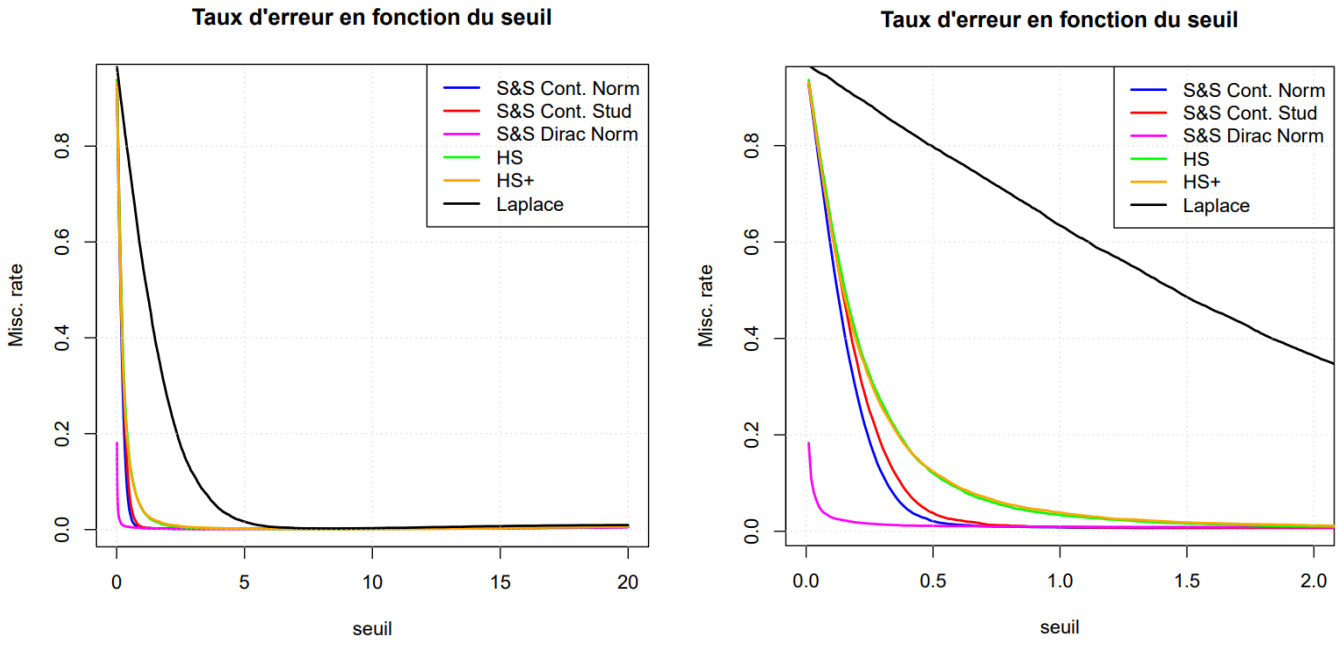

FIGURE 9 – Évolution du taux d'erreur en fonction du seuil sur 100 datasets en absence de corrélation, $\rho_\Sigma = 0$ (figure zoomée à droite)

