Inférence de réseaux de régulation orientés pour les facteurs de transcription d'Arabidopsis thaliana et création de groupes de co-régulation

Yann Vasseur

NNT : 2017SACLS475

THÈSE DE DOCTORAT

de

L'UNIVERSITÉ PARIS-SACLAY

École doctorale de mathématiques Hadamard (EDMH, ED 574)

Établissement d'inscription : Université Paris Sud

Laboratoires d'accueil : Laboratoire de mathématiques d'Orsay, UMR 8628 CNRS
MIA-Paris UMR 518 AgroParisTech / INRA

Spécialité de doctorat : Mathématiques appliquées

Yann VASSEUR

Inférence de réseaux de régulation orientés pour les facteurs de transcription d'*Arabidopsis thaliana* et création de groupes de co-régulation.

Date de soutenance : 08 Décembre 2017

Après avis des rapporteurs : CHRISTOPHE BIERNACKI (Université de Lille 1)
FRANCK PICARD (Université Claude-Bernard-Lyon-1)

Jury de soutenance :

CHRISTOPHE BIERNACKI	Université de Lille 1	Rapporteur
GILLES CELEUX	INRIA Saclay	Directeur de thèse (excusé)
SIMON DE GIVRY	INRA Centre de Toulouse	Examinateur
MARIE-L. MARTIN-MAGNIETTE	Institute of Plant Sciences - Saclay	Co-directrice de thèse
PASCAL MASSART	Université Paris-Sud	Président du jury
FRANCK PICARD	Université Claude-Bernard-Lyon-1	Rapporteur

Thèse préparée au
Laboratoire de Mathématiques d'Orsay
Bât. 425 Université Paris Sud
91405 Orsay CEDEX

université PARIS-SACLAY | École doctorale de mathématiques Hadamard (EDMH)

Titre : Inférence de réseaux de régulation orientés pour les facteurs de transcription d'*Arabidopsis thaliana* et création de groupes de co-régulation.

Mots Clefs : Grande dimension - Réseaux de gènes - Sélection de modèles - Régression pénalisée - Classification de graphes orientés - Indices de comparaison de couples de partitions.

Résumé :
Dans cette thèse, nous cherchons à caractériser les facteurs de transcription de la plante *Arabidopsis thaliana*, gènes importants pour la régulation de l'expression du génome. À l'aide de données d'expression, notre objectif biologique est de classer ces facteurs de transcription en groupes de gènes co-régulateurs et en groupes de gènes co-régulés. Nous procédons en deux phases pour y parvenir. La première phase consiste à construire un réseau de régulation entre les facteurs de transcription. La seconde phase consiste en la classification des facteurs de transcription selon les liens de régulation établis par ce réseau.

D'un point de vue statistique, les facteurs de transcription sont les variables et les données d'expression sont les observations. Nous représentons le réseau à inférer par un graphe orienté dont les noeuds sont les variables. L'estimation de ses arêtes est vue comme un problème de sélection de variables en grande dimension avec un faible nombre d'unités statistiques. Nous traitons ce problème à l'aide de régressions linéaires pénalisées de type LASSO. Une approche préliminaire qui consiste à sélectionner un ensemble de variables du chemin de régularisation par le biais de critères de vraisemblance pénalisée s'avère être instable et fournit trop de variables explicatives. Pour contrecarrer cela, nous proposons et mettons en compétition deux procédures de sélection, adaptées au problème de la haute dimension et mêlant régression linéaire pénalisée et rééchantillonnage. L'estimation des différents paramètres de ces procédures a été effectuée dans le but d'obtenir des ensembles de variables stables. Nous évaluons la stabilité des résultats à l'aide de jeux de données simulés selon notre modèle graphique.

Nous faisons appel ensuite à une méthode de classification non supervisée sur chacun des graphes orientés obtenus pour former des groupes de nœuds vus comme contrôleurs et des groupes de nœuds vus comme contrôlés. Pour évaluer la proximité entre les classifications doubles des nœuds obtenus sur différents graphes, nous avons développé un indice de comparaison de couples de partition dont nous éprouvons et promouvons la pertinence. D'un point de vue pratique, nous proposons une méthode de simulation en cascade, exigée par la complexité de notre modèle et inspirée du bootstrap paramétrique, pour simuler des jeux de données en accord avec notre modèle. Nous avons validé notre modèle en évaluant la proximité des classifications obtenues par application de la procédure statistique sur les données réelles et sur ces données simulées.

Title : Inference of directed regulatory networks on the transcription factors of *Arabidopsis thaliana* and setting up of co-regulation groups.

Keys words : High dimension - Gene network - Model selection - Penalized regression - Directed graphs clustering - Comparison index for pairs of partitions.

Abstract :
This thesis deals with the characterisation of key genes in gene expression regulation, called transcription factors, in the plant *Arabidopsis thaliana*. Using expression data, our biological goal is to cluster transcription factors in groups of co-regulator transcription factors, and in groups of co-regulated transcription factors. To do so, we propose a two-step procedure. First, we infer the network of regulation between transcription factors. Second, we cluster transcription factors based on their connexion patterns to other transcriptions factors.

From a statistical point of view, the transcription factors are the variables and the samples are the observations. The regulatory network between the transcription factors is modelled using a directed graph, where variables are nodes. The estimation of the nodes can be interpreted as a problem of variables selection. To infer the network, we perform LASSO type penalised linear regression. A preliminary approach selects a set of variable along the regularisation path using penalised likelihood criterion. However, this approach is unstable and leads to select too many variables. To overcome this difficulty, we propose to put in competition two selection procedures, designed to deal with high dimension data and mixing linear penalised regression and subsampling. Parameters estimation of the two procedures are designed to lead to select stable set of variables. Stability of results is evaluated on simulated data under a graphical model.

Subsequently, we use an unsupervised clustering method on each inferred oriented graph to detect groups of co-regulators and groups of co-regulated. To evaluate the proximity between the two classifications, we have developed an index of comparaison of pairs of partitions whose relevance is tested and promoted. From a practical point of view, we propose a cascade simulation method required to respect the model complexity and inspired from parametric bootstrap, to simulate data under our model. We have validated our model by inspecting the proximity between the two classifications on simulated and real data.

Remerciements

Mes premiers mots vont tout naturellement à Gilles Celeux et Marie-Laure Martin-Magniette. Gilles, je te remercie beaucoup d'avoir accepté de diriger ma thèse. Hormis les fructueuses discussions scientifiques que nous avons eues et la construction progressive et raisonnée du projet, j'ai beaucoup apprécié la liberté que tu m'as laissée durant ces années, ta bienveillance paternelle et ton humanité. Un grand merci pour avoir cru en moi. Je te souhaite beaucoup de bonnes choses pour la suite. Marie-Laure, je te remercie grandement pour avoir initié ce projet de thèse. J'ai été très chanceux de faire partie d'une aventure où le sujet était parfaitement cadré et tu y es pour beaucoup. Merci aussi pour tes conseils, pour les connaissances biologiques et statistiques que tu m'as apportées, ainsi que pour ton implication et ton soutien durant ces années. Je te souhaite bonne chance pour tes futurs encadrements et ne doute pas de la réussite des projets que tu entreprendras.

Je tiens aussi à remercier chaleureusement Guillem Rigaill pour sa collaboration. Un grand merci Guillem pour les nombreuses pistes que tu as sues proposer au cours de mes travaux, pour tes intuitions et pour ta gentillesse.

Je remercie aussi grandement Valérie Robert pour la co-écriture de l'article que nous avons soumis : merci Valou pour ton aide et pour ta fructueuse collaboration. Par la même occasion, je remercie Gilles et Christine Keribin pour avoir initié ce travail avec Valérie.

Je suis honoré que Franck Picard et Christophe Biernacki aient accepté de rapporter ma thèse. Je vous remercie beaucoup pour l'intérêt et l'oeil critique très enrichissant que vous avez portés sur mes travaux. Je remercie également Simon de Givry pour avoir volontiers consacré de son temps pour examiner ma thèse ainsi que Pascal Massart pour avoir assumé la présidence de mon jury.

Je tiens également à exprimer ma gratitude envers l'École Doctorale de Mathématiques de la région Paris-Sud et le laboratoire de Mathématiques d'Orsay pour m'avoir offert la possibilité de réaliser ce doctorat et pour m'avoir permis de découvrir l'enseignement par le biais du monitorat à la Faculté Jean Monnet puis du poste d'A.T.E.R à l'IUT de Sceaux. Je tiens aussi à souligner l'efficacité et la sympathie de Valérie Blandin-Lavigne, Florence Rey, Christelle Pires, Katia Evrat, Olga Mwana Mobulakani, Nathalie Carrière, Christine Bailleul, Catherine Ardin et Marie-Christine Myoupo dans la parfaite gestion du côté administratif. Merci également à Olivier Chaudet, Sandrine Lecouteux, Adrien Ramparison, Jérémy Gosse, Mathilde Rousseau et Sylvain Faure d'avoir su garder votre calme pour résoudre les différentes équations informatiques que je vous ai vilement posées. Vous êtes dotés d'une patience, d'une écoute et d'une gentillesse qui forcent mon admiration.

J'en viens maintenant aux personnes que j'ai eu le privilège de cotoyer lors de mes différentes expériences dans l'enseignement. Il me semblait indispensable de remercier Luc Joseph, au côté de qui j'ai fait mes premières gammes. Ensuite, que dire de la formidable année d'A.T.E.R que j'ai passée au sein du département TC2 à l'IUT de Sceaux ? Stéphane, Élise, Marielle, Nadège, Alberto, Carole, Jean-Marc et tous les autres, un grand merci pour l'ambiance formidable qui régnait au sein du département. Merci aussi à Patrick Pamphile pour avoir pris un peu de son temps pour me divulguer de précieux conseils pédagogiques. Je remercie aussi Malek, Ridha, Alexandre, Corentin, Caroline, Abdel et Éliane qui ont grandement facilité mon intégration au Lycée du Parc de Vilgenis de Massy où j'exerce aujourd'hui.

J'en viens désormais aux doctorants d'Orsay. J'ai une pensée toute particulière pour Magda, mais aussi pour Ignacio, Sébastien, Jana, Julien, Jeanne, Céline, Claire, Jade, Rémi, Clément et Eddie. Un énorme merci pour avoir ensoleillé toutes ces années passées à vos côtés. Et bien sûr comment ne pas mentionner l'incontournable Team composée de Mélina, Florence, Valou et Vincent ? Je vous remercie très sincèrement pour l'entraide fraternelle et tous les supers moments passés en votre compagnie. Je remercie également l'ensemble des doctorants des bâtiments 425 et 430 que j'ai eu la chance de côtoyer. Je voudrais décerner aussi une mention spéciale à l'équipe de foot du labo, emmenée par Rachid, capitaine emblématique, buteur né et tacticien hors paire qui nous a mené tant et tant de fois au succès. Cette équipe m'a accueillie en son sein et je le lui rends plutôt bien en ne trouvant absolument jamais le chemin des filets. Plus sérieusement, des matchs du mercredi, aux urban foots, en passant par les soirées arrosées et les troisièmes mi-temps où la bonne humeur et la rigolade priment, cette équipe occupe pour moi une place tout particulière.

En outre, il coulait de source, que dis-je, il sonnait comme une extrême évidence de remercier l'AgroParisTech, mon second laboratoire d'accueil et tous les personnes que j'y ai rencontrées. J'y ai certes passé un temps relativement restreint, mais amplement suffisant pour constater qu'il est difficile d'imaginer meilleur environnement pour s'épanouïr scientifiquement et humainement. L'osmose qui y régne entre les permanents, les doctorants ou autres post-doctorants a un caractère tout-à-fait unique. Que ce soit dans les locaux, en pause café, lors des incontournables quizzs ou lors d'enrichissantes conférences, je tiens à remercier très chaleureusement toutes les membres du labo avec lesquels j'ai partagés des moments privilégiés. Un petit clin d'oeil oblige aux doctorants et encadrants "first generation" Anna, Sylvain, Loïc, Marie.C, Pierre.B, Pierre.G, Paul ou encore Laure avec lesquels j'ai découvert la communauté scientifique qui s'offrait à moi ainsi que tous les doctorants "new generation" avec une attention toute particulière aux habiles Timothée, Rana, Marie.P, Mathieu, Félix et Raphaëlle dont la rencontre en valait la chandelle.

L'accomplissement de cette thèse a été fortement facilitée aussi par toutes les personnes hors du cercle des mathématiques qui me sont chères. Tout d'abord une immense pensée pour Alexandre, Lydie, Claire et Michael, mes fidèles amis de ma belle région natale. Les excursions boulonnaises, wimereusiennes et dunkerquoises à vos côtés sont toujours source de détente assurée. Ensuite, je remercie très fortement le Tennis Club de Villebon qui me permet d'exercer ma passion et qui m'a permis de rencontrer des gens formidables. Jess, Carine, Olivier, Florence, Ghislain, Éric, Élise et Ben, un très sincère merci pour tous les moments que j'ai passés avec chacun d'entre vous. Je remercie également Lucas, Ezequiel, Benoït, Arthur, JB, Julien et Sylvain pour les duels fratricides que nous nous sommes livrés raquette en main mais aussi pour l'excellente ambiance qui règne lors des matchs par équipe. Je tiens également à remercier très chaleureusement Cécile, puis Yohan, Jonathan et David pour toutes les soirées parisiennes et palaisiennes ainsi que pour nos excursions sportives.

Ensuite, comment ne pas mentionner les membres du Rlajmnythep, à savoir Tony mon fidèle colloc, Henri, Jérémie, Lucie et Adrien. Toutes ces années d'études passées à vos côtés, ces inoubliables soirées, des sorties au Player's aux 6 heures de Gravigny en passant par nos mémorables virées à vélo, il n'y a rien à jeter. Que de souvenirs qui resteront à jamais gravés dans ma mémoire et qui j'espère vont continuer à s'accumuler. Un immense merci !

Et enfin, j'aimerais avoir une énorme pensée pour les membres de ma famille, et plus particulièrement mes parents et mes grand-parents. Je mesure la chance que j'ai eue de grandir dans cette petite commune de 15 000 habitants de l'agglomération boulonnaise à vos côtés. Votre générosité, votre soutien et votre affection ont toujours été sans failles. Je voudrais vous dire merci pour tout ce que vous avez fait pour moi et pour tous les moments que j'ai partagés avec vous. Ensuite, comment ne pas mentionner my sweet and so funny sister, avec qui j'ai et j'aurais toujours une complicité sans égal. Enfin une grosse pensée à Aurélien, mon cousin pour tous les délires que l'on a eus et qui nous sont propres. Vous avez tous énormément compté pour moi et je vous en remercie grandement.

Table des matières

Chapitre 1

Contexte biologique et statistique

À l'heure actuelle, l'approfondissement de la connaissance des génomes constitue l'un des centres d'intérêt majeur en biologie moléculaire. Une manière de mieux comprendre le fonctionnement d'un organisme est de déterminer le rôle de chacun des gènes ainsi que la manière dont ils interagissent. Les progrès technologiques depuis une vingtaine d'années permettent désormais l'accès à une multitude de données moléculaires. Pour faciliter les avancées et concentrer les efforts, certains organismes ont été choisis par la communauté internationale : ils sont appelés organismes modèles. Parmi, ces espèces modèles, on trouve notamment la bactérie *Bacillus subtilis*, des levures telles que *S.cerevisiae*, des champignons tels que *Neurospora crassa*, des animaux vertébrés tels que le cobaye (*Cavia porcellus*) ou encore des plantes telles que le riz (*Oryza sativa*) ou encore *Arabidopsis thaliana*. Dans le cadre de cette thèse, nous avons cherché à mieux caractériser les facteurs de transcription d'*Arabidopsis thaliana*, qui sont des gènes importants pour la régulation de l'expression du génome. Dans l'introduction sur le contexte biologique, nous présenterons plus précisement les facteurs de transcription, les moyens techniques disponibles pour les étudier puis nous définirons l'objectif biologique de cette thèse. La dernière partie de ce chapitre expose la façon dont nous allons modéliser statistiquement la question biologique.

1 Gène d'intérêt : le facteur de transcription

1.1 Rôle du gène

L'information génétique, permettant à chaque individu de présenter les caractères de son espèce avec des variations qui lui sont propres, est portée par l'ADN, macromolécule située dans le noyau de chaque cellule d'un organisme. Cette information est transmise hors du noyau des cellules sous forme de protéines, molécules régissant la vie des cellules et de l'organisme au niveau phénotypique. La transmission de l'information génétique hors du noyau se fait en deux étapes (voir figure 1.1) :

— une étape dite de *transcription* lors de laquelle un morceau de brin d'ADN correspondant à la séquence d'un gène est copié en ARN messager (ARNm). Une fois synthétisé, l'ARNm migre à l'extérieur du noyau de la cellule vers son cytoplasme. Un ARNm est aussi appelé un transcrit.

— une étape dite de *traduction* lors de laquelle l'ARNm est traduit par un ribosome en protéine dans le cytoplasme de la cellule.

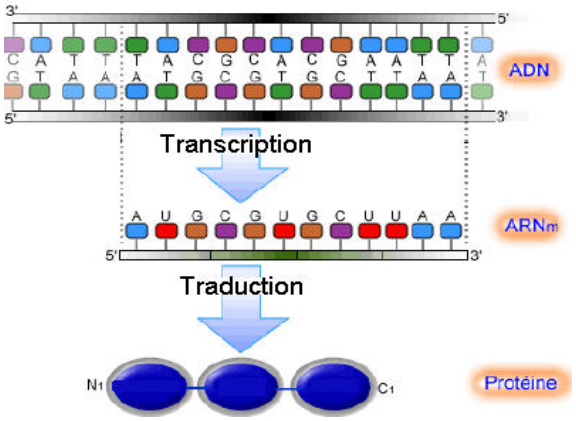

Figure 1.1 – Transmission de l'information génétique

Les facteurs de transcription (FTs) sont des gènes particuliers. Ils jouent un rôle majeur dans les étapes de *transcription* des brins d'ADN. En effet, une fois qu'un facteur de transcription est transcrit puis traduit, la protéine créée lors de ce processus réintégre le noyau de la cellule, puis se fixe sur la région de l'ADN indispensable à la *transcription* d'un gène, appelée promoteur ou région promotrice d'un gène. On dit alors que ce gène est une cible du facteur de transcription. Notons qu'un facteur de transcription peut-être lui-même une cible d'un autre facteur de transcription. Notons aussi qu'une protéine issue d'un même facteur de transcription peut se fixer au niveau des régions promotrices de différents gènes. Les facteurs de transcription possèdent donc plusieurs gènes cibles. De la même façon, un gène est la cible de plusieurs facteurs de transcription.

En réalité, pour démarrer la *transcription* d'un gène, il faut que les protéines formées par l'étape de *transcription-traduction* de tous les facteurs de transcription dont il est la cible forment des complexes protéiques. Ces complexes viennent ensuite se fixer sur la région promotrice du gène cible. Une fois seulement que toutes les protéines attendues sont présentes dans le complexe protéique, une enzyme nommée ARN polymérase, va démarrer la transcription du gène cible. On dit que les facteurs de transcription travaillent en module. De tels complexes peuvent être formés d'une bonne quantité de protéines, ne limitant pas spécialement le nombre de facteurs de transcription dont un gène peut être la cible (voir figure 1.2, [63]).

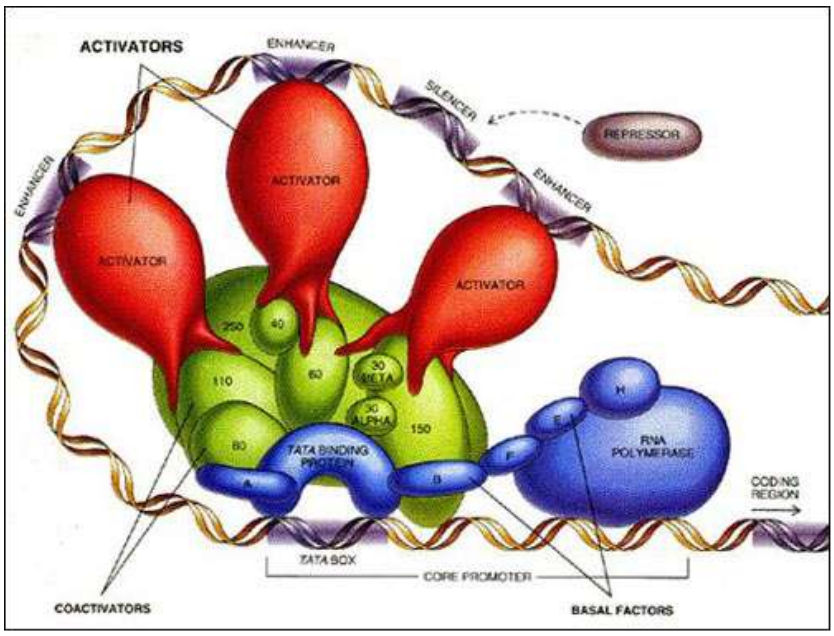

Figure 1.2 – Complexe protéique reconnaissant la double hélice d'ADN

Ces arguments nous amènent à considérer que les facteurs de transcription sont des pièces maîtresses des réseaux de régulation de gènes. Une connaissance approfondie de l'activité des facteurs de transcription résulterait donc en une connaissance approfondie de l'activité des gènes du même organisme.

1.2 Les facteurs de transcription d'Arabidopsis thaliana

Issue de la famille des brassicacées, regroupant essentiellement des plantes herbacées de l'hémisphère Nord, *Arabidopsis thaliana* (*At*), exposée figure 1.3, est une plante chlorophyllienne vasculaire servant de représentant pour l'un des deux grand groupes de plantes à fleurs, à savoir les dicotylédones.

Figure 1.3 – *Arabidopsis thaliana*

Certaines de ses caractéristiques ont motivé ce choix :

— sa petite taille (15 à 20 cm à l'âge adulte) facilitant sa culture en laboratoire.
— son cycle de vie très court (environ 3 semaines) facilitant les expériences.
— c'est un des plus petits génomes connus dans le monde végétal.
— ses cellules qui se multiplient particulièrement bien.

En 2000, *Arabidopsis thaliana* fut le premier génome végétal séquencé. L'absence d'intérêt économique concernant *At* a facilité le partage de la connaissance à son sujet. Actuellement, cette connaissance est mise à disposition sur un site international dédié à la plante, The Arabidopsis Information Resource (TAIR) ([58] , https ://www.arabidopsis.org/). Concernant les facteurs de transcription d'*Arabidopsis thaliana*, il y en a 2210 et sont regroupés en 79 familles structurales selon les motifs du domaine de fixation à l'ADN de leur protéine ([11]).

1.3 Mesure de l'activité des facteurs de transcription

Les expériences d'immunoprécipitation de chromatines (ChIP) (figure 1.4) dont le principe est d'isoler une protéine spécifique pour identifier ses lieux de fixation sur l'ADN, sont un moyen expérimental, au premier abord, intéressant pour éudier la régulation des gènes. Appliqué à une protéine produite par un facteur de transcription, cela permet d'identifier les cibles du facteur de transcription. Cependant, à cause de la spécificité des anticorps, il n'est pas possible de voir l'activité des facteurs de transcription ensemble mais uniquement facteur de transcription par facteur de transcription. En conséquence, les ChIPs ne peuvent nous permettre de détecter la combinaison de facteurs de transcription

nécessaire à l'activation d'un gène cible et par conséquent d'avoir une vision globale de la régulation des gènes d'*At*. Il faut s'appuyer sur d'autres mesures pour espérer quantifier l'activité des facteurs de transcription.

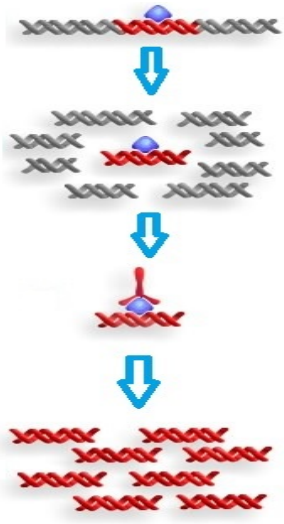

Figure 1.4 – Les ChIP s'opèrent en 5 étapes : (1) L'ADN est extraite de la cellule. (2) L'ADN est coupée en plusieurs brins courts. (3) Ajout d'un anticoprs spécifique à la protéine d'intérêt. Précipitation des complexes ADN-protéine-anticorps. (4) Élimination des brins d'ADN non associés à la protéine d'intérêt. (5) Étude par hybridation sur une puce (ChIP-chip) ou par séquençage (chIP-seq).

Le transcriptome correspond au niveau d'expression d'un gène, c'est-à-dire aux mesures de la quantité de transcrits de ce gène à un temps donné. Certaines technologies récentes telles que le séquençage du transcriptome entier (RNA-seq, [48], [13]) ou la technique des puces à ADN (ou hybridations, [37]) permettent de recueillir des données transcriptomiques. Or, considéré pour sa fonction première, le transcriptome ne permet pas de mesurer directement des interactions protéines-ADN. Cependant, les données transcriptomiques ont le mérite d'être nombreuses et de donner une vue globale de l'activité transcriptionnelle d'un génôme. Intéressons-nous au recueil de ces données par la technique des puces à ADN.

Les puces à ADN permettant de fournir des données transcriptomes se sont développées dans les années 90. Le principe est le même que celui de la transcription. Lors de l'étape de transcription, un fragment d'ADN dont la structure est une double hélice composée de deux brins complémentaires, est copié. Pour ce faire, la séquence de nucléotides, correspondant à un gène, d'un des deux brins du fragment d'ADN va être copiée par complémentarité et grâce à l'action de l'ARN polymérase, sous forme d'ARNm. L'ARNm a donc pour séquence de nucléotides celle du brin complémentaire à celui qui a été copié.

Les puces à ADN forment un dispositif permettant de capter les ARNm d'une cellule de l'organisme étudié en utilisant ce principe de complémentarité.

Une hybridation sur une puce 2 couleurs se déroule en 4 étapes (voir figure 1.5, [43]) :

1. *Fabrication des puces.* Des milliers de fragments d'ADN simple brin de l'organisme étudié sont disposés de manière ordonnée sur une lame de verre pour former plusieurs points appelés sondes, chaque sonde étant spécifique d'un gène de la plante. Ces fragments sont fixés à la lame puis amplifiés.
2. *Préparation des cibles.* On extrait les ARNm issus de la plante soumise à deux conditions différentes. On procède à une étape dite de transcription inverse lors de laquelle la séquence de chaque ARNm va être copiée par complémentarité en une séquence d'ADNc. Les brins d'ADNc issus de la première condition sont alors marquées en rouge et ceux issus de la seconde condition en vert.
3. *Hybridation.* Les brins d'ADNc des deux conditions sont mélangés puis disposés sur la puce à ADN fabriquée lors de l'étape 1. Chaque brin d'ADNc va s'associer avec son brin d'ADN complémentaire, situé au niveau d'une sonde, pour reformer la double hélice d'ADN.
4. *Lecture des résultats.* Chacune des sondes de la puce est soumise à un laser dont on récupère la fluorescence émise. On obtient alors deux images en niveau de gris. En remplaçant, les niveaux de gris de la première image par des niveaux de vert et ceux de la seconde par des niveaux de rouge, on obtient, après superposition des deux images, une image en couleur des sondes allant du vert au rouge.

Les résultats d'une hybridation sont donc des intensités. Ce sont donc des données continues. Une fois les intensités mesurées libérées au mieux de leurs biais techniques (par exemple ceux liés à la lame ou aux fluorochromes, [66]), on obtient donc les données transcriptomiques issues de l'expérience réalisée. Plus cette intensité est élevée, plus la quantité de transcrit est élevée.

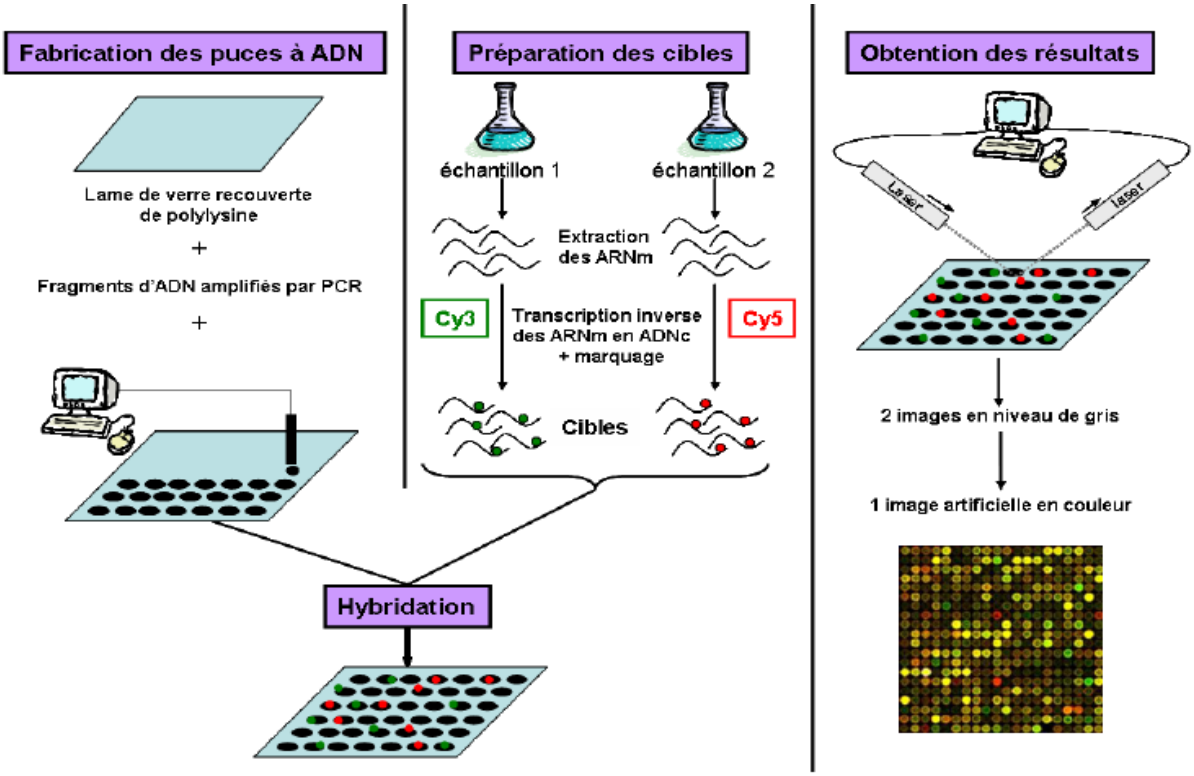

Figure 1.5 – Principe des puces à ADN

Les données transcriptomiques sont déposées dans des dépôts internationaux appelés National Center of Biotechnology Information (NCBI) ([22] , https ://www.ncbi.nlm.nih.gov/) et ArrayExpress ([49] , https ://www.ebi.ac.uk/arrayexpress/). Dans le cadre de cette thèse, j'ai travaillé à partir de données transcriptomiques générées par la plateforme de l'Unité de Recherche en Génomique Végétale d'Evry (URGV), qui a récemment rejoint l'Institut des Sciences et des Plantes de Paris-Saclay (IPS2). La puce à ADN utilisée est une puce conçue par l'initiative européenne CATMA (Complete Arabidopsis Transcriptome MicroArray), qui permet d'étudier tous les gènes codant les protéines simultanément. Toutes les données générées par la plateforme sont organisées dans la base de données CATdb ([21] , http ://tools.ips2.u-psud.fr/CATdb) et pour cette thèse, j'ai eu accès aux données publiques qui représentent 2670 mesures d'expression de 24 576 gènes de la plante, incluant 1937 facteurs de transcription.

1.4 Pertinence de l'utilisation du transcriptome des facteurs de transcription

Il est généralement énoncé que les facteurs de transcription d'un organisme étaient des gènes faiblement transcrits comparativement aux autres gènes. Si un tel dogme était vérifié, cela limiterait fortement l'intérêt de l'étude de l'organisation au sein de l'ensemble des facteurs de transcription d'un organsime par le biais de données d'expression. Or, Mitsuda ([47]) a soutenu alors expérimentalement, que pour *At*, le niveau d'expression des facteurs de transcription n'était pas significativement plus faible que celui des autres gènes. En effet, les résultat expérimentaux attestent effectivement d'une proportion plus importante de gènes non-FTs que de facteurs de transcription ayant des quantités de transcrits élevées mais en moyenne les facteurs de transcription présentent des niveaux d'expression tout-à-fait comparables à ceux des gènes non-FTs (voir figure 1.6).

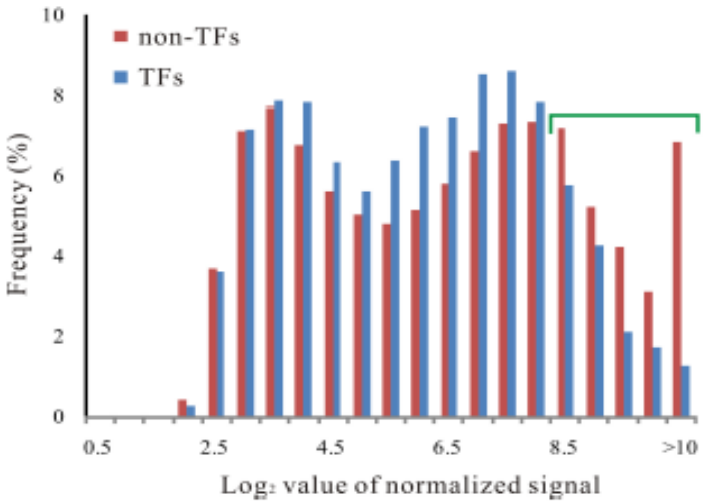

Figure 1.6 – Niveaux d'expression des gènes d'*At* recensés dans les expérimentations de [47]

Nous sommes également assurés de la pertinence de l'utilisation de nos données transcriptome en analysant les niveaux d'expression des facteurs de transcription sur un sous-ensemble d'expériences du jeu de données CATMA. Cela a consisté à étudier les résultats d'analyses différentielles réalisées par la plateforme transcriptome. Ces analyses differentielles, dont le principe est détaillé dans [43], sont des t-tests dans lesquels la variance est modélisée sur l'ensemble des gènes pour gérer le faible nombre d'observations par gène. Elles ont pour objectif de déterminer, pour une expérience d'hybridation et pour un gène, si la différence de niveau d'expression est significative. Si les facteurs de transcription sont toujours faiblement transcrits alors ils devraient être très peu différentiellement exprimés sur l'ensemble des 424 expériences considérées.

1.5 Résultats sur nos données transcriptomes

Nous avons recueilli les p-values des analyses différentielles réalisées sur tous les gènes d'*At*. Parmi celles-ci, nous ne considérons que celles des facteurs de transcription, en faisant fi des autres gènes. La figure 1.7 illustre, pour chaque facteur de transcription, le nombre d'expériences pour lesquelles le facteur de transcription est différentiellement exprimé. Les résultats rapportent que seuls 372 des 1937 facteurs de transcriptions ne sont jamais différentiellement exprimés sur l'ensemble des 424 expériences liées aux stress. L'hypothèse nulle est donc rejetée pour la majorité des facteurs de transcription d'*Arabidopsis thaliana*, traduisant le fait qu'il y a bien de l'information à extraire de l'analyse du transcriptome des facteurs de transcription d'*At*, comme le laissait présager Mitsuda ([47]).

Une première étude consiste, par curiosité, à regarder si l'information portée par le transcriptome des facteurs de transcription de la plante est semblable à la connaissance déjà établie sur les facteurs de transcription de la plante, à savoir la connaissance structurale. La figure 1.8 nous informe que plus une famille structurale de facteurs de transcription est volumineuse, plus celle-ci possède un nombre de facteurs de transcription différentiellement exprimés sur au moins une des expériences de stress conséquent. Le lien est même totalement linéaire. L'utilisation des données transcriptomes que l'on va utiliser dans cette thèse dans le but de détecter le mode de fonctionnement présumé de la régulation des facteurs de transcription entre eux nous apportera des informations autres que la connaissance structurale déjà parfaitement établie sur la plante.

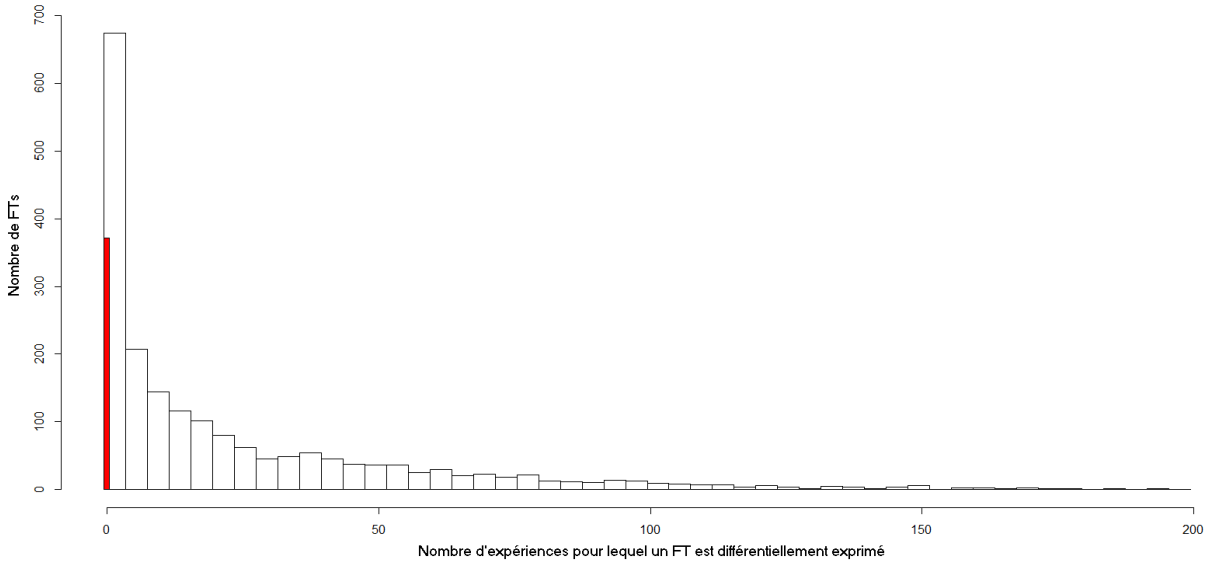

Figure 1.7 – Répartition du nombre d'expériences pour lequel chaque FT est différencié

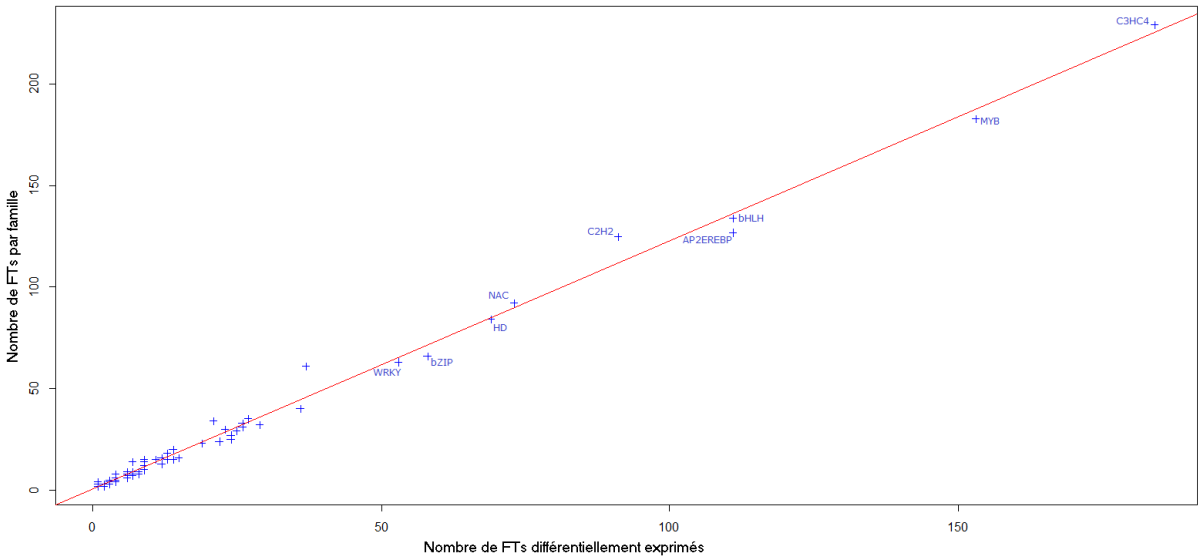

Figure 1.8 – Nombre de FTs différenciés par famille structurale en fonction de la taille de la famille

2 Objectif biologique de la thèse

Dans le cadre de cette thèse, nous nous intéressons à la régulation de l'expression des facteurs de transcription d'*Arabidopsis thaliana*. Nous avons vu précédemment que les facteurs de transcription travaillent en module (voir paragraphe 1.1). Un facteur de transcription sera donc activé par le complexe de protéines crées par l'ensemble de ses facteurs de transcription régulateurs. Nous appellerons par la suite ces complexes protéiques, de manière abusive, complexes de facteurs de transcription. Un facteur de transcription est donc régulé par un groupe de facteurs de transcription. De par ce mode de fonctionnement, nous allons émettre l'hypothèse suivante quant à l'organisation qui règne au sein de l'ensemble des facteurs de transcription :

— Un facteur de transcription vu comme régulateur agit toujours avec les mêmes facteurs de transcription partenaires pour activer certains facteurs de transcription. De tels facteurs de transcription partenaires sont appelés co-régulateurs. On peut ainsi diviser l'ensemble des facteurs de transcription d'*Arabidopsis thaliana* en plusieurs groupes de facteurs de transcription co-régulateurs.

— Un facteur de transcription vu comme régulé est toujours activé simultanément avec les mêmes autres facteurs de transcription. De tels facteurs de transcription activés simultanément sont appelés co-régulés. On peut ainsi diviser l'ensemble des facteurs de transcription d'*Arabidopsis thaliana* en plusieurs groupes de facteurs de transcription co-régulés.

Au final, la hiérarchie supposée fait état d'actions simultanée de groupes de facteurs de transcription vu comme régulateurs pour agir sur des groupes de facteurs de transcription vu comme régulés. (voir figure 1.9). L'objectif biologique de cette thèse consiste en la mise en évidence de cette supposée organisation et par conséquent en la détection de deux types de classification des facteurs de transcription : une première les regroupant en groupes de gènes co-régulateurs et une seconde en groupes de gènes co-régulés.

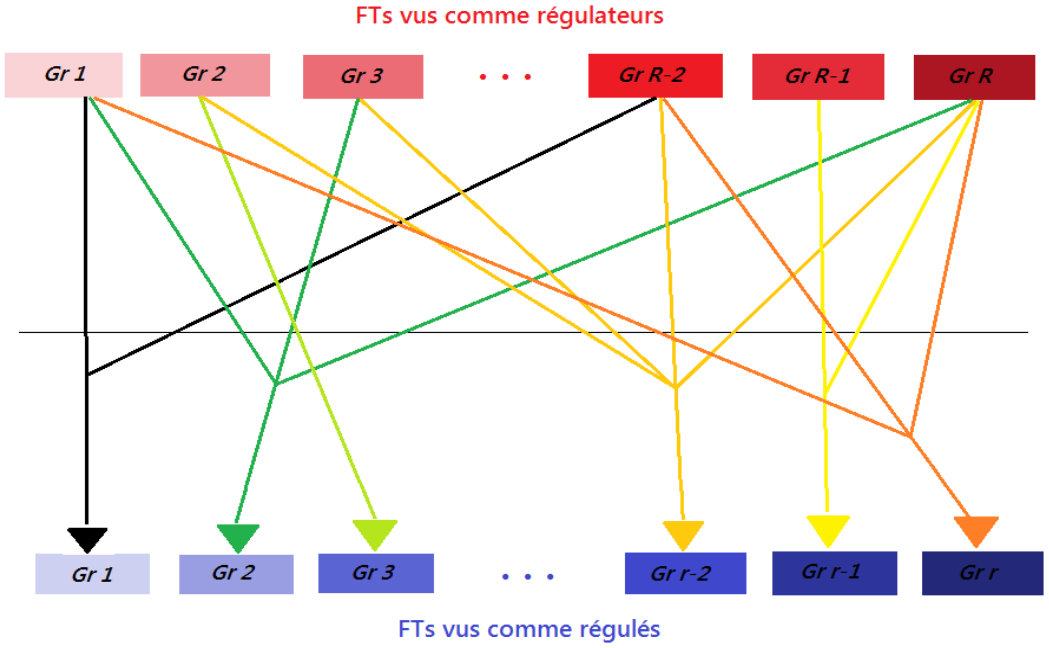

Figure 1.9 – Hypothèse d'organisation au sein de l'ensemble des FTs d'*At*

Pour résumer, notre objectif biologique est de mettre en évidence l'organisation présumée existant au sein de l'ensemble des facteurs de transcription d'*At*. Notre travail consiste donc en la mise en place et l'utilisation de méthodes statistiques dans le but de former les groupes de facteurs de transcription co-régulateurs et les groupes de facteurs de transcription co-régulés escomptés. L'utilisation de données transcriptome pour parvenir à cet objectif semble tout à fait pertinente selon cette étude préliminaire et devrait fournir des groupes de facteurs de transcription co-régulés et co-régulateurs différents des familles structurales de facteurs de transcription déjà établies.

3 Contexte statistique

Au vu de l'objectif biologique annoncé, il nous faut mettre en place une procédure statistique permettant à partir des données transcriptomes de regrouper les facteurs de transcription d'*Arabidopsis thaliana* en les deux types de classification souhaités. Classer ces gènes en groupes de gènes co-régulés et co-régulateurs sous-entend de connaître au

préalable les liens de régulation qui existent au sein de l'ensemble des facteurs de transcription. Or nous ne connaissons pas ces liens. Pour cela, nous avons décidé d'articuler notre méthodologie statistique en deux phases :

1. Une première consistant à former un réseau de régulation entre les facteurs de transcription d'*At*.
2. Une seconde ayant pour but de regrouper les facteurs de transcription en classes au vu des liens de régulation établis.

3.1 Formation d'un réseau de facteurs de transcription

Les variables statistiques sont les $p = 1937$ facteurs de transcription d'*Arabidopsis thaliana* pour lesquels nous avons des mesures d'expression et les observations sont les $n = 2670$ données transcriptomiques. Nous sommes dans un cadre de haute dimension statistique où $n \approx p$ et dans lequel nous devons inférer un réseau de régulation sur ces p facteurs de transcription.

3.1.1 Modélisation du problème

Depuis longtemps, ce type de problème est majoritairement traité à l'aide de modèles graphiques. La modélisation graphique consiste en l'utilisation d'un graphe pour représenter un modèle. Nous représenterons donc le réseau de facteurs de transcription à inférer par un graphe $\mathcal{G}$. Un noeud du graphe représente un des p facteurs de transcription. Les arêtes de ce graphe représentent les liens de régulation entre les facteurs de transcription. De par notre volonté de créer deux types de classification selon que les facteurs de transcription sont vus comme régulateurs ou régulés, nous devons représenter le réseau à inférer par un graphe orienté, c'est-à-dire dont les arêtes sont orientées. Une arête partant d'un premier noeud et orientée vers un second noeud traduit le fait que le facteur de transcription associé au premier noeud régule le facteur de transcription associé au second.

Concernant la modélisation, un des modèles graphiques les plus couramment utilisés est le modèle graphique gaussien ([36], [16] et [64]). Dans ce modèle, chaque noeud du graphe est modélisé par une variable gaussienne. Le vecteur formé de l'ensemble de ces variables gaussiennes est donc un vecteur gaussien de loi $\mathcal{N}(0, \Sigma)$. Le point fort de cette modélisation est que les coefficients de la matrice de précision $K = \Sigma^{-1}$ sont proportionnels aux coefficients de corrélation partielle, donc aux liens directs entre deux noeuds. Estimer l'ensemble des arêtes du graphe revient alors dans ce modèle à estimer cette matrice de précision. Celle-ci étant symétrique, le graphe que l'on estime par le biais de ce modèle est un graphe non orienté, c'est-à-dire qu'une arête n'est pas orientée. La présence

d'une arête entre deux noeuds traduit le fait qu'il y a lien de régulation entre les facteurs de transcription associés à ces noeuds sans savoir lequel des deux régule l'autre.

Ceci représente un défaut de ce modèle vis-à-vis de notre objectif biologique. Nous ne l'utiliserons donc pas pour modéliser notre réseau à inférer. Nous décidons néanmoins de modéliser chacun des noeuds de notre graphe $\mathcal{G}$ par une variable gaussienne centrée réduite $\mathcal{N}(0,1)$. Cependant pour estimer l'ensemble des arêtes du graphe, nous ne chercherons pas à estimer la matrice de précision du vecteur gaussien formé par les variables gaussiennes mais utiliserons une procédure de sélection de variables comme décrite ci-dessous.

Nous adopterons les notations suivantes :

— $\mathcal{I} = \{1, \ldots, n\}$ l'ensemble des n observations.

— $i \in \mathcal{I}$ représente une observation.

— $j \in \{1, \ldots, p\}$ représente un noeud du graphe.

— $X_j \sim \mathcal{N}(0,1)$ représente la variable gaussienne associée au noeud j .

— X est notre matrice de données de taille $n \times p$.

— Pour $\mathcal{M} \subset \{1, \ldots, p\}, X^{\mathcal{M}}$ correspond à la matrice de taille $n \times |\mathcal{M}|$, restriction de X aux variables X_j pour $j \in \mathcal{M}$. Si $\mathcal{M} = \{j\}$, on posera $X^j = X^{\{j\}}$ (si $\mathcal{M} = \{1, \ldots, p\} \backslash \{j\}$, on posera $X^{\mathcal{M}} = X^{-j}$). X^j est donc un n-échantillon de la variable X_j.

3.1.2 Concordance des données réelles avec le modèle graphique

La matrice de données d'expression X que nous traitons est log-normalisée. À partir de la matrice brute recueillie sur les hybridations, deux transformations successives sont ainsi effectuées :

1. une transformation logarithmique des données permettant une distribution "en cloche" des données, c'est-à-dire centrée et symétrique.
2. une renormalisation en colonne permettant d'homogénéiser les données pour chaque variable et de pouvoir les comparer entre elles.

Nous allons juger de la concordance des données ainsi transformées avec le modèle. Il s'agit de vérifier la normalité du n-échantillon X^j de chacune des variables X_j qui sont censées être gaussiennes. La densité de certains de ces n-échantillons est représentée en figure 1.10. Pour ce faire, nous nous appuyons sur le test de Kolmogorov-Smirnov(KS, [38]). Le but de ce test est de comparer la distribution des fréquences relatives cumulées établies sur l'échantillon d'observartions avec celle d'une loi normale avec moyenne et écart-type estimés. Dans notre cadre, l'hypothèse H_0 consiste à affirmer que l'échantillon provient d'un tirage d'une variable aléatoire de loi $\mathcal{N}(0,1)$.

Ce test réalisé au niveau $\alpha = 0.05$ et aux p-values réajustées selon Bonferroni rejette l'hypothèse H_0 pour 733 des $p = 1937$ n-échantillons testés, ce qui n'est pas négligeable. Le modèle graphique utilisé pour cette étape d'inférence de réseau ne concorde pas tout à fait avec les données que nous traitons. Cependant, l'objectif statistique de cette thèse n'est pas de proposer des modèles de façon à minimiser l'erreur d'approximation du modèle vis-à-vis des données mais plutôt à partir des modèles utilisées, de réaliser des estimations stables. Nous employons donc ce modèle graphique en sachant qu'il ne coïncide pas tout à fait avec notre jeu de données.

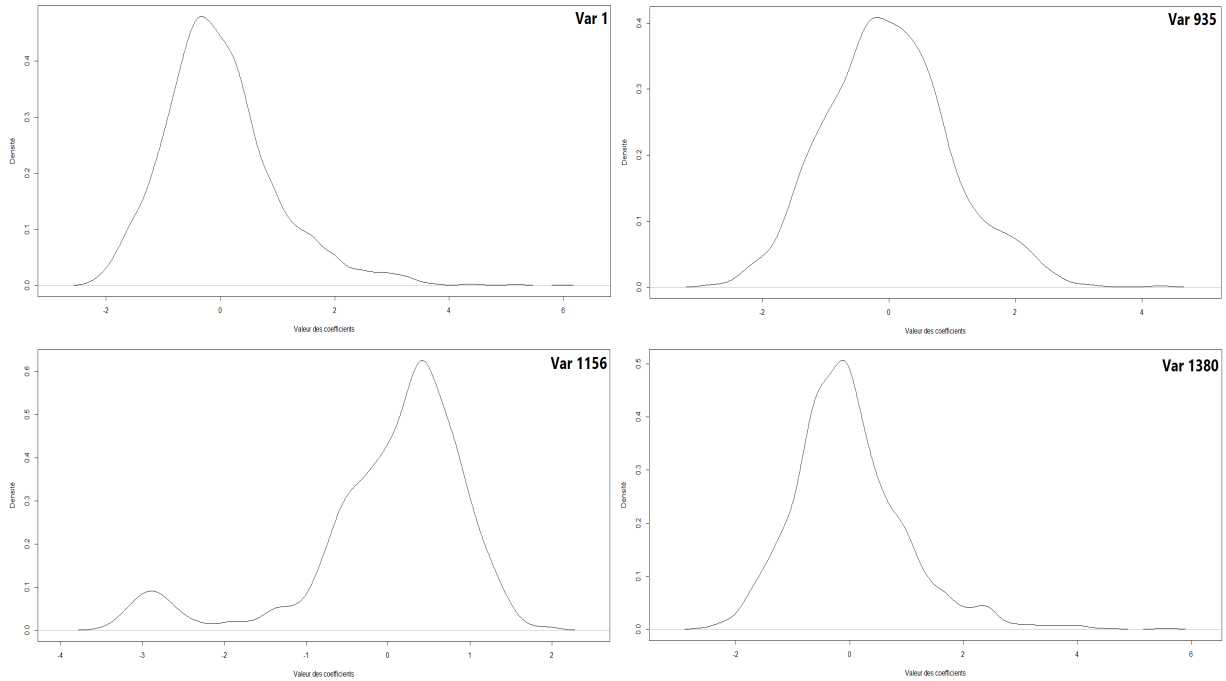

Figure 1.10 – Densité du n-échantillon observé de quelques variables

3.1.3 Sélection de variables : pénalisation LASSO

Reconstituer le réseau de régulation de facteurs de transcription revient donc, pour chaque noeud j, à estimer l'ensemble des noeuds j' possédant une arête orientée vers j. Ceci est un problème de sélection de variables. Sélectionner un ensemble de variables pertinentes pour le noeud j revient à sélectionner un sous-ensemble de $\{1, \ldots, p\} \backslash \{j\}$ qui auraient une arête orientée vers j. Il y a 2^{p-1} tels sous-ensembles. Une recherche exhaustive du sous-ensemble de variables le plus pertinent n'est pas envisageable. Dans notre cadre statistique de grande dimension, il est nécessaire d'employer des procédures statistiques de sélection de variables algorithmiquement faisables.

Meinschausen and Bühlman ([45]) proposent de détecter ces arêtes orientées à l'aide des p modèles de régression linéaire pénalisées d'une variable sur les autres traités indépendamment les uns des autres. La méthode de pénalisation utilisée est la pénalisation LASSO

(Least Absolute Shrinkage and Selection Operator), implémentable via l'algorithme LARS (Least-angle regression, [17]). Introduit par Tibshirani (1996, [59]) en régression linéaire, le LASSO correspond à une pénalisation ℓ_1 de la vraisemblance des modèles de régression linéaire. Plusieurs résultats de consistance ont été prouvés à son sujet, notamment la consistance asymptotique en signe ([68]) ou en sélection ([69]) une fois qu'une condition dite d'irreprésentabilité est vérifiée. Plusieurs corrections du LASSO ont d'ailleurs été proposés notamment dans [69], [3] (avec le recours au bootstrap) ou [60] pour pallier ce problème de condition à vérifier.

Partant du modèle de régression linéaire d'une variable sur les autres,

$$X^j = X^{-j}\Theta_j + \epsilon_j \text{ avec } \begin{cases} \Theta_j = \left\{\theta_{j,j'} \text{ pour } j' \in \{1,...,p\}\backslash j\right\}^T \\ \epsilon_j = \{\epsilon_{j,1},..,\epsilon_{j,n}\}^T \text{ tq } \{\epsilon_{j,i}\}_{i\in\{1,..,n\}} \text{ i.i.d de loi } \mathcal{N}(0,\sigma^2) \end{cases} \quad (1.1)$$

l'estimateur de Θ_j par le LASSO est :

$$\widehat{\Theta}_j^{Las}(\lambda_j) = \underset{\Theta_j}{\operatorname{argmin}} \left(\frac{1}{2}||X^j - X^{-j}\Theta_j||_2^2 + \lambda_j||\Theta_j||_1 \right). \quad (1.2)$$

Dépendant d'une pénalité λ_j, le LASSO permet de réduire bon nombre de coefficients $\widehat{\theta}_{j,j'}^{Las}$ à 0. En règle générale, plus la pénalité λ_j est élevée, plus le nombre de tels coefficients nuls l'est également. L'algorithme LARS fournit un nombre fini de pénalités λ_j pour lesquelles les coefficients $\widehat{\theta}_{j,j'}^{Las}(\lambda_j)$ nuls de l'estimateur du vecteur de coefficients de régression $\widehat{\Theta}_j^{Las}(\lambda_j)$ sont différents. À une pénalité λ_j, correspond le sous-ensemble de variables candidat $\left\{j' \in \{1,\ldots,p\}\backslash\{j\} \text{ tq } \widehat{\theta}_{j,j'}^{Las}(\lambda_j) \neq 0\right\}$. Le LASSO évince donc la plupart des 2^{p-1} ensembles de variables explicatives candidats au noeud j et ne conserve que les plus pertinents.

Nous adopterons dans cette thèse cette méthode de sélection de variables. Tout le problème résidera alors en la calibration de la pénalité LASSO pour chacun des modèles de régression d'une variable sur les autres, pour sélectionner le sous-ensemble de variables le plus pertinent parmi ceux déjà présélectionnés par le LASSO. Pour ce faire, nous aurons notamment recours à des critères de vraisemblance pénalisés. Quelqu'uns d'entre eux sont l'Akaike Information Criterium (AIC, [1]), le Bayesian Information Criterium (BIC, [56]), le C_p de Mallows ([40]), la validation croisée ([51]), l'heuristique de pente ([8] , [9]) ou encore le critère LINselect ([4]). Nous mettrons en place des méthodes de calibration des pénalités LASSO fondées en partie sur certains de ces critères dans un but de sélectionner des ensembles de variables les plus stables possibles. Nous insisterons dans le chapitre suivant sur le fait que le recours au rééchantillonnage est nécessaire pour cet objectif statistique.

3.1.4 Problèmes liés à la haute dimension

Une des difficultés majeures que nous allons rencontrer et notamment dans la première des deux étapes de notre procédure statistique de formation de groupes de variables réponses et régressées est le problème de la haute dimension. Cette difficulté requiert l'utilisation de méthodes de sélection peu lourdes applicables à chacun des p modèles de régression linéaire de l'équation (1.1). Dans un souci de sélection de sous-ensembles de variables pertinentes stables, il sera difficile dans ce cadre d'étudier cette stabilité pour nos méthodes employées sur chacun des p modèles de régression. Nous aurons alors régulièrement recours à l'utilisation d'une poignée seulement de ces modèles de régression linéaires, ceux associé à des variables expliquées que nous jugerons représentatives du panel. Ces quelques variables typiques nous servirons de variables tests pour les méthodes de sélection que nous établirons.

Un autre problème découlant de la haute dimension est celui du risque de réaliser des estimations en ultra haute dimension statistique. La ultra haute dimension est un cadre statistique dans lequel il est impossible de valider les estimations réalisées, rendant vain tout type d'estimation. Ce cadre apparaît lorsque $p >> n$. Pour la sélection de variables en modèles de régression, Verzelen ([61]) a établit un lien entre le nombre de variables p, le nombre d'observations n et la taille k des sous-ensembles sélectionnés. Ce lien établit que le cadre de la ultra-haute dimension est atteint si un sous-ensemble de variables sélectionné composé de k variables vérifie :

$$k > k^* \text{ où } k^* = \min\left\{k \in \mathbb{N} \text{ tq } 2k \times \log\left(\frac{p}{k}\right) \geq n\right\}. \tag{1.3}$$

Autrement dit, il est nécessaire de sélectionner des ensembles de variables de taille inférieure à k^* car notre modèle à p variables et n observations est mal adapté pour l'estimation de sous-ensembles de variables de tailles supérieures à k^*.

Nous avons tracé, en figure 1.11 la fonction $f : k \mapsto 2k \times \log\left(\frac{p}{k}\right)$ avec le nombre de variables correspondant à celui de notre cadre statistique ($p = 1937$). On s'aperçoit que pour le nombre d'observations dont nous disposons ($n = 2670$), la valeur minimale k^* de l'équation (1.3) n'est jamais atteinte. Cela signifie que le nombre d'observations dont nous disposons est suffisamment grand par rapport au nombre de variables pour ne jamais se situer en ultra haute dimension, ce qui rend notre cadre statistique plutôt avantageux. En revanche, lorsque nous aurons recours à des procédures de rééchantillonnage, le nombre d'observations sera divisé par 2 et vaudra $n/2 = 1335$. Pour le même nombre de variables p, un sous-ensemble de variables explicatives estimé devra être muni d'au plus 475 variables, sous peine d'être dans un cadre de ultra haute dimensions (voir figure 1.11). Nous devrons nous assurer par la suite que les ensembles de variables jugées comme

étant pertinentes pour la variable régressée, selon des procédures de sélection fondées sur le rééchantillonnage, soient de taille ne dépassant pas cette valeur.

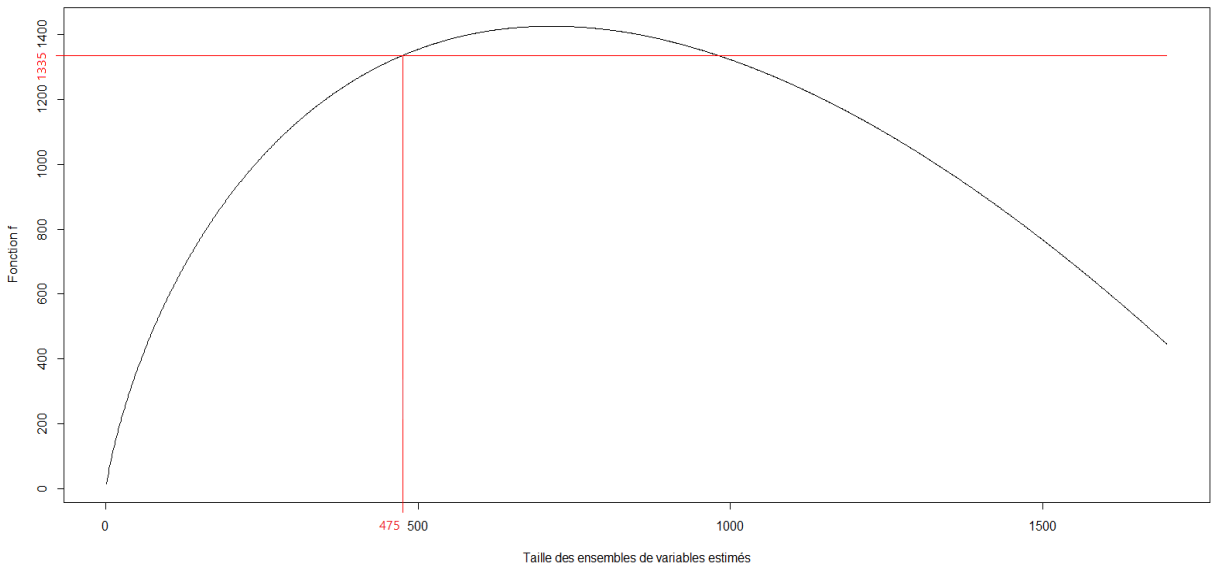

Figure 1.11 – Évolution de $f(k) = 2k \log\left(\frac{p}{k}\right)$ en fonction du nombre de variables sélectionnées k

3.2 Classification double des facteurs de transcription

Une fois estimées les arêtes du graphe $\mathcal{G}$, représentant le réseau de facteurs de transcription à inférer, il s'agit de regrouper ses noeuds en deux types de classification. Pour cet objectif, nous emploierons des méthodes de classification non supervisée. Dans le cadre de modèles de mélange, tout l'enjeu est de déterminer les labels inconnus des variables. Notre objectif biologique induit quant à lui l'obligation d'une méthode de classification non supervisée faisant intervenir deux types de labels. Nous préférerons en cela des modèles de classification de graphes tels que les modèles à blocs latents (LBM, [24]) plutôt que des modèles à blocs stochastiques (SBM, [20] et [29]), ces derniers modèles étant avant tout conçus pour la classification de graphes non orientés.

La difficulté principale de cette partie résidera en notre capacité à comparer les classifications doubles établies sur différents graphes formés par l'application de diverses procédures de sélection de variables sur notre jeu de données transcriptomiques. Plusieurs indices de comparaison de partitions doubles existent déjà, tels que l'Erreur de classification ([39]). Nous en proposerons un nouveau, fondé sur l'Ajusted Rand Index (ARI, [31]) qui est à la base un indice de comparaisons de partitions simples. Nous présenterons les avantages de ce nouvel indice par rapport à ceux déjà existant.

Nous chercherons également dans cette partie à évaluer la stabilité des résultats, en l'occurrence des classifications doubles établies à partir du jeu de données réelles. Ceci nous permettrait de valider les groupes de facteurs de transcription co-régulés et co-régulateurs correspondant. Dans ce but, nous avons mis au point une procédure de validation qui dans l'esprit relève du bootstrap paramétrique. Vu la complexité de notre démarche en plusieurs étapes, elle exige des simulations en cascade que nous pouvons esquisser ainsi. Une fois de tels jeux de données simulés, nous relancerons la procédure statistique globale (étape d'estimation des arêtes du graphe puis étape de double classification de ses noeuds) sur eux puis évaluerons la proximité des doubles classifications ainsi créées avec celle obtenue sur le jeu réel grâce à l'indice de comparaison que nous avons mis en place. Une procédure statistique globale stable devra former des classifications doubles variant le moins possible si l'on perturbe le jeu de données initial.

Chapitre 2

Procédures stables de sélection de variables

L'un des deux objectifs statistiques principaux de la thèse est la construction d'un graphe orienté modélisant le réseau de régulation à inférer. Comme annoncé lors du chapitre précédent, détecter les arêtes de ce graphe est un problème de sélection de variables. Il s'agit de déceler pour chaque noeud j du graphe, l'ensemble des autres noeuds qui le contrôlent. L'objectif de cette partie consiste en la mise en place de procédures stables de sélection de variables visant à estimer ces ensembles de noeuds contrôleurs. Nous entendons par stables des procédures sélectionnant des ensembles de variables variant peu lorsque les jeux de données traités sont modifiés. Les procédures que nous expérimenterons sont fondées sur les régressions linéaires pénalisées avec emploi de pénalités de type LASSO. Nous verrons que ne considérer qu'un seul chemin de régularisation par variable régressée induit une forte instabilité dans la sélection. Ceci nous incite à multiplier les chemins de régularisation à l'aide du rééchantillonnage. Toute la difficulté de cette partie consiste alors en l'ajustement des paramètres de ces procédures de façon à obtenir des ensembles de variables sélectionnées de taille raisonnable et stable. Le graphe construit à partir des arêtes sélectionnées par ces procédures sera, comme escompté, stable et prêt à être soumis aux méthodes de classifications de graphes orientés.

Dans toute la suite, nous désignerons par support du noeud j, l'ensemble S_j des variables explicatives à la variable régressée X_j.
Pour un noeud j, $S_j = \{j' \in \{1,..,p\} \backslash \{j\} \text{ tq. } \theta_{j,j'} \neq 0\}$. Il s'agit d'estimer les p supports S_j pour $j \in \{1,..,p\}$ à l'aide de procédures stables.

1 Gauss-LASSO

1.1 Description

La première procédure fondée sur la régression pénalisée que nous avons expérimentée est le Gauss-LASSO. Proposée par Meinshausen.N et Bühlmann.P ([45]), elle effectue p régressions linéaires pénalisées indépendantes d'une variable sur les autres. Nous avons testé la calibration de la pénalité λ du LASSO à l'aide de deux critères de vraisemblance pénalisée : la validation croisée (10-fold) et le critère BIC. Cette procédure s'applique sur chaque modèle de régression de manière indépendante et s'articule en 3 étapes (voir détail de la procédure avec choix du lambda par BIC en Annexe A) :

1. Obtention d'une collection de supports par sélection à l'aide du LASSO
2. Ré-estimation des coefficents de régression des variables de chacun des supports de la collection par la méthode des moindres carrés ordinaires
3. Choix du support de la collection à l'aide de la validation croisée ou du critère BIC

Notons que contrairement à [45], nous décidons de ne pas choisir la même pénalité LASSO pour chaque régression. Biologiquement, les FTs jouent des rôles différents et ne sont pas susceptibles d'être contrôlés par le même nombre d'autres FTs. Une pénalité λ pourra, de cette manière, être plus forte dans le cadre de la régression pénalisée d'une variable représentant un FT contrôlé par peu de FTs que dans le cadre d'une régression faisant état d'une variable modélisant un FT contrôlé par beaucoup d'autres.

Nous nous attendons à ce que Gauss-LASSO sélectionne des supports très peu stables. C'est ce qui est annoncé dans Bolasso ([3]). En effet, le choix d'une procédure de sélection fondée sur de la régression pénalisée mais ne faisant intervenir qu'un seul chemin de régularisation permet, sous certaines conditions que doit vérifier la pénalité λ, de sélectionner les vraies variables mais également un bon nombre de variables non pertinentes. La stabilité d'une telle procédure est alors limitée. Vérifions cette supposition sur notre jeu de données.

1.2 Résultats sur notre jeu de données

Selon l'histogrammes en figure 2.1 et le tableau 2.1 de la section 2.1.3, le choix des pénalités LASSO à l'aide de la validation croisée résulte en une sélection trop peu sévère.

En effet, la sélection par validation croisée n'élimine, en moyenne sur les p régressions, que deux tiers des variables explicatives. Ceci s'explique notamment par le fait que la validation croisée a pour objectif de sélectionner de bons modèles prédictifs, à défaut de modèles parcimonieux. Cette méthode de calibration de la pénalité LASSO n'est donc pas satisfaisante.

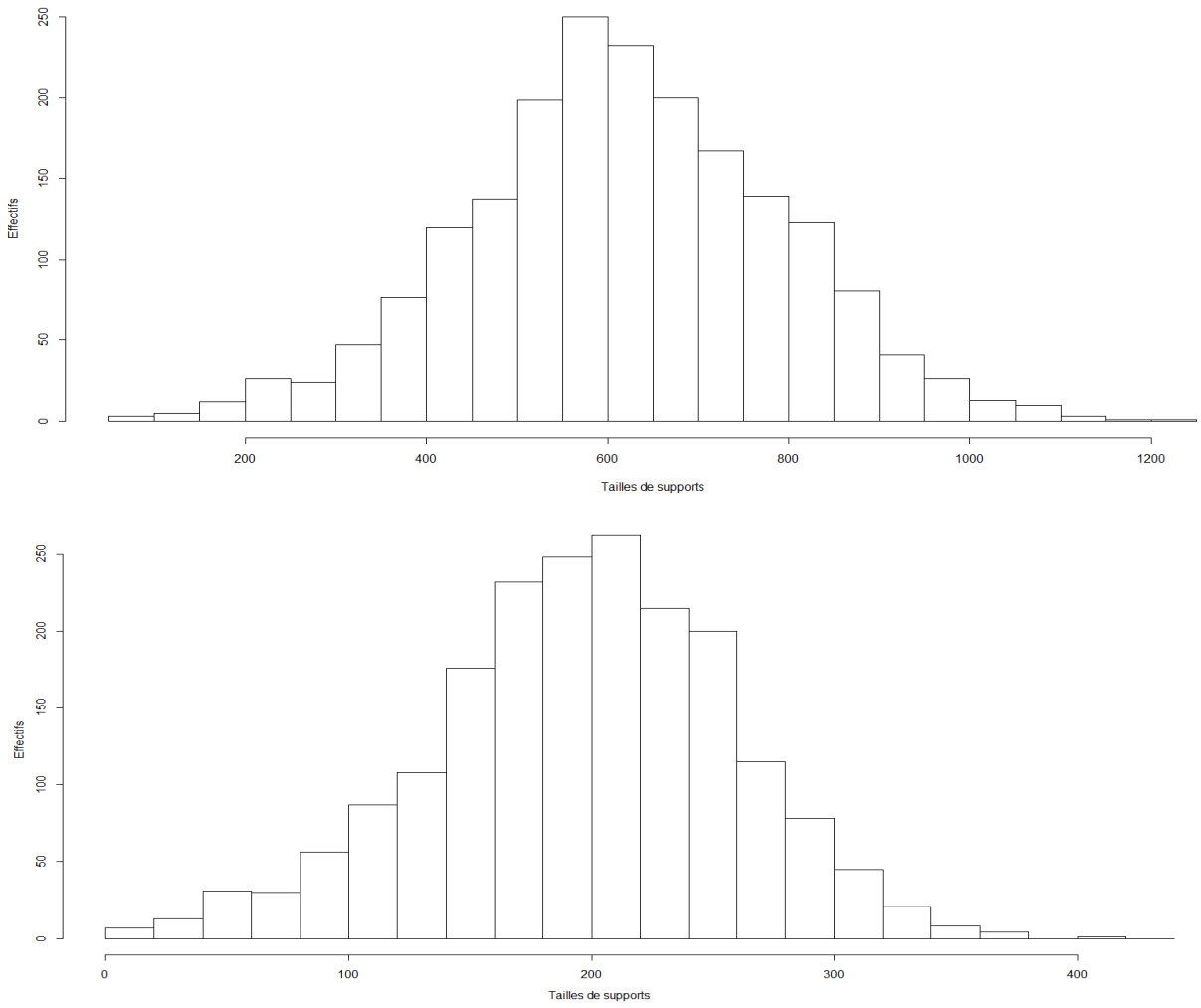

Figure 2.1 – Distribution des p tailles de supports estimées par Gauss-LASSO + CV (histogramme du dessus) et Gauss-LASSO + BIC (histogramme du dessous)

La sélection à l'aide du critère BIC est plus sévère que celle par validation croisée comme en attestent la figure 2.1 et le tableau 2.1 (situé en partie 2.1.3). Pour chaque variable régressée, un bon nombre de variables explicatives sont éliminées par Gauss-LASSO. De plus, l'allure satisfaisante des courbes BIC (voir figure 2.2 pour quatre des p variables) faisant état d'un minimum net tend à conforter l'utilisation de ce critère. En outre, d'un point de vue de l'interprétation biologique, comme il était détaillé dans le premier chapitre, le fait qu'un gène soit contrôlé par environ 200 FTs comme c'est le cas en moyenne selon les résultats statistiques de cette procédure, parait tout à fait raisonnable.

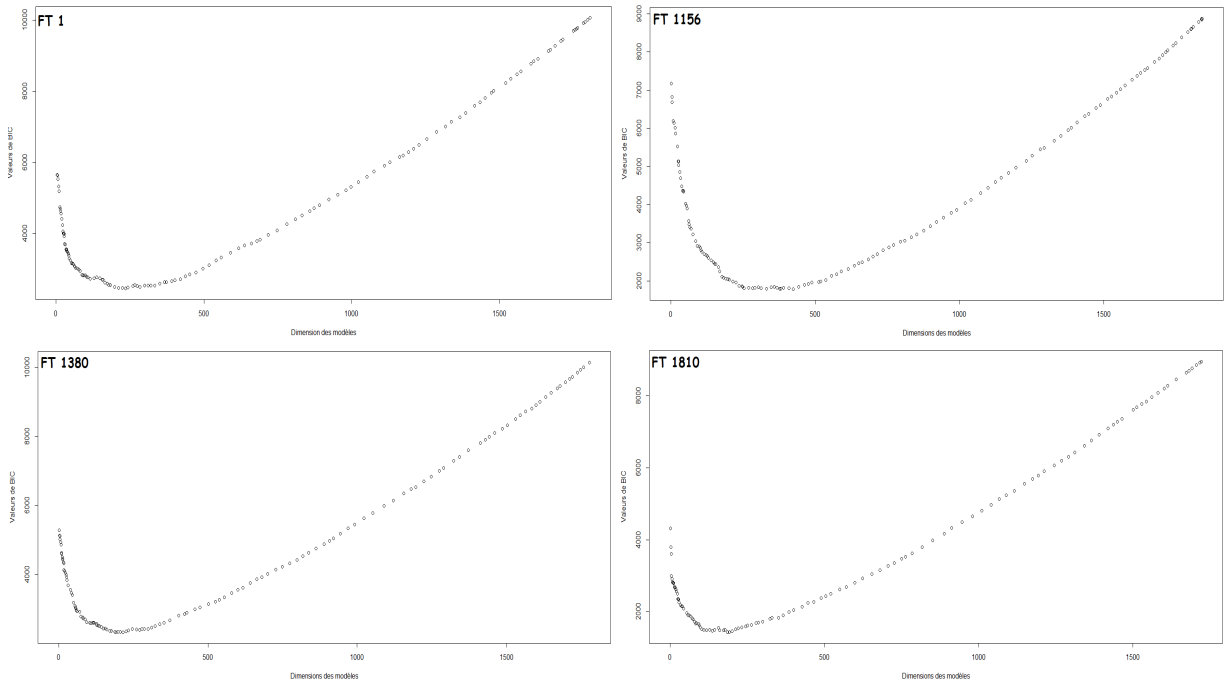

Figure 2.2 – Evolution du critère BIC en fonction des dimensions de modèles issus de Gauss-LASSO

S'il s'avère, en plus, que cette procédure de sélection est stable, celle-ci nous servirait de procédure de référence. Dans la suite, nous appellerons cette procédure Gauss-LASSO + BIC. Pour cela, nous décidons de créer des jeux de données issus du modèle gaussien qui sous-tend le modèle Gauss-LASSO.

1.3 Evaluation de la stabilité de la procédure

Pour évaluer la stabilité des supports estimés, nous décidons de créer des jeux de données suivant notre modèle graphique. Notre matrice de données peut être interprétée comme étant le tirage d'une loi gaussienne multivariée $\mathcal{N}\big(0, \Sigma\big)$. Nous générons alors des jeux de données de taille $n \times p$ selon la loi $\mathcal{N}\big(0, \widehat{\Sigma}_r\big)$ où $\widehat{\Sigma}_r$ est la matrice de variance-covariance estimée sur les données réelles. Une fois ces jeux générés, nous appliquons la procédure Gauss-LASSO + BIC sur chacun d'eux et comparons les résultats avec ceux de notre jeu de données.

En pratique, nous avons généré 10 jeux. Pour s'assurer qu'ils ont été correctement simulés, nous étudions la proximité entre $\widehat{\Sigma}_r$ et chacune des 10 matrices de variance covariance estimées sur les jeux simulés $\widehat{\Sigma}_{s_1}, ..., \widehat{\Sigma}_{s_{10}}$. Nous avons pour cela calculé, pour chaque matrice simulée ${}^{(k)}X$, la matrice ${}^{(k)}D = \widehat{\Sigma}_r - \widehat{\Sigma}_{s_k}$ différence entre sa matrice de variance covariance associée et celle estimée sur le jeu réel. Les coefficients des matrices ${}^{(k)}D$ sont très proches de 0 (cf. figure 2.3), ce qui atteste de la fiabilité des simulations.

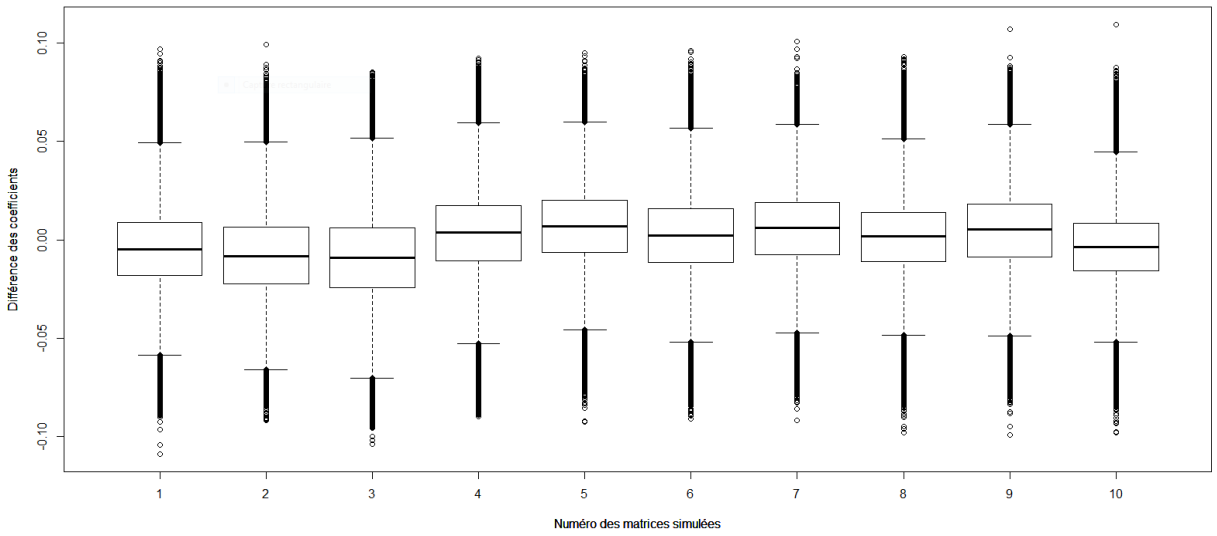

Figure 2.3 – Répartition des coefficients des matrices ${}^{(k)}D = \widehat{\Sigma}_r - \widehat{\Sigma}_{s_k}$

Nous appliquons la procédure Gauss-LASSO + BIC à chacun des jeux de données simulés. Nous décidons de restreindre l'évaluation de la stabilité aux modèles de régressions de cinq variables parmi les p que nous choisissons pour leur taille de support estimé par la procédure :

— la première de la liste (variable 1)
— celle de taille de support minimale (variable 935)
— celle de taille de support maximale (variable 1156)
— deux variables de taille de support moyenne (variables 1380 et 1810)

Dans la suite, nous appellerons ces cinq variables "variables typiques" et aurons régulièrement recours à elles pour évaluer la fiabilité de nos procédures. Il s'avère que, pour chacune des cinq variables typiques, les supports estimés par la procédure sur les onze jeux dont nous disposons (le jeu réel et les dix simulés) ont des tailles d'un ordre de grandeur similaire. Néanmoins, les contenus de ces onze supports sont très hétérogènes comme en témoigne le faible nombre de variables appartenant à leur intersection (cf. figure 2.4). Ceci met l'accent sur l'instabilité de la procédure. Une légère variation du jeu de données étudié résulte en des sélections relativement différentes. L'utilisation d'un seul chemin de régularisation dans l'optique d'obtenir une sélection stable semble peu sûr. Le recours à plusieurs chemins de régularisation est nécessaire pour pallier ce problème.

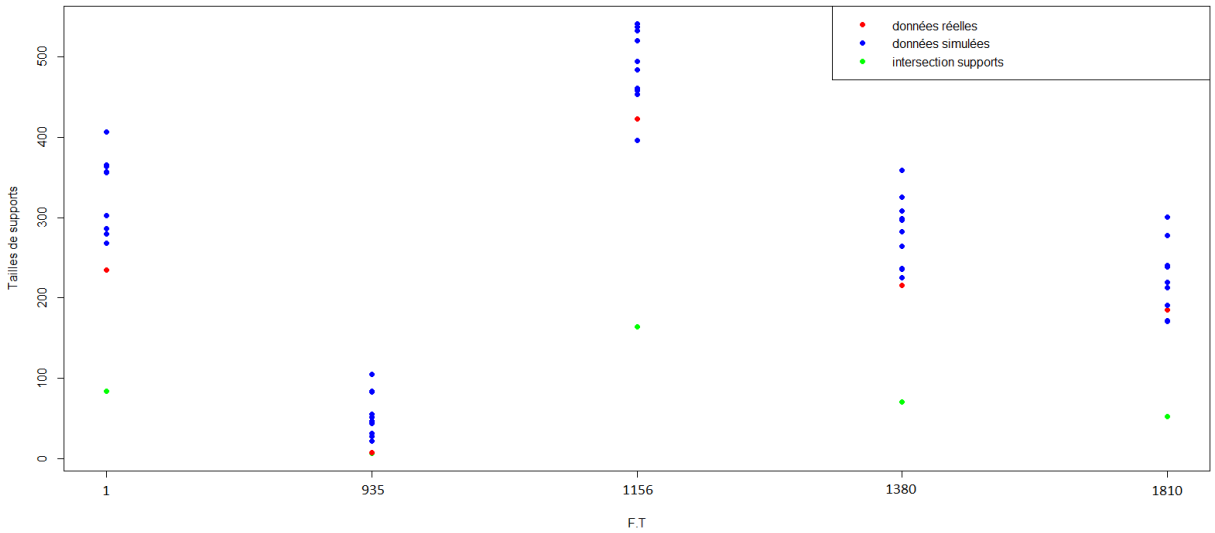

Figure 2.4 – Comparaison des supports estimés par Gauss-LASSO+BIC sur le jeu réel et les simulations

1.4 Stabilisation par rééchantillonnage

Pour pallier le problème d'instabilité du Gauss-LASSO + BIC, nous faisons appel au rééchantillonnage. Le but du rééchantillonnage est, en perturbant les données, de fournir non plus un seul mais une multitude de chemins de régularisation. Cela revient à proposer, pour chaque variable régressée j, non plus un ensemble de supports candidats comme le fait le Gauss-LASSO, mais plusieurs. Pour diversifier au maximum ces ensembles de supports, nous proposons de rééchantillonner par Sample-Splitting (SS), inspiré de TIGRESS ([28]) :

- Tirage de $m/2$ sous-ensembles $\mathcal{J}_1, , .., \mathcal{J}_{m/2}$ du jeu $\mathcal{I}$, tq. $|\mathcal{J}_1| = ... = |\mathcal{J}_{m/2}| = \lfloor \frac{n}{2} \rfloor$.
- Ajout des $m/2$ ensembles complémentaires $\left(\forall k, \mathcal{J}_{m/2+k} = (\mathcal{J}_k)^C\right)$ dans $\mathcal{I}$.
- Restriction de X à chacun des m jeux créés : $\forall k \in \{1, .., m\}$, on pose ${}^{(k)}X = X_{|\mathcal{J}_k}$.
- Création de m sous-modèles pour X_j : $\forall k \in \{1, .., m\}$, ${}^{(k)}X^j = {}^{(k)}X^{-j}.\Theta_j + \epsilon_j$.

Cette technique, comparée à du rééchantillonnage classique, permet de mieux contrôler le nombre de variables sélectionnées à tort. La prise en compte de sous-jeux de données tirés aléatoirement ainsi que leurs sous-jeux complémentaires renforce la légitimité des variables sélectionnées. En effet, une variable sélectionnée un bon nombre de fois parmi les m chemins de régularisation associés aux sous-modèles proposés, le sera sur les chemins issus des sous-jeux tirés aléatoirement ainsi que ceux issus de leurs sous-jeux de données complémentaires donc disjoints. Cette technique de rééchantillonnage est en cela tout à fait en adéquation avec notre volonté de stabiliser les supports.