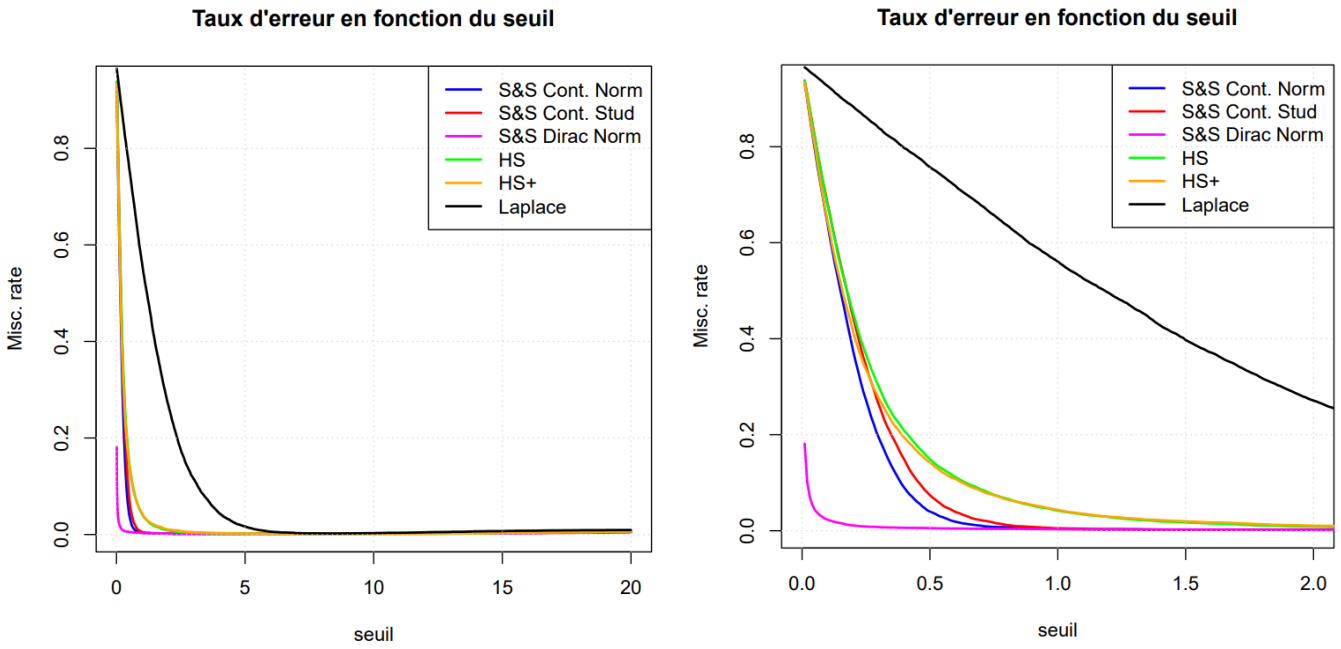

FIGURE 10 – Évolution du taux d'erreur en fonction du seuil sur 100 datasets avec corrélation, $\rho_\Sigma = 0.3$ (figure zoomée à droite)

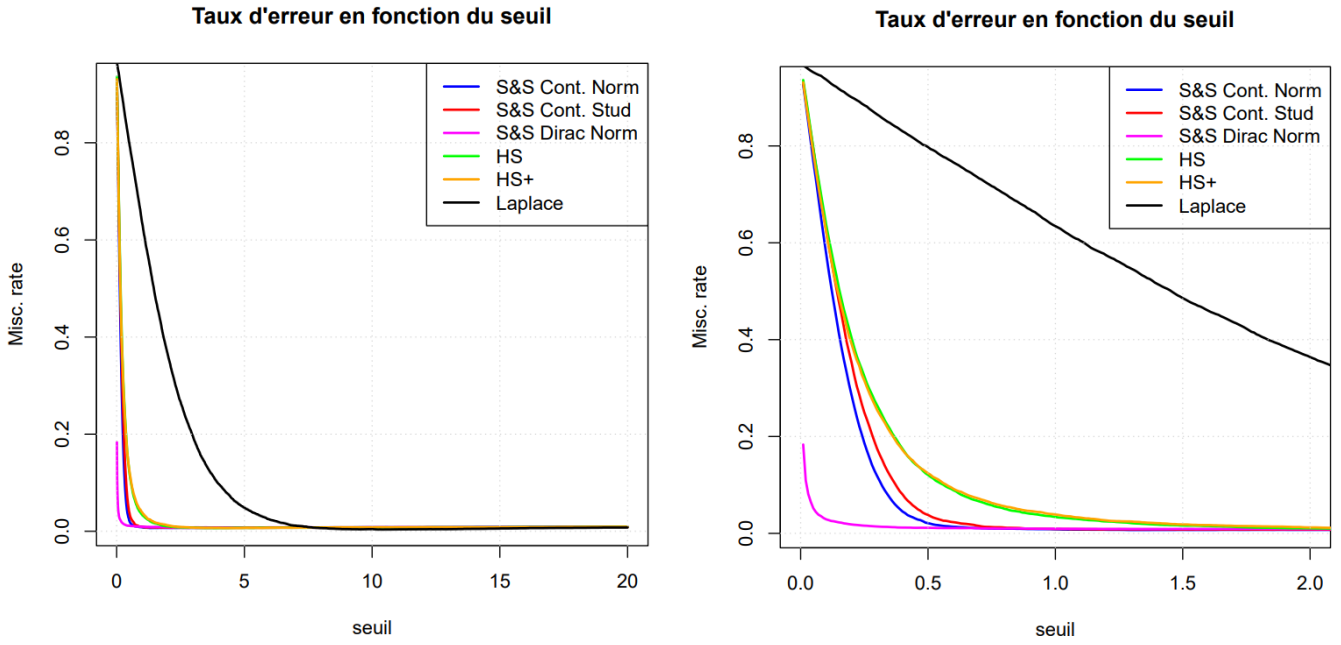

FIGURE 11 – Évolution du taux d'erreur en fonction du seuil sur 100 datasets avec corrélation, $\rho_\Sigma = 0.9$ (figure zoomée à droite)

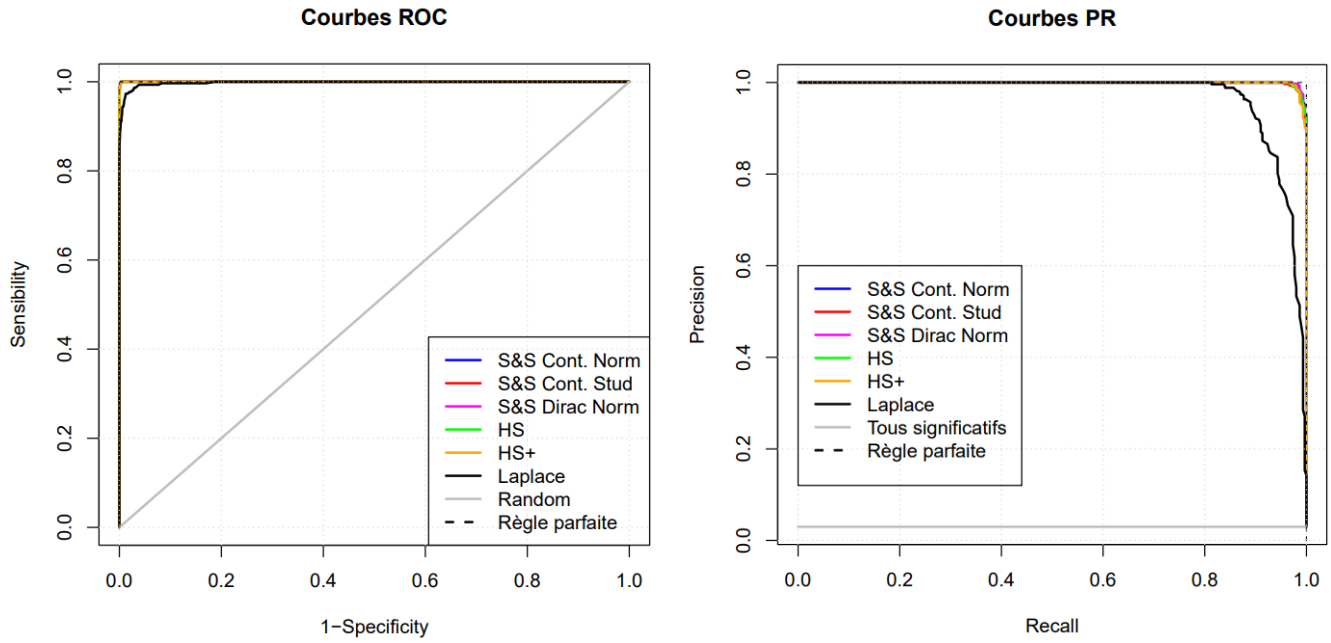

FIGURE 12 – Courbes ROC et PR calculées sur 100 datasets en absence de corrélation ($\rho_\Sigma = 0$)