Pour gagner en stabilité, nous exploitons par la suite ces m sous-modèles de régression proposés par le SS avec deux procédures, Gauss-LASSO *stabilisé* et Gauss-LASSO *enrichi*, construites différemment dont nous calibrerons les paramètres et comparerons les résultats.

2 Gauss-LASSO stabilisé

La procédure fondée sur le rééchantillonnage que nous proposons et que nous appellerons Gauss-LASSO stabilisé est basée sur le principe de stabilité dans la sélection élaborée par Meinshausen.N et Bühlmann.P ([46]). La procédure de sélection Bolasso proposée par F.Bach ([3]) est également fondée sur ce principe et a inspiré la procédure Gauss-LASSO stabilisée que nous avons mise en place. La première différence majeure entre la procédure de sélection que nous proposons et Bolasso est la méthode de rééchantillonnage utilisée, Bolasso s'appuyant sur du rééchantillonnage par bootstrap tandis que Gauss-LASSO stabilisé s'appuie sur du Sample Splitting pour pénaliser plus sévèrement l'instabilité de la sélection. La procédure Gauss-LASSO stabilisé se présente d'ailleurs de cette manière, pour une variable régressée j (Voir détail de la procédure en Annexe B) :

1. Étape de Sample Splitting : génération de $m/2$ sous-ensembles des observations, de cardinal $n/2$, et prise en compte des sous-ensembles complémentaires
2. Restriction des données à chacun de ces sous-ensembles. Obtention de m sous-modèles de régression
3. Application de la procédure Gauss-LASSO + BIC à chacun des m sous-modèles
4. Obtention d'un vecteur de scores correspondant à la fréquence d'apparition de chaque variable explicative parmi les m supports obtenus
5. Restriction des supports finaux aux variables de scores supérieurs à un seuil s choisi

Par ce procédé, une variable explicative qui pourrait être sélectionnée exclusivement par Gauss-LASSO+BIC sur un chemin de régularisation, sera difficilement sélectionnée sur plusieurs chemins de régularisation, qui plus est associés à des sous-jeux formés par Sample Splitting. Son score sera donc faible. Cette procédure, de par sa construction est donc forcément apte à estimer des supports plus stables que Gauss-LASSO+BIC. Néanmoins, elle est conditionnée par deux paramètres, à savoir le nombre de sous-modèles créés m et le seuil de scores s. Il s'agit désormais d'évaluer l'impact de ces deux paramètres sur les supports finaux estimés et de les calibrer pour une stabilité optimale.

2.1 Importance des paramètres

Bolasso ([3]) propose, dans chacun des m sous-modèles créés par rééchantillonnage, de sélectionner un support par Gauss-LASSO et de choisir une pénalité λ identique pour chacun des sous-modèles. Le choix d'une même pénalité LASSO sur chaque sous-modèle semble justifié puisque chacun d'entre eux a pour fin l'estimation du support de la même variable régressée. Cependant, pour choisir cette valeur de λ commune, [46] réalise du rééchantillonage sur des intervalles de pénalité. Une fois cette étape réalisée, l'article démontre que la valeur du λ choisi, pourvue qu'elle ne soit ni trop petite ni trop grande a un impact limité sur le support final estimé par une telle procédure de stabilisation. En effet, dans le but de contrôler le nombre de variables sélectionnées à tort, le choix de λ peut être compensé par la valeur de s.

Dans notre procédure Gauss-LASSO stabilisé, la pénalité est choisie sur chaque sous-jeu à l'aide du critère BIC, correspondant à un choix de pénalité différent selon les sous-jeux. Ceci n'est pas pertinent au premier abord. Mais, le choix d'un tel critère pénalisé a l'avantage de nous affranchir du choix de la pénalité LASSO commune aux m sous-modèles. Or appliquée sur notre jeu de données réel pour quatre variables typiques régressées, les m valeurs de λ sélectionnées par BIC sont très proches les unes des autres (cf. figure 2.5). La variable 935 fait exception mais, malgré un panel de pénalités plus varié, le nombre de variables sélectionnées sur chaque sous-modèle reste faible et quasiment identique. Nous omettrons souvent de la mentionner car présentant peu d'intérêt.

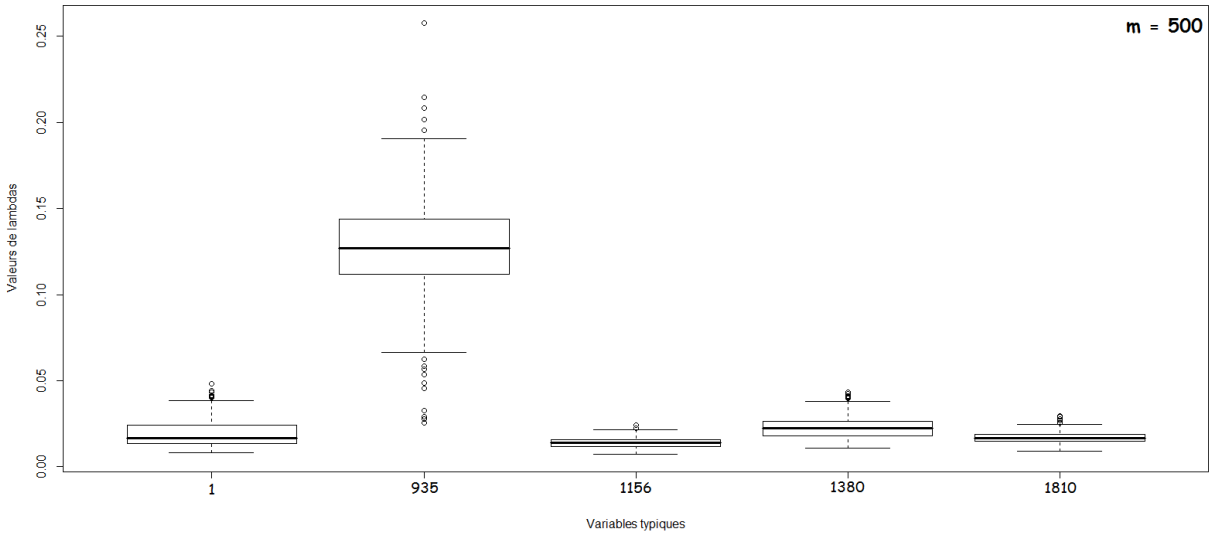

Figure 2.5 – Répartition des valeurs des pénalités LASSO choisies par BIC sur les m chemins de régularisation

Notons également qu'aucun des supports estimés sur les m jeux de taille $n/2$ n'est de taille supérieure à 475, ce qui nous aurait confronté au problème de la ultra-haute dimension selon [61] (voir chapitre précédent).

2.1.1 Importance de m pour Gauss-LASSO stabilisé($s = 1$)

La procédure Bolasso munit le support final d'une variable régressée comme l'ensemble des variables sélectionnées dans chacun des chemins de régularisation proposés par le rééchantillonnage, c'est-à-dire les variables ayant un score de 1. Nous fixons dans un premier temps le paramètre s de Gauss-LASSO stabilisé à 1 et étudions la variabilité des supports estimés lorsque m varie. Notons $\widehat{S}_j^{GLstab}(m, s)$ le support estimé par la procédure pour la variable régressée j. Selon les figures 2.6 et 2.7, les supports $\widehat{S}_1^{GLstab}(m, s = 1)$ et $\widehat{S}_{1156}^{GLstab}(m, s = 1)$ de deux variables typiques se stabilisent très rapidement. Une dizaine de sous-échantillons seulement permettent d'épurer considérablement les supports et quelques centaines permettent de stabiliser leur taille.

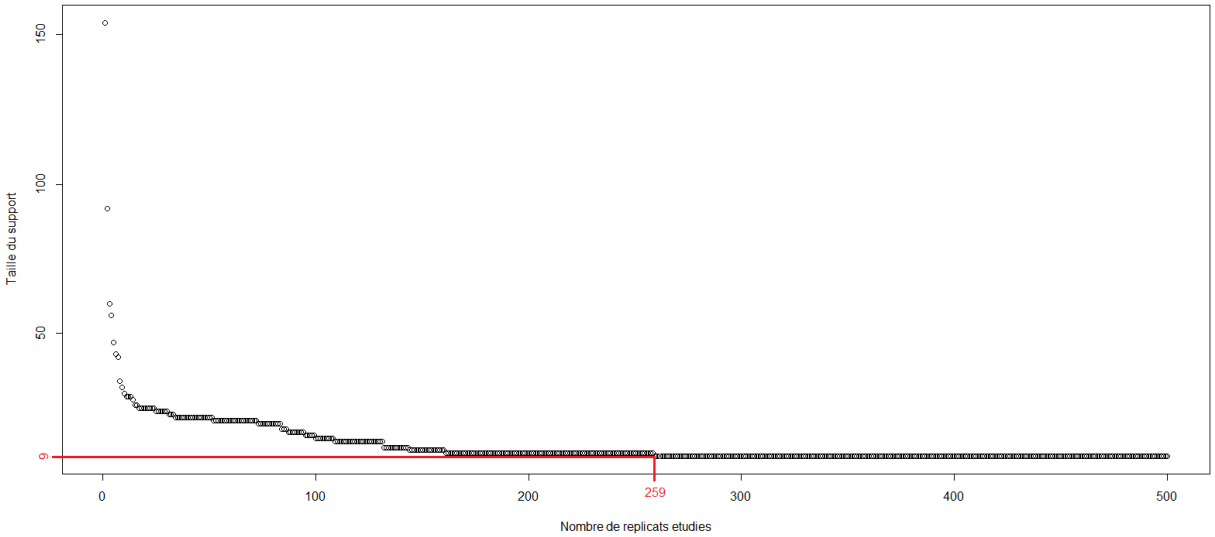

Figure 2.6 – Evolution de $\left|\widehat{S}_1^{GLstab}(m, s = 1)\right|$, support de la variable 1 en fonction de m

Prenons comme référence $m = 500$. La figure 2.8 détaille la répartition sur les p variables régressées du nombre m de sous-échantillons suffisants pour atteindre le support obtenu pour la valeur $m = 500$. La figure 2.9 représente la répartition sur les p variables régressées du nombre de sous-échantillons suffisants pour obtenir un support correspondant au support de référence (pour $m = 500$) plus une variable. On y apprend qu'en moyenne sur les p variables, les supports pour $m = 338$ et $m = 550$ sont identiques. De plus, 240 sous-échantillons sont suffisants en moyenne pour obtenir le support de référence plus une variable.

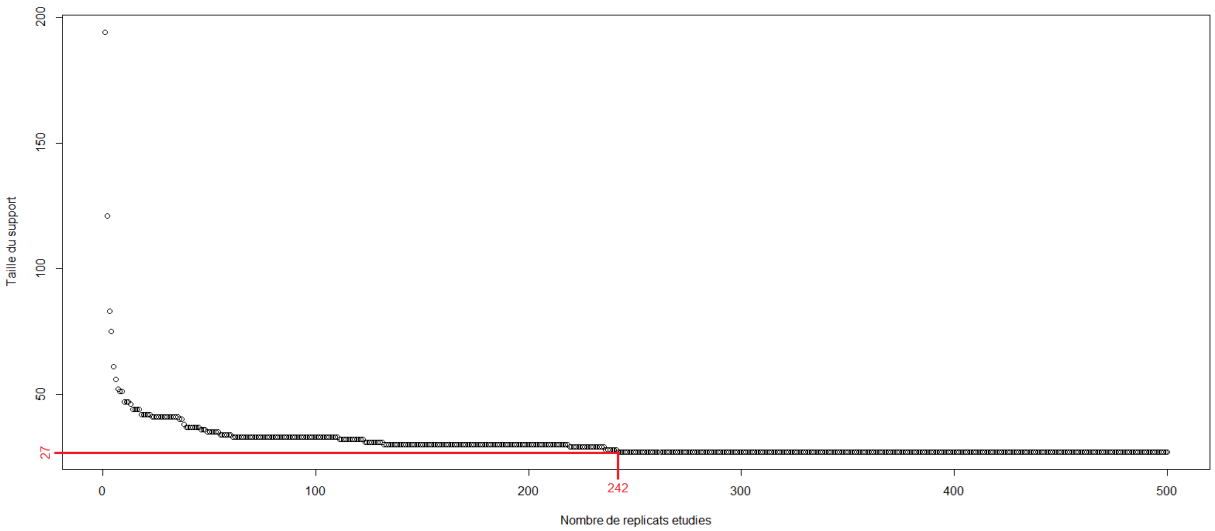

Figure 2.7 – Evolution de $\left|\widehat{S}_{1156}^{GLstab}(m, s = 1)\right|$, support de la variable 1156 en fonction de m

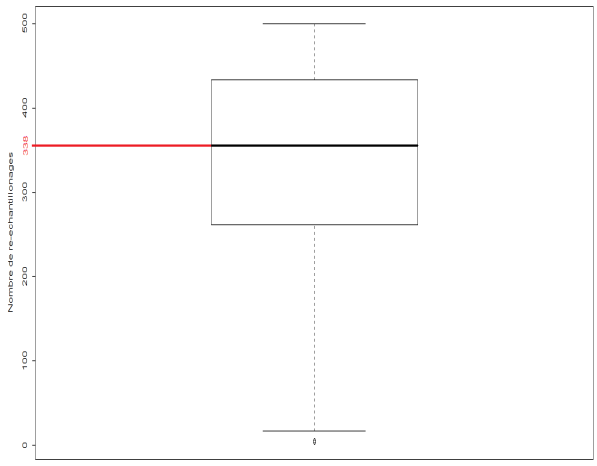

Figure 2.8 – Répartition du nombre m suffisant pour atteindre le support à $m = 500$

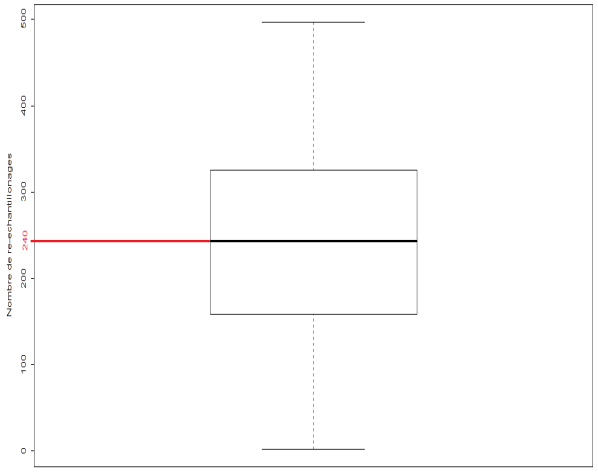

Figure 2.9 – Répartition du nombre m suffisant pour atteindre le support (+1 var) à $m = 500$

Le paramètre m de Gauss-LASSO stabilisé n'a un impact que très faible sur la taille des supports estimés pour $s = 1$. Étudions si tel est toujours le cas à seuil s quelconque.

2.1.2 Importance de m pour Gauss-LASSO stabilisé à s quelconque

La figure 2.10 (resp. 2.11) fait état de courbes représentant l'évolution de $f_s : m \mapsto |\widehat{S}_1^{GLstab}(s, m)|$ pour des seuils s fixés entre 0.1 et 1 par pas de 0.1 (resp. entre 0.8 et 1 par pas de 0.02). Ces courbes se stabilisent très rapidement et la valeur de m tant qu'elle n'est pas choisie trop petite, n'influence que très peu la taille du support estimé par Gauss-LASSO stabilisé comme il est représenté pour la variable 1. Ceci est également

vérifié pour les quatre autres variables typiques. En conséquence, nous pouvons choisir librement une valeur de m supérieure à 200 sans que les supports estimés soient impactés.

Nous fixons donc dans la suite m à 300 et désormais s est considéré comme le seul paramètre de la procédure. Le support estimé par cette procédure pour la variable j sera désormais noté $\widehat{S}_j^{GLstab}(s)$. Il reste à savoir si le rôle de s est lui aussi mineur.

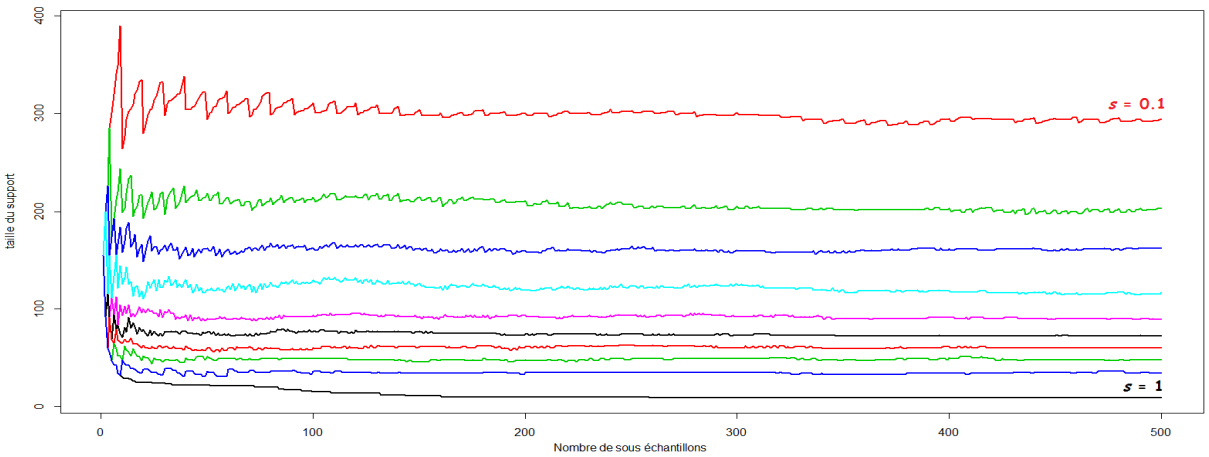

Figure 2.10 – Evolution de $m \mapsto \left|\widehat{S}_1^{GLstab}(m,s)\right|$ à s fixé (pas de 0.1)

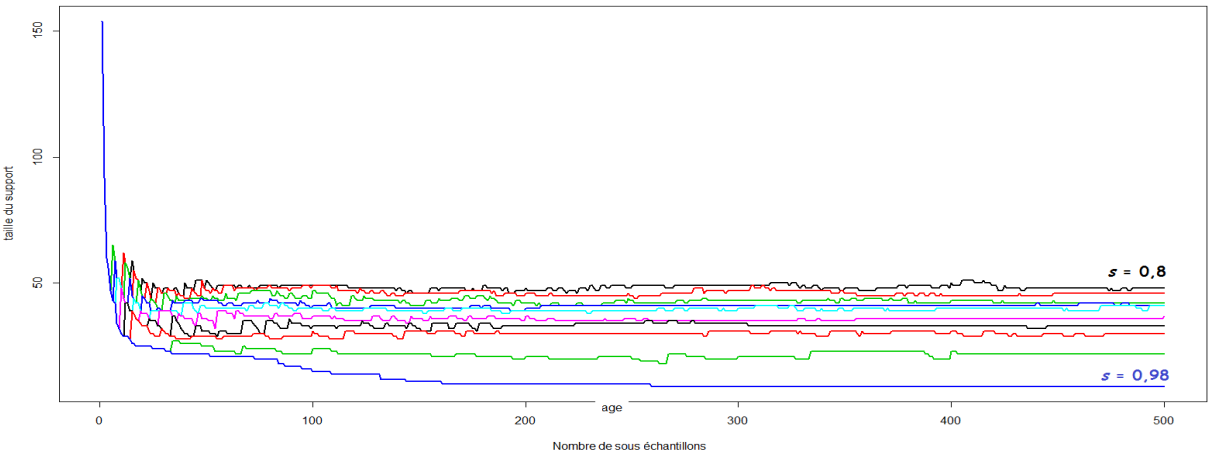

Figure 2.11 – Evolution de $m \mapsto \left|\widehat{S}_1^{GLstab}(m, s=1)\right|$ à s fixé (pas de 0.02)

2.1.3 Importance du choix de s

Dans l'article de F.Bach ([3]), la procédure proposée Bolasso fixe le seuil de scores à 1 et appliquée à quelques jeux de données simulés, elle donne de bons résultats prédictifs comparativement à d'autres méthodes de sélection telles que LASSO ou RIDGE. Il y réside cependant quelques cas où l'erreur de prédiction de la procédure explose. Diminuer la valeur de s subvient alors à ce problème. Dans *Stability Selection* ([46]), des simulations

réalisées sur la procédure de rééchantillonnage, font intervenir un seuil moins restrictif (souvent $s \simeq 0.6$). Les seuils de scores utilisés dans [3] et [46] diffèrent selon les cas de figure. Il est donc difficile de fixer un seuil universel. Nous envisageons donc, en cela, de mettre en place une méthode de calibration de s pour Gauss-LASSO stabilisé permettant une bonne stabilité des supports estimés.

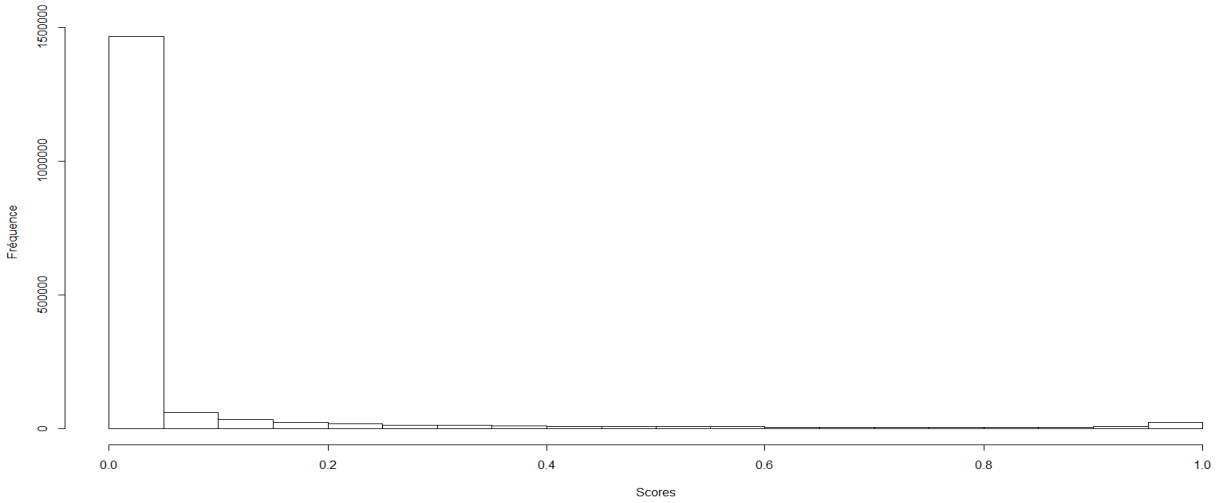

Figure 2.12 – Repartition de l'ensemble des scores issus de notre procedure

Lorsque nous appliquons la procédure Gauss-LASSO stabilisé à notre matrice de données réelles, la très forte majorité des scores sont nuls ou très proches de 0 comme l'illustre la figure 2.12. Cela signifie que dans le cadre de l'estimation du support d'une variable j, la grande majorité des variables explicatives n'apparaissent que très rarement parmi les m supports $\left\{ {}^{(k)}\widehat{S}_j^{GL}(\lambda_j^{BIC}) \right\}_{k\in\{1,\dots,m\}}$ associés aux sous-modèles de régression. Un seuil de scores s même peu sévère permettrait d'épurer drastiquement les supports estimés par la procédure. Cependant, la présence d'une quantité non négligeable de variables aux scores compris entre 0.9 et 1 (cf . figure 2.12) prouve qu'une faible quantité de variables sont très stables puisque sélectionnées quasi-systématiquement sur les sous-modèles associés aux sous-échantillons créés par Sample Splitting. On s'attend à ce qu'une variation même infime de s autour de 1 puisse modifier la sélection de manière non négligeable.

Si l'on choisit, comme pour Bolasso, un seuil de scores égal à 1, les supports estimés par Gauss-LASSO stabilisé sont munis d'une dizaine de variables en moyenne sur les p variables régressées. Cependant, la légère baisse de ce seuil à $s = 0.97$ double en moyenne la taille des supports estimés. (cf. tableau 2.1 et figure 2.13). Ceci confirme bien ce qui était présagé. Le choix de s, compensant notamment le choix de la pénalité λ dans les m sous-modèles de régression, est un paramètre déterminant de la procédure. Sa bonne calibration pour une sélection stable représente l'enjeu majeur de la suite de cette partie.

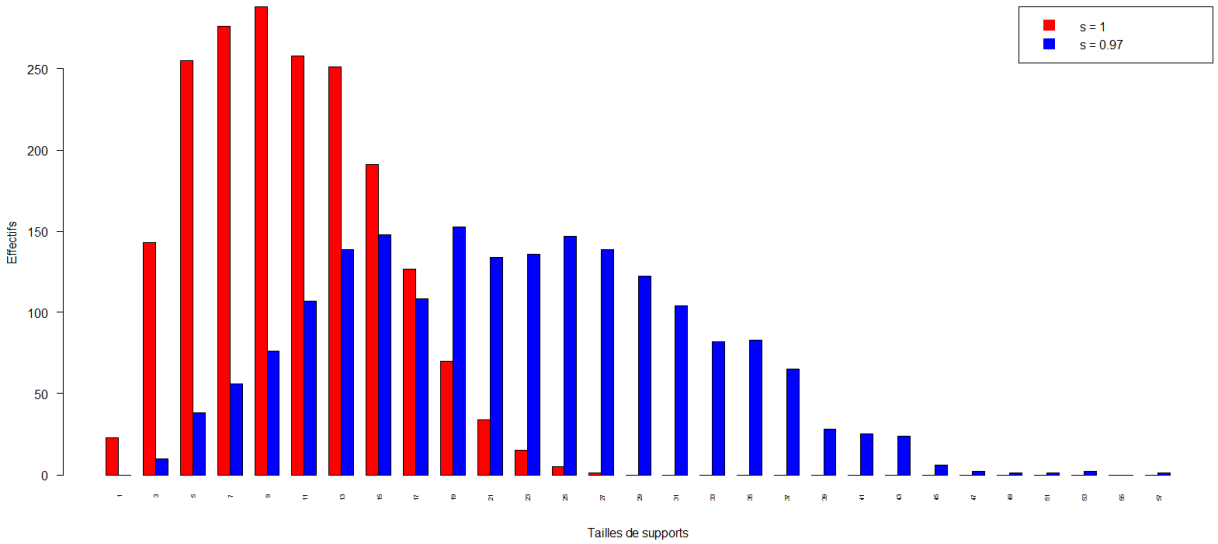

Figure 2.13 – Distribution des p tailles de support estimés par Gauss-LASSO stabilisé(s)

Procédures	**Minimum**	**Moyenne ± Ecart-type**	**Maximum**
Gauss-LASSO + CV	82	618.72 ± 173.14	1238
Gauss-LASSO + BIC	8	196.40 ± 62.10	402
GL stabilisé ($s = 1$)	0	9.77 ± 4.80	27
GL stabilisé ($s = 0.97$)	2	21.82 ± 9.38	57
GL stabilisé ($\widehat{s}_{BIC}$)	7	100.48 ± 35.49	230
GL stabilisé ($\widehat{s}_{inter}$)	7	121.76 ± 36.23	211
GL stabilisé($\widehat{s}_{HP}$)	6	93.51 ± 35.35	258
GL enrichi($l = 2$)	3	96.17 ± 33.11	303

Table 2.1 – Taille des p supports estimés par nos différentes procédures

2.2 Comparaison des erreurs de prédiction

Gauss-LASSO stabilisé est une procédure de sélection. Elle n'a donc pas été mise en place dans un but prédictif. Néanmoins, si les erreurs de prédictions de cette procédure sont trop importantes, il sera difficile de lui accorder du crédit. Notre première idée consiste, pour nos cinq variables typiques, à évaluer l'erreur de prédiction effective ainsi que la taille des supports estimés par Gauss-LASSO stabilisé en fonction du seuil s choisi.

2.2.1 Résultats pour s proche de 1

Le calcul des erreurs de prédiction a pour objectif de constater l'écart entre les données des variables régressées et leurs valeurs estimées. Il se fera à l'aide de la 10-fold Cross-Validation (voir détail de la méthode de calcul en Annexe C).

On choisit dans un premier temps, de lancer ces procédés pour $s = 1$ et $s = 0.97$, de façon à apprécier l'impact d'une lègère baisse de seuil sur les erreurs de prédiction :

Procédure	**Informations**	**V 1**	**V 935**	**V 1156**	**V 1380**	**V 1810**
Gauss-LASSO + BIC	Taille support	230	8	402	196	196
	Erreur de prédiction	5.38	3.00	4.43	5.36	4.76
Gauss-LASSO stabilisé ($s = 1$)	Taille support	9	3	23	7	4
	Erreur de prédiction	9.52	3.06	9.93	9.33	6.87
Gauss-LASSO stabilisé ($s = 0.97$)	Taille support	22	3	53	20	17
	Erreur de prédiction	7.84	3.06	7.51	7.87	6.27
Gauss-LASSO stabilisé($\widehat{s}_{BIC}$)	Taille support	146	7	230	79	76
	Erreur de prédiction	6.31	3.00	5.15	6.03	5.09
	$\widehat{s}_{BIC}$	0.33	0.35	0.325	0.57	0.535
Gauss-LASSO stabilisé ($\widehat{s}_{inter}$)	Taille support	157	10	203	129	95
	Erreur de prédiction	6.14	3.00	5.03	5.86	5.01
	$\widehat{s}_{inter}$	0.31	0.18	0.42	0.35	0.42
Gauss-LASSO stabilisé ($\widehat{s}_{HP}$)	Taille support	120	10	142	79	76
	Erreur de prédiction	6.58	3.00	5.57	6.01	5.09
	$\widehat{s}_{HP}$	0.39	0.23	0.70	0.59	0.54
GL enrichi($l = 1$)	Taille support	290	170	398	376	295
GL enrichi($l = 2$)	Taille support	79	7	147	79	110

Table 2.2 – Erreurs de prédictions et tailles de supports estimés pour les 5 variables typiques

Les procédures Gauss-LASSO stabilisé($s = 1$) et Gauss-LASSO stabilisé($s = 0.97$) ne sélectionnent que très peu de variables. Ceci explique le fait qu'elles présentent des erreurs de prédictions, pour les cinq variables typiques, dégradées comparées à celles de Gauss-LASSO+BIC. Néanmoins, la baisse de s de 1 à 0.97 double certes la taille des supports estimés mais diminue plus significativement la valeur des erreurs de prédiction. On peut ainsi légitimement penser que les quelques variables rajoutées aux supports estimés par Gauss-LASSO stabilisé par cette baisse de seuil sont des variables explicatives pertinentes.

La diminution du seuil de scores engendre une baisse de l'erreur de prédiction. Cependant, selon que l'on fasse varier s autour de valeurs élevées ou faibles, cette baisse de l'erreur de prédiction pourrait être plus ou moins prononcée. Levons cette interrogation en dressant, pour les variables typiques, les courbes de ces erreurs de prédiction en fonction des tailles de supports estimés par Gauss-LASSO stabilisé pour différents seuils.

2.2.2 Impact de la variation de s sur les erreurs de prédiction

Pour nos variables typiques, les fonctions $s \in]0,1] \mapsto Err_j^{CV}\left(\widehat{\Theta}_j^{GLstab}(s)\right)$ représentées en figures 2.14 et 2.15, sont majoritairement croissantes. Les erreurs de prédiction de Gauss-LASSO stabilisé se dégradent bien au fur et à mesure que $s > 0$ augmente. On remarquera en outre que les erreurs de prédiction de Gauss-LASSO stabilisé($s = 1$), sont comparables à celles de Gauss-LASSO($s = 0$), et donc mauvaises. Ceci tend à confirmer que, pour $s = 1$, la procédure sélectionne beaucoup trop sévèrement.

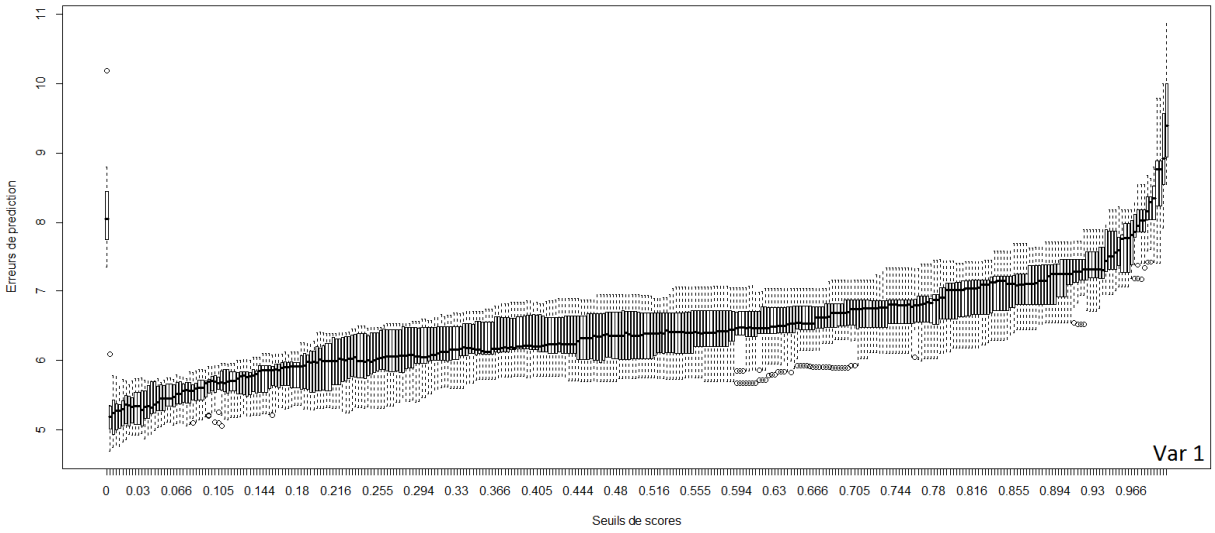

Figure 2.14 – Évolution de l'erreur de prédiction en fonction de s - variable 1

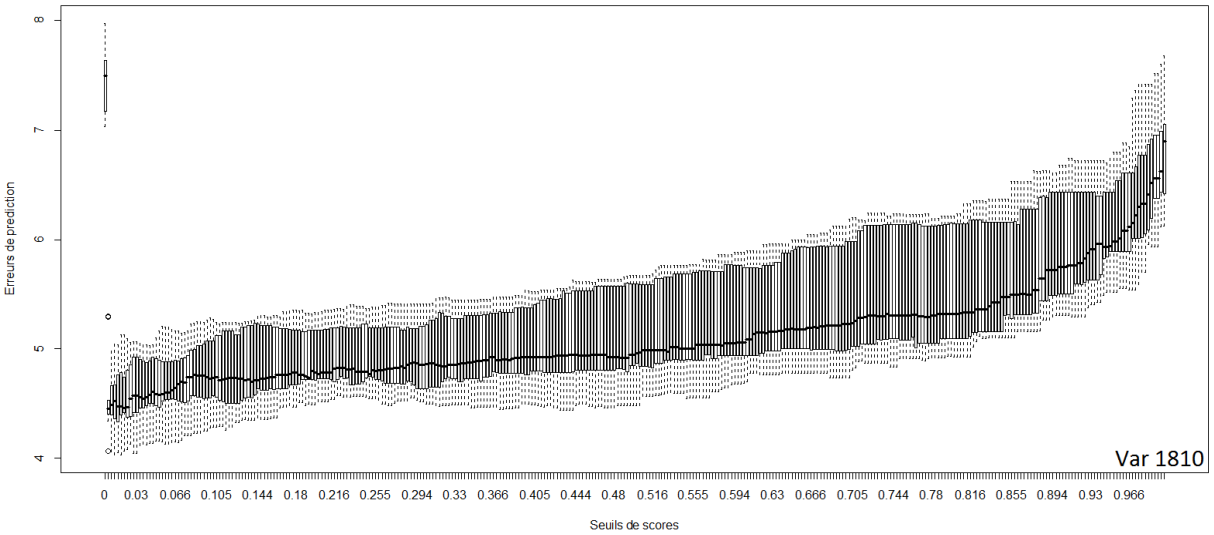

Figure 2.15 – Évolution de l'erreur de prédiction en fonction de s - variable 1810

Nous décidons d'approfondir l'étude en mesurant, au fur et à mesure que s diminue, l'impact de la baisse des erreurs de prédictions sur l'augmentation des tailles de supports estimés. Les figures 2.16 et 2.17 illustrent que, pour les variables typiques, la baisse de s à partir de $s = 1$ entraîne une chute de l'erreur de prédiction conséquente comparativement au nombre de variables ajoutées dans les supports. À partir d'un certain seuil, les variables ajoutées dans les supports estimés par Gauss-LASSO stabilisé ne permettent plus aux erreurs de prédiction de diminuer significativement.

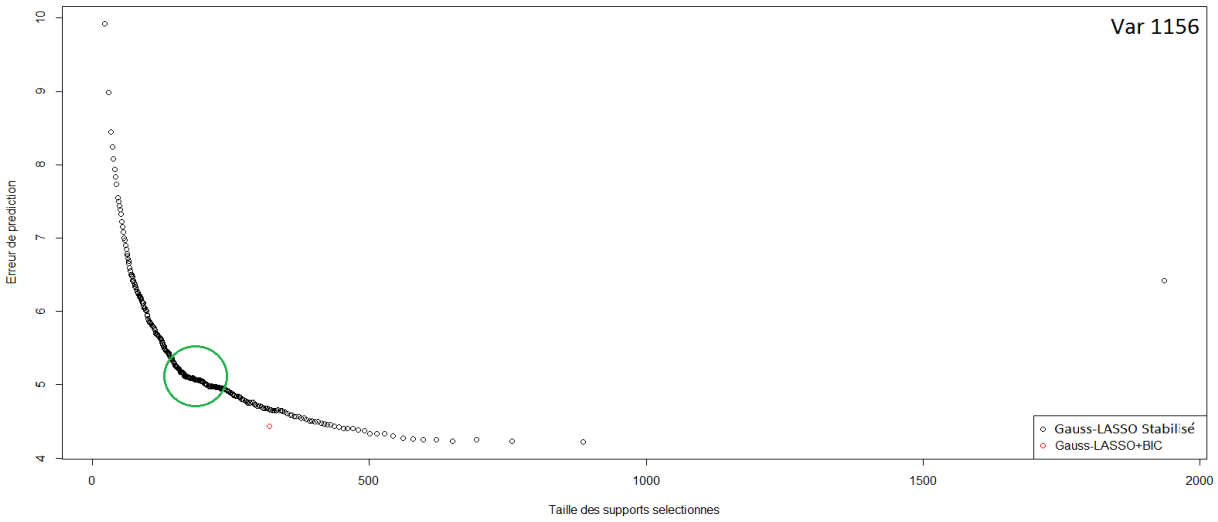

Figure 2.16 – Erreurs de prédiction VS Tailles des supports lorsque s varie - FT 1156

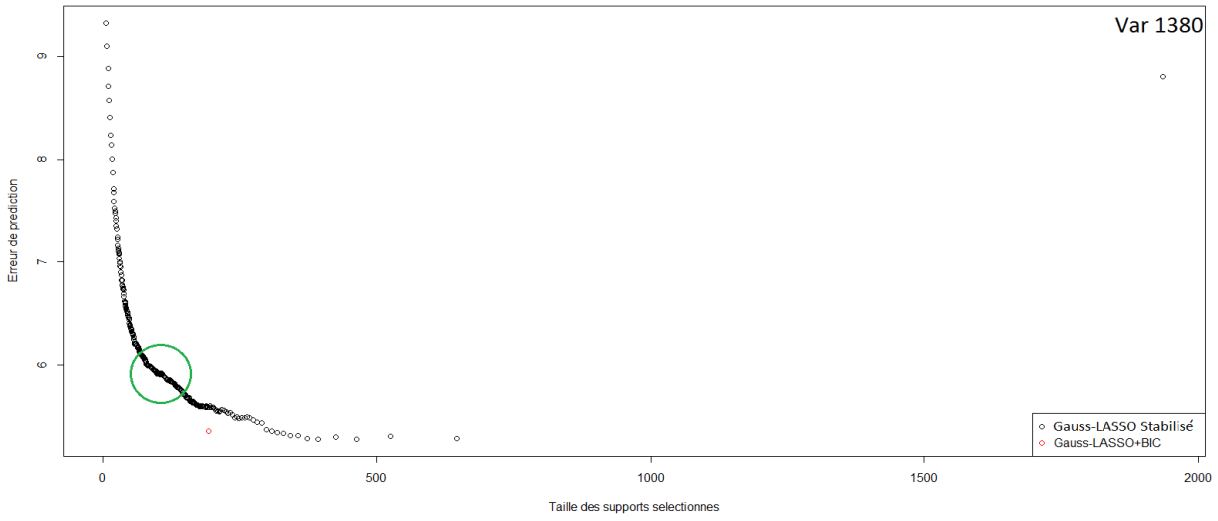

Figure 2.17 – Erreurs de prédiction VS Tailles des supports lorsque s varie - FT 1380

L'idéal serait de calibrer, pour chaque variable régressée, s de façon à respecter un bon compromis entre prédiction et parcimonie (zones entourées en vert sur les figures 2.16 et 2.17). Il reste à trouver un procédé permettant systématiquement de choisir un tel seuil.

2.2.3 Minimisation de l'erreur de prédiction avec écarts-types

Les erreurs de prédiction de Gauss-LASSO stabilisé sont minimales pour des seuils proches de 0. Dans le cadre de la régression de la variable j, on note le seuil minimal s_{Emin}, c'est-à-dire le seuil vérifiant :

$$s_{Emin} = \underset{s\in[0,1]}{\text{argmin}} Err_j^{CV}\left(\widehat{\Theta}_j^{GLstab}(s)\right)$$

. Le considérer comme seuil de référence n'a aucun sens. L'intérêt même du rééchantillonnage est d'estimer la stabilité de chacun des régresseurs potentiels. En autorisant dans les supports la présence de variables aux scores tout proches de 0, donc quasiment jamais sélectionnées parmi les m sous-supports estimés $\left\{{}^{(k)}\widehat{S}_j^{GL}(\lambda_{BIC}^j)\right\}_{k\in\{1,\dots,m\}}$, on munirait les supports de variables non pertinentes.

En revanche, en considérant le seuill $s_{\alpha.std}$ dont l'erreur de prédiction associée vérifie :

$$Err_j^{CV}\left(\widehat{\Theta}_j^{GLstab}(s_{\alpha.std})\right) = Err_j^{CV}\left(\widehat{\Theta}_j^{GLstab}(s_{Emin})\right) + \alpha\times\sigma\left(\left\{Err_j^{CV}\left(\widehat{\Theta}_j^{GLstab}(s)\right)\right\}_s\right)$$

c'est-à-dire l'erreur de prédiction minimale plus α fois l'écart-type de l'ensemble des erreurs, nous aspirons à ce que Gauss-LASSO stabilisé pour ce choix de seuil estime des supports permettant un bon compromis prédiction-parcimonie.

Procédures	Informations	V 1	V 935	V 1156	V 1380	V 1810
Gauss-LASSO stabilisé(s_{Emin})	Taille support	660	51	886	393	429
	Erreur de prédiction	5.20	2.98	4.22	5.27	4.53
	Seuil $s_{Err.min}$	0.006	0.003	0.003	0.015	0.012
Gauss-LASSO stabilisé($s_{1.std}$)	Taille support	189	9	177	90	97
	Erreur de prédiction	5.85	3.00	5.09	5.96	4.99
	Seuil $s_{1.std}$	0.18	0.22	0.465	0.48	0.38
Gauss-LASSO stabilisé($s_{2.std}$)	Taille support	59	7	99	40	47
	Erreur de prédiction	6.51	3.02	5.96	6.64	5.45
	Seuil $s_{2.std}$	0.64	0.372	0.79	0.843	0.71

Table 2.3 – Taille des supports pour des seuils choisis par la règle de l'erreur minimale + écarts-type

Selon le tableau 2.3, les tailles des supports estimés par Gauss-LASSO stabilisé ($s = s_{1.std}$) coïncident avec les zones vertes des figures 2.16 et 2.17 pour quatre variables typiques. Pour $s = s_{2.std}$ la procédure est plus parcimonieuse mais fournit des erreurs de prédiction trop dégradées. La procédure Gauss-LASSO stabilisé ($s = s_{1.std}$) doit nous

servir de procédure de référence. Malheureusement, le calcul des erreurs de prédiction est trop lourd pour pouvoir être réalisé en temps convenable sur les p variables.