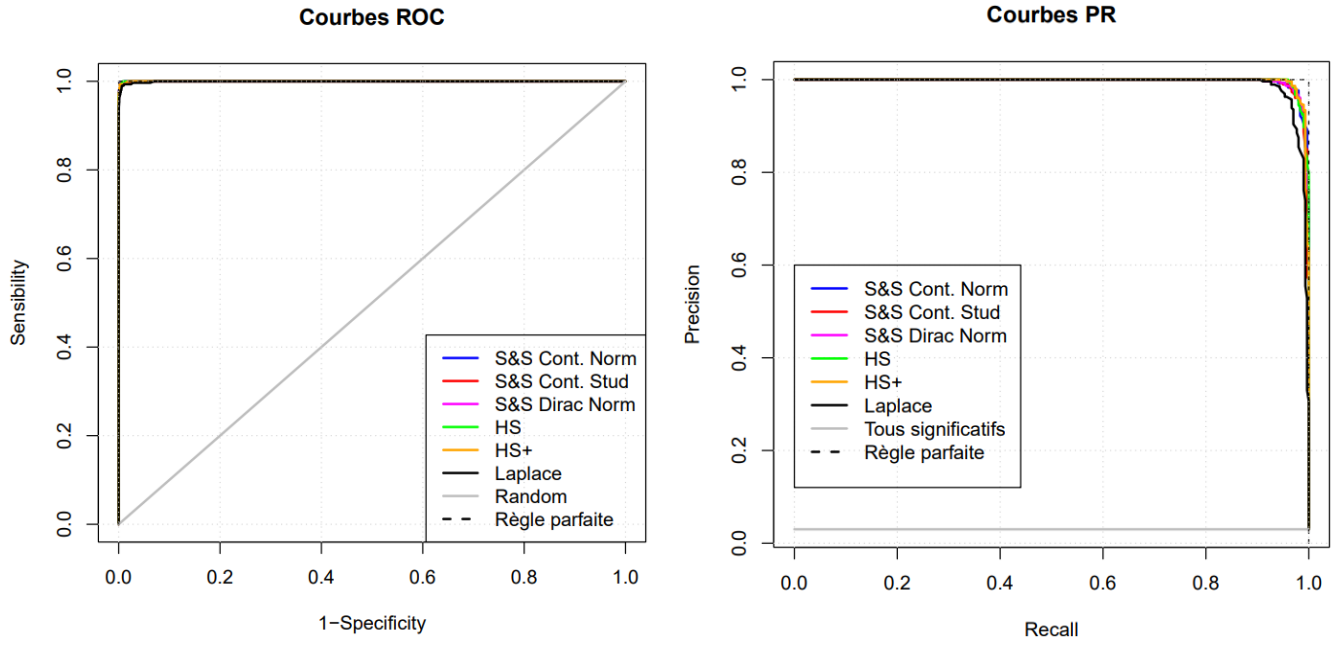

FIGURE 13 – Courbes ROC et PR calculées sur 100 datasets avec corrélations ($\rho_\Sigma = 0.3$)

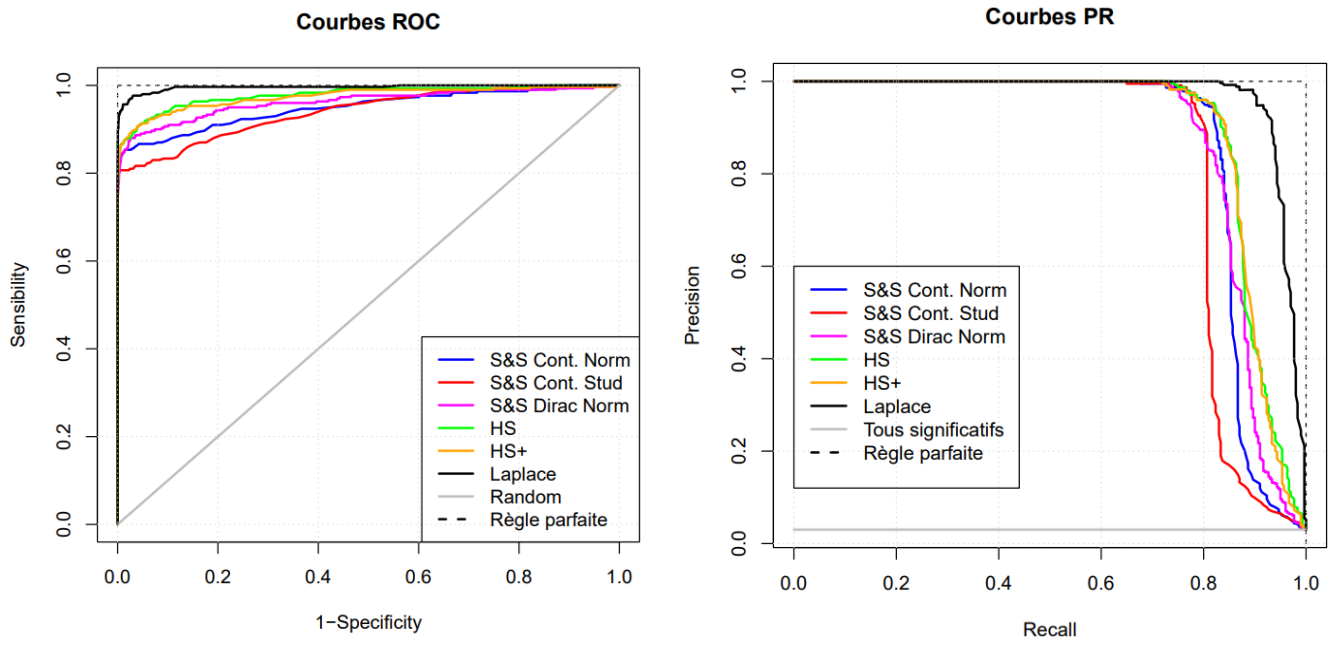

FIGURE 14 – Courbes ROC et PR calculées sur 100 datasets avec corrélations ($\rho_\Sigma = 0.9$)