Notre prochain objectif est donc d'obtenir une sélection stable permettant un bon compromis entre erreurs de prédiction et tailles de supports estimés. Pour ce faire, il nous faut trouver un moyen de calibrer s en ce sens mais sans passer par le calcul des erreurs de prédiction. Il s'agit de trouver une méthode systématique de calibration de s, donnant des seuils voisins de $s_{1.std}$ pour les 5 variables typiques et peu lourde pour pouvoir être appliquée à l'ensemble des p variables régressées.

Nous envisageons pour cela de nous appuyer sur la vraisemblance des modèles estimés, peu coûteuse à calculer. L'évolution de la log-vraisemblance maximisée en ses paramètres des modèles estimés par Gauss-LASSO stabilisé en fonction de s pourrait nous être utile.

2.3 Procédures de calibration du seuil basées sur la Log-Vraisemblance

Pour un noeud j, on a un panel de supports estimés par Gauss-LASSO stabilisé dépendant de s :

$$\mathcal{S}_j^{GLstab} = \left\{ \widehat{S}_j^{GLstab}(s) \right\}_{s\in[0,1]}.$$

Une collection de modèles de régression linéaire en découle :
$\mathcal{M}_j^{GLstab} = \left\{ M_j^{GLstab}(s) \right\}_{s\in[0,1]}$ où $M_j^{GLstab}(s)$ correspond à $X^j = X^{\widehat{S}_j^{GLstab}(s)}\Theta_j + \epsilon_j$.

Pour chaque modèle de $\mathcal{M}_j^{GLstab}$, on peut calculer la log-vraisemblance maximisée en ses paramètres sur l'ensemble des n observations. Pour un seuil s :

$$V_{max}\left(M_j^{GLstab}(s)\right) = \frac{n}{2}\left(log\left(\frac{n}{2\pi}\right) - 1 - log\left(\left\|X^j - X^{\widehat{S}_j^{GLstab}(s)}\widehat{\Theta}_j^{GLstab}(s)\right\|^2\right)\right)$$

où $\widehat{\Theta}_j^{GLstab}(s)$ est l'estimateur de Θ_j obtenu par moindres carrés ordinaires.

Nous disposons d'une collection de modèles $\mathcal{M}_j^{GLstab}$ estimés par notre procédure et dépendant de s. Nous allons exposer trois méthodes de calibration de s utilisant la log-vraisemblance des modèles de cette collection, la première par l'application d'un critère de vraisemblance pénalisé asymptotique (BIC), la seconde fondée sur la comparaison des courbes de vraisemblance de Gauss-LASSO et Gauss-LASSO stabilisé et la dernière par l'application d'un critère de vraisemblance pénalisé non asymptotique (l'heuristique de pente détaillée dans [8] et [9]). Dans tous les cas, ces trois méthodes impliquent que les seuils choisis seront différents selon la variable régressée étudiée. Nous noterons désormais s_j le seuil associé à la variable j.

2.3.1 Sélection du support par Gauss-LASSO stabilisé($\widehat{s}_{BIC}$)

Une démarche naturelle est de pénaliser ces log-vraisemblances dans le but de sélectionner un modèle de $\mathcal{M}_j^{GLstab}$, et par conséquent un seuil s_j.

On utilise dans un premier temps un critère asymptotique, à savoir BIC :

$$\widehat{s}_j^{BIC} = \underset{s\in[0,1]}{\text{argmin}} \left(-2V_{max}\left(M_j^{GLstab}(s)\right) + \left|\widehat{S}_j^{GLstab}(s)\right| \log(n)\right).$$

Cette procédure consistant à choisir le seuil de Gauss-LASSO stabilisé à l'aide du critère BIC sera appelée Gauss-LASSO stabilisé($\widehat{s}_{BIC}$). Appliquée à notre jeu de données réel, les p seuils $\{\widehat{s}_j^{BIC}\}_{j\in\{1,..,p\}}$ choisis par BIC sont en moyenne proches de 0.5 (figure 2.18). Gauss-LASSO stabilisé($\widehat{s}_{BIC}$) préconise donc des sélections moins drastiques que Gauss-LASSO stabilisé pour $s = 1$ ou 0.97 mais plus sévères que Gauss-LASSO + BIC.

La procédure Gauss-LASSO stabilisé($\widehat{s}_{BIC}$) semble être une procédure de référence pour nous :

1. elle réduit, avec un rapport de 2, les p supports estimés par Gauss-LASSO+BIC (cf. tableau 2.1 et figure 2.19), tout en ne détériorant que très peu ses erreurs de prédiction sur les cinq variables typiques (cf tableau 2.2)
2. le rapport entre taille des unions et des intersections des supports estimés par la procédure et Gauss-LASSO+BIC vaut en moyenne 2 également (cf. figure 2.20), ce qui implique que les variables de $\widehat{S}_j^{GLstab}(\widehat{s}_j^{BIC})$ sont en partie dans $\widehat{S}_j^{GL}(\lambda_j^{BIC})$.
3. L'objectif qui est d'obtenir des résultats proches de Gauss-LASSO stabilisé($s = s_{1.std}$) en termes d'erreurs de prédiction et de tailles de supports estimés, pour les cinq variables typiques, est rempli (cf. tableaux 2.3 et 2.2).
4. L'utilisation du critère BIC semble bien adapté ici. L'allure des courbes BIC, lisses et au minimum net (cf. figure 2.21) pour quatre variables typiques, nous pousse à lui faire confiance.

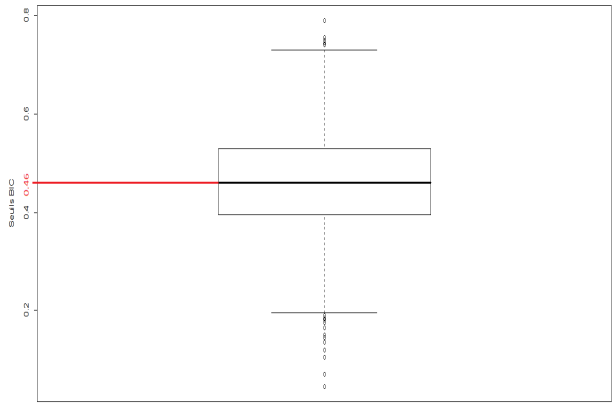

Figure 2.18 – Boxplot des p seuils $\widehat{s}_j^{BIC}$

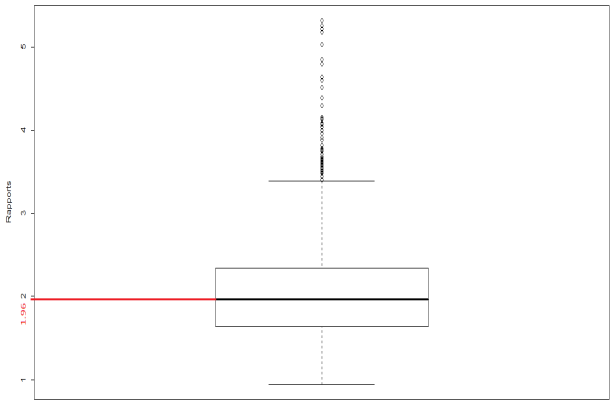

Figure 2.19 – Boxplot des p rapports $\frac{\left|\widehat{S}_j^{GL}(\lambda_j^{BIC})\right|}{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{BIC})\right|}$

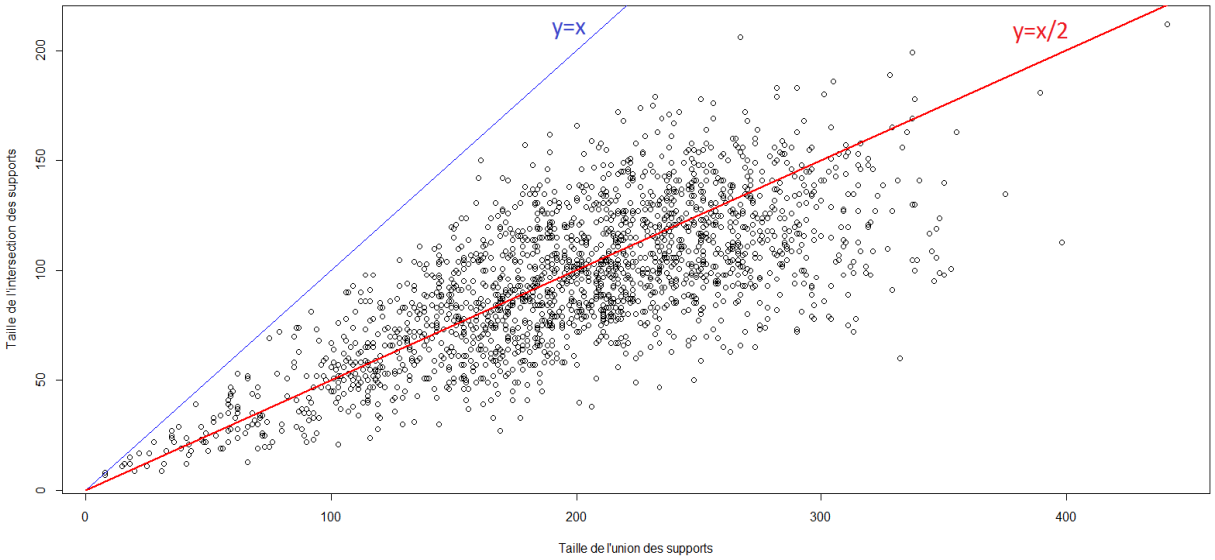

Figure 2.20 – Taille de l'intersection en fonction de la taille de l'union des supports Gauss-LASSO+BIC et Gauss-LASSO stabilisé($\widehat{s}_{BIC}$)

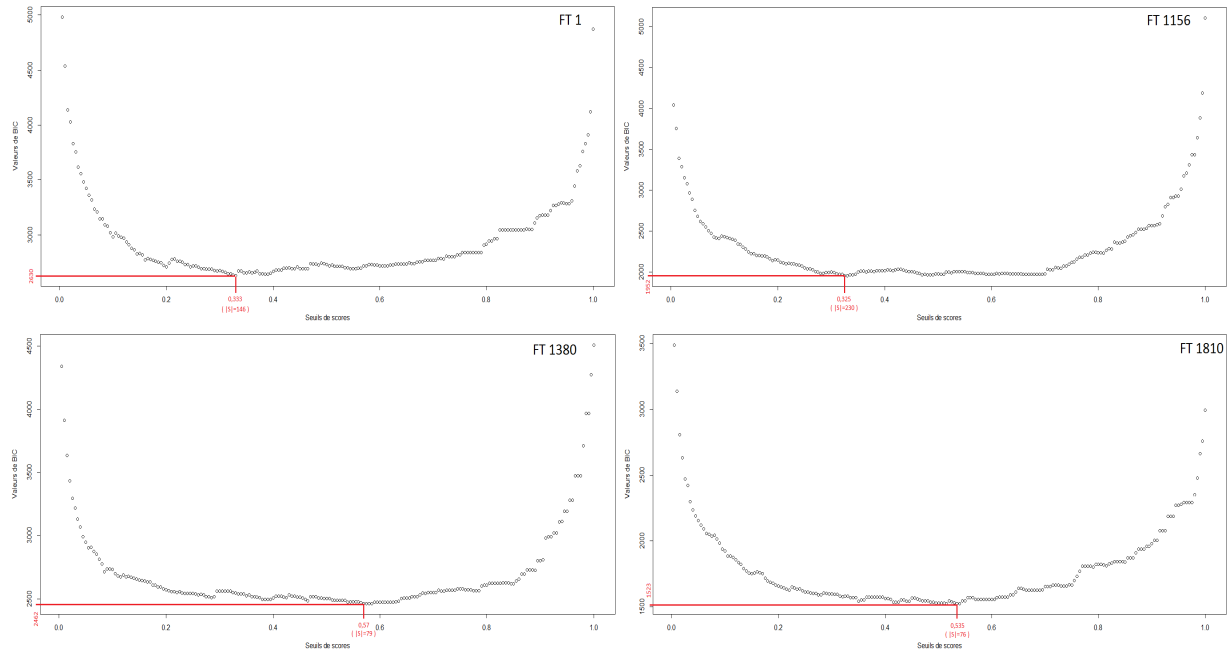

Figure 2.21 – Evolution du critère BIC des modèles de Gauss-LASSO stabilisé(s) en fonction del s

Cette méthode de calibration de s nous sert de référence. Une fois les autres méthodes de calibration exposées, nous garderons celle proposant les supports les plus stables.

2.3.2 Sélection du support par Gauss-LASSO stabilisé($\widehat{s}_{HP}$)

La seconde méthode de calibration que nous testons consiste en l'application de l'heuristique de pente (HP) ([8] et [9]). Le choix de l'heuristique de pente comme critère de

vraisemblance pénalisée est motivé par son caractère non asymptotique lui permettant d'adapter la pénalité de la régression linéaire à la collection de modèles étudiée. En théorie, lorsqu'un modèle est bien adapté à un jeu de données, son biais tend vers 0 lorsque la taille du jeu tend vers l'infini. Dans le cas d'un modèle jugé grossier, ce qui est le cas du nôtre, elle a la vertu de compenser le fort biais du modèle, à condition que celui-ci se stabilise au voisinage de l'infini. Dans notre cadre d'estimation des supports, cette condition se traduit par le caractère linéaire de la courbe de log-vraisemblance d'une collection de modèles étudiée que l'on détecte pour des modèles de grande dimension. La pente de cette partie linéaire sera notée κ_j pour la variable j.

Le seuil choisi par cette méthode de calibration vérifiera alors :

$$\widehat{s}_j^{HP} = \underset{s\in[0,1]}{\operatorname{argmin}} \left(-V_{max}\left(M_j^{GLstab}(s)\right) + 2 \times \kappa_j \times \left|\widehat{S}_j^{GLstab}(s)\right| \log(n) \right).$$

Cette procédure consistant à estimer le seuil de Gauss-LASSO stabilisé par le biais de l'heuristique de pente sera appelée Gauss-LASSO stabilisé ($\widehat{s}_{HP}$). Sur notre jeu de données, les p seuils $\{\widehat{s}_j^{HP}\}_{j\in\{1,..,p\}}$ choisis valent en moyenne 0.51 (figure 2.23). Ses résultats sont, en conséquence, très proches d'une sélection du seuil à l'aide de BIC en termes de taille de supports estimés et d'erreurs de prédiction (cf. tableaux 2.2 et 2.1).

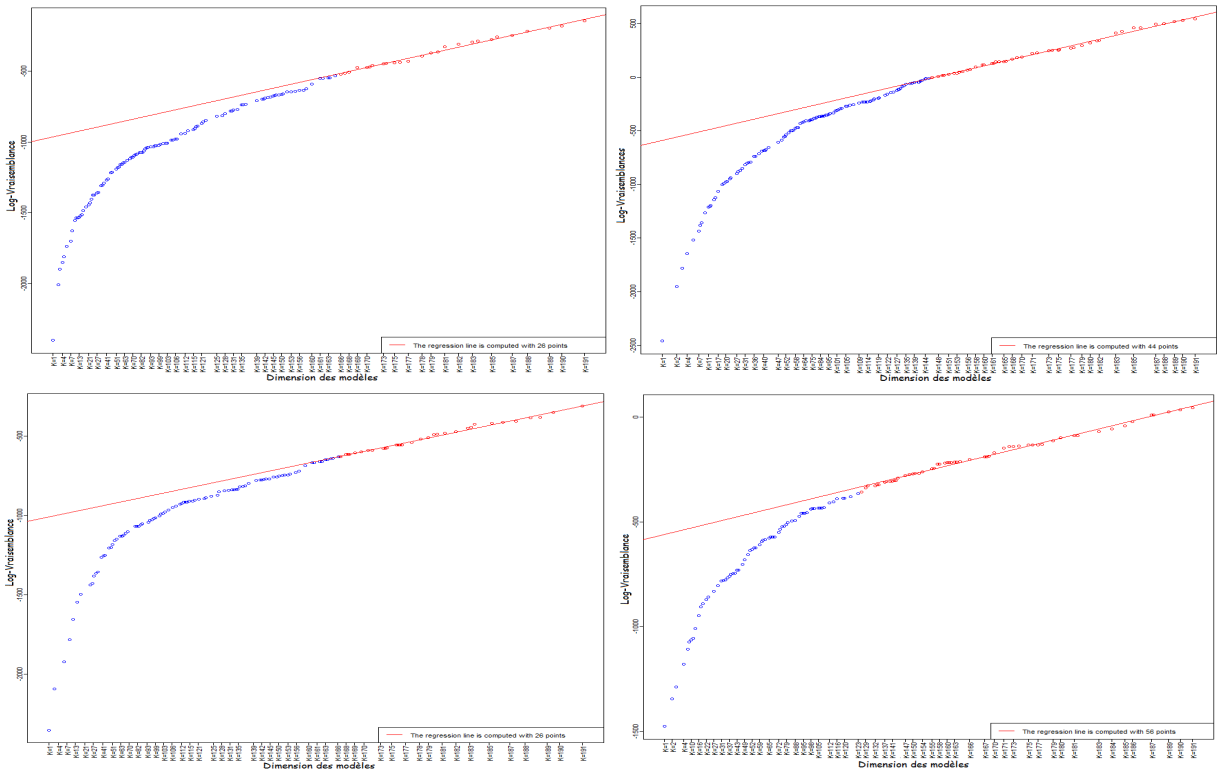

Figure 2.22 – Détection par HP de la partie linéaire de $V_{max}\left(M_j^{GLstab}(s)\right)$ en fonction de la dimension des modèles pour variables typiques

La procédure Gauss-LASSO stabilisé($\widehat{s}_{HP}$) semble également être une procédure de référence pour nous :

1. Ses résultats étant similaires à Gauss-LASSO ($\widehat{s}_{BIC}$), les comparaisons de ses supports estimés avec ceux estimés par Gauss-LASSO + BIC affichent les mêmes constats (cf. figures 2.24 et 2.25).
2. L'allure des courbes de log-vraisemblances qui croissent de manière constante à partir d'une certaine dimension de modèle permettant la détection d'une pente nette (cf. figure 2.22) pour quatre variables typiques, nous pousse à lui faire confiance.

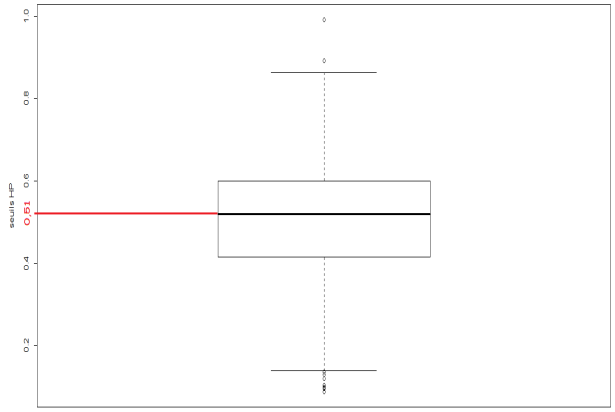

Figure 2.23 – Boxplot des p seuils $\widehat{s}_j^{HP}$

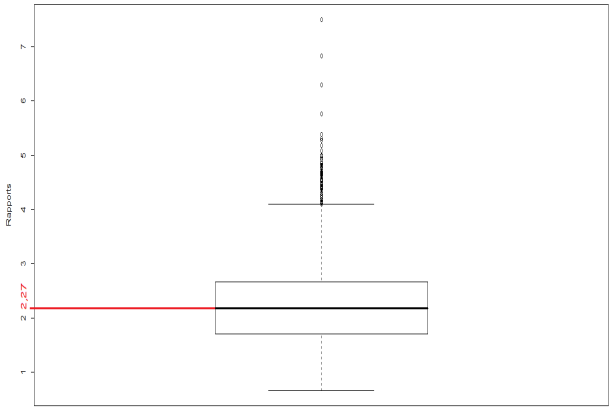

Figure 2.24 – Boxplot des p rapports $\frac{\left|\widehat{S}_j^{GL}(\lambda_{BIC}^j)\right|}{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{HP})\right|}$

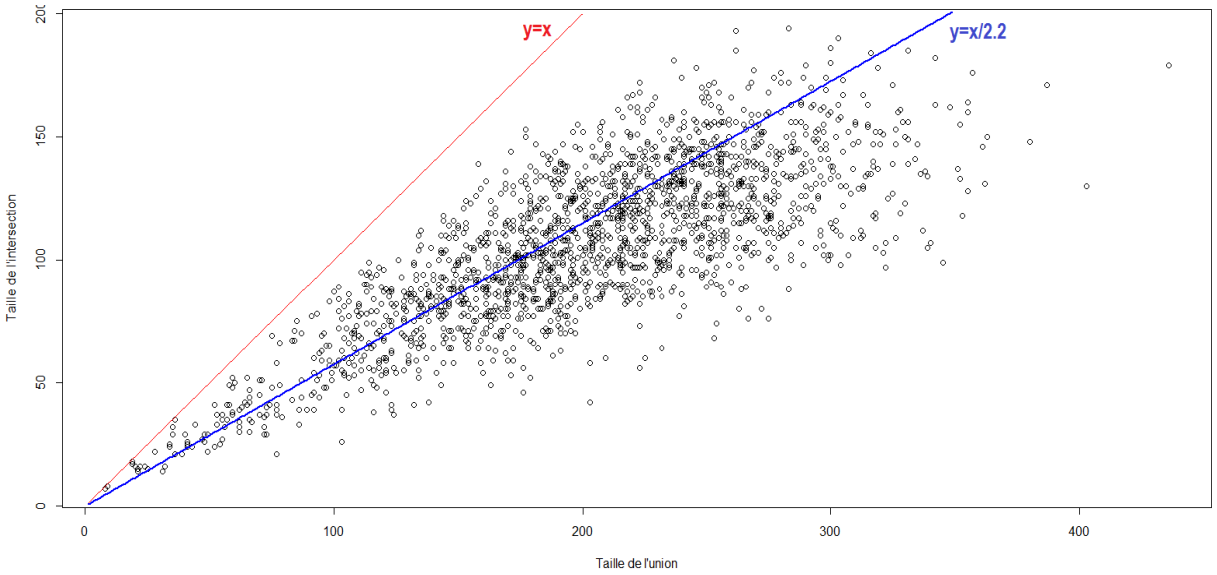

Figure 2.25 – Taille de l'intersection en fonction de la taille de l'union des supports Gauss-LASSO+BIC et Gauss-LASSO stabilisé($\widehat{s}_{HP}$)

2.3.3 Sélection du support par Gauss-LASSO stabilisé($\widehat{s}_{inter}$)

Pour diversifier les méthodes de calibration du seuil, nous avons cherché à en établir une fondée sur la log-vraisemblance des modèles de la collection $\mathcal{M}_j^{GLstab}$, sans avoir recours à de la régression pénalisée.

La log-vraisemblance d'un modèle permet d'estimer la concordance du modèle avec le jeu de données étudié. Le recours au rééchantillonnage doit stabiliser les supports. Nous sommes en droit de penser que, dans le cadre de la régression d'une variable j, les supports estimés par Gauss-LASSO stabilisé possèdent à dimension égale moins de variables sélectionnées à tort que ceux estimés par Gauss-LASSO. Nous aspirons en cela à ce que les modèles de la collection $\mathcal{M}_j^{GLstab}$ estimés par Gauss-LASSO stabilisé aient, à dimension égale, une vraisemblance maximisée en ses paramètres plus importante que les modèles de la collection $\mathcal{M}_j^{GL}$ estimés par Gauss-LASSO. Notons que les vraisemblances des modèles de ces deux collections sont comparables puisque nous les calculons sur le jeu $\mathcal{I}$ entier. Vérifions cette conjecture.

Pour tout modèle $M_j^{GL}(\lambda) \in \mathcal{M}_j^{GL}$, sa log-vraisemblance maximisée en ses paramètres s'écrit de cette manière :

$$V_{max}\left(M_j^{GL}(\lambda_j)\right) = \frac{n}{2}\left(log\left(\frac{n}{2\pi}\right) - 1 - log\left(\left\|X^j - X^{\widehat{S}_j^{GL}(\lambda_j)}\widehat{\Theta}_j^{GL}(\lambda_j)\right\|^2\right)\right)$$

où $\widehat{\Theta}_j^{GL}(\lambda_j)$ est l'estimateur de Θ_j obtenu par moindres carrés ordinaires.

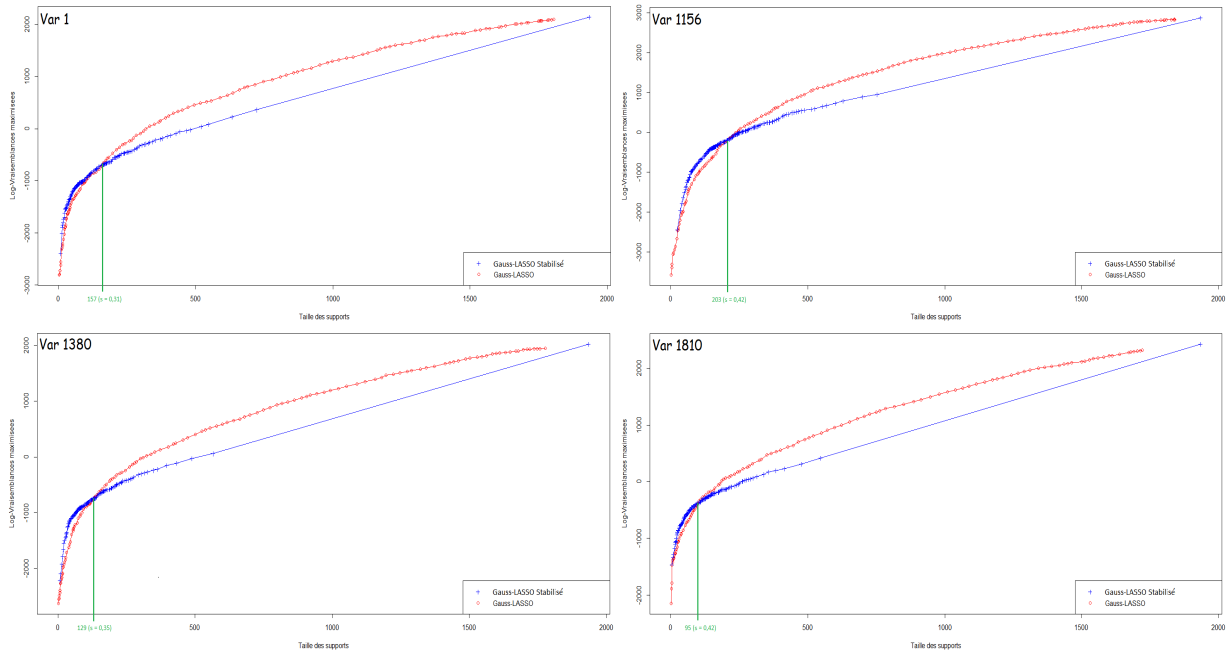

Figure 2.26 – Comparaison de l'évolution de $V_{max}\left(M_j^{GLstab}(s_j)\right)$ et de $V_{max}\left(M_j^{GL}(\lambda_j)\right)$ en fonction de la taille des support associés aux modèles pour 4 des variables typiques

Le comportement de ces log-vraisemblances (cf. figure 2.3.3) est systématique pour les p variables régressées. Les modèles associés aux seuils s ou aux pénalités λ de dimensions $\left|\widehat{S}_j^{GLstab}(s_j)\right|$ et $\left|\widehat{S}_j^{GL}(\lambda_j)\right|$ identiques et petites, vérifient :

$$V_{max}\left(M_j^{GL}(\lambda_j)\right) < V_{max}\left(M_j^{GLstab}(s_j)\right)$$

Puis à partir d'une certaine dimension associée à un seuil et une pénalité que l'on notera $\widehat{s}_j^{inter}$ et $\widehat{\lambda}_j^{inter}$ tels que $\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{inter})\right| = \left|\widehat{S}_j^{GL}(\widehat{\lambda}_j^{inter})\right|$, la tendance s'inverse.

La cohérence de ces courbes est due au fait que l'esprit de ces deux méthodes de sélection est différent. Pour une dimension de modèle $\alpha \in \mathbb{N}^*$, Gauss-LASSO sélectionne le groupe de $\left|\widehat{S}_j^{GL}(\lambda_j)\right| = \alpha$ variables les plus corrélées avec la variable expliquée sous contrainte de proximité entre ces variables, tandis que Gauss-LASSO stabilisé sélectionne lui les $\left|\widehat{S}_j^{GLstab}(s_j)\right| = \alpha$ variables les plus stables puisqu'elle classe les variables explicatives selon la probabilité qu'elles ont d'être liées avec la variable régressée.

Ainsi, pour des dimensions α proches de 0, les variables sélectionnées par Gauss-LASSO stabilisé ont des scores très élevés et forment un support dont le modèle de $\mathcal{M}_j^{GLstab}$ a une vraisemblance maximisée en ses paramètres plus élevée que le modèle associé au support formé des variables sélectionnée par Gauss-LASSO. En revanche, plus α est grand, plus Gauss-LASSO stabilisé munit ses supports de variables au score faible. Au contraire, Gauss-LASSO réactualise ses supports en rejetant des variables si nécessaire, de façon à obtenir le groupe de α variables le plus corrélé avec la variable expliquée au vu du jeu de données utilisé.

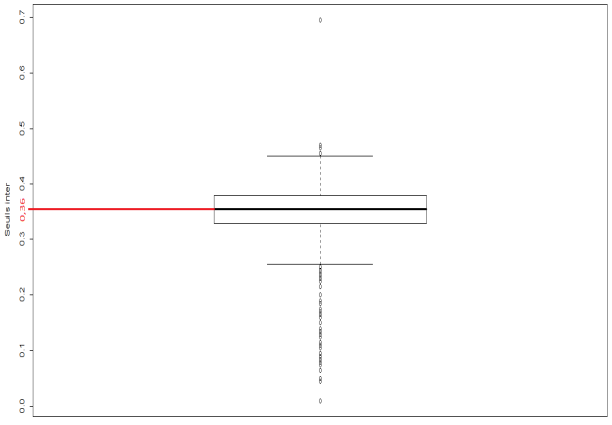

Figure 2.27 – Boxplot des p seuils $\widehat{s}_j^{inter}$

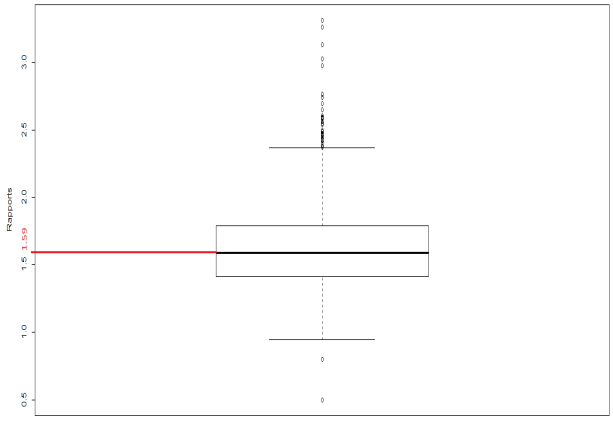

Figure 2.28 – Boxplot des p rapports $\frac{\left|\widehat{S}_j^{GL}(\lambda_j^{BIC})\right|}{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{inter})\right|}$

Appliquée à nos données, les p seuils $\{\widehat{s}_j^{inter}\}_{j\in\{1,..,p\}}$ sont en moyenne proches de 0.35 (cf. figure 2.27), ce qui procure des sélections moins sévères et en des erreurs de prédiction légèrement meilleures que par le biais de critères pénalisés (cf. tableaux 2.2 et 2.1).

Néanmoins, Gauss-LASSO stabilisé($\widehat{s}_{inter}$) est une procédure de sélection correspondant à nos attentes, sous réserve qu'elles sélectionne des supports stables :

1. Elle épure tout de même les supports estimés par Gauss-LASSO+BIC tout en ne détériorant que très peu ses erreurs de prédiction (rapport valant environ 1.6 en moyenne sur les p variables).
2. En moyenne, le rapport entre taille des unions et des intersections des supports estimés par la procédure et Gauss-LASSO+BIC (cf. figure 2.29) correspond quasiment au rapport entre les tailles de supports estimés par ces deux procédures (cf. figure 2.28), ce qui implique que les variables de $\widehat{S}_j^{GLstab}(\widehat{s}_j^{inter})$ sont en grande partie dans $\widehat{S}_j^{GL}(\lambda_j^{BIC})$.
3. L'objectif qui est d'obtenir des résultats proches de Gauss-LASSO stabilisé($s = s_{1.std}$) en termes d'erreurs de prédiction et de tailles de supports estimés, pour les cinq variables typiques, est rempli (cf. tableaux 2.3 et 2.2).

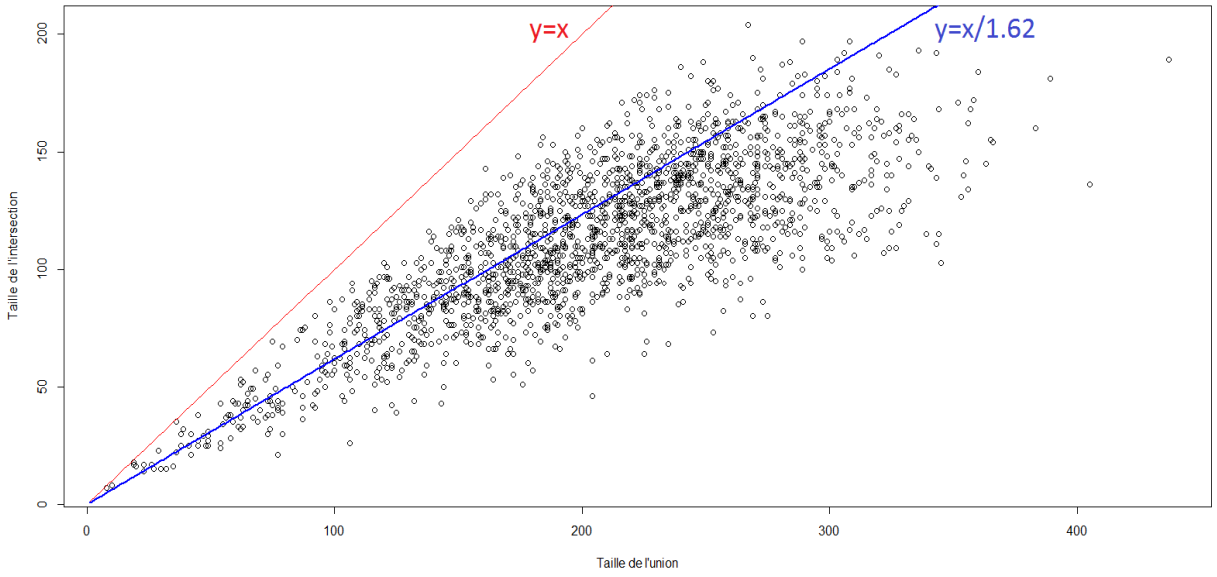

Figure 2.29 – Taille de l'intersection en fonction de la taille de l'union des supports Gauss-LASSO+BIC et Gauss-LASSO stabilisé ($\widehat{s}_{inter}$)

La calibration du seuil par intersection des vraisemblances pourrait tout comme la calibration à l'aide de critères pénalisés nous servir de méthode de référence. Il s'agira de s'en assurer en étudiant la stabilité des supports estimés par Gauss-LASSO stabilisé calibré avec cette méthode.

2.3.4 Comparaisons des méthodes de calibration de s

Les supports estimés par Gauss-LASSO stabilisé pour les trois types de sélection de seuils $(\widehat{s}_{BIC}, \widehat{s}_{HP}, \widehat{s}_{inter})$ ont un ordre grandeur similaire (environ une centaine de variables sélectionnées dans les supports estimés par Gauss-LASSO stabilisé pour les trois choix de seuils). De plus, la croissance linéaire affichée du nuage de points représenté figure 2.31 témoigne de la bonne cohésion entre les supports estimés par trois méthodes de calibration de s à savoir que plus une variable a un support estimé selon Gauss-LASSO stabilisé$(\widehat{s}_{BIC})$ conséquent, plus son support suivant Gauss-LASSO stabilisé$(\widehat{s}_{HP})$ ou $(\widehat{s}_{inter})$ le sera également.

Néanmoins, Gauss-LASSO stabilisé $(\widehat{s}_{BIC})$ et $(\widehat{s}_{HP})$ estiment des supports, en moyenne sur les p variables régressées, de taille légèrement inférieure à ceux estimés par Gauss-LASSO stabilisé$(\widehat{s}_{inter})$ (cf. tableau 2.1 et figure 2.30). Ceci est également appuyé par les boites à moustache (figure 2.32) qui illustrent le fait que la majorité des variables régressées ont des supports estimés par Gauss-LASSO stabilisé $(\widehat{s}_{BIC})$ et $(\widehat{s}_{HP})$ de taille très proche tandis que leurs supports estimés par Gauss-LASSO stabilisé $(\widehat{s}_{inter})$ sont de tailles plus grands. La proximité entre les seuils estimés par BIC et par l'heuristique de pente est également appuyée par la figure 2.33 où chaque point du nuage représente une variable j et a pour coordonnées $(\widehat{s}_j^{BIC}, \widehat{s}_j^{HP}, \widehat{s}_j^{inter})$. On y distingue la dépendance linéaire apparente entre les seuils $\widehat{s}_{BIC}$ et $\widehat{s}_{HP}$ alors qu'aucune structure particulière ne se distingue dans la comparaison des seuils $\widehat{s}_j^{HP}$ et $\widehat{s}_j^{inter}$ ou $\widehat{s}_j^{BIC}$ et $\widehat{s}_j^{inter}$.

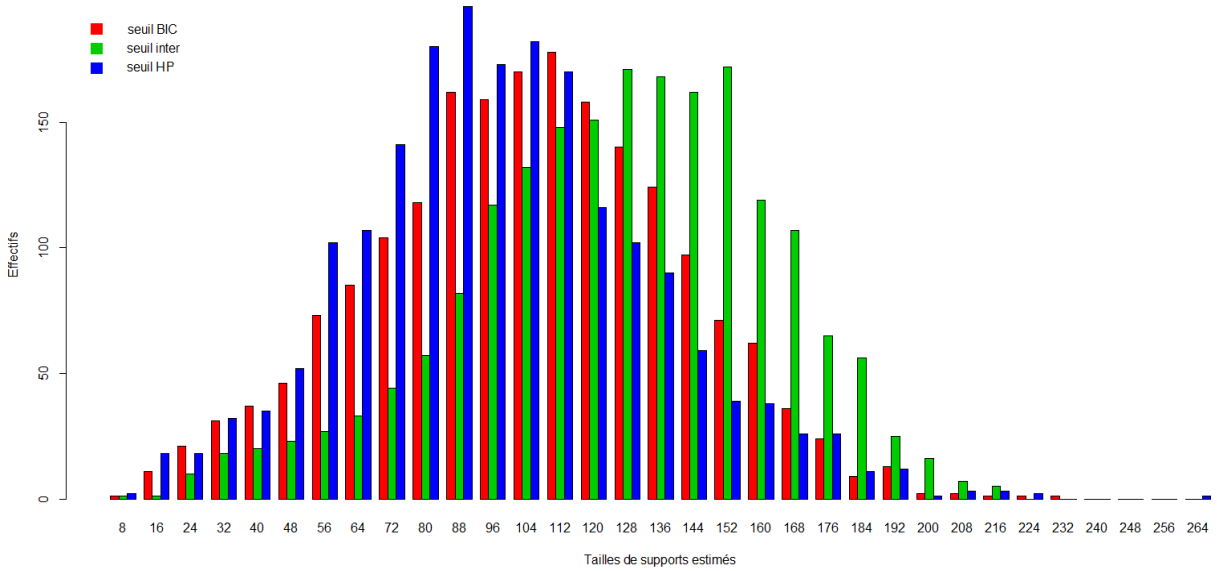

Figure 2.30 – Répartition des tailles des p supports estimés par Gauss-LASSO stabilisé pour les trois méthodes de calibration de s

En outre, pour une variable régressée j, les supports associés à la collection de modèles $\mathcal{M}_j^{GLstab}$ sont emboîtés. En effet, dans le cadre de la régression d'une variable j, si deux seuils vérifient $s_j^1 < s_j^2$, les modèles $M_j^{GLstab}(s_j^1)$ et $M_j^{GLstab}(s_j^2)$ ont leurs supports estimés associés qui vérifient :

$$\widehat{S}_j^{GLstab}(s_j^2) \subset \widehat{S}_j^{GLstab}(s_j^1)$$

Cela implique que les supports estimés par Gauss-LASSO stabilisé, pour une variable j et avec les seuils calibrés par BIC et HP $\left(\widehat{S}_j^{GLstab}(s_j^{BIC}), \widehat{S}_j^{GLstab}(s_j^{HP})\right)$ sont du fait de tailles très proches et de contenus quasiment similaires. De plus, la majorité des variables régressées vérifient, de la même manière $\widehat{S}_j^{GLstab}(s_j^{BIC}) \subset \widehat{S}_j^{GLstab}(s_j^{inter})$ et $\widehat{S}_j^{GLstab}(s_j^{HP}) \subset \widehat{S}_j^{GLstab}(s_j^{inter})$.

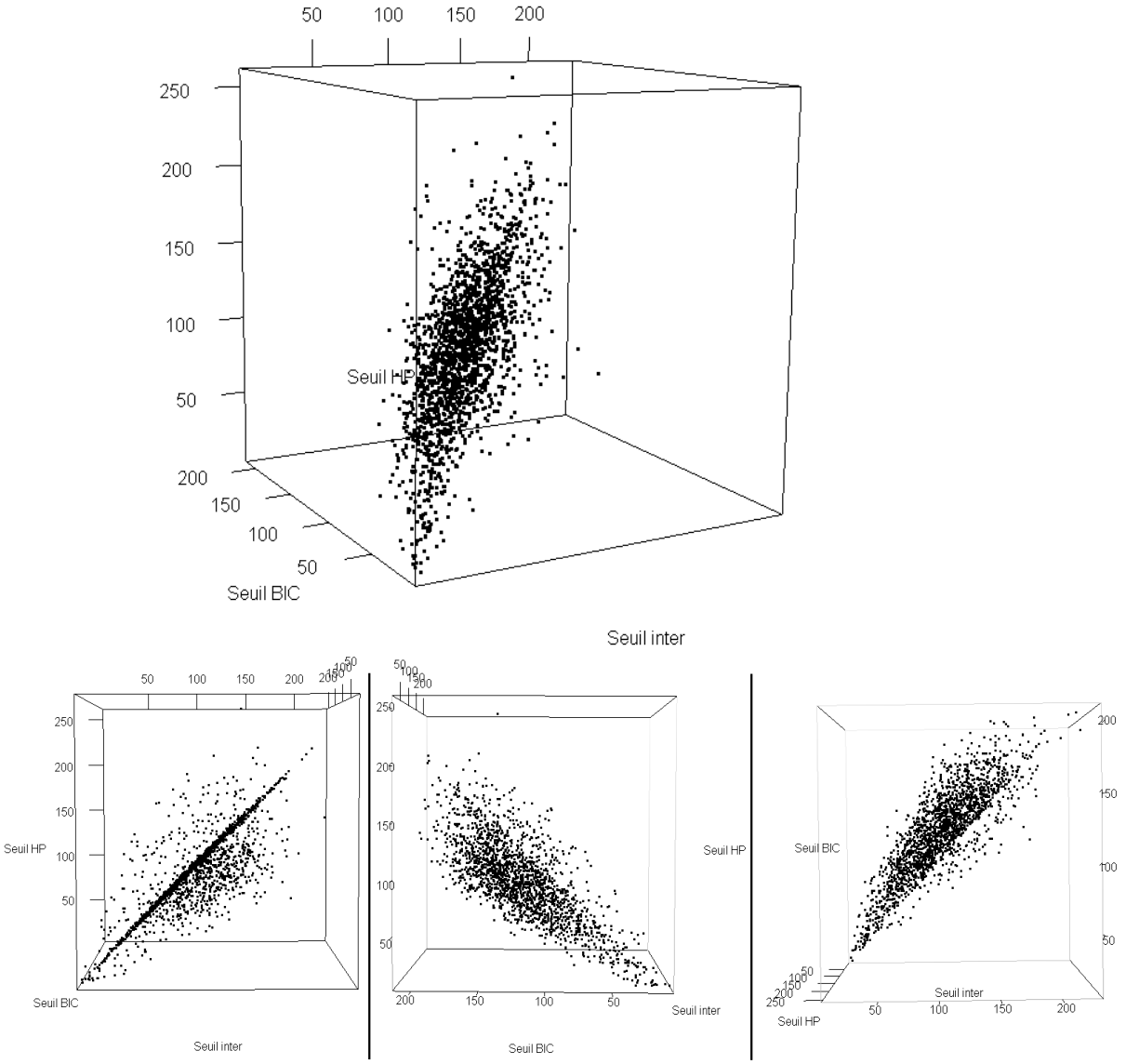

Figure 2.31 – Répartition des tailles de supports estimés par Gauss-LASSO stabilisé pour les trois seuils

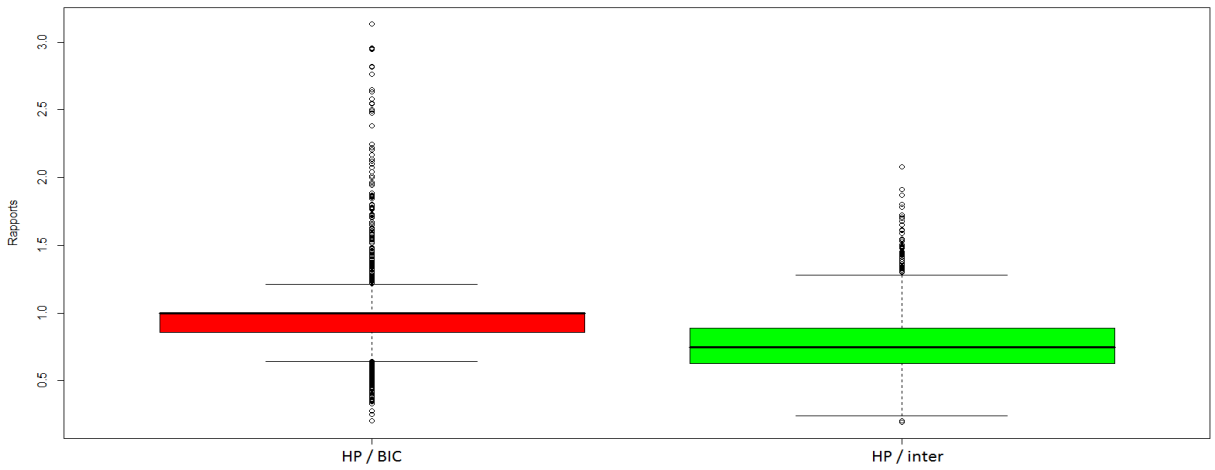

Figure 2.32 – Boxplots des p rapports $\frac{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{HP})\right|}{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{BIC})\right|}$ et $\frac{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{HP})\right|}{\left|\widehat{S}_j^{GLstab}(\widehat{s}_j^{inter})\right|}$

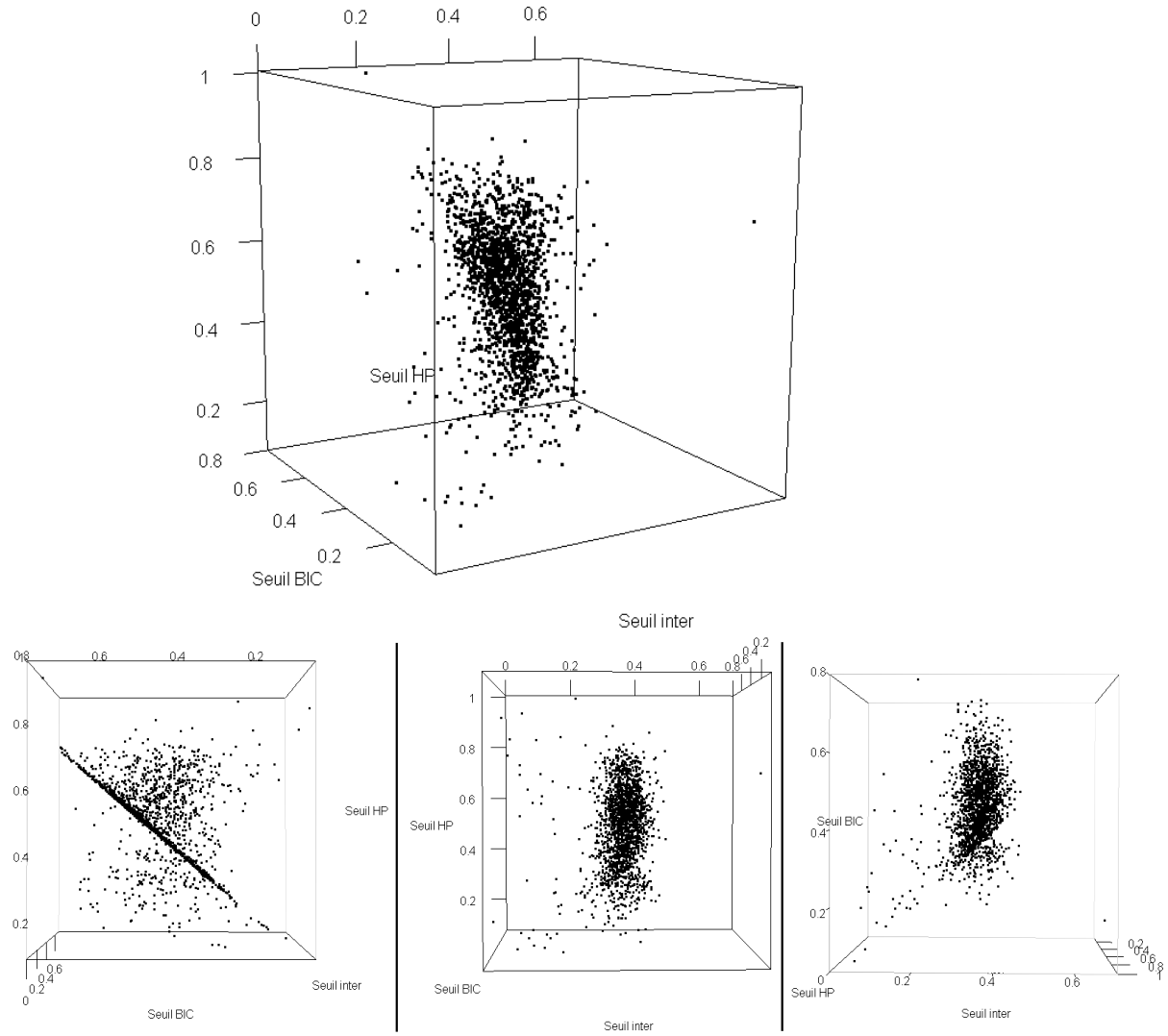

Figure 2.33 – Répartition des trois seuils sélectionnés pour chaque variable

En résumé, Gauss-LASSO stabilisé ($\widehat{s}_{BIC}$) et Gauss-LASSO stabilisé($\widehat{s}_{HP}$) munissent, en moyenne, les supports des variables explicatives sélectionnées plus de 50% du temps sur les m sous-modèles de régression issus du Sample Splitting. Gauss-LASSO stabilisé($\widehat{s}_{inter}$) sélectionne en moyenne les mêmes variables agrémentées de quelques variables aux scores plus faibles, c'est-à-dire sélectionnées moins d'une fois sur deux sur ces m sous-modèles. Le risque encouru est que ces variables en surplus soient sélectionnées à tort. L'application de Gauss-LASSO stabilisé sur les jeux de données simulés créés lors de la partie 1.3 pourrait mettre en lumière l'instabilité de ces variables ajoutées et nous faire préférer une méthode de calibration de s.

2.4 Evaluation de la stabilité de la procédure

Cette étude a pour fin de choisir la méthode de calibration de s parmi les trois qui sera jugée la plus stable. Le graphe $\mathcal{G}$ modélisant notre réseau de gènes sera construit à partir des supports estimés par cette procédure.

Nous allons évaluer la stabilité de Gauss-LASSO stabilisé à l'aide des 10 jeux de donnés simulés qui avaient été générés pour l'évaluation de la stabilité de Gauss-LASSO + BIC dans la partie 1 du chapitre. Appliqué à ces dix jeux de données simulés, les supports estimés par Gauss-LASSO stabilisé pour nos cinq variables typiques et pour chaque méthode de calibration de seuil sont beaucoup moins hétérogènes que ceux estimés par Gauss-LASSO+BIC. (cf. figure 2.34) De plus, le tableau des indices de Jaccard (cf. tableau 2.4) correspondant, pour chaque variable typique et pour chaque méthode de calibration de s, à la taille de l'intersection des 11 supports estimés par Gauss-LASSO stabilisé divisé par la taille de leur union donne un léger avantage à Gauss-LASSO stabilisé($\widehat{s}_{HP}$) et Gauss-LASSO stabilisé($\widehat{s}_{BIC}$) sur la méthode de calibration par intersection des vraisemblances. Ceci tend à appuyer le fait que, en moyenne sur l'ensemble des p variables, les quelques variables explicatives acceptées par Gauss-LASSO stabilisé ($\widehat{s}_{inter}$) dans les supports mais rejetées par les deux autres méthodes de calibration de s ne sont pas stables et sont certainement sélectionnées à tort.

C'est pourquoi nous décidons de privilégier Gauss-LASSO stabilisé avec calibration du seuil par le biais d'un deux critères pénalisés. Entre ces deux méthodes de calibration, l'étude de la stabilité de la procédure ne les départageant pas spécialement, nous décidons de privilégier la calibration par l'heuristique de pente de par son caractère non asymptotique. Ainsi, l'estimation des pénalités LASSO dans les m sous-modèles de régression se fait avec BIC, tandis que le seuil s est choisi avec l'heuristique de pente. Nous conservons donc comme procédure de référence Gauss-LASSO stabilisé ($\widehat{s}_{HP}$). Nous appellerons par la suite de manière abusive cette procédure Gauss-LASSO stabilisé.

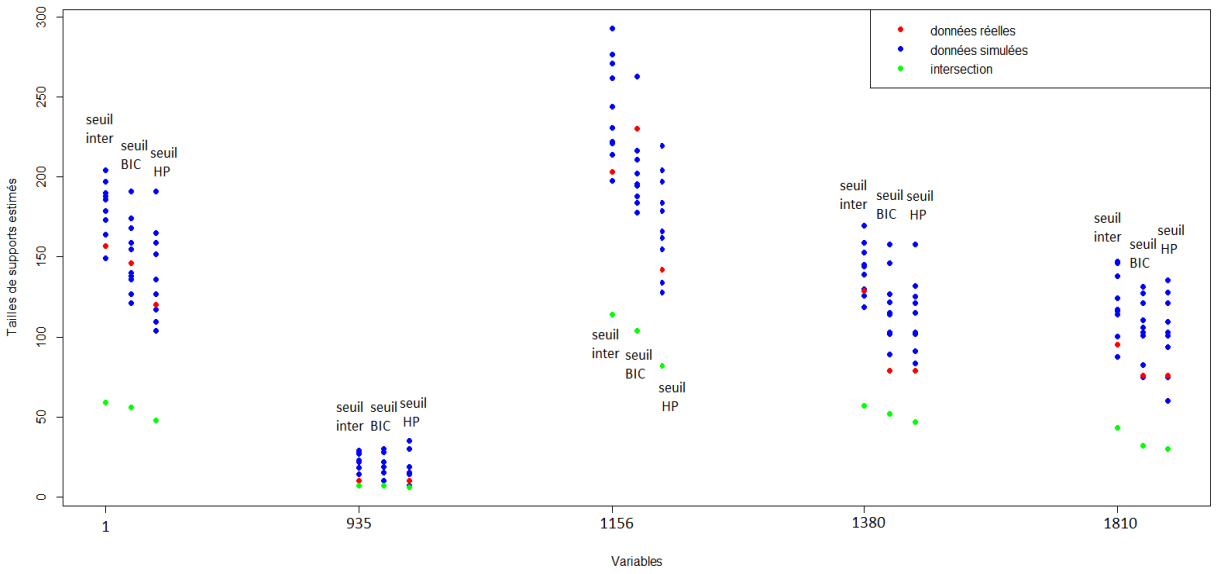

Figure 2.34 – Comparaison des supports estimés par Gauss-LASSO stabilisé sur le jeu réel et les simulations

Seuils	Informations	V 1	V 935	V 1156	V 1380	V 1810
$\widehat{s}_{inter}$	|Support réel|	157	10	203	129	95
	|Support simulé| moyen	184	25	252	140	122
	Indice de Jaccard	0.152	0.117	0.259	0.210	0.160
$\widehat{s}_{BIC}$	|Support réel|	146	7	230	79	76
	|Support simulé| moyen	156	21	208	123	97
	Indice de Jaccard	0.168	0.137	0.282	0.215	0.153
$\widehat{s}_{HP}$	|Support réel|	120	10	142	79	76
	|Support simulé| moyen	139	15	164	115	89
	Indice de Jaccard	0.162	0.143	0.277	0.217	0.153

Table 2.4 – Comparaison des supports estimés par Gauss-LASSO stabilisé sur les jeux rééls et simulés

2.5 Extrapolation à des jeux de donnés de petite taille

Le travail fourni se situe dans un cadre statistique où $n \simeq p$ et même, dans notre cas de figure, n est légèrement supérieur à p. Nous avons réduit le nombre d'observations dont nous disposons à $n/4$ pour évaluer la pertinence de Gauss-LASSO stabilisé dans ce cadre. Appliqué à une matrice de données de dimension $n/4 \times p$, restriction de notre matrice de données réelle X, nous pouvons constater qu'une procédure de rééchantillonnage telle que Gauss-LASSO stabilisé n'est pas satisfaisante dans ce cadre statistique où $n << p$.

En effet, la répartition des scores issus de la procédure appliquée aux variables typiques régressées (cf. figure 2.35) est beaucoup plus hétérogène que dans le cas où l'on prend en

compte les n observations. De plus, il n'y a quasiment pas de variables aux scores proche de 1 contrairement à l'étude réalisée sur les n observations. Gauss-LASSO stabilisé sanctionne donc beaucoup moins sévèrement les variables non pertinentes et aura tendance, quel que soit le choix de s à sélectionner des variables instables. Ceci est illustré par la figure 2.36, où l'on s'aperçoit que pour les variables typiques, la log-vraisemblance des modèles de la collection $\mathcal{M}_j^{GL}$ est même plus importante que celle des modèles de la collection $\mathcal{M}_j^{GLstab}$ pour des modèles de petite dimension.

Or, Gauss-LASSO stabilisé a pour fin première de sélectionner les variables les plus stables. Comme précédemment pour la même étude où les n observations sont prises en compte, les supports estimés par Gauss-LASSO stabilisé devraient être associés à des modèles de petite dimension de vraisemblance maximisée en ses paramètres plus importante que les modèles estimés par Gauss-LASSO+BIC, matérialisant la stabilité des variables qu'elle sélectionne. Ceci n'est pas le cas ici, ce qui semble indiquer que le rééchantillonnage n'est pas une solution pour stabiliser la sélection de variables dans un cadre statistique où $n << p$.

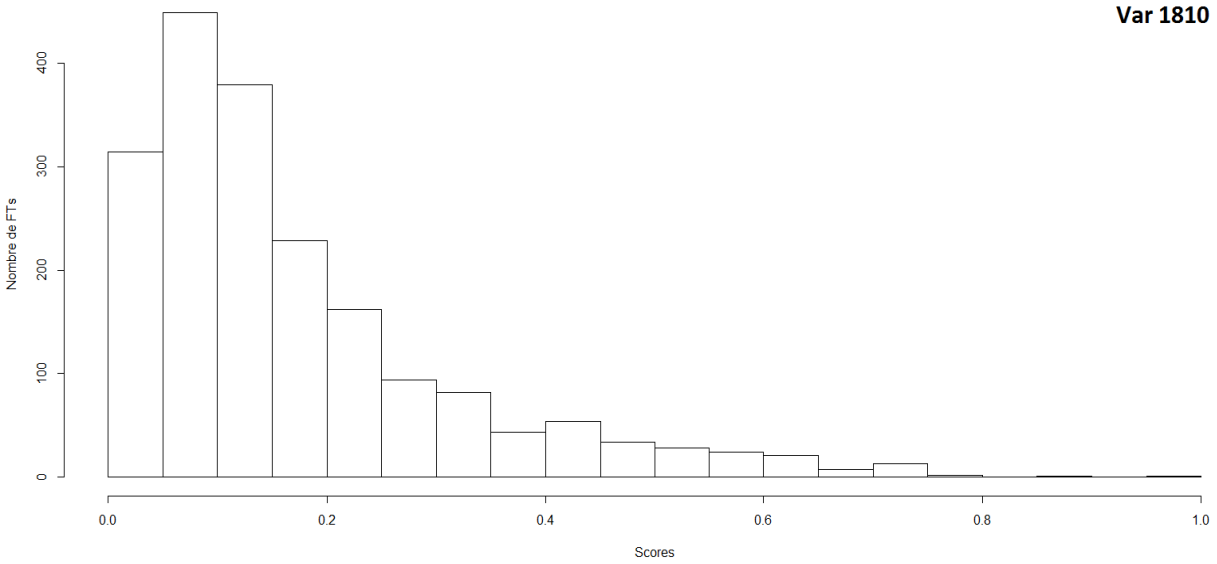

Figure 2.35 – Répartition des scores de chaque variable explicative à la variable typique 1810 régressée pour $n/4$ observations

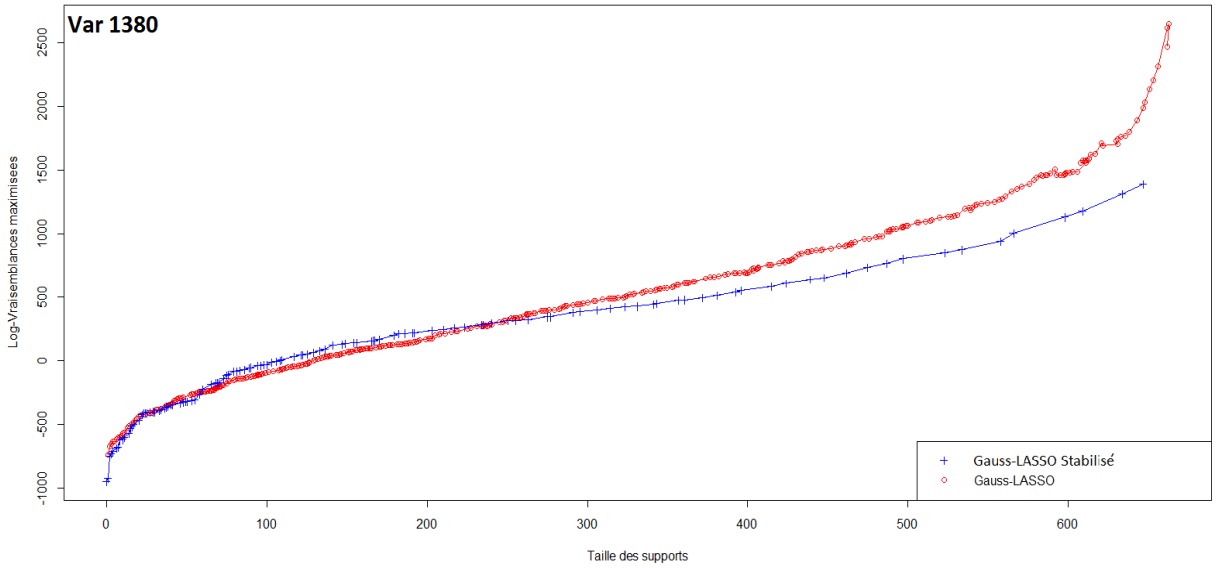

Figure 2.36 – Comparaison des vraisemblances des modèles estimés par Gauss-LASSO et Gauss-LASSO stabilisé pour une variable typique avec un nombre d'observations égal à $n/4$

3 Gauss-LASSO enrichi

La procédure Gauss-LASSO stabilisée nous satisfait en termes de parcimonie et de stabilité des supports estimés. Cependant, elle possède un défaut dans sa construction. Contrairement aux procédures présentées dans Stability Selection ([46]), où les pénalités LASSO λ des sous-modèles de régression et le seuil s sont calibrés simultanément pour contrôler le nombre de variables sélectionnées à tort ou à Bolasso où λ est le seul paramètre à calibrer puisque s est fixé à 1, Gauss-LASSO stabilisé calibre les pénalités LASSO à l'aide de BIC puis une fois ces pénalités fixées, calibre le second paramètre s à l'aide de l'heuristique de pente. Certes l'impact du choix des pénalités LASSO sur les m sous-modèles est limité, mais cela peut constituer une faille dans la procédure. C'est pour cela que nous avons décidé de mettre en place une seconde procédure de sélection, que nous appellerons Gauss-LASSO enrichi, fondée également sur le Sample Splitting, mais esquivant ce problème de calibration successive des paramètres. Elle a pour fin l'enrichissement de la collection de modèles proposée par Gauss-LASSO appliqué au jeu réel en une collection formée des meilleurs modèles issus de Gauss-LASSO appliqués aux sous-échantillons, d'où l'origine de son nom.

3.1 Description

La procédure Gauss-LASSO enrichi estime le support d'une variable régressée j de cette manière (Voir détail de la procédure en Annexe D) :

1. Étape de Sample Splitting : génération de $m/2$ sous-ensembles des observations, de cardinal $n/2$, et prise en compte des sous-ensembles complémentaires
2. Restriction des données à chacun de ces sous-ensembles. Obtention de m sous-modèles de régression
3. Application de Gauss-LASSO sur chaque sous-modèle : obtention d'un chemin de régularisation et d'une collection de modèles candidats sous-jacente par sous-modèle.
4. Choix du meilleur des m chemins de régularisation et de la meilleure collection de modèles associée au sens de la log-vraisemblance maximisée en ses paramètres.
5. Application de l'heuristique de pente sur ce meilleur chemin pour sélectionner un modèle de la collection dont le support associé sera le support final estimé $\widehat{S}_j^{GLenri}$.

En pratique, nous désignerons par "meilleur" chemin de régularisation le chemin correspondant à l'enveloppe convexe des m courbes de log-vraisemblances maximisées en leurs paramètres (chaque courbe étant associée à un chemin de régularisation) en fonction de la dimension de leurs modèles associés. Les modèles candidats du meilleur chemin comme nous l'appelons ne proviennent donc pas tous du même chemin de régularisation. L'enveloppe convexe peut en effet être composée de morceaux de courbes de log-vraisemblances associés à différents chemins de régularisation. Notre cherchons donc à tirer un profit maximal des m chemins de régularisation en créant une collection de modèles optimale composée, pour chaque dimension, d'un modèle et d'un seul. Ce modèle sera celui qui aura la plus grande log-vraisemblance maximisée en ses paramètres parmi les modèles candidats de tous les chemins de régularisation, pour cette dimension.

L'esprit de Gauss-LASSO enrichi est différent de celui de Gauss-LASSO stabilisé. Pour chaque variable régressée j, Gauss-LASSO stabilisé utilise le rééchantillonnage en sélectionnant sur chacun des m chemins de régularisation qui en découlent un support. Le support final $\widehat{S}_j^{GLstab}$ estimé par la procédure contient des variables dont la fréquence d'apparition parmi ces m supports est jugée suffisante pour pouvoir être considérées comme variables pertinentes. Gauss-LASSO enrichi utilise le rééchantillonnage en mettant en compétition les m chemins de régularisation qui en découlent pour former le meilleur chemin possible. Le support final $\widehat{S}_j^{GLenri}$ estimé par la procédure parmi les supports associés à ce chemin est celui sélectionné par un critère de vraisemblance pénalisé.

L'avantage de Gauss-LASSO enrichi est que le recours à un critère pénalisé pour calibrer un paramètre n'apparaît qu'à l'ultime étape de la procédure. Sa construction apparaît plus propre que celle de Gauss-LASSO stabilisé. Il reste à vérifier si, appliquée au jeu de données réel, Gauss-LASSO enrichi donne des résultats aussi satisfaisant que Gauss-LASSO stabilisé.

3.2 Vraisemblances calculées sur le jeu entier

Notons bien que, comme détaillé dans l'annexe D, chacun des m chemins de régularisation présente une collection ${}^{(k)}\mathcal{M}_j^{GL}$ de modèles qui ont été sélectionnés en tenant compte uniquement des données du sous-jeu $\mathcal{J}_k$ associé au chemin de régularisation. En revanche, les log-vraisemblances de ces modèles sont calculées et maximisées en leurs paramètres en tenant compte des données du jeu entier $\mathcal{I}$. Ainsi, il est possible de comparer proprement chacun des modèles candidats par évaluation de leurs log-vraisemblances maximisées puisque celles-ci sont toutes calculées sur le même jeu de données.

Cependant, comme l'illustre la figure 2.37 pour quatre variables typiques, les pentes ${}^{(l=1)}\kappa_j$ associées aux log-vraisemblances maximisées sont peu prononcées, ce qui résulte en une sélection peu sévère. En effet, le tableau 2.2 atteste bien que sur les cinq variables typiques, Gauss-LASSO enrichi (cas $l = 1$) sélectionne beaucoup trop de variables comparativement à Gauss-LASSO stabilisé.

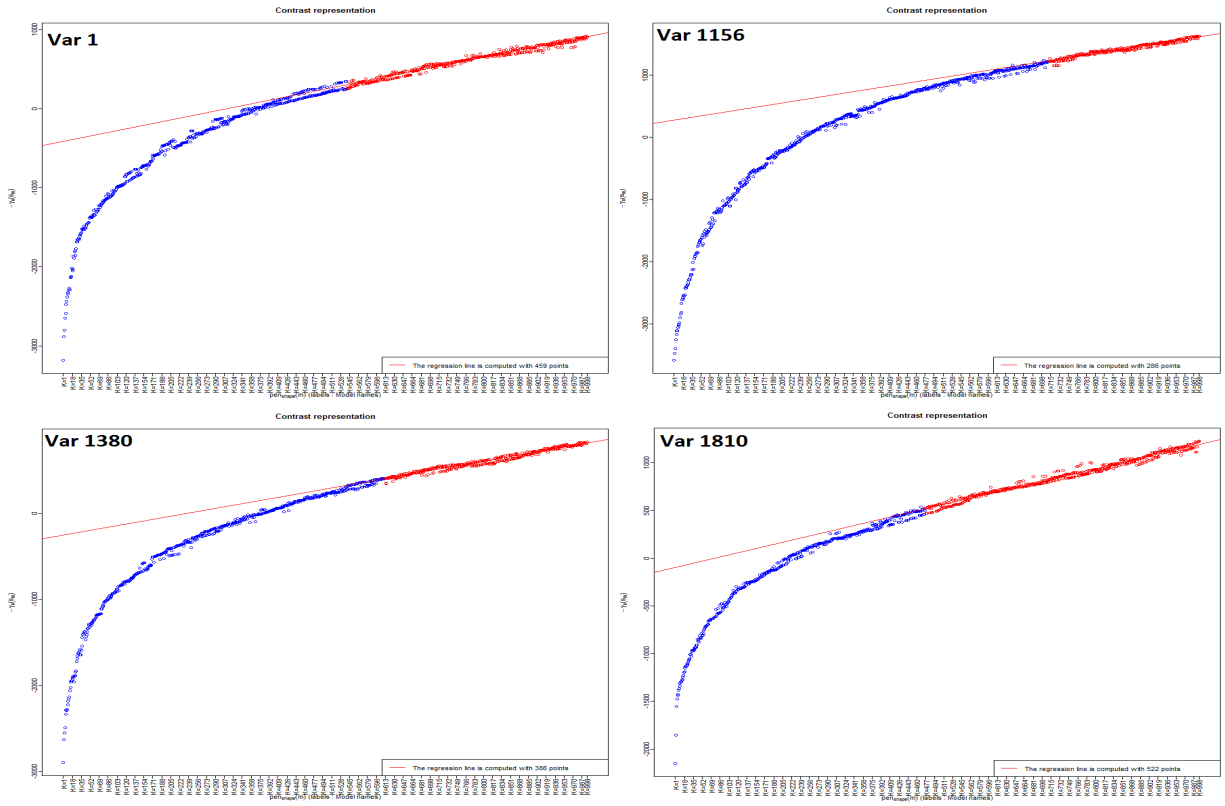

Figure 2.37 – Détection par heuristique de pente de la partie linéaire de $V_{max}(M)$ pour $M \in \mathcal{M}_j^{max}(l = 1)$ en fonction de la dimension des modèles pour quatre variables typiques

Nous soupçonnons à nouveau qu'un grand nombre de variables sont sélectionnés à tort dans les supports estimés par la procédure. Ceci est appuyé par le fait que Gauss-LASSO+BIC que nous avons jugée comme peu fiable quant à sa capacité à ne sélectionner que des ensembles de variables stables, estime des supports de taille moins important que cette nouvelle procédure proposée.

Pour vérifier cette crainte, nous appliquons Gauss-LASSO enrichi aux dix jeux simulés. Les résultats de stabilité sont mauvais, comme escompté (cf. figure 2.38 et tableau 2.5 pour le cas $l = 1$). En conséquence, cette nouvelle proposition ne peut pas nous servir de procédure de référence.

Le manque de sévérité est peut être dû au fait que les modèles de chaque chemin de régularisation sont sélectionnés à partir des données du sous-jeu associé au chemin tandis que les log-vraisemblances sont calculées sur un jeu différent et de taille deux fois plus importante. Calculer les log-vraisemblances des modèles directement en utilisant uniquement les données de son sous-jeu de sélection et non en extrapolant au jeu entier est une possibilité que nous avons décidé d'étudier en espérant éviter ce problème.

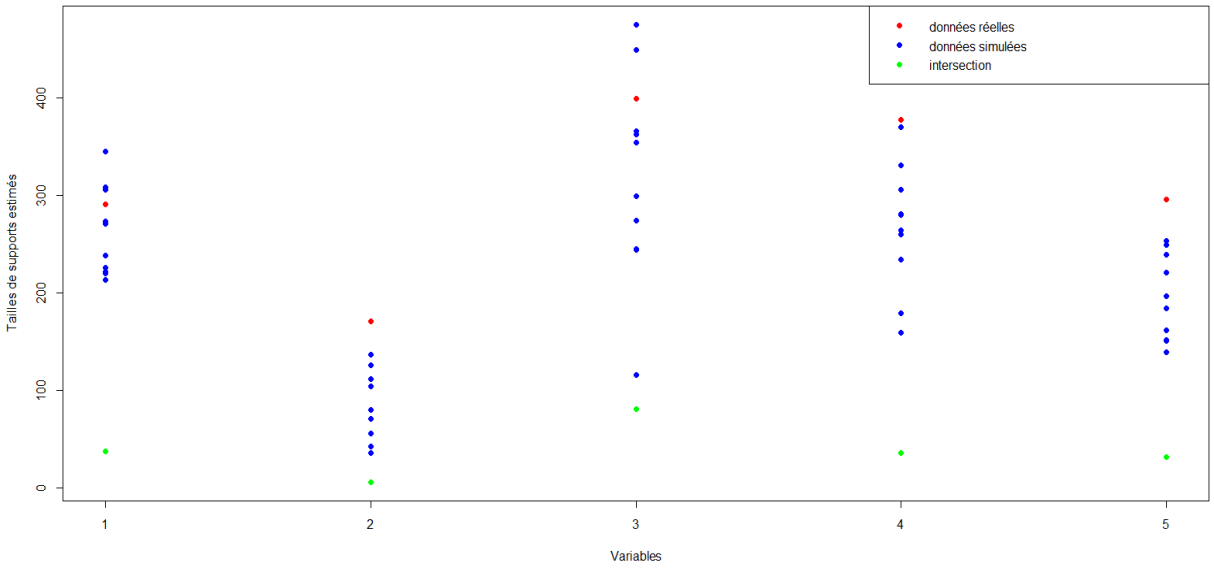

Figure 2.38 – Comparaison des supports estimés par Gauss-LASSO enrichi ($l = 1$) sur le jeu réel et les simulations

Cas	Informations	V 1	V 935	V 1156	V 1380	V 1810
$l = 1$	|Support réel|	290	170	398	376	295
	|Support simulé| moyen	260	80	337	263	195
	Indice de Jaccard	0.051	0.016	0.107	0.045	0.055
$l = 2$	|Support réel|	79	7	147	79	110
	|Support simulé| moyen	94	13	143	90	86
	Indice de Jaccard	0.105	0.119	0.163	0.144	0.140

Table 2.5 – Comparaison des supports estimés par Gauss-LASSO enrichi sur les jeux réél et simulés

Nous distinguerons à l'avenir ces deux cas de figure. Le cas $l = 1$ correspond au cas, que nous venons d'étudier, où les vraisemblances des modèles sont calculées à l'aide des données du jeu entier $\mathcal{I}$ tandis que le cas $l = 2$ correspondra, quant à lui, au cas où les vraisemblances des modèles sont calculées à l'aide des données du sous-jeu de sélection associé au modèle.

3.3 Vraisemblances calculées sur les sous-jeux de sélection

Nous décidons, pour chacun des m chemins de régularisation de calculer les log-vaisemblances sur le sous-jeu de sélection associé (voir détail en annexe D). L'esprit de la procédure évaluée dans le cas $l = 2$ est différent du cas $l = 1$. Pour $l = 1$ et dans le cadre de la régression d'une variable j, nous placions tous les modèles de $\mathcal{M}_j^{GLenri}$ sur le même pied d'égalité en évaluant leur log-vraisemblance maximisée calculée sur le même jeu $\mathcal{I}$. Les modèles conservés étaient ceux le plus en adéquation avec l'ensemble $\mathcal{I}$ de toutes observations. La faiblesse est que ces modèles ne sont pas sélectionnés à partir du jeu entier lui-même. Nous pouvons craindre que ce choix n'est pas le plus approprié. Et effectivement, hormis le manque de stabilité des supports estimés par Gauss-LASSO enrichi ($l = 1$), les comparaisons, pour les variables typiques, entre les courbes de log-vraisemblances maximisées en leur paramètres calculées sur le jeu entier des modèles $\mathcal{M}_j^{GL}$ de Gauss-LASSO directement sélectionnés sur ce jeu et ceux de $\mathcal{M}_j^{max}(l = 1)$ (cf. figures 2.39 et 2.40) vont dans le sens de cette crainte.

Pour des dimensions D de modèles faibles, les modèles appartenant à $\mathcal{M}_j^{GL}$ sont de vraisemblances maximisées similaires à ceux de $\mathcal{M}_j^{max}(l = 1)$, puis pour des dimensions plus importantes, de vraisemblances maximisées bien plus faibles. De plus, selon le tableau 2.5, les supports estimés par la procédure dans le cas $l = 1$ sont, pour les variables typiques, de tailles élevées. À taille de supports identique, le modèle estimé par Gauss-LASSO est de meilleur vraisemblance selon ces mêmes figures. Ceci limite l'intérêt de l'utilisation de Gauss-LASSO enrichi ($l = 1$).

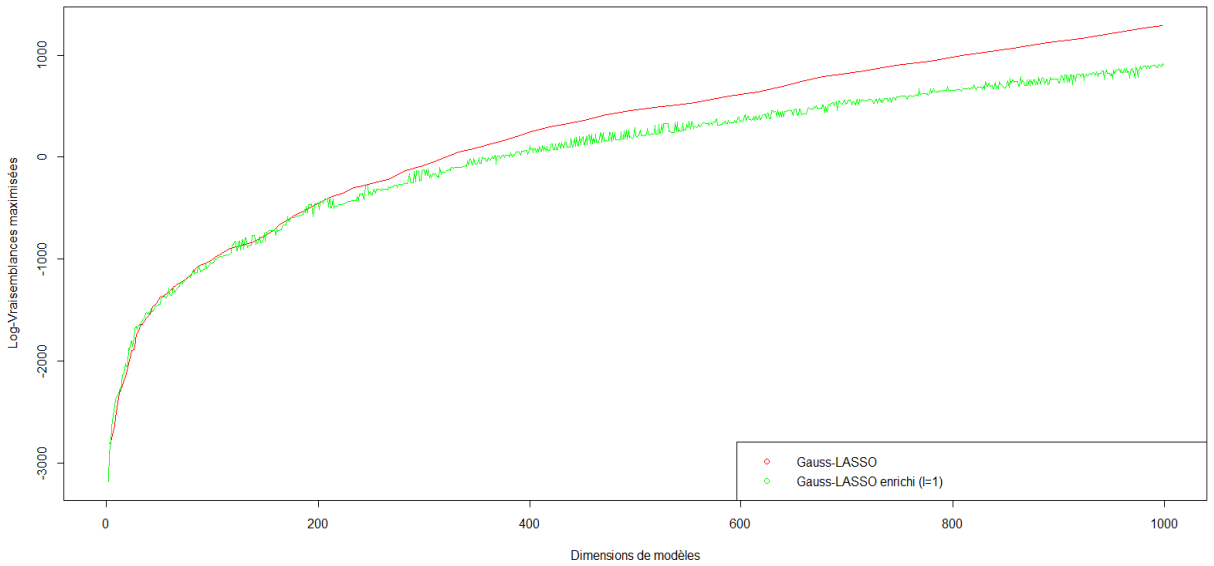

Figure 2.39 – Comparaison des log-vraisemblances maximisées des modèles sélectionnés par Gauss-LASSO enrichi ($l = 1$) et Gauss-LASSO pour la variable 1

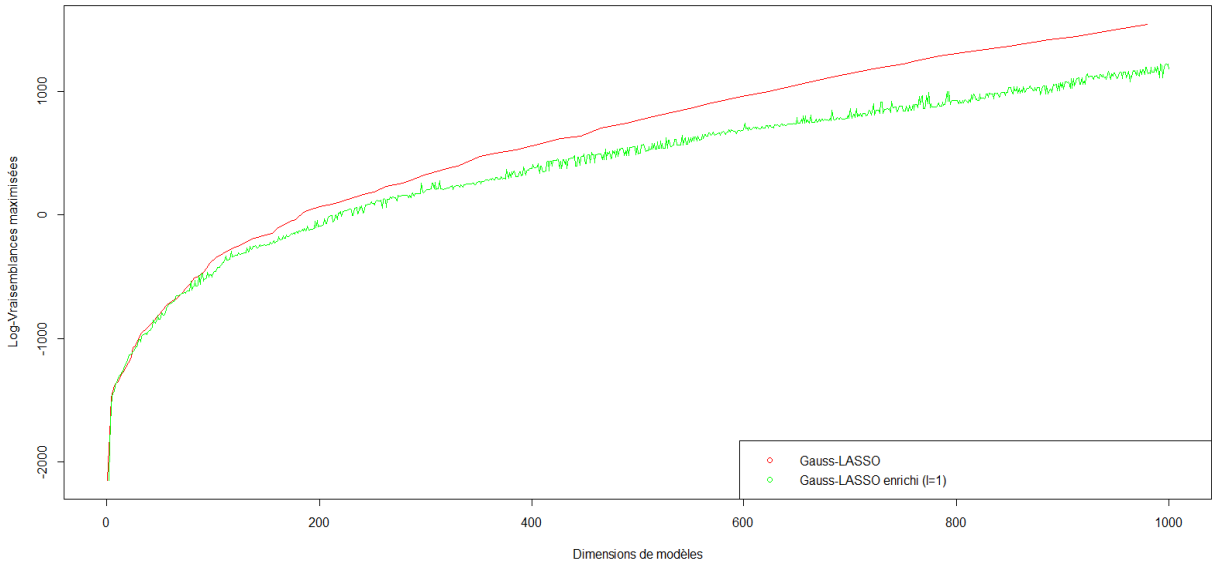

Figure 2.40 – Comparaison des log-vraisemblances maximisées des modèles sélectionnés par Gauss-LASSO enrichi ($l = 1$) et Gauss-LASSO - pour la variable 1810

Dans le cas $l = 2$, pour une variable régressée j, les modèles de $\mathcal{M}_j^{GLenri}$ sont sélectionnés à partir des observations du sous-échantillon associé et leurs log-vraisemblances maximisées sont calculées sur ces mêmes sous-jeux. La comparaison directe des log-vraisemblances maximisées de ces modèles semble à premier abord peu pertinente car calculées sur des jeux de données différents. Cependant, la log-vraisemblance maximisée d'un modèle en ses paramètres est un outil permettant de quantifier son adéquation avec

le jeu de données sur laquelle elle est calculée. Dans le cas $l = 2$, la comparaison de cette fonction pour plusieurs modèles possédant le même nombre de paramètres permet de déterminer celui d'entre eux dont l'adéquation avec le sous-jeu sur lequel elle est calculée est la meilleure. Il est, en outre, nécessaire pour cela que les différents sous-jeux de calcul de ces log-vraisemblances soient de même taille. Cette condition est bien respectée par notre procédure puisque les sous-échantillons $\mathcal{J}_1, .., \mathcal{J}_m$ sont tous de taille $n/2$. Or, comme évoqué ci-dessus, chaque sous-jeu de calcul de log-vraisemblance d'un modèle de $\mathcal{M}_j^{GLenri}$ est également celui qui a été pris en compte dans la sélection de ce même modèle via Gauss-LASSO. Dans ce cadre, plus la log-vraisemblance maximisée d'un modèle est élevée, plus sa présence sur le chemin de régularisation du sous-jeu associé est fondée. Comparer des modèles de même dimension à partir de ces fonctions dans le cas $l = 2$, permet de classer ces modèles selon la force de leur sélection sur le sous-jeu associé. En résumé, pour une dimension de modèle, nous munissions $\mathcal{M}_j^{max}(l = 1)$ du meilleur modèle comparé aux autres par évaluation sur un même jeu $\mathcal{I}$ correspondant à l'union des différents sous-jeux de sélection $\mathcal{J}_1, ..\mathcal{J}_m$ tandis que nous munissons $\mathcal{M}_j^{max}(l = 2)$ du modèle qui, comparé aux autres, est le plus en adéquation avec son propre sous-jeu de sélection.

L'esprit de ces deux méthodes est différent. Les résultats appliqués sur le jeu réel le sont également sensiblement. Le tableau 2.5 montre une estimation de supports de tailles plus faibles par Gauss-LASSO enrichi dans ce cas $l = 2$ comparé au cas $l = 1$ pour les cinq variables typiques. Les pentes ${}^{(l=2)}\kappa_j$ plus prononcées des courbes de log-vraisemblances maximisées pour ces variables sont à l'origine de cette sélection sévère de l'heuristique de pente dans le cas $l = 2$. (cf. figure 2.42). Ce constat n'est pas surprenant puisque l'apport de données nouvelles non utilisées lors de l'étape de sélection des modèles pour le calcul de la log-vraisemblances maximisées de ces mêmes modèles nous rapproche d'un cadre asymptotique. Ceci permet au terme de variance du risque quadratique pour des modèles de grande dimensions d'être plus faible, le terme de biais étant alors constant. Ce terme de variance étant directement proportionnel à la pente des courbes de log-vraisemblances maximisées, le constat est donc bien cohérent.