9.2.2 Annexe B2 : Variations des hyperparamètres

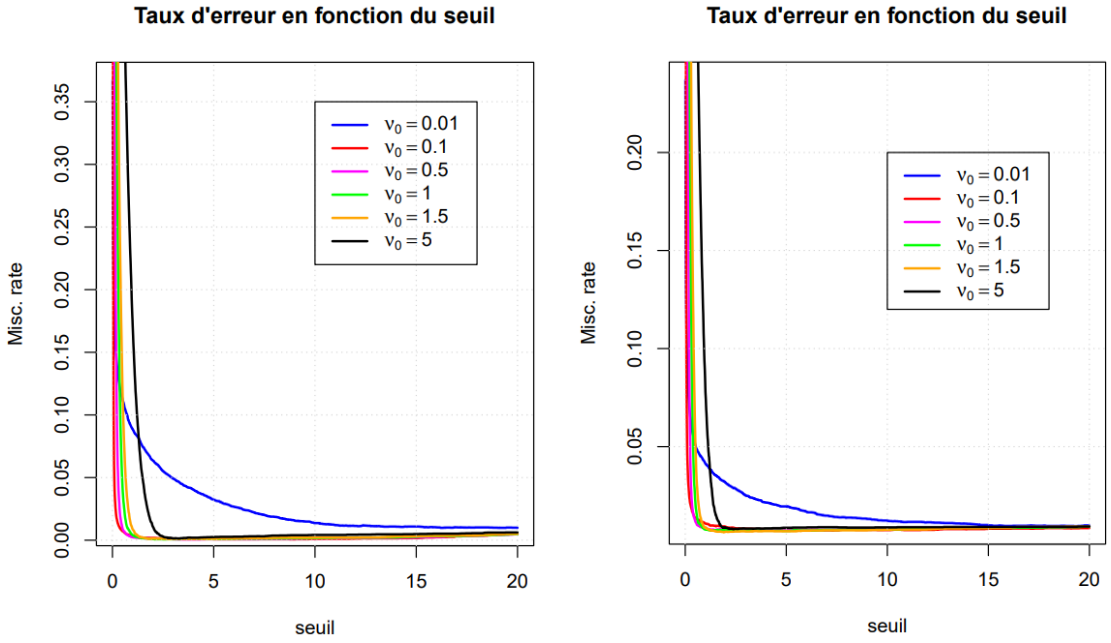

FIGURE 15 – Évolution du taux d'erreur en fonction du seuil sur 100 datasets pour le Spike and Slab continu normal (covariables indépendantes à gauche, corrélées à droite)

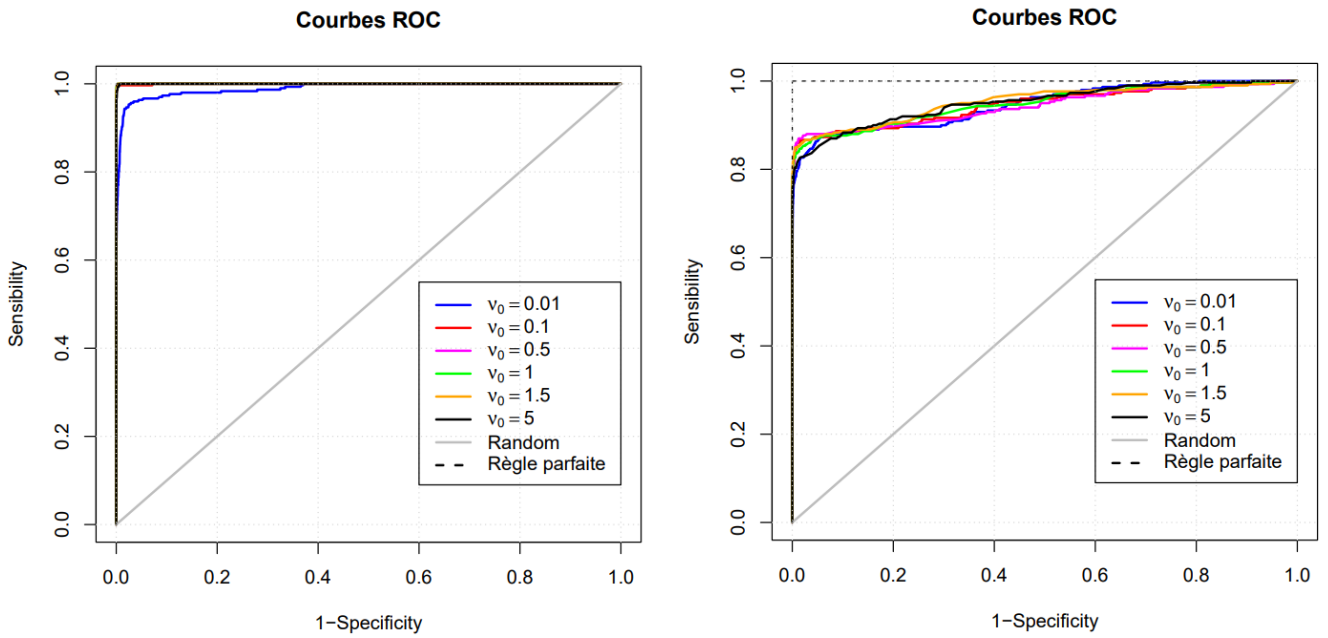

FIGURE 16 – Courbes ROC pour 100 datasets pour le Spike and Slab continu normal (covariables indépendantes à gauche, corrélées à droite)