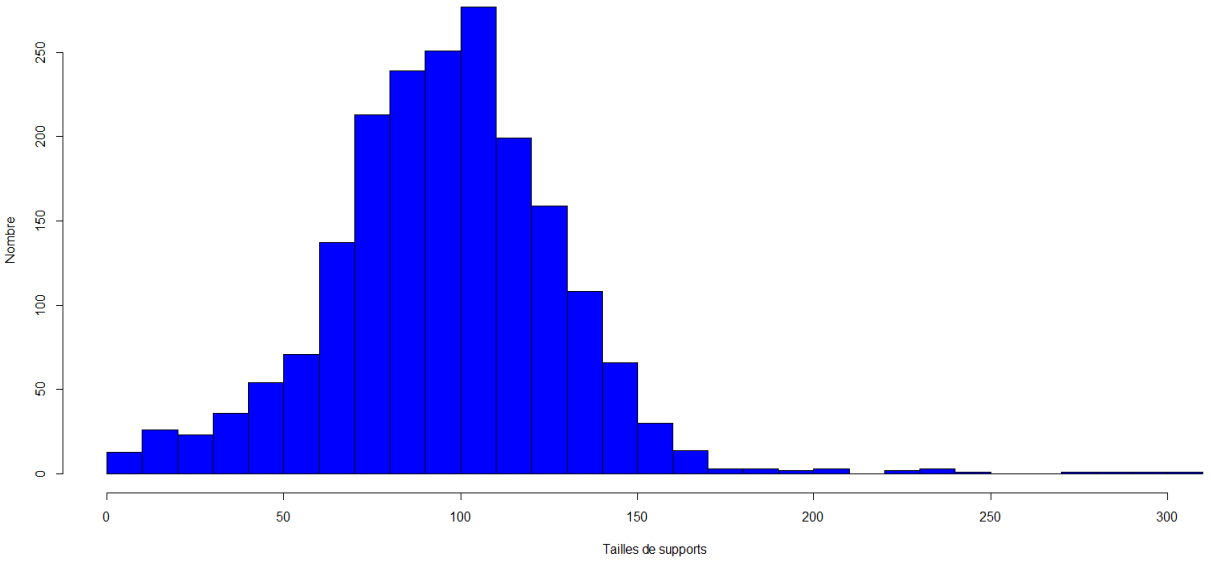

Figure 2.41 – Répartition des p tailles de supports estimés par Gauss-LASSO enrichi

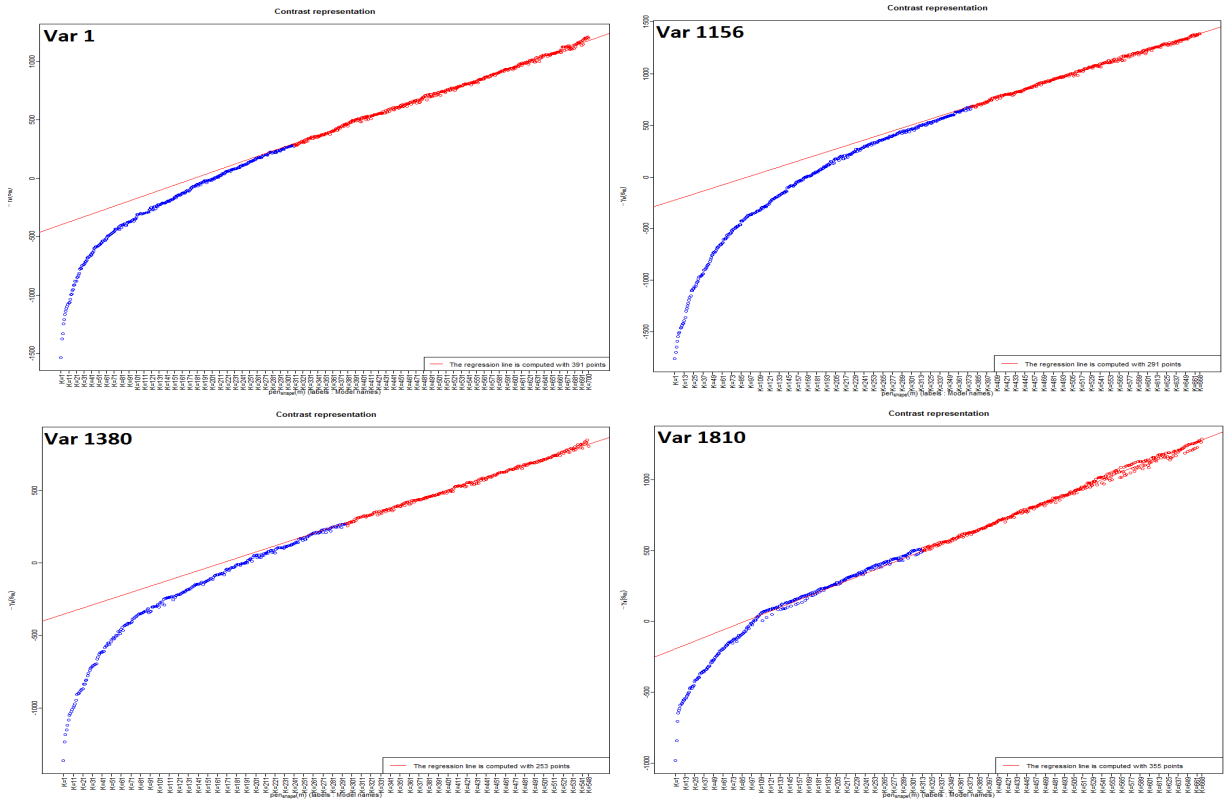

Figure 2.42 – Détection par HP de la partie linéaire de $V_{max}(M)$ pour $M \in \mathcal{M}_j^{max}(l=2)$ en fonction de la dimension des modèles pour quatre variables typiques

Appliquée à nos données, Gauss-LASSO enrichi ($l = 2$) fournit des supports de taille convenable. Les supports estimés par Gauss-LASSO+BIC sont fortement épurés avec des rapports $\frac{|\widehat{S}_j^{GL}(\lambda_{BIC}^j)|}{|\widehat{S}_j^{GLenri}(l=2)|}$ valant environ 2.3 en moyenne sur les p variables (cf. figure 2.43).

De plus, en moyenne sur les p variables régressées, le rapport entre taille des unions et des intersections des supports estimés par la procédure et Gauss-LASSO+BIC (cf. figure 2.44) correspond quasiment au rapport entre les tailles de supports estimés par ces deux procédures (cf. figure 2.43). Ceci implique que les variables de $\widehat{S}_j^{GLenri}(l=2)$ sont en grande partie dans $\widehat{S}_j^{GL}(\lambda_j^{BIC})$.

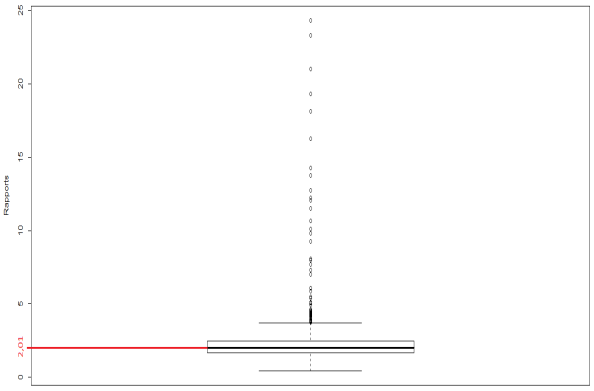

Figure 2.43 – Boxplot des p rapports $\frac{|\widehat{S}_j^{GL}(\lambda_{BIC}^j)|}{|\widehat{S}_j^{GLenri}(l=2)|}$

Enfin, la procédure Gauss-LASSO stabilisé, qui est considérée comme notre actuelle procédure de référence, estime des supports de tailles similaires, en moyenne sur les p variables régressées, à Gauss-LASSO enrichi ($l = 2$) (cf. tableau 2.1). Ces arguments tendent à valider cette dernière procédure.

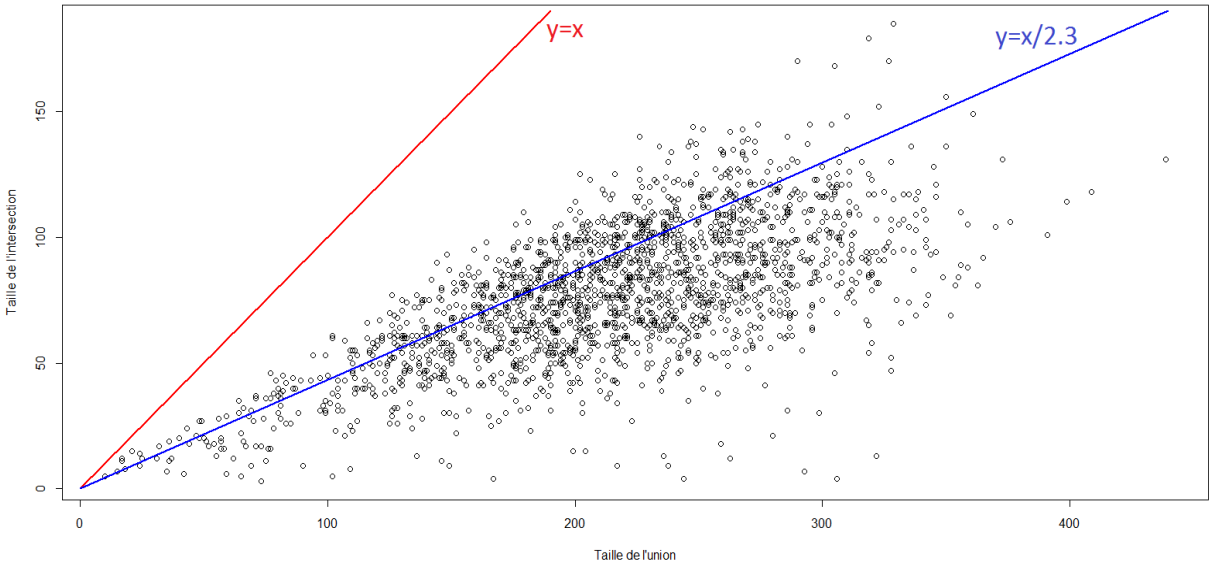

Figure 2.44 – Taille de l'intersection VS taille de l'union des supports Gauss-LASSO+BIC et Gauss-LASSO enrichi($l = 2$)

Enfin, d'un point de vue stabilité des supports estimés par la procédure, le changement de jeux de données impliqués dans les calculs de log-vraisemblances maximisées en leurs paramètres des modèles considérés, permet un gain de stabilité appréciable. En effet, l'application de la procédure dans le cas $l = 2$ sur les dix jeux de données simulés précédemment expérimentés par Gauss-LASSO+BIC et Gauss-LASSO stabilisé, résultent

pour chacune des variables typiques en des supports estimés de contenus beaucoup plus proches que pour le cas $l = 1$. (cf. tableau 2.5 et illustration en figure 2.45).

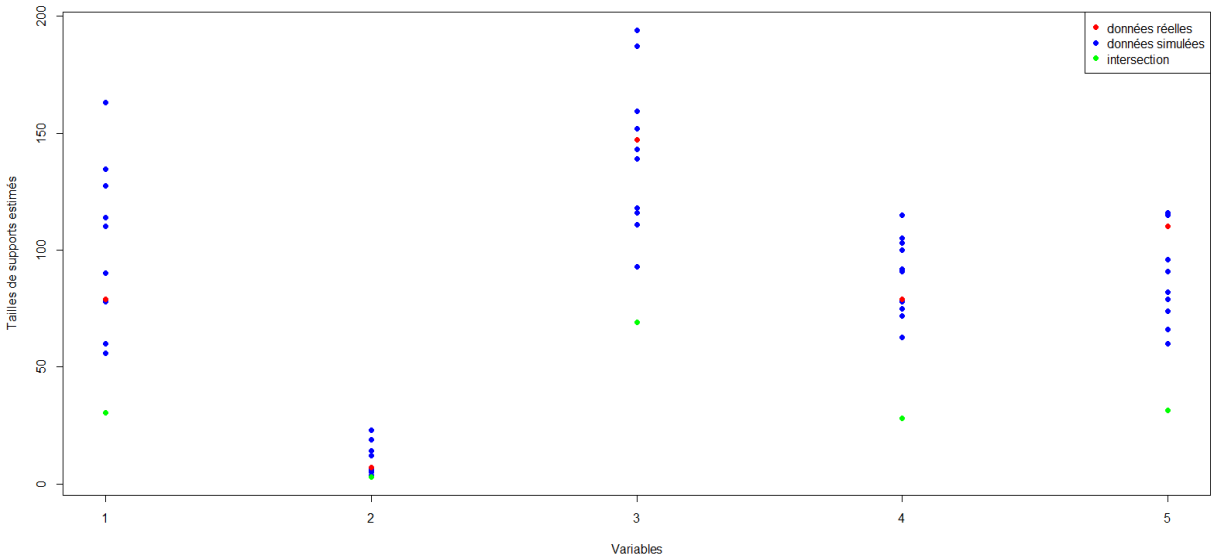

Figure 2.45 – Comparaison des supports estimés par Gauss-LASSO enrichi ($l = 2$) sur le jeu réel et les simulations

Gauss-LASSO enrichi ($l = 2$) est également pour nous une procédure de référence. Nous mentionnerons désormais par Gauss-LASSO enrichi, la procédure Gauss-LASSO enrichi effectuée dans le cas $l = 2$. Nous noterons tout de même que comparé aux résultats de stabilité de Gauss-LASSO stabilisé (cf. tableau 2.4), les sélections réalisées par Gauss-LASSO enrichi sur nos jeux de données simulés sont moins stables. La construction même de Gauss-LASSO stabilisé, visant exclusivement en la sélection, pour chaque variable régressée, des variables les plus stables possibles en est la principale explication. Néanmoins ces deux procédures, dont les paramètres ont été calibrés pour une stabilité optimale, estiment des supports de tailles similaires. Vérifions si leurs contenus coïncident.

4 Comparaison des deux procédures

4.1 Supports estimés

Estimer des supports de tailles voisines (cf. figure 2.46) n'implique pas forcément estimer des supports aux contenus voisins. Certes, en moyenne, plus le support estimé $\widehat{S}_j^{GLstab}$ d'une variable j par Gauss-LASSO stabilisé est fourni, plus son support estimé $\widehat{S}_j^{GLenri}$ par Gauss-LASSO enrichi l'est également (cf.figure 2.47). Mais, la construction de Gauss-LASSO stabilisé et celle de Gauss-LASSO enrichi, bien que fondées toutes les deux sur le Sample Splitting, sont différentes. Il n'est donc pas évident, au premier abord, que les sélections réalisées par ces procédures soient similaires.

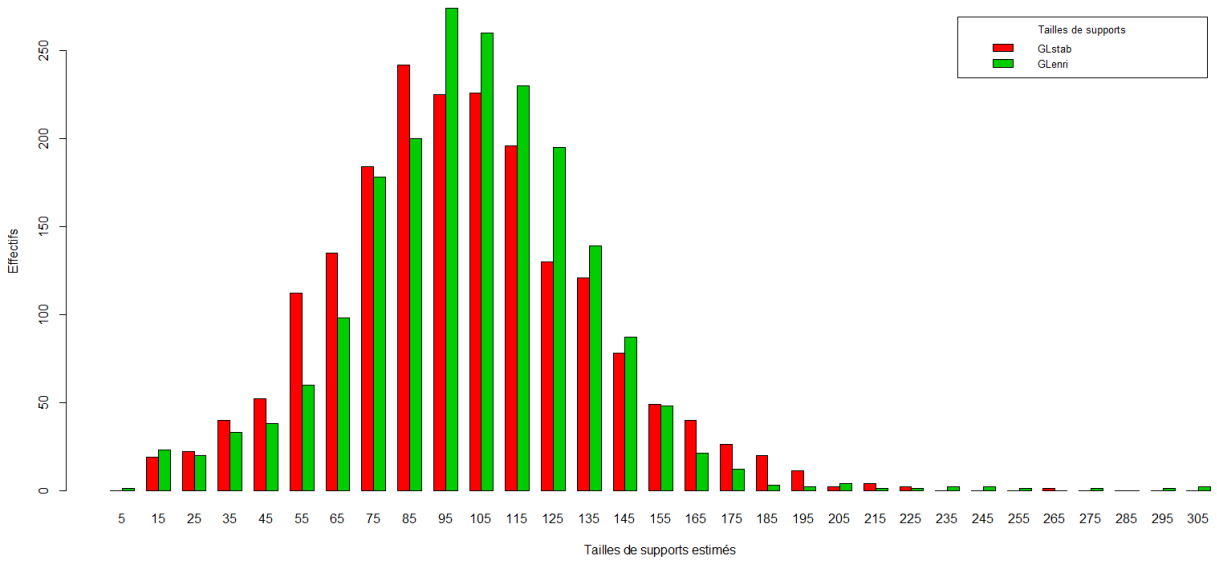

Figure 2.46 – Comparaison de la taille des supports estimés par Gauss-LASSO stabilisé et enrichi

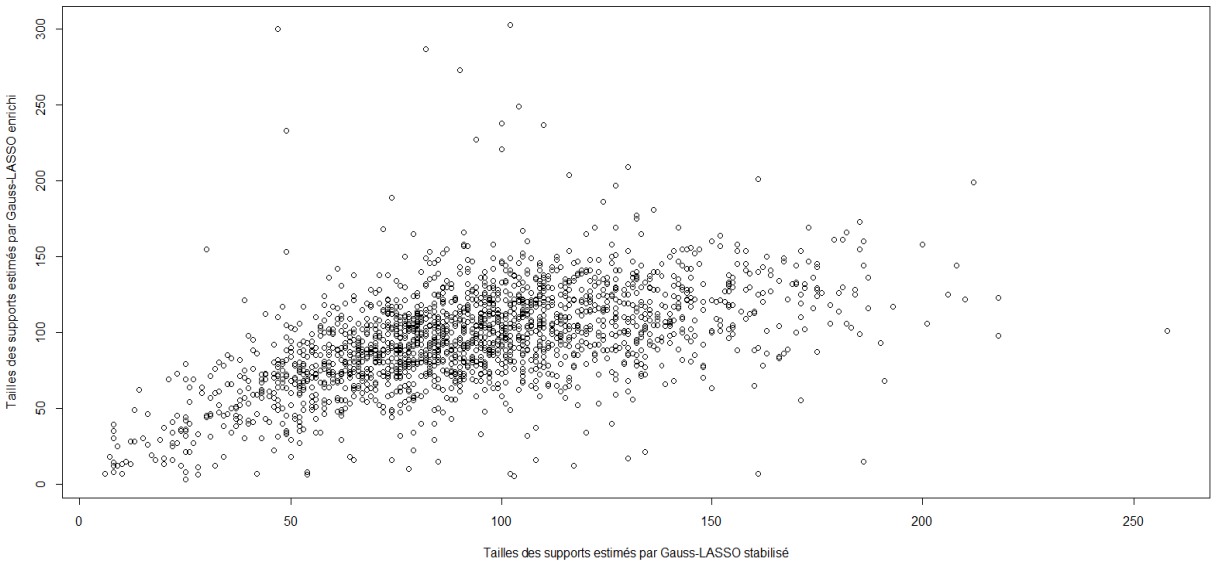

Figure 2.47 – Tailles des supports estimés $\widehat{S}_j^{GLenri}$ en fonction de celles de $\widehat{S}_j^{GLstab}$

En s'attardant, pour quatre variables typiques, aux supports estimés par Gauss-LASSO+BIC, Gauss-LASSO stabilisé et Gauss-LASSO enrichi, nous nous apercevons que les variables communes à ces trois procédures sont en nombre relativement important (cf. figure 2.48). Peu de variables sont uniquement sélectionnées par Gauss-LASSO enrichi ou stabilisé tandis que Gauss-LASSO+BIC a tendance à sélectionner des variables exclues des supports estimés par les deux autres procédures. Ceci est dû à la grande taille des supports qu'elle estime comparativement à celle des supports estimés par les deux procédures fondées sur le Sample Splitting. En ce qui concerne Gauss-LASSO stabilisé

et enrichi, l'intersection pour une variable régressée j, de ses supports estimés $\widehat{S}_j^{GLstab}$ et $\widehat{S}_j^{GLenri}$ semble relativement importante au vu de ces diagrammes pour les variables typiques. L'extrapolation de ce résultat sur l'ensemble des p variables régressées est représenté en figure 2.49. Celle-ci nous informe qu'en moyenne, le nombre de variables présentes dans l'intersection des supports $\widehat{S}_j^{GLstab} \cap \widehat{S}_j^{GLenri}$ d'une variable régressée j représente 79% des variables présentes dans le plus petit des deux supports estimés. Ceci tend à suggérer le fait que les procédures Gauss-LASSO stabilisé et Gauss-LASSO enrichi estiment des supports aux contenus très proches, comme il était espéré.

Néanmoins, il faut s'assurer au mieux que les variables sélectionnées exclusivement par l'une des deux procédures ne soient pas des variables explicatives capitales. Pour juger de l'importance de ces variables exclusives, nous allons nous attarder sur leurs scores et leurs coefficients de régression.

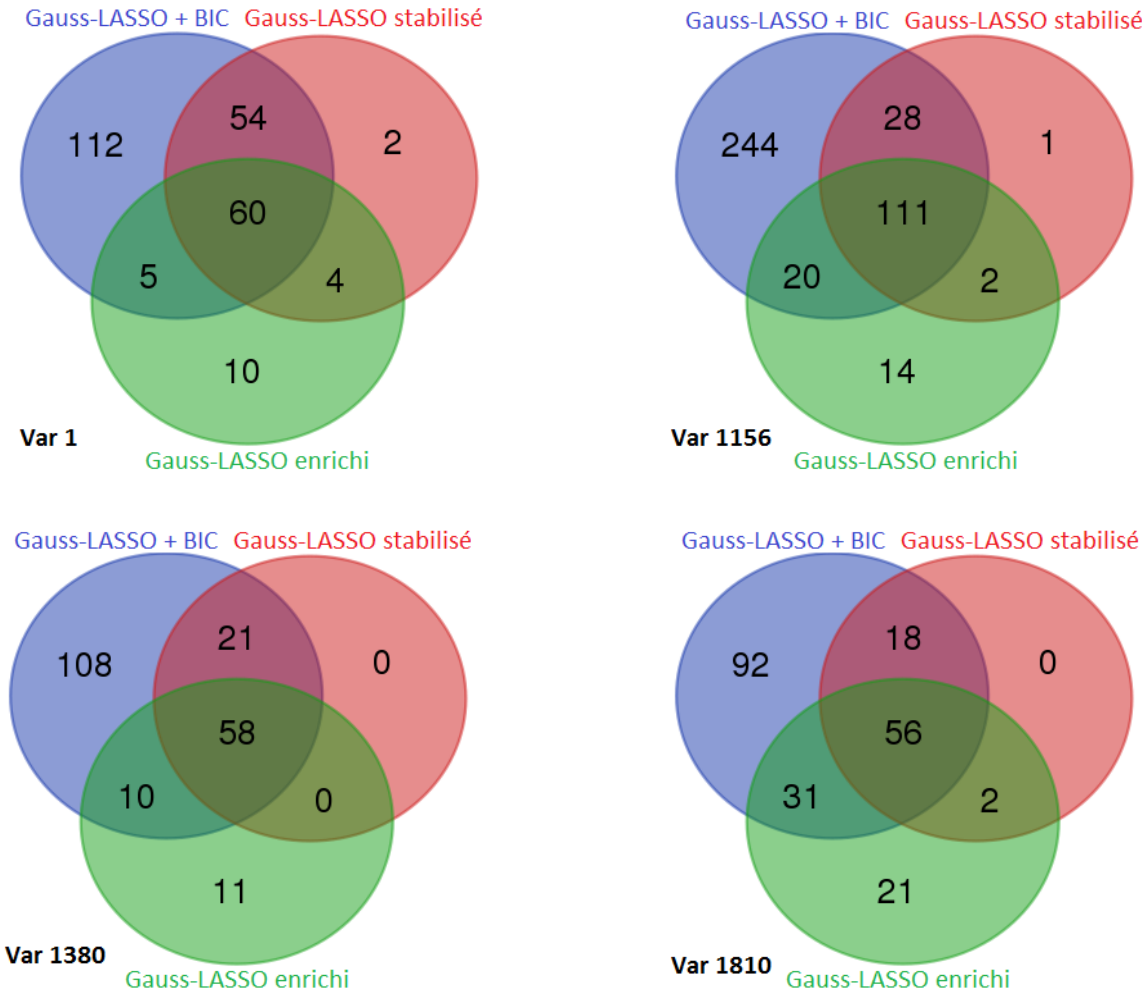

Figure 2.48 – Diagrammes de Venn des supports estimés par nos procédures

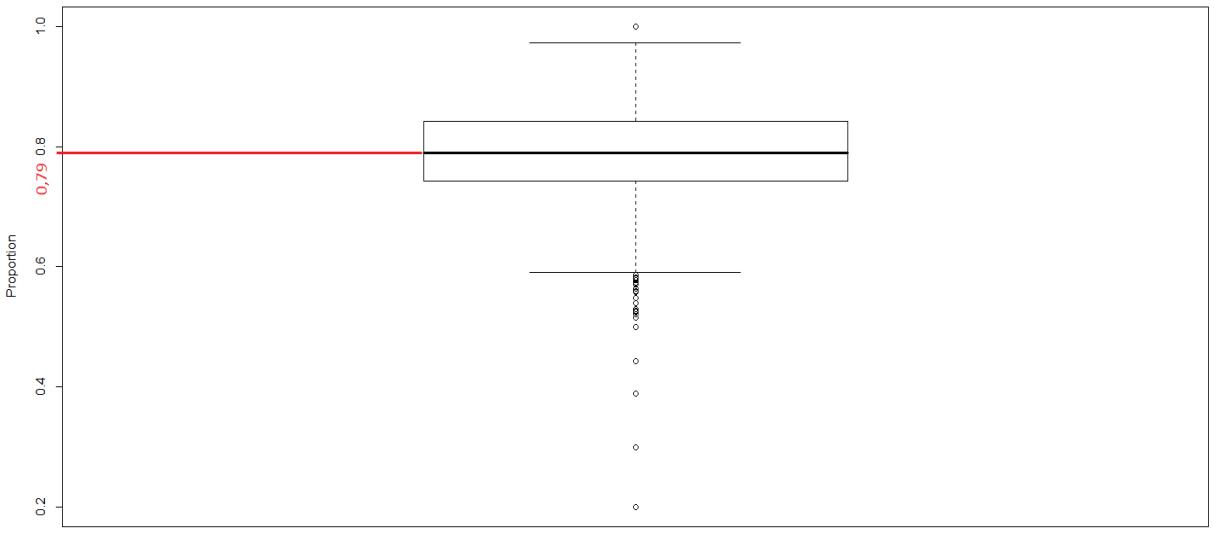

Figure 2.49 – Proportions des variables de $\widehat{S}_j^{GLstab} \cap \widehat{S}_j^{GLenri}$ dans le plus petit des 2 supports

Pour vérifier si les sélections réalisées par nos deux procédures de référence sont proches, nous allons nous attarder, pour chaque variable régressée j, sur la valeur des scores $\mathbb{S}(j, j')$ des variables $j' \in \widehat{S}_j^{GLstab} \backslash \widehat{S}_j^{GLenri}$. Nous espérons que les scores de ces variables exclusives à Gauss-LASSO stabilisé ne soient pas nettement plus grands que la valeur du seuil $\widehat{s}_j^{HP}$ associé à la variable. La figure 2.50 illustre la valeur des scores de ces variables exclusives à Gauss-LASSO stabilisé. Une majorité d'entre elles ont un score proche du seuil estimé par heuristique de pente associé. Cependant, on peut remarquer aussi qu'un nombre d'entre elles ont des scores relativement élevés. Ceci atteste du fait qu'un petit nombre de variables jugées très stables par Gauss-LASSO stabilisé ne sont pas importantes aux yeux de Gauss-LASSO enrichi.

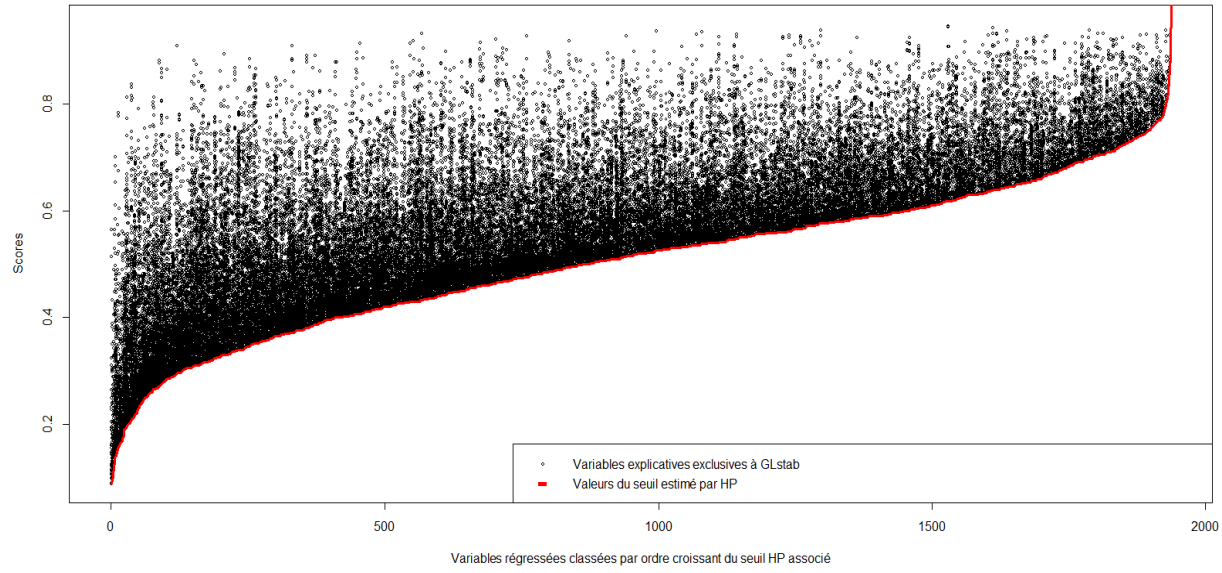

Figure 2.50 – Scores des variables sélectionnées par GL stabilisé et non par GL enrichi

Pour évaluer l'importance des variables exclusivement sélectionnées dans les supports par Gauss-LASSO enrichi, nous faisons appel aux coefficients de régression linéaire de chaque variable régressée sur celles de son support associé estimé par Gauss-LASSO enrichi. Pour une variable régressée j, nous estimons le vecteur de coefficients du modèle de régression restreint aux données du sous-jeu $\mathcal{J}_k$ à partir duquel le support $\widehat{S}_j^{GLenri}$ a été estimé : $\widehat{\Theta}_j^{Glenri} = \underset{\Theta}{\operatorname{argmin}} ||{}^{(k)}X^j - {}^{(k)}X^{\widehat{S}_j^{Glenri}}\Theta||_2^2$.

Si les variables exclusives à Gauss-LASSO enrichi font partie des variables aux coefficients de régression de plus forte valeur absolue parmi toutes celles du support estimé par la procédure, cela appuierait leur importance. La figure 2.51 fait justement état, pour les variables typiques, de la valeur absolue des coefficients de $\widehat{\Theta}_j^{Glenri}$ classés par ordre décroissant. La valeur absolue des coefficients des variables de $\widehat{S}_j^{GLenri}$ exclusivement sélectionnées par Gauss-LASSO enrichi sont représentées en rouge. On peut clairement remarquer que ces variables ne font jamais partie des variables aux coefficients de régression de valeurs absolues très élevées. Néanmoins, quelques unes de ces variables sélectionnées uniquement par Gauss-LASSO enrichi sont bien positionnées dans le classement des variables du support selon la valeur de leur coefficient de régression.

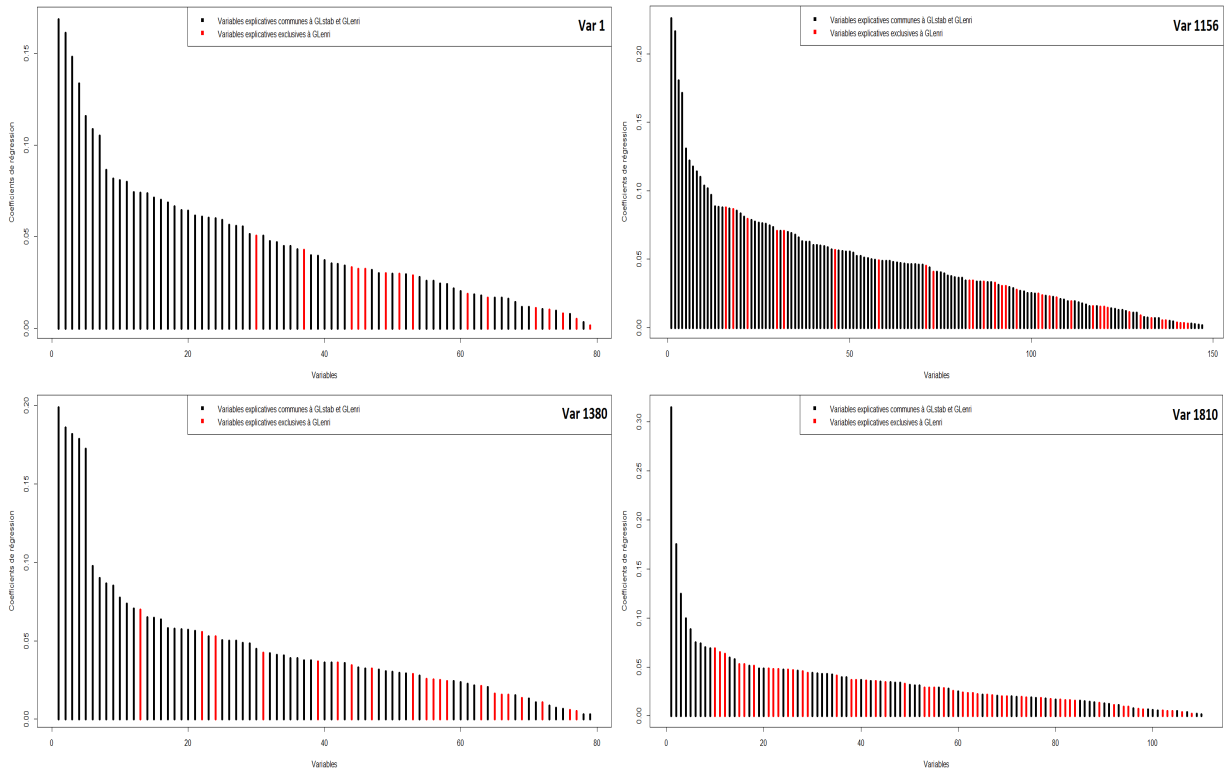

Figure 2.51 – Coefficients de régression linéaires des variables du support estimé par Gauss-LASSO enrichi pour les variables typiques

Le constat sur l'ensemble des coefficients des variables des p supports est le même. Sur la figure 2.52, où tous les coefficients de régressions pour les p variables régressées sont représentés, les coefficients des variables des supports estimés communes aux deux procédures sont dans l'ensemble clairement de valeurs absolues plus importantes que ceux des variables exclusivement sélectionnées par Gauss-LASSO enrichi. Il est également notable que certains coefficients de telles variables sont de valeurs absolues importantes, appuyant le fait que certaines variables des supports somme toute importante aux yeux de Gauss-LASSO enrichi, ne sont pas sélectionnées par Gauss-LASSO stabilisé.

Notons que nous avons omis les coefficients extrêmes de valeur absolue supérieure à 0.6 associés à des variables sélectionnées communément par les deux procédures, pour faciliter la lecture de la figure 2.52.

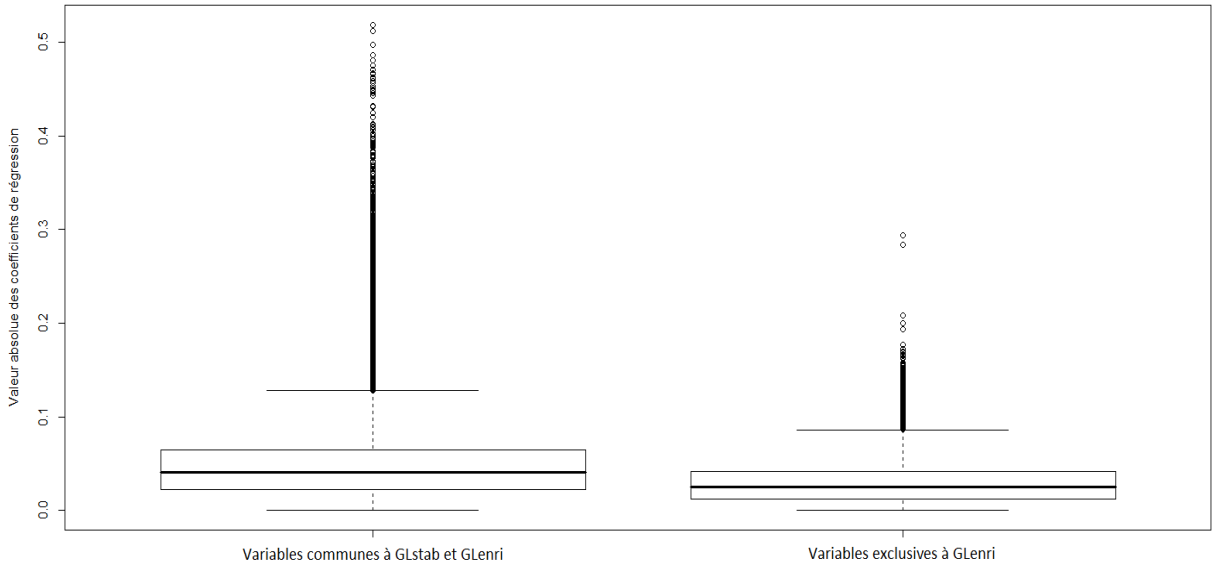

Figure 2.52 – Répartition des valeurs absolues des coefficients de régressions pour les p variables régressées

Nous pouvons donc conclure de cette comparaison des contenus des différents supports estimés par les deux procédures que, globalement, pour chaque variable régressée j, les supports $\widehat{S}_j^{GLenri}$ et $\widehat{S}_j^{GLstab}$ sont de contenus fort similaires. Néanmoins, quelques variables importantes de poids selon Gauss-LASSO stabilisé ne sont pas sélectionnées par Gauss-LASSO enrichi et réciproquement.

4.2 Matrices d'adjacence

Une procédure de sélection estime un support pour chacune des variables régressées. Les variables de ce support correspondent aux variables explicatives pertinentes de la variable régressée selon la procédure. D'un point de vue biologique, ces variables explicatives pertinentes représentent les FTs régulateurs du FT représenté par la variable régressé.

Dans le graphe $\mathcal{G}$ modélisant le réseau de FTs, les variables du support correspondent exactement aux noeuds présentant une arête dirigée vers le noeud de la variable régressée. L'ensemble des arêtes de ce graphe est résumé dans la matrice d'adjacence A du graphe $\mathcal{G}$. Cette matrice d'adjacence est de taille $p \times p$ et ses coefficients sont établis de cette manière :

— $A_{j,j'} = 1 \Leftrightarrow$ le FT j' régule le FT $j \Leftrightarrow j'$ appartient au support de j.

— $A_{j,j'} = 0$ sinon.

Gauss-LASSO stabilisé et enrichi proposent chacune un graphe et une matrice d'adjacence associée. Nous désignerons par $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ les graphes respectifs établis par chacune de ces procédures et A^{GLstab} et A^{GLenri} leurs matrices d'adjacence associées. Le tableau 2.6 représente les coefficients de la matrice $B = A^{GLstab} - A^{GLenri}$. Les $p \times p$ coefficients de cette matrice sont, en très forte majorité égaux à 0. Les coefficients égaux à -1 et ceux égaux à 1, caractérisant respectivement les liens de régulation exclusivement établis par Gauss-LASSO enrichi et par Gauss-LASSO stabilisé apparaissent en très faibles quantités, illustrant comme dans le paragraphe précédent la cohérence entre les sélections qu'elles réalisent.

Valeur	-1	0	1
Proportion (%)	1.6	96.9	1.5

Table 2.6 – Répartition de la valeur des coefficients de la matrice B

En outre, d'un point de vue biologique, si un FT j régule un FT j', l'inverse n'est pas forcément vrai. Cependant, les FTs agissent en groupe pour activer d'autres gènes et il n'est pas rare de constater que des FTs qui s'activent mutuellement, à savoir qu'un FTj peut avoir besoin que le FT j' soit activé lui-même pour s'activer et réciproquement. Détecter sur les graphes bâtis par nos procédures de sélections, une bonne proportion de paires de FTs (j, j') qui s'activent mutuellement serait encourageant.

De plus, d'un point de vue statistique, on peut penser que si une variable j' permet d'expliquer une variable j via le modèle de régression $X^j = X^{-j}\Theta_j + \epsilon_j$, inversement, la variable j permettra d'expliquer la variable j' via le modèle $X^{j'} = X^{-j'}\Theta_{j'} + \epsilon_{j'}$. Nous allons donc étudier "le degré de symétrie" des matrices adjacences des graphes issus de Gauss-LASSO + BIC, Gauss-LASSO stabilisé et Gauss-LASSO enrichi, à savoir la proportion de couples de variables (j, j') qui s'activent mutuellement. Une bonne procédure de sélection devrait former des matrices d'adjacences au degré de symétrie bien supérieur à des matrices formée par une sélection au hasard.

Soit A la matrice d'adjacence d'une procédure. On désignera, par la suite :
— un couple de noeuds (j, j') vérifiant $A(j, j') = A(j', j) = 1$ par une paire.
— l'ensemble des matrices binaires de taille $p \times p$ ayant ses éléments diagonaux nuls et possédant k coefficients égaux à 1 par $\mathcal{M}(p, k)$.

Chacune de nos procédures de sélection établit un nombre k d'arêtes orientées dans $\mathcal{G}$. A possède donc k coefficients égaux à 1 en dehors de sa diagonale. En cela, $A \in \mathcal{M}(p, k)$. Nous comparons, pour chacune de nos procédures, le nombre de paires estimées $\widehat{N}_{paires}(k)$ au nombre moyen de paires $\mathbb{E}(N_{paires}(p, k))$ des matrices de $\mathcal{M}(p, k)$. Nous espérons que $\widehat{N}_{paires}(k) \gg \mathbb{E}(N_{paires}(k))$. Calculons le nombre moyen de paires des matrices de $\mathcal{M}(p, k)$.

Soit l'expérience aléatoire : "Tirage d'une matrice de $\mathcal{M}(p, k)$" dans l'espace probabilisé $(\Omega, \mathcal{A}, P)$ où :
— $\Omega = \mathcal{M}(p, k)$.
— $\mathcal{A}$ l'ensemble des événements possibles.
— P la mesure de probabilité associée.

Dans ce cadre, il y a équiprobabilité des événements élémentaires puisque chaque matrice $M \in \mathcal{M}(p, k)$ a une probabilité $P(M) = \frac{1}{|\Omega|}$ d'être tirée. D'ailleurs, $|\Omega| = C^k_{p(p-1)}$ puisque les k coefficients de M égaux à 1 font partie des $p(p-1)$ élements non diagonaux.