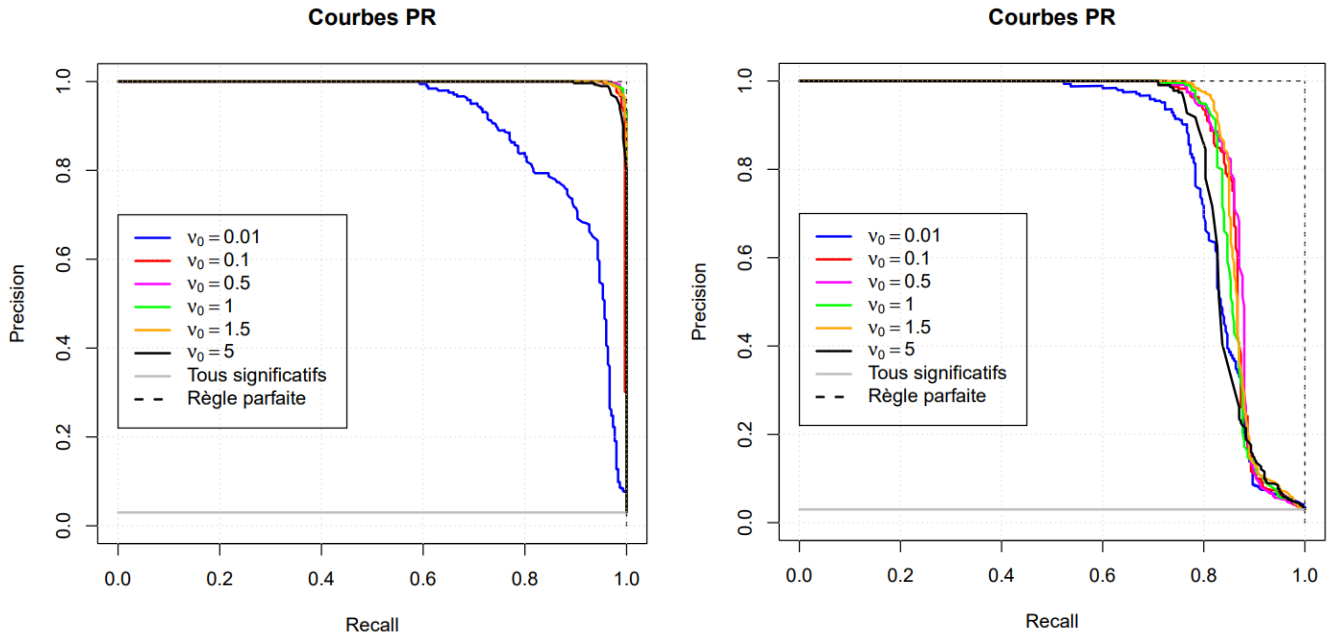

FIGURE 17 – Courbes PR pour 100 datasets pour le Spike and Slab continu normal (covariables indépendantes à gauche, corrélées à droite)

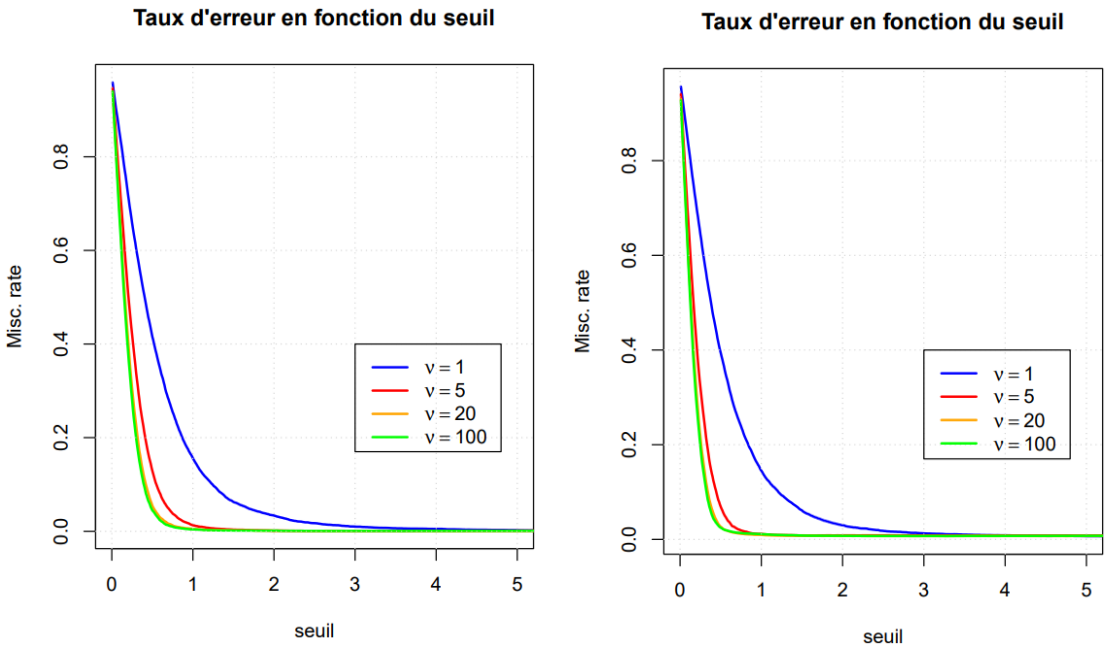

FIGURE 18 – Évolution du taux d'erreur en fonction du seuil sur 100 datasets pour le Spike and Slab continu Student (covariables indépendantes à gauche, corrélées à droite)

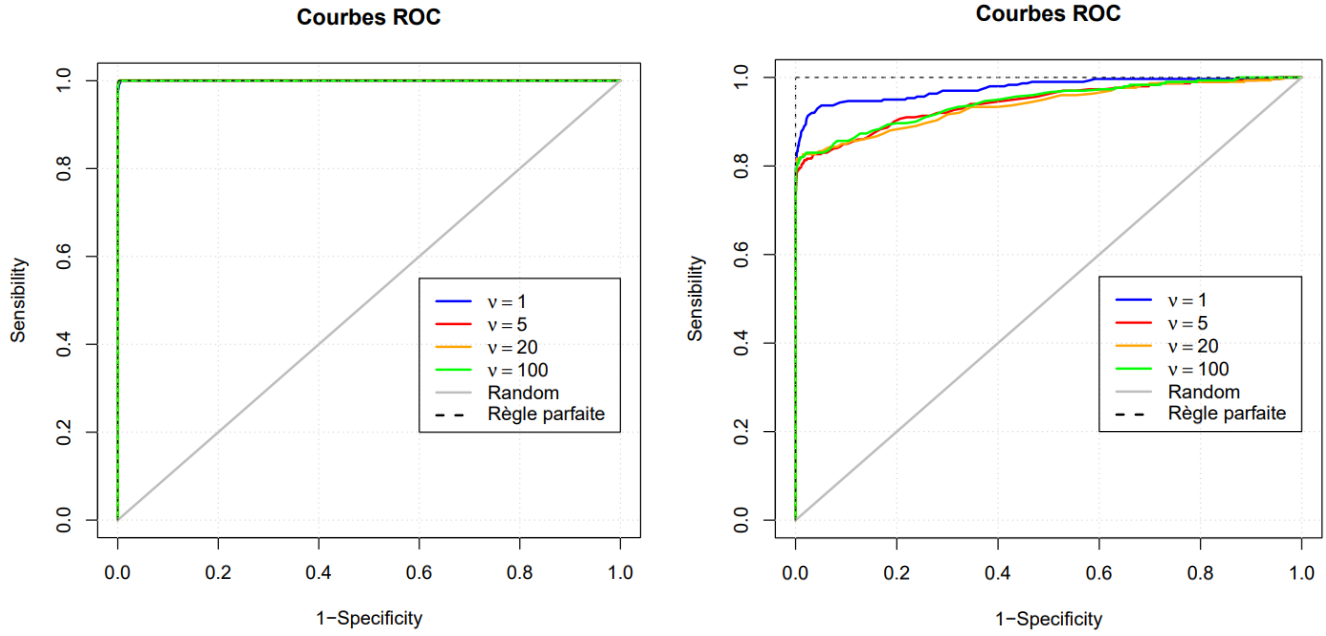

FIGURE 19 – Courbes ROC pour 100 datasets pour le Spike and Slab continu Student (covariables indépendantes à gauche, corrélées à droite)