On considére alors la variable aléatoire $N_{paire}(M)$ correspondant au nombre de paires d'une matrice M tirée aléatoirement dans Ω. Par définition, $N_{paire}(M) = \sum_{i<j} 1_{M_{i,j}=M_{j,i}=1}$. Le nombre moyen de paires des matrices de $\mathcal{M}(p, k)$ correspond à l'espérance de cette variables aléatoire :

$$\mathbb{E}(N_{paire}(M)) = \sum_{i<j} \mathbb{E}(1_{M_{i,j}=M_{j,i}=1}) = \sum_{i<j} P(M_{i,j} = M_{j,i} = 1).$$

Or, de par le cadre d'équiprobabilité :

$$P(M_{i,j} = M_{j,i} = 1) = \frac{|\{M \in \mathcal{M}(p,k),\ \text{tq}\ M_{i,j}=M_{j,i}=1\}|}{|\Omega|} = \frac{C^{k-2}_{p(p-1)-2}}{C^k_{p(p-1)}}.$$

On en déduit que :

$$\begin{aligned}\mathbb{E}(N_{paire}(M)) &= \sum_{1 \le i<j \le p} \frac{C^{k-2}_{p(p-1)-2}}{C^k_{p(p-1)}} = \frac{p(p-1)}{2} \times \frac{(p(p-1)-2)!}{(p(p-1))!} \times \frac{k!}{(k-2)!} \\ &= \frac{k(k-1)}{2(p(p-1)-1)}.\end{aligned}$$

Le nombre moyen de paires des matrices de $\mathcal{M}(p,k)$ vaut donc :

$$\boxed{\mathbb{E}(N_{paire}(M)) = \frac{k(k-1)}{2(p(p-1)-1)}}$$

Il reste à comparer ce nombre moyen de paires au nombre de paires estimées par les procédures de sélection. Notre jeu de données présente $p = 1937$ variables. Le nombre k de coefficients valant 1 dans une matrice de $\mathcal{M}(p,k)$ varie entre 0 et $p(p-1) = 3750032$. On considère $\mathbb{E}(N_{paire}(k))$ comme une fonction dépendant de k, puisque nous fixons p à 1937.

Pour la matrice d'adjacence associée à chacune de nos procédures de sélection, nous recensons le nombre k de ses coefficients égaux à 1, le nombre $\mathbb{E}(N_{paire}(k))$ correspondant au nombre moyen de paires d'une matrice de $\mathcal{M}(p,k)$, la proportion attendue $P_{\mathbb{E}(N)}(k) = 2\mathbb{E}(N_{paire}(k))/k$ de coefficients égaux à 1 faisant partie d'une paire pour une matrice de $\mathcal{M}(p,k)$, le nombre de paires $\widehat{N}_{paires}(k)$ effectif de la matrice d'adjacence, et enfin la proportion $P_{\widehat{N}}(k) = 2\widehat{N}_{paires}(k)/k$ de coefficients égaux à 1 faisant partie d'une paire de la matrice d'adjacence en question. les résultats sont résumés dans le tableau 2.7.

Procédure	k	$\mathbb{E}(N_{paire}(k))$	$P_{\mathbb{E}(N)}(k)$	$\widehat{N}_{paires}(k)$	$P_{\widehat{N}}(k)$
Gauss-LASSO + BIC	380 421	19 295.8	10.14 %	98 534	51.8 %
Gauss-LASSO stabilisé	181 135	4374.6	4.83 %	44 138	48.7 %
Gauss-LASSO enrichi	186 278	4626.5	4.97 %	36 697	39.4 %

Table 2.7 – Évaluation du nombre de paires des matrices d'adjacence estimées par nos procédures

Pour les trois procédures, on constate bien que $\widehat{N}_{paires}(k) \gg \mathbb{E}(N_{paires}(k))$. Les matrices d'adjacence estimées présentent effectivement, dans des mesures différentes, beaucoup plus de paires que la normale. Dans l'optique de trouver des procédures qui bâtissent des matrices d'adjacence au degré de symétrie élevé, les trois procédures de sélection sont bien adaptées. Gauss-LASSO+BIC bâtit une matrice d'adjacence où la proportion de coefficients égaux à 1 faisant partie d'une paire ($P_{\widehat{N}}(k)$) est la plus élevée parmi les 3 procédures. Ceci était attendu, au vu de la valeur de $P_{\mathbb{E}(N)}(k)$ associée. Cependant, on constate que Gauss-LASSO stabilisé bâtit une matrice au degré de symétrie $P_{\widehat{N}}(k)$ plus important que celle établie par Gauss-LASSO enrichi, alors que l'on s'attendait à ce qu'elles présentent des degrés très proches, au vu de $P_{\mathbb{E}(N)}(k)$.

La comparaison directe des graphes orientés $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ nous fait constater que les sélections réalisées par les deux procédures sont proches mais possèdent quelques différences comme illustré dans ce paragraphe 4. Nous rappelons que l'objectif principal de

notre procédure méthodologique globale est de former des groupes de variables régressées et explicatives aux profils similaires. L'étape de sélection ayant permis de former les graphes orientés correspond à la première partie du travail, l'étape de classification des graphes à la seconde. Nous pourrons juger de l'impact des quelques différences observées dans l'étape de sélection entre nos différentes procédures, sur l'étape de classification des graphes. Nous verrons alors si ces quelques différences conduisent à la création de groupes de variables aux contenus très différents lors cette étape de classification.

5 Annexes

5.1 Annexe A : Procédure Gauss-LASSO + Choix de pénalité par BIC

Pour un X_j, Gauss-LASSO fournit un chemin de régularisation. Le support estimé correspondra à celui dont la pénalité associée λ_j est choisie par BIC. En voici le détail :

1. Régression pénalisée par LASSO : $\widehat{\Theta}_j^{Las}(\lambda_j) = \underset{\Theta}{\operatorname{argmin}} \left(\frac{1}{2}||X^j - X^{-j}\Theta||_2^2 + \lambda_j||\Theta||_1\right)$.
   - Obtention des pénalités de changement de supports : $\Lambda_j : \lambda_j^1 > ... > \lambda_j^{L_j} > 0$.
   - Collection de modèles obtenue : $\mathcal{M}_j^{Las} = \left\{X^j = X^{-j}\widehat{\Theta}_j^{Las}(\lambda_j) + \widehat{\epsilon}_j^{Las}\right\}_{\lambda_j \in \Lambda_j}$.
   - Ensemble de supports associés : $\mathcal{S}_j^{Las} = \left\{\widehat{S}_j^{Las}(\lambda_j) = \left\{j' \text{ tel que } \widehat{\theta}_{j,j'}^{Las}(\lambda_j) \neq 0\right\}\right\}_{\lambda_j \in \Lambda_j}$.
2. Réestimation des coefficients dans les supports par moindres carrés ordinaires :
$$\forall \lambda_j \in \Lambda_j, \left(\widehat{\theta}_{j,j'}^{GL}(\lambda_j)\right)_{j' \in \widehat{S}_j^{Las}(\lambda_j)} = \underset{\Theta}{\operatorname{argmin}} ||X^j - X^{\widehat{S}_j^{Las}(\lambda_j)}\Theta||_2^2.$$
   Cette étape ne modifie pas les supports mais la vraisemblance des modèles associés :
   - Nouvelle collection de modèle : $\mathcal{M}_j^{GL} = \left\{M_j^{GL}(\lambda_j)\right\}_{\lambda_j \in \Lambda_j} = \left\{X^j = X^{-j}\widehat{\Theta}_j^{GL}(\lambda_j) + \widehat{\epsilon}_j^{GL}\right\}_{\lambda_j \in \Lambda_j}$.
   - Ensemble de supports associés inchangé : $\mathcal{S}_j^{GL} = \mathcal{S}_j^{Las}$.
3. Sélection par le critère BIC d'un λ_j et par conséquent du support de $\mathcal{S}_j^{GL}$ associé :
$$\lambda_j^{BIC} = \underset{\lambda_j \in \Lambda_j}{\operatorname{argmin}} \left(-2V_{max}\left(M_j^{GL}(\lambda_j)\right) + \left|\widehat{S}_j^{GL}(\lambda_j)\right| \log(n)\right).$$
   où $V_{max}\left(M_j^{GL}(\lambda_j)\right) = \frac{n}{2}\left(\log\left(\frac{n}{2\pi}\right) - 1 - \log\left(\left\|X^j - X^{-j}\widehat{\Theta}_j^{GL}(\lambda_j)\right\|^2\right)\right)$ est la log-vraisemblance maximisée en ses paramètres d'un modèle appartenant à $\mathcal{M}_j^{GL}$.
4. Alternative : sélection par la 10-fold validation croisée (CV) d'un λ_j.

— Découpage de $\mathcal{I}$ en 10 sous-ensembles $\mathcal{J}_1, ..., \mathcal{J}_{10}$, avec $|\mathcal{J}_1| = ... = |\mathcal{J}_{10}| = \lfloor \frac{n}{10} \rfloor$ ou $\lfloor \frac{n}{10} \rfloor + 1$ (ici de taille 267).
— Pour $k \in \{1, ..., 10\}$:
  — On pose $\mathcal{J}_k$ l'échantillon test et $\mathcal{J}_k^C$ l'échantillon d'apprentissage : $T = \mathcal{J}_k$ et $A = \mathcal{J}_k^C$.
  — Considération du modèle de régression lié au jeu d'apprentissage : $X_A^j = X_A^{-j}\Theta_{j,A} + \epsilon_{j,A}$ avec X_A matrice X restreinte aux données de A.
  — Estimation des coefficients du modèle par Gauss-LASSO pour les pénalités $\lambda_j \in \Lambda_j$ obtenues sur le jeu entier. Collection de modèles obtenue : $\mathcal{M}_{j,A}^{GL} = \left\{ M_{j,A}^{GL}(\lambda_j) \right\}_{\lambda_j \in \Lambda_j} = \left\{ X_A^j = X_A^{-j} \widehat{\Theta}_{j,A}^{GL}(\lambda_j) + \widehat{\epsilon}_{j,A}^{GL} \right\}_{\lambda_j \in \Lambda_j}$.
— Calcul des erreurs de validation croisée pour chaque pénalité appartenant à Λ_j : $Err_j^{CV}(\lambda_j) = \frac{1}{10} \sum_{k \in \{1,..10\}} ||X_{\mathcal{J}_k}^j - X_{\mathcal{J}_k}^{-j} \widehat{\Theta}_{j,\mathcal{J}_k^C}^{GL}(\lambda_j)||_2$.
— Choix de la pénalité λ_j^{CV} telle que $\lambda_j^{CV} = \underset{\lambda_j \in \Lambda_j}{\text{argmin}} \big(Err_j^{CV}(\lambda_j) \big)$.

5.2 Annexe B : Gauss-LASSO stabilisé

On considère les m sous-modèles de régression d'une variable X_j formés par SS : $\forall k \in \{1, .., m\}$, ${}^{(k)}X^j = {}^{(k)}X^{-j}.{}^{(k)}\Theta_j + {}^{(k)}\epsilon_j$.

— Application de Gauss-LASSO sur chacun d'eux. Obtention de m collections de modèles ${}^{(k)}\mathcal{M}_j^{GL} = \left\{ {}^{(k)}M_j^{GL}({}^{(k)}\lambda_j) \right\}_{{}^{(k)}\lambda_j}$ et ensembles de supports ${}^{(k)}\mathcal{S}_j^{GL} = \left\{ {}^{(k)}\widehat{S}_j^{GL}({}^{(k)}\lambda_j) \right\}_{{}^{(k)}\lambda_j}$.
— $\forall k \in \{1, .., m\}$, choix dans ${}^{(k)}\mathcal{S}_j^{GL}$, du support ${}^{(k)}\widehat{S}_j^{GL}\left({}^{(k)}\lambda_j^{BIC}\right)$ désigné par BIC.
— Calcul du score de chaque variable explicative candidate, à savoir sa fréquence d'apparition dans ces m supports : $\forall j' \in \{1, .., p\} \backslash j$, $\mathbb{S}(j, j') = \frac{1}{m} \sum_{k=1}^{m} 1_{j' \in {}^{(k)}\widehat{S}_j^{GL}({}^{(k)}\lambda_j^{BIC})}$.
— Création du support final avec les variables candidates aux scores plus élevés qu'un seuil s_j à calibrer : $\widehat{S}_j^{GLstab}(s_j) = \left\{ j' \in \{1, .., p\} \backslash j, \text{ tq } \mathbb{S}(j, j') \geq s_j \right\}$.

5.3 Annexe C : Calcul des erreurs de prédictions

Le principe, pour la régression de la variable j, est le suivant :

— Découpage aléatoire du jeu $\mathcal{I}$ en 10 sous-ensembles $\mathcal{J}_1, \mathcal{J}_2, ..., \mathcal{J}_{10}$ de cardinaux $\lfloor \frac{n}{10} \rfloor$ ou $\lfloor \frac{n}{10} \rfloor + 1$ (ici de taille 267).
— Pour $k \in \{1, ..., 10\}$:
  1. On pose $\mathcal{J}_k$ l'échantillon test et $\mathcal{J}_k^C$ l'échantillon d'apprentissage : $T = \mathcal{J}_k$ et $A = \mathcal{J}_k^C$.

2. On considère le modèle de régression lié au jeu d'apprentissage :
   $X_A^j = X_A^{-j}\Theta_{j,A} + \epsilon_A$ avec X_A matrice X restreinte aux données de A.

3. On estime les coefficients du modèle par la procédure Gauss-LASSO + BIC :
   $X_A^j = X_A^{-j}\widehat{\Theta}_{j,A}^{GL}(\lambda_{BIC}) + \epsilon_A^{GL}$, avec :
   — $\widehat{\Theta}_{j,A}^{GL}(\lambda)$ l'estimateur Gauss-LASSO de $\Theta_{j,A}$ pour une pénalité LASSO λ.
   — λ_{BIC} la pénalité choisie par le critère BIC.

4. On estime les coefficients du modèle par le biais de Gauss-LASSO stabilisé :
   — Construction de m sous-ensembles $\{A_1, ..., A_m\}$ de A de taille $\lfloor \frac{|A|}{2} \rfloor$ par Sample Splitting.
   — Mise en place de m sous-modèles de régression au modèle de régression en restreignant les observations à $A_1, A_2, ...$ et A_m.
   — Application de la procédure de sélection Gauss-LASSO + BIC à chacun des m sous-modèles.
   Obtention de m supports $\left\{ {}^{(k)}\widehat{S}_{j,A}^{GL}(\lambda_{BIC}^j) \right\}_{k\in\{1,...,m\}}$.
   — Création du vecteur de scores $\mathbb{S}_A$ tel que $\forall j' \in \{1,..,p\}\backslash j$, $\mathbb{S}_A(j,j') = \frac{1}{m}\sum_{k=1}^m 1_{j' \in {}^{(k)}\widehat{S}_{j,A}^{GL}(\lambda_{BIC}^j)}$
   — Obtention du support $\widehat{S}_{j,A}^{GLstab}(s) = \left\{ j' \in \{1,..,p\}\backslash j,\ \text{tq } \mathbb{S}_A(j,j') \geq s \right\}$.
   — Estimation des coefficients des variables du support par Moindre carrés ordinaires :

$$\widehat{\Theta}_{j,A}^{GLstab}(s) = \underset{\beta}{\text{argmin}} ||X_A^j - X_A^{\widehat{S}_{j,A}^{GLstab}(s)}\beta||_2$$

— Calcul de l'erreur de Cross-Validation pour chacune de ces procédures :
$Err_j^{CV}(\widehat{\Theta}_j) = \frac{1}{10}\sum_{k\in\{1,..10\}} ||X_{\mathcal{J}_k}^j - X_{\mathcal{J}_k}^{-j}\widehat{\Theta}_{j,\mathcal{J}_k^C}||_2$ pour :
— $\widehat{\Theta}_j = \widehat{\Theta}_j^{GLStab}(s)$.
— $\widehat{\Theta}_j = \widehat{\Theta}_j^{GL}(\lambda_{BIC})$.

5.4 Annexe D : Gauss-LASSO enrichi

On considère les m sous-modèles de régression d'une variable X_j formés par SS : $\forall k \in \{1,..,m\}$, ${}^{(k)}X^j = {}^{(k)}X^{-j}.{}^{(k)}\Theta_j + {}^{(k)}\epsilon_j$.

— Application de Gauss-LASSO sur chacun d'eux.
Obtention d'un ensemble de pénalité par sous -modèle : ${}^{(k)}\lambda_j^1 < .... < {}^{(k)}\lambda_j^{{}^{(k)}L_j}$.
Obtention de m collections de modèles ${}^{(k)}\mathcal{M}_j^{GL} = \left\{ {}^{(k)}M_j^{GL}({}^{(k)}\lambda_j) \right\}_{{}^{(k)}\lambda_j}$ et en-

sembles de supports candidats ${}^{(k)}\mathcal{S}_j^{GL} = \left\{ {}^{(k)}\widehat{S}_j^{GL}\left({}^{(k)}\lambda_j\right)\right\}_{{}^{(k)}\lambda_j}$.

— Réunion des m collections en une grande collection $\mathcal{M}_j^{GLenri} = \bigcup_{k=1}^{m} {}^{(k)}\mathcal{M}_j^{GL}$ et des m ensembles de support en un grand $\mathcal{S}_j^{GLenri} = \bigcup_{k=1}^{m} {}^{(k)}\mathcal{S}_j^{GL}$.

— Regroupement des modèles de même dimension en sous-collections :
$\forall D \geq 1, \mathcal{M}_j^{GLenri}(D) = \left\{ M_j \in \mathcal{M}_j^{GLenri}, \left|\widehat{S}_{M_j}\right| = D \right\}$ avec $\widehat{S}_{M_j} \in \mathcal{S}_j^{GLenri}$ support associé à M_j.

— Choix dans chaque sous-collection du modèle de plus grande log-vraisemblance :
$\forall D \geq 1, {}^{(l)}M_j^{max}(D) = \underset{\mathcal{M}_j^{GLenri}(D)}{\operatorname{argmax}} \{V_{max}(M_j)\}$, avec 2 cas possibles ($l = 1$ ou 2)

— $l = 1$, $V_{max}(M_j) = V_{max}^{(n)}(M)$ calculée sur le jeu entier $\mathcal{I}$.
Si $M_j \in {}^{(k)}\mathcal{M}_j^{GL}$ (avec $\widehat{S}_{M_j} \in {}^{(k)}\mathcal{S}_j^{GL}$ support et ${}^{(k)}\lambda_j$ pénalité associés),

$$V_{max}^{(n)}(M_j) = \frac{n}{2}\left(\log\left(\frac{n}{2\pi}\right) - 1 - log\left(\left\|X^j - X^{\widehat{S}_{M_j}}.\widehat{\Theta}_j^{GL}\left({}^{(k)}\lambda_j\right)\right\|^2\right)\right).$$

où $\widehat{\Theta}_j^{GL}\left({}^{(k)}\lambda_j\right)$ est l'estimateur de Θ_j obtenu par moindres carrés ordinaires.

— $l = 2$, $V_{max}(M_j) = V_{max}^{(k)}(M_j)$ calculée sur les sous-jeux de sélection :
Si $M_j \in {}^{(k)}\mathcal{M}_j^{GL}$ (avec $\widehat{S}_{M_j} \in {}^{(k)}\mathcal{S}_j^{GL}$ support et ${}^{(k)}\lambda_j$ pénalité associés),

$$V_{max}^{(k)}(M_j) = \frac{n}{4}\left(\log\left(\frac{n}{4\pi}\right) - 1 - log\left(\left\|{}^{(k)}X^j - {}^{(k)}X^{\widehat{S}_{M_j}}.{}^{(k)}\widehat{\Theta}_j^{GL}\left({}^{(k)}\lambda_j\right)\right\|^2\right)\right).$$

où ${}^{(k)}\widehat{\Theta}_j^{GL}\left({}^{(k)}\lambda_j\right)$ est l'estimateur de ${}^{(k)}\Theta_j$ obtenu par moindres carrés ordinaires.

— Établissement de la meilleure collection de modèles $\mathcal{M}_j^{max}(l) = \left\{{}^{(l)}M_j^{max}(D)\right\}_{D\geq 1}$ et de l'ensemble des supports associés $\mathcal{S}_j^{max}(l) = \left\{{}^{(l)}\widehat{S}_j^{max}(D)\right\}_{D\geq 1}$.

— Sélection du modèle $M_j^{GLenri}(l) \in \mathcal{M}_j^{max}(l)$ et de son support associé $\widehat{S}_j^{GLenri}(l) \in \mathcal{S}_j^{max}(l)$ par l'heuristique de pente. En notant ${}^{(l)}\kappa_j$, la pente de la partie linéaire de la courbe de log-vraisemblances, le modèle choisi vérifie :

$$M_j^{GLenri}(l) = \underset{M_j \in \mathcal{M}_j^{max}(l)}{\operatorname{argmin}} \left(-V_{max}(M_j) + 2 \times {}^{(l)}\kappa_j \times \left|\widehat{S}_{M_j}\right| \log(n)\right).$$

Chapitre 3

Classification des graphes orientés

L'étape de sélection de variables traitée dans le chapitre précédent a permis la construction de graphes orientés $\mathcal{G}$. Les deux méthodes développées ont été appliquées aux données transcriptomes concernant les facteurs de trancription d'*Arabidopsis thaliana*. Nous disposons ainsi de deux graphes représentant le même réseau biologique. Le second objectif statistique de la thèse consiste alors en la classification de ces graphes. Biologiquement, ceci permettrait la classification des facteurs de transcription en groupes de gènes régulateurs et en groupes de gènes régulés. Par classification d'un graphe, nous entendons la répartition de ses noeuds en groupes selon leurs arêtes communes. Les graphes traités étant orientés, deux types de classification des noeuds sont possibles : une première s'appuyant sur les arêtes dirigées vers les noeuds et une seconde sur les arêtes partant des noeuds. Nous utiliserons les modèles à blocs latents pour la classification de ces graphes orientés. Ce modèle, en plus de permettre cette double classification, met en évidence les liens de contrôle entre les deux types de classification. Nous passerons en revue les algorithmes existant permettant l'estimation de ses paramètres et adopterons l'algorithme *V-Bayes*, basé sur une inférence bayésienne pour mener à bien cette estimation. Une fois les doubles classifications effectuées, la difficulté réside en l'évaluation de leur qualité. Plusieurs mesures existent déjà pour évaluer la similitude entre deux couples de partitions. Nous en proposons une nouvelle qui est une extension de l'Ajusted Rand Index, indice visant à comparer deux simples partitions, à des couples de partitions et mettrons en évidence les points forts de cette dernière. Ce nouvel indice, que nous appelons Coclustering Ajusted Rand Index, nous permettra de comparer les classifications effectuées sur nos graphes.

1 Détection de l'hétérogénéité des graphes

Le travail réalisé dans le chapitre précédent a permis la création de deux graphes $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$. Ces graphes orientés sont les représentations que nous proposons du réseau de facteurs de transcription d'*At*. Nous désignerons par la suite :

— une arête sortante d'un noeud comme étant une arête issue de ce noeud et dirigée vers un autre noeud.

— une arête entrante d'un noeud comme étant une arête issue d'un autre noeud et dirigée vers ce noeud.

Les figures 3.1 et 3.2 illustrent la forte hétérogénéité de nos deux graphes. Certains noeuds présentent beaucoup d'arêtes sortantes et d'autres très peu. Le constat est le même pour les arêtes entrantes. On notera aussi que la répartition des arêtes sortantes de $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ est encore plus hétérogène que celle des entrantes.

Dans l'optique de classer les noeuds en groupes selon leurs connectivités, nous cherchons à appliquer sur nos graphes un modèle probabiliste permettant de détecter leur hétérogénéité. Matias et Robin [42] ont passé en revue les méthodes existantes fondées sur une structure latente et permettant d'atteindre cet objectif. Le modèle à blocs stochastiques (SBM) y est tout particulièrement exposé. Issu des travaux de Frank et Harary (1982) [20] et de Holland *et al.* (1983) [29], le SBM permettait à l'origine la classification des noeuds de graphes non orientés en groupes de noeuds homogènes. Plusieurs généralisations importantes du modèle ont vu le jour telles que l'extension du modèle à des graphes pondérés ou orientés [41].

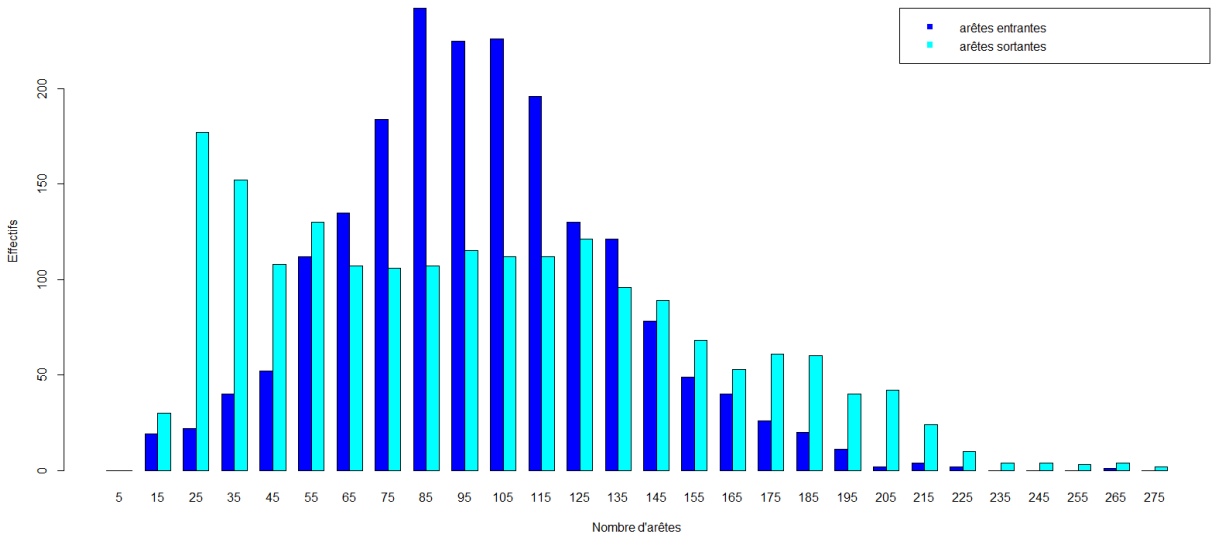

Figure 3.1 – Répartition du nombre d'arêtes des noeuds du graphe construit par Gauss-LASSO stabilisé

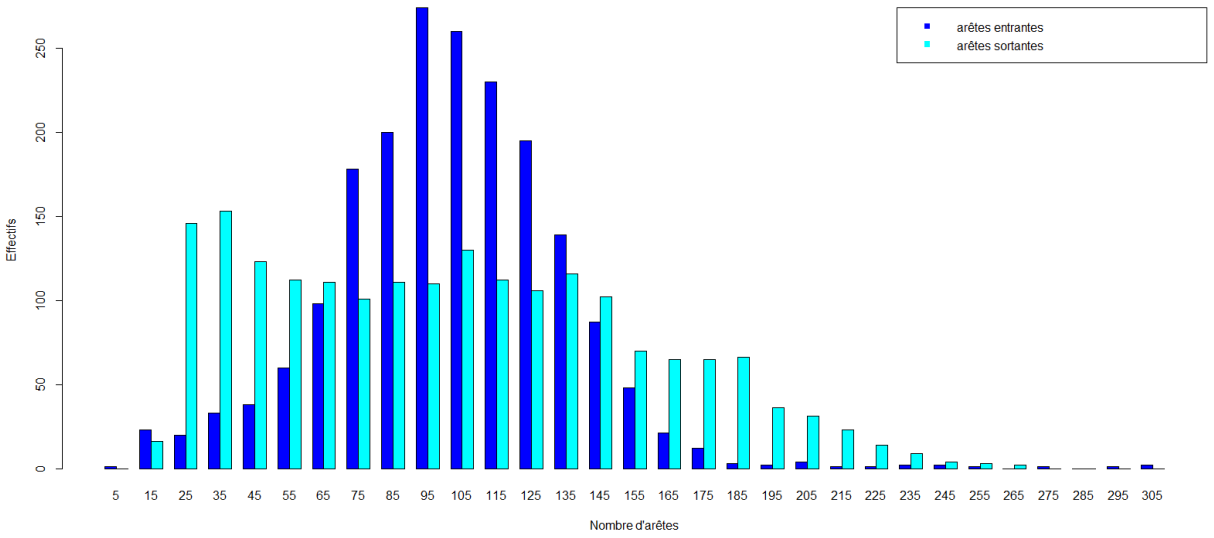

Figure 3.2 – Répartition du nombre d'arêtes des noeuds du graphe construit par Gauss-LASSO enrichi

Cependant ce modèle ne convient pas à notre objectif de double classification. En effet, sa structure latente à un seul type de label résulterait en une classification unique des noeuds. De part notre objectif biologique qui est de classer les facteurs de transcription en groupes de gènes régulés et de gènes régulateurs, nous nous devons d'appliquer aux graphes orientés $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ un modèle fondé sur une structure latente double pour mettre à profit l'information supplémentaire donnée par l'orientation des arêtes. Pour répondre à nos besoins, nous avons recours aux modèles à blocs latents (LBM), également mentionnés dans [42]. Contrairement au SBM, le LBM ne confond pas en un seul un noeud émetteur d'une arête et un noeud receveur d'une arête.

2 Modèles à blocs latents pour des données binaires

Les LBM peuvent être appliqués à des matrices de taille quelconque. Nous nous placerons par la suite dans ce cadre général en considérant une matrice quelconque $A = \{A_{jj'}; j = 1, \ldots, J; j' = 1, \ldots, J'\}$. Les lignes et colonnes de la matrice traitée seront réorganisées selon la double classification établie transformant cette dernière en une matrice présentant des blocs contrastés.

La figure 3.3 illustre un exemple où une matrice binaire de taille $(J, J') = (200, 400)$ est résumée en une matrice par blocs après application du LBM. Un coefficient de la matrice valant 1 est représenté par un point noir. Le modèle classe ici les lignes de la matrice en $H = 3$ groupes homogènes et ses colonnes en $L = 2$ groupes. Le bloc le plus sombre est celui possédant la plus forte densité de coefficients égaux à 1. Nous pourrions en déduire que les éléments du groupe ligne №2 et du groupe colonne №1 sont fortement connectés.

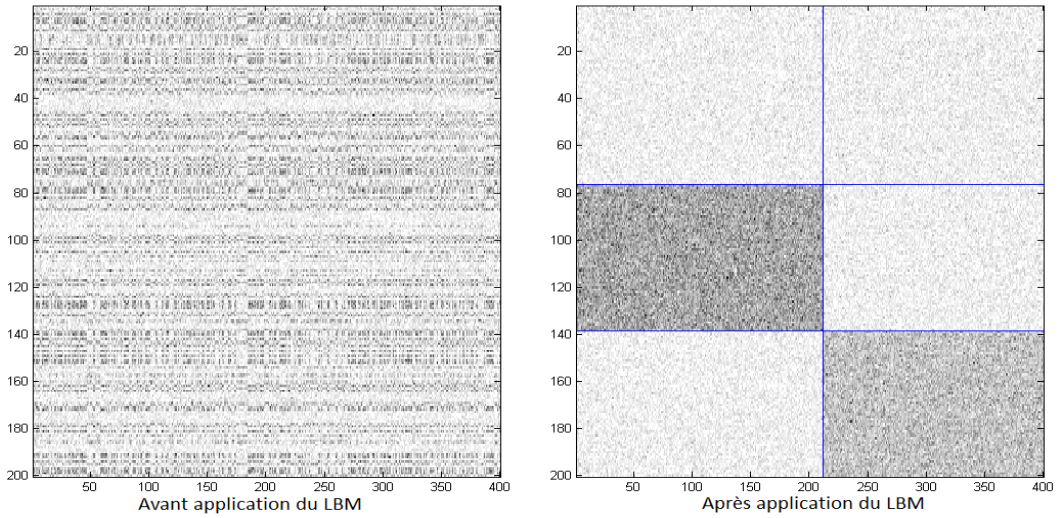

Figure 3.3 – Exemple d'application du LBM sur une matrice binaire

Nous allons dans un premier temps présenter le modèle LBM puis exposer les résultats que nous obtenons sur nos graphes $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$.

2.1 Présentation du modèle

On considère que la matrice A est une réalisation d'une variable aléatoire $C = (C_{jj'})$.

Le LBM, introduit par G.Govaert et M.Nadif [24], repose sur plusieurs hypothèses :

- Il existe une structure en blocs des données et ces blocs sont obtenus par le produit cartésien d'une partition des lignes en H composantes représentée par $v = (v_{jh}; j = 1, \ldots, J; h = 1, \ldots, H)$ et d'une partition des colonnes en L composantes représentée par $w = (w_{j'\ell}; j' = 1, \ldots, J'; \ell = 1, \ldots, L)$. Ces partitions sont définies par :
  — $\forall j \in J, v_{jh} = 1 \Leftrightarrow$ le noeud j appartient à la classe h en ligne.
  — $\forall j' \in J', w_{j'\ell} = 1 \Leftrightarrow$ le noeud j' appartient à la classe ℓ en colonne.
- Les variables latentes V et W sont indépendantes :

$$\forall (v, w) \in \mathcal{V} \times \mathcal{W}, \quad p(v, w) = p(v)p(w),$$

  avec $p(v) = \prod_{j,h} \rho_h^{v_{jh}}$ et $p(w) = \prod_{j',\ell} \tau_\ell^{w_{j'\ell}}$, où $\rho_h = \mathbb{P}(v_{jh} = 1), h = 1, \ldots, H$ et $\tau_\ell = \mathbb{P}(w_{j'\ell} = 1), \ell = 1, \ldots, L$ sont les proportions des composantes en ligne et en colonne.
- Les variables aléatoires $C_{jj'}$ sont indépendantes conditionnellement à v et w. De plus, ces variables $C_{jj'}$ suivent une loi paramétrique notée ϕ qui dépend de la nature des données modélisées. En effet, le LBM peut s'appliquer à plusieurs types de données : aux données binaires ([24]) en utilisant la loi de Bernoulli, aux données

réelles en utilisant la loi gaussienne ([39]), aux données catégorielles en utilisant la loi multinomiale ([35]), aux données ordinales en utilisant le modèle *BOS* (*Binary Ordinal Search*, [7]) et aux données de comptage ([26], [2]) en utilisant la loi de Poisson.

Nos matrices d'adjacence étant binaires, nous considérerons par la suite l'application développée dans [24]. Ainsi, la distribution paramétrique conditionnelle $\phi(A_{jj'};\alpha_{h\ell})$ de la variable $C_{jj'}$ sachant que v_{jh} et $w_{j'\ell}$ valent 1, est supposée être une loi de Bernoulli $\mathcal{B}(\alpha_{h\ell})$ où $\alpha_{h\ell}$ représente l'interaction à l'intérieur du bloc $h\ell$.

La densité conditionnelle pour l'observation $A_{jj'}$ du bloc $h\ell$ s'écrit alors :

$$\phi(A_{jj'};\alpha_{h\ell}) = \alpha_{h\ell}^{A_{jj'}} \times (1-\alpha_{h\ell})^{1-A_{jj'}},$$

Par conséquent, la densité marginale de A peut être vue comme une densité de mélange

$$\begin{aligned} p(A;\theta) &= \sum_{(v,w)\in\mathcal{V}\times\mathcal{W}} p(v;\theta)p(w;\theta)p(A|v,w;\theta) \\ &= \sum_{(v,w)\in\mathcal{V}\times\mathcal{W}} \prod_{j,h} \rho_h^{v_{jh}} \prod_{j',\ell} \tau_\ell^{w_{j'\ell}} \prod_{h,j,j',\ell} \phi(A_{jj'};\alpha_{h\ell})^{v_{jh}w_{j'\ell}}, \end{aligned} \tag{3.1}$$

où $\mathcal{V}$ et $\mathcal{W}$ représentent l'ensemble des partitions possibles pour les lignes et les colonnes, et $\theta = (\rho, \tau, \alpha)$ le vecteur de paramètres du modèle à estimer.

2.2 Estimation des paramètres

Pour effectuer l'estimation des paramètres du LBM, on cherche à évaluer la vraisemblance complétée des données A. L'algorithme *Espérance Maximisation* (*EM*), introduit par Dempster et al. ([14]), est l'algorithme classique d'estimation des paramètres d'un modèle comportant des variables latentes.

2.2.1 Algorithme EM

Après une étape d'initialisation des paramètres du modèle $\theta^{(0)}$, cet algorithme alterne deux étapes successives à chaque itération (d) :

- **Espérance** : La première étape consiste à maximiser l'espérance de la log vraisemblance complétée conditionnellement aux observations A et aux paramètres courants estimés notés $\theta_{H,L}^{(d)}$:

$$\mathcal{Q}(\theta_{H,L}|\theta_{H,L}^{(d)}) = \mathbb{E}\left[\log \ell(A;v,w)|A,\theta_{H,L}^{(d)}\right].$$

Pour les modèles de mélange, cette étape revient à calculer les probabilités conditionnelles, notées $t_{jh}^{(d)}$ et $r_{j'\ell}^{(d)}$, que l'observation $A_{jj'}$ appartient à la composante h en ligne et ℓ en colonne sachant les données A et les paramètres courant du modèle $\theta_{H,L}^{(d)}$

- **Maximisation** : La deuxième étape consiste à évaluer $\theta_{H,L}^{(d)}$, à savoir les paramètres maximisant la quantité $\mathcal{Q}(\theta_{H,L}|\theta_{H,L}^{(d)})$ en $\theta_{H,L}$.

Le bon fonctionnement de l'algorithme est dû à la propriété fondamentale, garantissant que la maximisation de $\mathcal{Q}(\theta_{H,L}|\theta_{H,L}^{(d)})$ permet d'augmenter la log vraisemblance $\ell(A|\theta_{H,L})$. Ainsi, l'algorithme alterne les étapes d'Espérance et de Maximisation et met à jour les paramètres du modèle à chaque itération (d) jusqu'à convergence, parfois lente, vers un maximum local. Celui-ci n'est d'ailleurs pas forcément le maximum global de la fonction de vraisemblance.

Néanmoins, si nous reprenons le calcul de l'espérance conditionnellement aux observations et $\theta^{(d)}$ de la log-vraisemblance complétée utilisé dans l'algorithme EM, nous avons :

$$\mathcal{Q}(\theta;\theta^{(d)}) = \sum_{v,w} \log p(A,v,w;\theta^{(d)}) p(v,w|A;\theta^{(d)})$$

et

$$p(A,v,w;\theta) = p(A|v,w;\theta)p(v,w;\theta).$$

D'où

$$\mathcal{Q}(\theta;\theta^{(d)}) \;=\; \sum_{j,h} r_{jh}^{(d)} \log \rho_h + \sum_{j',\ell} t_{j'\ell}^{(d)} \log \tau_\ell + \sum_{j,h,j',\ell} e_{jhj'\ell}^{(d)} \log \phi(A_{jj'};\alpha_{j'\ell})$$

où :

$$\begin{aligned} r_{jh}^{(d)} &= \mathbb{P}(V_{jh}=1|A;\theta^{(d)}), \\ t_{j'\ell}^{(d)} &= \mathbb{P}(W_{j'\ell}=1|A;\theta^{(d)}), \\ e_{jj'h\ell}^{(d)} &= \mathbb{P}(V_{jh}=1, W_{j'\ell}=1|A;\theta^{(d)}). \end{aligned}$$

Le calcul de $e_{jj'h\ell}$ est infaisable en un temps fini raisonnable car les variables latentes conditionnellement aux observations ne sont pas indépendantes. Pour pallier ce problème, [25] suggèrent l'algorithme VEM qui propose de faire cette approximation d'indépendance conditionnelle.

2.2.2 Algorithme VEM

Le principe de VEM est d'écrire la log-vraisemblance en introduisant une distribution libre des variables latentes $q_{vw}(v,w)$. La log-vraisemblance devient alors :

$$L(\theta) = \underbrace{\mathbb{E}_{(V,W)\sim q_{vw}}\left[\log\left(\frac{p(A,V,W;\theta)}{q_{vw}(V,W)}\right)\right]}_{:=\mathcal{F}(q_{vw};\theta)} + D_{KL}\big(q_{vw}||p(v,w|A;\theta)\big).$$

avec $\mathcal{F}$ l'énergie libre et D_{KL} la divergence de Kullback-Leibler. Cette dernière étant positive, $\mathcal{F}$ est un minorant de la log-vraisemblance. $L(\theta) = \mathcal{F}(q_{vw};\theta)$ si et seulement si $q_{vw} = p(v,w|A;\theta^{(d)})$. En cela, calculer la loi $p(v,w|A;\theta^{(d)})$ dans l'étape E revient à maximiser $\mathcal{F}(q_{vw})$ en q_{vw}. Ceci est possible à condition que la fonction q_{vw} soit recherchée parmi l'ensemble des lois possibles. A partir de là, l'idée est de faire une approximation variationnelle dite en champ moyen, consistant à maximiser l'énergie libre à $\theta^{(d)}$ fixé en supposant que les distributions libres q_{vw} se factorisent comme suit :

$$q_{vw}(v,w) = q_v(v)q_w(w).$$

Cette simplification permet de calculer facilement les mises à jour de la loi. Ainsi, l'énergie libre à maximiser, s'écrit pour une itération (d) :

$$\begin{aligned}\mathcal{F}(q_{vw};\theta^{(d)}) \;=\; & \sum_{j,h} r_{jh}^{(d)} \log\rho_h + \sum_{k,\ell} t_{k\ell}^{(d)} \log\tau_\ell + \sum_{j,k,h,\ell} r_{jh}^{(d)} t_{k\ell}^{(d)} \log\phi(c_{jk};\mu_j\nu_k\gamma_{h\ell}) \\ & - \sum_{j,h} r_{jh}^{(d)} \log r_{jh}^{(d)} - \sum_{k,\ell} t_{k\ell}^{(d)} \log t_{k\ell}^{(d)}.\end{aligned}$$

L'algorithme VEM fonctionne ainsi et est décrit dans son intégralité en Annexe A. Une fois les résultats stabilisés, nous obtenons des probabilités conditionnelles $\widehat{r}_{jh}^{VEM} = \mathbb{P}(V_{jh} = 1|A;\widehat{\theta}^{VEM})$ et $\widehat{t}_{j'\ell}^{VEM} = \mathbb{P}(W_{j'\ell} = 1|A;\widehat{\theta}^{VEM})$ d'appartenance de l'observation $A_{jj'}$ à une classe d'appartenance en ligne et en colonne définies par les partitions v et w. Nous utilisons la règle du Maximum A Posteriori (MAP) pour affecter à cette observation une classe en ligne et une classe en colonnes :

$$\widehat{v}_{jh}^{VEM} = \begin{cases} 1 & \text{si } \arg\max_{h'} \widehat{r}_{jh'}^{VEM} = h \\ 0 & \text{sinon.} \end{cases} \quad \text{et } \widehat{w}_{j'\ell}^{VEM} = \begin{cases} 1 & \text{si } \arg\max_{\ell'} \widehat{t}_{j'\ell'}^{VEM} = \ell \\ 0 & \text{sinon.} \end{cases}$$

L'algorithme VEM par rapport à l'algorithme EM est donc implémentable en pratique. Cependant, l'inconvénient de cet algorithme est qu'il est très sensible à l'initialisation, plus encore que l'algorithme EM.

2.2.3 Échantillonneur de Gibbs

Une autre alternative à l'algorithme VEM, où n'est effectuée aucune approximation est l'échantillonneur de Gibbs proposé par C.Keribin ([34]) pour le LBM. L'étape d'estimation est remplacée par la génération d'un échantillon des données manquantes $(v^{(d)}, w^{(d)})$ sous la loi de ces données conditionnellement aux observations et au paramètre courant $\theta^{(d)}$. On obtient ainsi un pseudo-échantillon complet (étape S). L'étape de maximisation recherche le paramètre maximisant la vraisemblance complétée, dans laquelle les variables manquantes sont remplacées par leur tirage. L'échantillonneur de Gibbs permet alors de remédier au problème de l'impossibilité du calcul de $p(v, w|A, \theta^{(d)})$.