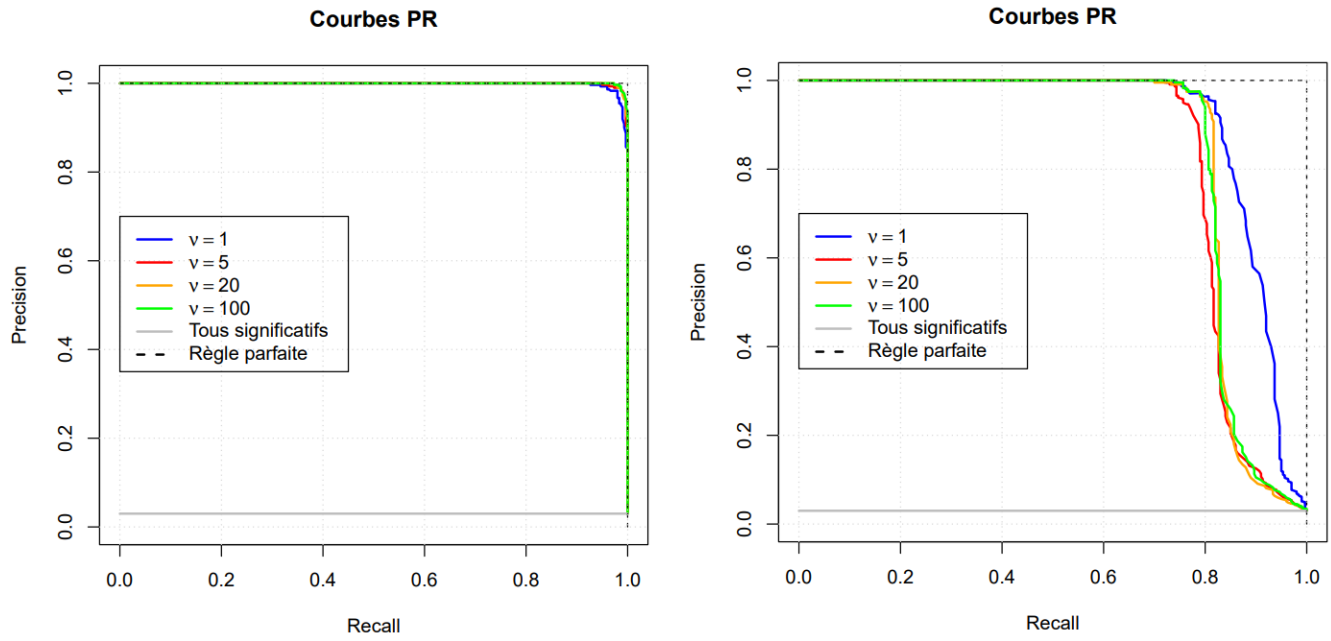

FIGURE 20 – Courbes PR pour 100 datasets pour le Spike and Slab continu Student (covariables indépendantes à gauche, corrélées à droite)

10 Références

[1] Marion Naveau, Guillaume Kon Kam King, Renaud Rincent, Laure Sansonnet, and Maud Delattre. Bayesian high-dimensional covariate selection in non-linear mixed-effects models using the saem algorithm. *arXiv preprint arXiv :2206.01012*, 2022.

[2] P. de Valpine, D. Turek, C.J. Paciorek, C. Anderson-Bergman, D. Temple Lang, and R. Bodik. Programming with models : writing statistical algorithms for general model structures with NIMBLE. *Journal of Computational and Graphical Statistics*, 26 :403–417, 2017.

[3] Anupreet Porwal and Adrian E Raftery. Comparing methods for statistical inference with model uncertainty. *Proceedings of the National Academy of Sciences*, 119(16) :e2120737119, 2022.

[4] Gertraud Malsiner-Walli and Helga Wagner. Comparing spike and slab priors for bayesian variable selection. *arXiv preprint arXiv :1812.07259*, 2018.

[5] Robert B O'hara and Mikko J Sillanpää. A review of bayesian variable selection methods : what, how and which. 2009.

[6] Carlos M Carvalho, Nicholas G Polson, and James G Scott. Handling sparsity via the horseshoe. In *Artificial intelligence and statistics*, pages 73–80. PMLR, 2009.

[7] Anindya Bhadra, Jyotishka Datta, Nicholas G Polson, and Brandon Willard. The horseshoe+ estimator of ultra-sparse signals. 2017.

[8] Juho Piironen and Aki Vehtari. Sparsity information and regularization in the horseshoe and other shrinkage priors. 2017.

[9] Yuchen Han. *Bayesian variable selection using Lasso*. PhD thesis, Case Western Reserve University, 2017.

[10] Quan Zhang. Predictor selection algorithm for bayesian lasso. 2014.

[11] P. de Valpine, C. Paciorek, D. Turek, N. Michaud, C. Anderson-Bergman, F. Obermeyer, C. Wehrhahn Cortes, A. Rodríguez, D. Temple Lang, W. Zhang, S. Paganin, and J. Hug. *NIMBLE User Manual*, 2023.

[12] Jean-Michel Marin, Christian P Robert, et al. *Bayesian core : a practical approach to computational Bayesian statistics*, volume 268. Springer, 2007.

[13] Yuzo Maruyama and Edward I George. Fully bayes factors with a generalized g-prior. 2011.

[14] Gareth O Roberts and Jeffrey S Rosenthal. Examples of adaptive mcmc. *Journal of computational and graphical statistics*, 18(2) :349–367, 2009.