Initié par S.Geman et D.Geman ([23]), cet échantillonneur simule alors une chaîne de Markov irréductible de cette manière :

1. Initialisation de $\theta^{(0)}$ et de $w^{(0)}$.
2. Pour $d = 0 \dots n_{iter}$:
   - Simulation de $v^{(d+1)}$ suivant la loi $p(v|c, w^{(d)}; \theta^{(d)})$.
   - Simulation de $w^{(d+1)}$ suivant la loi $p(w|c, v^{(d+1)}; \theta^{(d)})$.
   - Simulation de $\theta^{(d+1)}$ suivant la loi $p(\theta|c, v^{(d+1)}, w^{(d+1)})$.

L'échantillonneur de Gibbs est moins sensible aux initialisations que l'algorithme VEM. Cependant, V.Brault ([10]) a mis en évidence un nouveau type d'états absorbants pour la chaîne de Markov engendrée par l'algorithme propre à la configuration en blocs du modèle. Pour pallier ce problème et tirer profit de chacun des deux algorithmes présentés, un couplage d'algorithmes Gibbs+VEM peut être envisagé puisque l'échantillonneur de Gibbs fournit en amont une bonne initialisation pour l'algorithme VEM. Néanmoins, C.Keribin et al. ([35]) ont montré que cette combinaison estime parfois un nombre de classes inférieur à celui demandé. Pour contourner cette nouvelle difficulté, ils ont aussi proposé l'algorithme *V-Bayes*, version partiellement bayésienne de l'algorithme VEM.

2.2.4 Algorithme V-Bayes

Dans ce cadre bayésien, θ est supposé aléatoire. Les proportions de mélange sont munies d'une loi a priori de Dirichlet $\mathcal{D}$ et le paramètre α d'une loi bêta $\mathcal{B}e$:

$$\rho \sim \mathcal{D}(a, .., a) \quad \text{et} \quad \tau \sim \mathcal{D}(a, .., a) \quad \text{et} \quad \alpha \sim \prod_{h,l} \mathcal{B}e(b, b)$$

Le même hyperparamètre a est choisi pour toutes les distributions des proportions de mélange afin de ne favoriser aucune composante.

Pour estimer le mode de la loi a posteriori de θ, *V-Bayes* utilise la même démarche que celle de l'algorithme *EM* :

$$\begin{aligned}\log p(A;\theta) &= \mathcal{Q}(\theta|\theta^{(d)}) - H(\theta|\theta^{(d)}) + \log p(\theta) \\ &= \mathcal{F}_B(\theta|\theta^{(d)}) - H(\theta|\theta^{(d)}),\end{aligned}$$

où $p(\theta)$ est la loi a priori de θ définie ci-dessus.

L'algorithme *V-Bayes* cherche donc à maximiser une version "bayésienne" de l'énergie libre $\mathcal{F}(\theta)$ définie par :

$$\mathcal{F}_B(\theta) = \mathcal{F}(\theta) + \log p(\theta).$$

Le détail de l'algorithme est exposé en Annexe B.

C.Keribin et al. ([35]) ont montré que le couplage avec l'échantillonneur de Gibbs en amont, est la meilleure combinaison dans le cadre du modèle des blocs latents sur données catégorielles. En effet, l'échantillonneur de Gibbs permet de fournir une zone pertinente autour du bon mode a posteriori et présente l'avantage supplémentaire suivant : les états qui sont absorbants pour certains algorithmes ne le sont pas pour lui. ([10]).

C'est à l'aide de ce couplage échantillonneur de Gibbs et algorithme *V-Bayes* que nous allons estimer les paramètres de notre LBM appliqué à nos matrices de données. Nous choisissons comme hyperparamètres $a = 4$ et $b = 1$ comme il est préconisé dans [35]. De manière similaire au cadre du VEM, c'est par la règle du MAP que nous allons choisir une classe en lignes et en colonnes pour chaque observation $A_{jj'}$:

$$\widehat{v}_{jh}^{VB} = \begin{cases} 1 & \text{si } \arg\max_{h'} \widehat{r}_{jh'}^{VB} = h \\ 0 & \text{sinon.} \end{cases} \quad \text{et } \widehat{w}_{j'\ell}^{VB} = \begin{cases} 1 & \text{si } \arg\max_{\ell'} \widehat{t}_{j'\ell'}^{VB} = \ell \\ 0 & \text{sinon.} \end{cases}$$

Cependant, il reste une difficulté majeure. La présentation du modèle ainsi que l'estimation de ses paramètres ont été effectuées jusque là avec un nombre de classe en ligne H et en colonnes L fixés. Dans notre cas de figure, ces deux valeurs nous sont inconnues. Nous allons faire appel à l'Integrated Completed Likehood (ICL), critère de sélection pour estimer ces deux nouveaux paramètres.

2.3 Estimation du nombre de classes

2.3.1 Cas des modèles de mélange simples

Dans ce cadre, C.Biernacki et al. ([6]) proposent, comme alternative au critère BIC, le critère ICL qui se situe dans un cadre bayésien. En effet, le critère BIC est consistant lorsque le vrai modèle est dans la collection de modèles considérés ([33]) mais tend à

surestimer le nombre de composantes lorsque le modèle initial n'est pas dans la collection de modèles considérée. L'objectif du critère ICL est de trouver le modèle $\mathcal{M}$ caractérisé avec H composantes maximisant la vraisemblance intégrée complète :

$$(\widehat{H}, \widehat{\mathcal{M}}) \in \underset{\mathcal{M},H}{\operatorname{argmax}} \quad p(c, v^*|\mathcal{M}),$$

avec c matrice d'observations et v^* partition des données.

Ainsi, ICL vise à trouver un nombre de composantes bien séparées. En pratique, v^* est inconnue et on choisit son estimateur $\widehat{v}_{\mathcal{M}}$ par la règle du MAP :

$$\widehat{v}_{\mathcal{M}} \in \underset{v \in \mathcal{V}}{\operatorname{argmax}} \quad p(v|c, \widehat{\theta}_{\mathcal{M}}),$$

où $\widehat{\theta}_{\mathcal{M}}$ est une estimation de θ.

Le critère ICL sélectionne donc le modèle maximisant la log-vraisemblance complète intégrée en ayant remplacé v^* par $\widehat{v}$:

$$ICL(H, \mathcal{M}) = \log p(c, \widehat{v}_{\mathcal{M}}).$$

Pour simplifier les notations, nous remplacerons $ICL(H, \mathcal{M})$ par $ICL(H)$.

2.3.2 Cas des LBM pour données binaires

L'adaptation du critère ICL aux LBM pour des données catégorielles dont le cadre des données binaires est un cas particulier a été réalisée dans [35].

Nous reprenons les lois a priori définies précédemment pour l'explicitation de l'algorithme *V-Bayes* :

$$\rho \sim \mathcal{D}(a, .., a) \quad , \quad \tau \sim \mathcal{D}(a, .., a) \quad \text{et} \quad \forall h, \ell, \quad \alpha_{h\ell} \sim \mathcal{B}e(b, b).$$

Le critère $ICL(H, L)$ sélectionne alors le modèle maximisant la log-vraisemblance intégrée complète, une fois les partitions estimées par *V-Bayes* :

$$ICL(H, L) = \log p(A, \widehat{v}^{VB}, \widehat{w}^{VB}).$$

,
où $\widehat{v}^{VB} = \left\{\widehat{v}^{VB}_{jh}\right\}_{j \in \{1,...,J\}, h \in \{1,...,H\}}$ et $\widehat{w}^{VB} = \left\{\widehat{w}^{VB}_{j'\ell}\right\}_{j' \in \{1,...,J'\}, \ell \in \{1,...,L\}}$.

Pour calculer ce terme, nous utilisons l'hypothèse d'indépendance conditionnelle des partitions v et w par rapport au vecteur de paramètre θ, de façon à ce que :

$$p(A, v, w) = p(A|v, w)\ p(v)\ p(w).$$

Le critère ICL s'écrit ainsi :

$$ICL(H, L) = \log p\big(A|\widehat{v}^{VB}, \widehat{w}^{VB}\big) + \log p\big(\widehat{v}^{VB}\big) + \log p\big(\widehat{w}^{VB}\big). \tag{3.2}$$

La formule explicite du critère avec les lois a priori choisies (cf. [35]) donne :

$$\begin{aligned} ICL(H,L) &= \log\Gamma(H\times a) + \log\Gamma(L\times a) - (H+L)\log\Gamma(a) \\ &+ HL\left(\log\Gamma(2b) - 2\log\Gamma(b)\right) - \log\Gamma(J + H\times a) - \log\Gamma(J' + L\times a) \\ &+ \sum_h \log\Gamma\left(\sum_j \widehat{v}_{jh}^{VB} + a\right) + \sum_\ell \log\Gamma\left(\sum_{j'} \widehat{w}_{j'\ell}^{VB} + a\right) \\ &+ \sum_{h,\ell}\left[\sum_h \log\Gamma\left(b + \sum_{j,j'} \widehat{v}_{jh}^{VB}\widehat{w}_{j'\ell}^{VB}A_{jj'}\right) - \log\Gamma\left(2b + \sum_j \widehat{v}_{jh}^{VB}\sum_{j'}\widehat{w}_{j'\ell}^{VB}\right)\right] \\ &+ \sum_{h,\ell}\log\Gamma\left(b + \sum_j \widehat{v}_{jh}^{VB}\sum_{j'}\widehat{w}_{j'\ell}^{VB} - \sum_{j,j'}\widehat{v}_{jh}^{VB}\widehat{w}_{j'\ell}^{VB}A_{jj'}\right) \end{aligned} \tag{3.3}$$

Pour résumer, l'estimation des paramètres est réalisée par l'algorithme $V - Bayes$ à un nombre de couple de classes (H, L) fixé. Nous pouvons ainsi réaliser cette estimation pour une multitude de tels couples de classes. L'objectif est désormais de calculer la valeur du critère ICL pour ces différents couples et de choisir celui vérifiant :

$$(\widehat{H}^{ICL}, \widehat{L}^{ICL}) = \underset{(H,L)}{\operatorname{argmax}} \quad ICL(H, L).$$

Cependant, étant donné que nous devons effectuer ce calcul sur un nombre de classes en ligne et aussi en colonne, le nombre de couples à explorer devient très grand comparé au cas d'un mélange simple. Il est donc nécessaire d'adopter une stratégie d'exploration plus élaborée que celle du parcours exhaustif de chaque couple. Pour ce faire, nous utiliserons l'adaptation proposée par V.Robert ([52]) de l'approche "K minus 1" (*KM1*), élaborée dans le cas d'un mélange simple dans [5], au cadre des LBMs. Cette stratégie est appelée *Bi-KM1*.

2.3.3 Stratégie d'exploration des couples de classes

Le principe de cette adaptation est le suivant. Elle est basée sur l'hypothèse que le nombre de classes en ligne appartient à $\{H_{min}, \ldots, H_{max}\}$ et celui en colonnes appartient à

$\{L_{min}, \ldots, L_{max}\}$. L'initialisation récursive consiste à partir d'un couple (H, L) à explorer les deux couples $(H+1, L)$ et $(H, L+1)$ comme l'illustre la figure 3.4.

L'initialisation se fait au couple $(H^{(0)}, L^{(0)}) = (H_{min}, L_{min})$. L'estimation des partitions en ligne et en colonne $(\widehat{v}^{VB}, \widehat{w}^{VB})$ se fait par l'échantillonneur de Gibbs couplé à *V-Bayes*. Puis nous calculons $ICL(H_{min}, L_{min})$.

La première étape consiste à estimer les paramètres des modèles $(H_{min}+1, L_{min})$ et $(H_{min}, L_{min}+1)$ via le couplage échantillonneur de Gibbs - *V-Bayes*, de calculer le critère ICL de ces deux couples puis de ne garder que le couple $(H^{(1)}, L^{(1)})$ ayant la meilleure valeur d'ICL. L'autre couple est définitivement éliminé.

De manière analogue, au bout de l'itération d, un couple $(H^{(d)}, L^{(d)})$ est choisi. Le couple $(H^{(d+1)}, L^{(d+1)})$ sera celui parmi les deux couples $(H^{(d)}+1, L^{(d)})$ et $(H^{(d)}, L^{(d)}+1)$ qui présentera la meilleur valeur du critère ICL.

La dernière itération n_{iter} de cet algorithme nous permet de visiter le couple $(H^{(n_{iter})}, L^{(n_{iter})}) = (H_{max}, L_{max})$.

Parmi les $n_{iter}+1$ couples $(H^{(0)}, L^{(0)}); \ldots; (H^{(n_{iter})}, L^{(n_{iter})})$ retenus par l'algorithme *Bi-KM1*, le couple maximisant le critère ICL sera le couple $(\widehat{H}^{ICL}, \widehat{L}^{ICL})$ et le modèle à $\widehat{H}^{ICL}$ classes en lignes et $\widehat{L}^{ICL}$ classes en colonne sera le modèle final que nous garderons.

Remarques :
- En pratique, nous choisissons $(H_{min}, L_{min}) = (2, 2)$.
- Au lieu de parcourir une grille de taille $H_{max} \times L_{max}$, l'algorithme visite dans le pire des cas $H_{max} + L_{max}$ couples de classes en ligne et en colonne.

Pour résumer, la stratégie que nous adoptons pour estimer les paramètres du LBM est la suivante :

1. Stratégie de sélection des couples de nombre de classes (H, L) à explorer : *Bi-KM1*.
2. Pour chacun de ces couples, estimation des paramètres via échantillonneur de Gibbs et *V-Bayes*, des partitions via la règle du MAP puis calcul du critère ICL associé.
3. Choix du couple (H, L) associé maximisant le critère ICL.

Il reste à appliquer le LBM aux matrices d'adjacences de nos graphes orientés $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ puis d'en évaluer les paramètres via la stratégie ci-dessus.

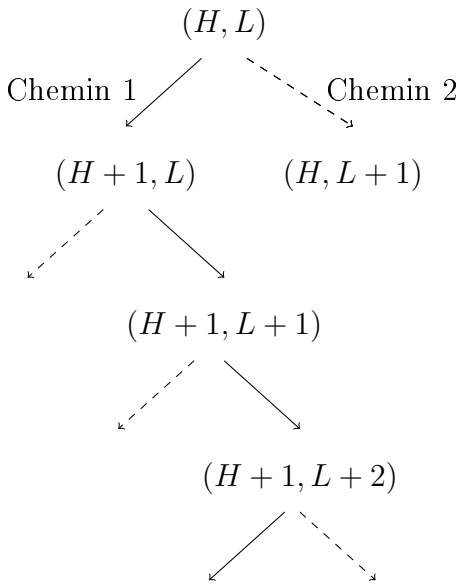

Figure 3.4 – Représentation de l'algorithme *Bi-KM1*.

2.4 Résultats

Dans notre cadre, $J = J' = p$, les matrices d'adjacences A^{GLstab} et A^{GLenri} étant carrées. Les résultats de l'application de LBM sur ces matrices puis de l'estimation de ses paramètres sont les suivants :

- La matrice A^{GLstab} est transformée en une matrice par blocs B^{GLstab} (voir figure 3.5) présentant $\widehat{H}^{GLstab} = 30$ groupes en ligne et $\widehat{L}^{GLstab} = 29$ groupes en colonne.
- La matrice A^{GLenri} est transformée en une matrice par blocs B^{GLenri} (voir figure 3.6) présentant $\widehat{H}^{GLenri} = 26$ groupes en ligne et $\widehat{L}^{GLenri} = 23$ groupes en colonne.

Au vu de ces figures, les matrices d'adjacence réorganisées B^{GLstab} et B^{GLenri} présentent des blocs aux densités de coefficients égaux à 1 très contrastées. Ceci témoigne du fort lien qui existe entre certains groupes de noeuds vus en tant que contrôleurs et d'autres vus en tant que contrôlés. On peut également remarquer que certaines colonnes (respectivement lignes) entières présentent une forte densité de coefficients valant 1 et d'autres une très faible densités. Ceci caractérise la présence de groupes de noeuds possédant des arêtes sortantes (respectivement entrantes) sur une bonne majorité des autres noeuds et de groupes de noeuds n'en possédant que très peu.

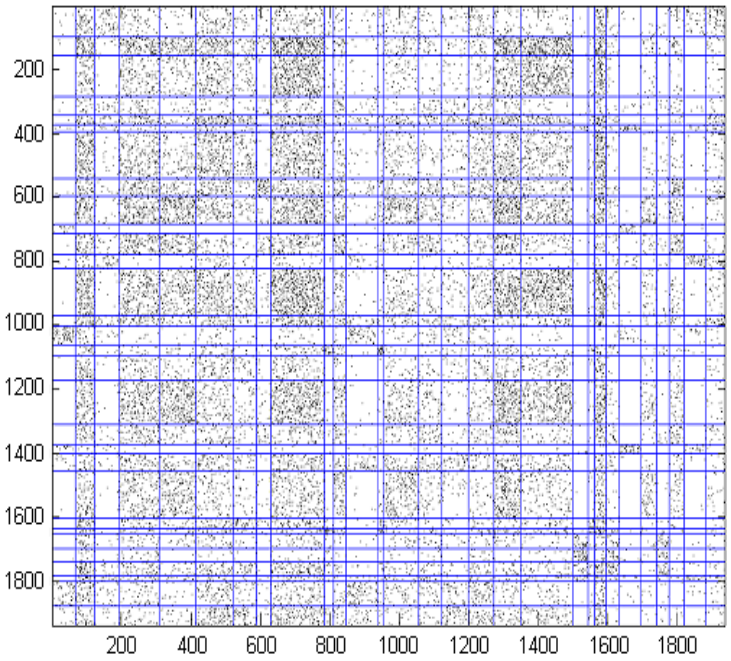

Figure 3.5 – Matrice par blocs B^{GLstab}

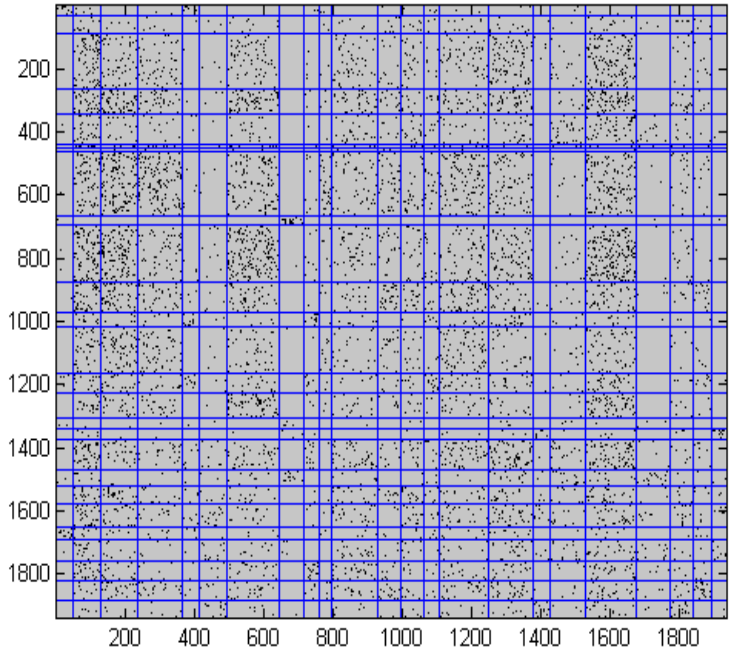

Figure 3.6 – Matrice par blocs B^{GLenri}

Ce constat est logique et rassurant. En effet, si une matrice d'adjacence traitée par le LBM présentait des blocs aux densités de coefficients valant 1 proches les unes des autres, cela aurait caractérisé une forte homogénéité des noeuds du graphe qui lui est associé. Biologiquement, cela aurait caractérisé une structure au niveau de la régulation des FTs sans hiérarchie clairement définie, chose que nous n'espérions pas. La forte hétérogénéité

de nos graphes $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$ traités par le LBM constatée au début du chapitre est donc bien en accord avec l'allure des matrices B^{GLstab} et B^{GLenri}.

Interrogeons nous maintenant sur la différence entre le nombre de classes en ligne et en colonne du modèle $(H^{GLstab} = 30, L^{GLstab} = 29)$ estimé pour $\mathcal{G}^{GLstab}$ et celui $(\widehat{H}^{GLenri} = 26, \widehat{L}^{GLenri} = 23)$ estimé pour $\mathcal{G}^{GLenri}$. Il serait en effet intéressant de savoir si le couple $(H = 26, L = 23)$ pouvait être également un couple candidat crédible pour le graphe $\mathcal{G}^{GLstab}$. Pour jauger de celà, la comparaison entre la valeur du critère ICL calculée pour le couple $(\widehat{H}^{GLstab} = 30, \widehat{L}^{GLstab} = 29)$ maximisant le critère et la valeur calculée pour le couple $(H = 26, L = 23)$ dans le cadre du graphe $\mathcal{G}^{GLstab}$ semblerait être un bon indicateur. Si ces deux valeurs sont proches, un modèle à 26 classes en ligne et 23 en colonne aurait pu être envisagé pour ce graphe.

Cependant, cette comparaison n'est possible que si le modèle au couple de classes $(H = 26, L = 23)$ est un modèle visité par l'algorithme *Bi-KM1* dans le cadre de l'estimation des paramètres du LBM appliqué à A^{GLstab}. La figure 3.7 illustre justement le chemin parcouru par l'algorithme et le couple $(H = 26, L = 23)$ est en effet visité. Les paramètres du modèle présentant ce nombre de classes ont bien été estimés et le critère ICL calculé.

La figure 3.8 présente les valeurs du critère ICL évalué sur les couples du chemin parcouru par *Bi-KM1*. Le couple $(\widehat{H}^{GLstab} = 30, \widehat{L}^{GLstab} = 29)$ maximise bien ce critère mais les couples (H, L) de ce chemin vérifiant $26 < H < 36$ et $23 < L < 33$ présentent des valeurs de critère ICL proches de ce maximum. Ces couples et donc notamment $(H = 26, L = 23)$ auraient tout-à-fait pu convenir en tant que nombre de classes du LBM appliqué à A^{GLstab}. Cet argument atténue la différence observée entre les couples de nombre de classes estimés pour $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$.

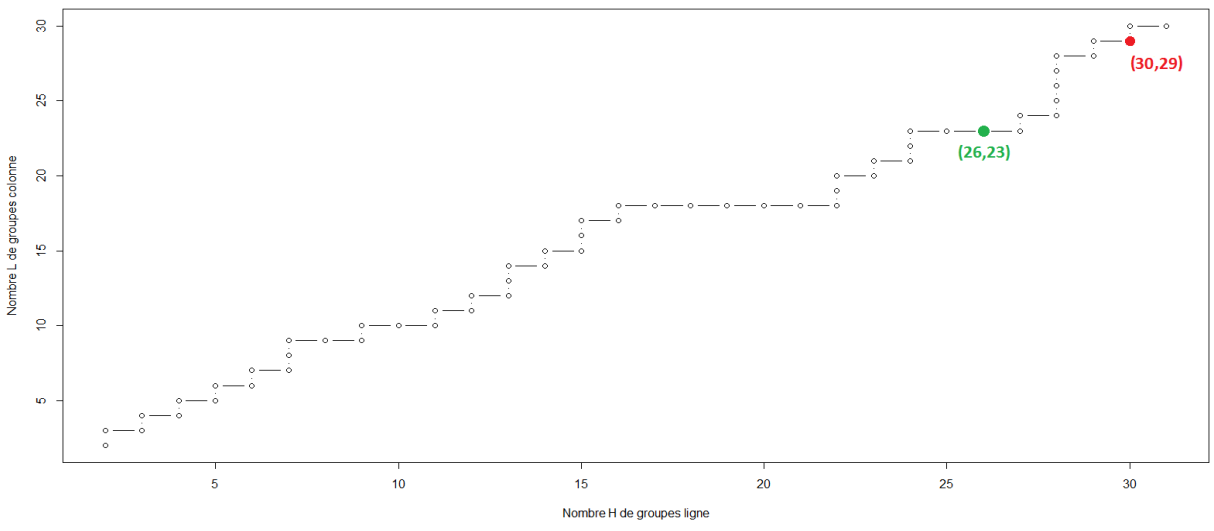

Figure 3.7 – Parcours suivi par l'algorithme *Bi-KM1* pour la matrice A^{GLstab}

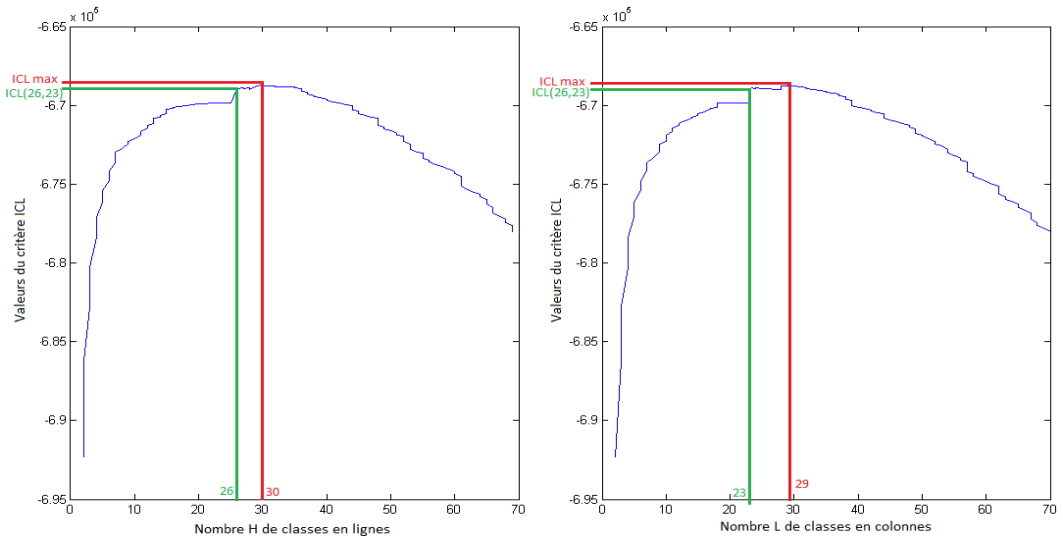

Figure 3.8 – Valeurs d'ICL en fonction de H (à gauche) et de L (à droite) pour $\mathcal{G}^{GLstab}$

Du point de l'estimation du modèle appliqué à $\mathcal{G}^{GLenri}$, la figure 3.10 illustre que la valeur maximale du critère ICL est effectivement atteinte pour le couple ($H^{GLenri} = 26, L^{GLenri} = 23$) mais qu'une petite quantité de couples (H, L) parcouru par *Bi-KM1* autour de celui-ci présentent une valeur du critère ICL quasiment identique à cette valeur maximale. Les couples (H, L) parcourus par l'algorithme pour $21 < H < 28$ et $21 < L < 29$ auraient légitimement pu être également choisis en tant que couples de nombre de classes estimé du LBM appliqué à A^{GLenri}. A noter que le couple $(H = 30, L = 29)$ n'a pas été parcouru par l'algorithme *Bi-KM1* (voir figure 3.9)

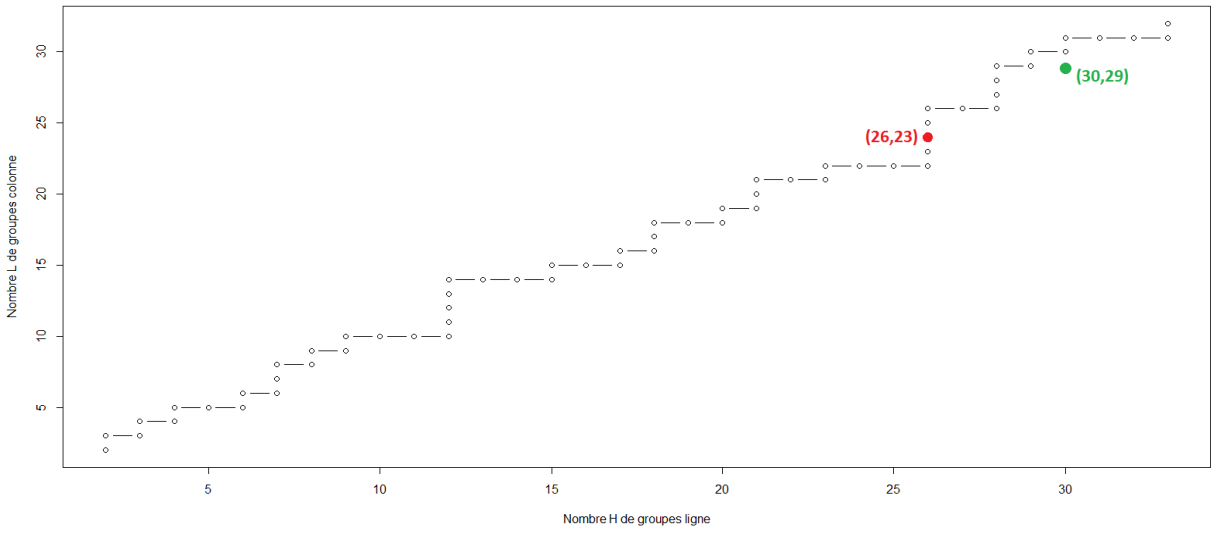

Figure 3.9 – Parcours suivi par l'algorithme *Bi-KM1* pour la matrice A^{GLenri}

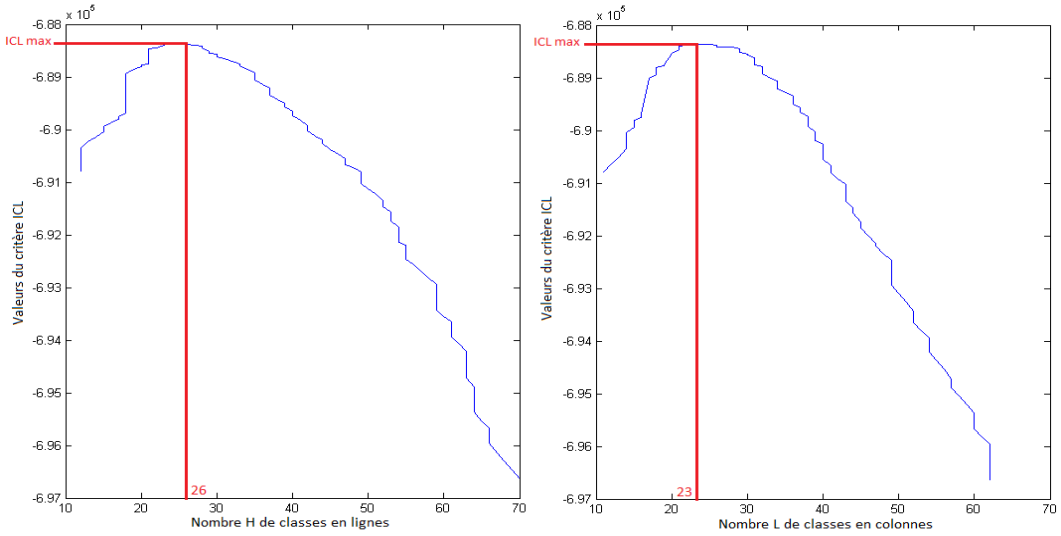

Figure 3.10 – Valeurs d'ICL en fonction de H (à gauche) et de L (à droite) pour $\mathcal{G}^{GLenri}$

L'analyse des résultats du critère ICL nous satisfait. Toutefois, le calcul du critère s'appuie sur les vraies partitions en lignes v^* et en colonnes w^* inconnues. Comme expliqué précédemment, nous pallions ce problème en les remplaçant par les partitions estimées $\widehat{v} = \left\{\widehat{v}_{jh}\right\}_{j,h}$ et $\widehat{w} = \left\{\widehat{w}_{j'l}\right\}_{j',l}$. Ces partitions sont obtenues par la règle du MAP une fois les probabilités d'appartenance $\widehat{r} = \left\{\widehat{r}_{jh}\right\}_{j,h}$ et $\widehat{t} = \left\{\widehat{t}_{j'l}\right\}_{j',l}$ estimées. Pour avoir confiance en le critère ICL, il faut que les classes attribuées aux noeuds par le MAP soient fiables. En effet, pour un noeud j, si ses probabilités d'appartenance aux classes en ligne $\widehat{r}_{j1}, \ldots, \widehat{r}_{jH}$ (respectivement si ses probabilités d'appartenance aux classes en colonne $\widehat{t}_{j1}, \ldots, \widehat{t}_{jL}$) sont proches les unes des autres et par conséquent toutes relativement faibles, le MAP va attribuer à j une classe par défaut à laquelle il a peu de chances d'appartenir. Si la plupart des noeuds sont dans ce cas de figure, les partitions $\widehat{v}$ et $\widehat{w}$ sont erronées et l'évaluation du critère ICL faussée.

Il convient donc de s'assurer, que la plupart des noeuds ont une probabilité d'appartenance à leur groupe en ligne et en colonne, désignés par la règle du MAP, élevée pour les classifications induites dans B^{GLstab} et B^{GLenri}.

Les figures 3.11 et 3.12 représentent la répartition des probabilités d'appartenance de chaque noeud à son groupe en ligne et à son groupe en colonne selon B^{GLstab}. On s'aperçoit qu'une forte majorité de ces probabilités, que ce soit pour les classifications en ligne ou en colonne, sont effectivement élevées confortant ainsi les partitions estimées par la règle du MAP. Le constat dans le cadre de la classification à $H^{GLenri} = 26$ groupes en ligne et $L^{GLenri} = 23$ groupes en colonne du graphe $\mathcal{G}^{GLenri}$ est le même (voir figures 3.13 et 3.14). Ce constat nous rassure donc quant à la fiabilité des valeurs du critère ICL sur les modèles proposés.

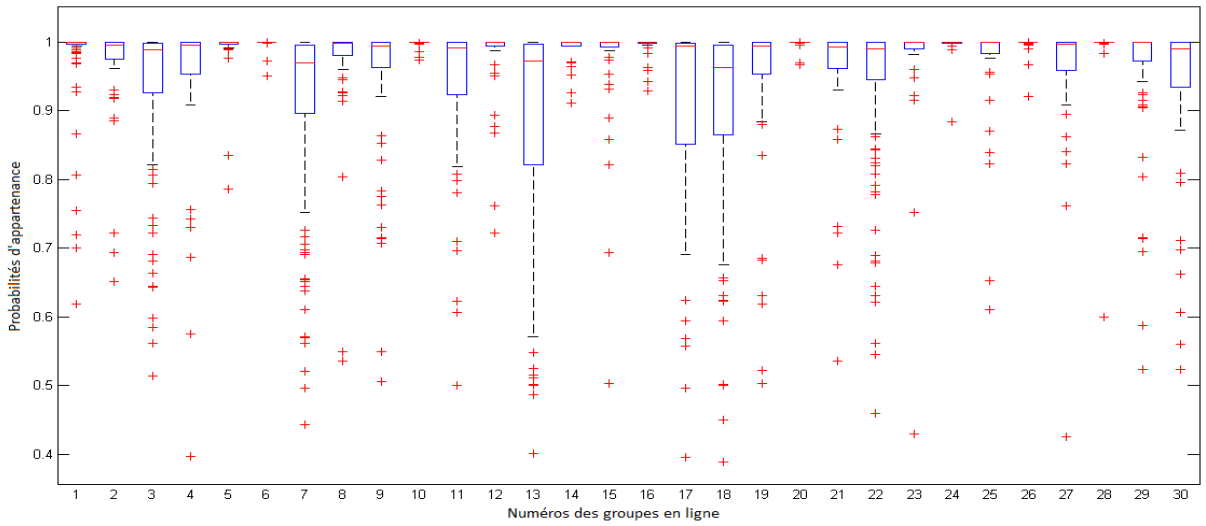

Figure 3.11 – Probabilités d'appartenance des noeuds à leur groupe en ligne pour $\mathcal{G}^{GLstab}$

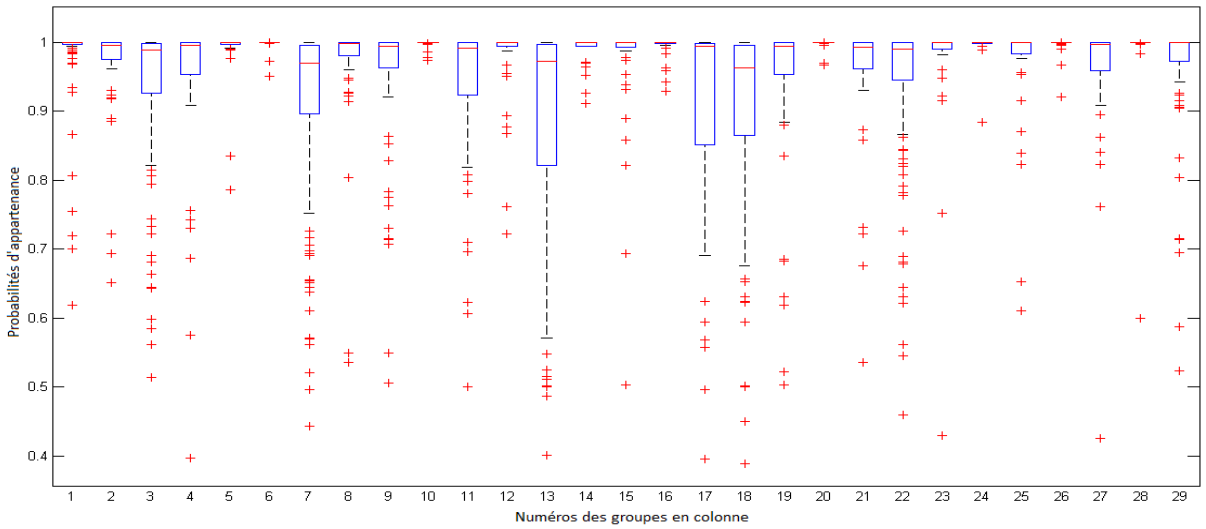

Figure 3.12 – Probabilités d'appartenance des noeuds à leur groupe en colonne pour $\mathcal{G}^{GLstab}$

Une fois ces résultats exposés, il s'agit de trouver un moyen de comparer les partitions proposées par le modèle de classification estimé pour le graphe $\mathcal{G}^{GLstab}$ et par celui du graphe $\mathcal{G}^{GLenri}$. En effet, au chapitre précédent, nous mettions en évidence les quelques différences existant entre les sélections réalisées par Gauss-LASSO enrichi et Gauss-LASSO stabilisé sur notre jeu de données. Nous nous demandions alors si celles-ci allaient avoir un impact conséquent sur l'étape de classification. Les résultats de cette étape, détaillée dans ce chapitre, viennent d'être exposés pour les deux procédures de sélection sur notre jeux de données. Pour attester ou non de l'impact que nous avons mentionné, il s'agit de trouver un moyen de comparer les deux couples de partitions $\left(\widehat{v}^{GLstab}, \widehat{w}^{GLstab}\right)$ et $\left(\widehat{v}^{GLenri}, \widehat{w}^{GLenri}\right)$.

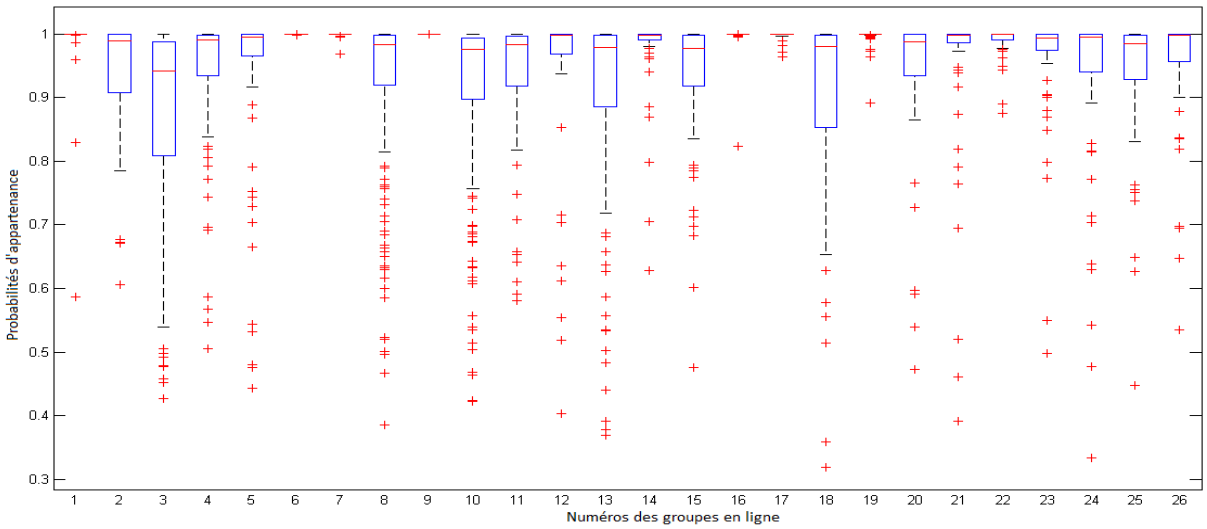

Figure 3.13 – Probabilités d'appartenance des noeuds à leur groupe en ligne pour $\mathcal{G}^{GLenri}$

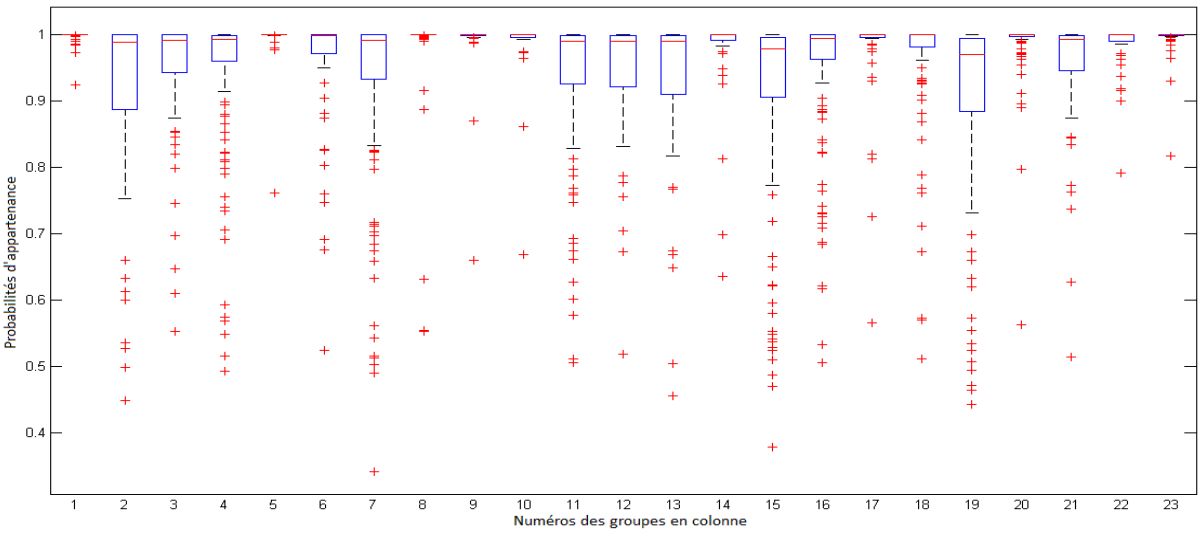

Figure 3.14 – Probabilités d'appartenance des noeuds à leur groupe en colonne pour $\mathcal{G}^{GLenri}$

Quelques méthodes existent déjà pour comparer des couples de partitions deux à deux. Nous en proposons une nouvelle, débatterons de ces avantages et l'utiliserons pour comparer les doubles classifications établies sur nos données.

3 Indice de comparaison des couples de partitions

3.1 Présentation de l'indice

Nous proposons comme indice de comparaison, le *Coclustering Ajusted Rand Index* ($CARI$). Nous voullions étendre l'*Ajusted Rand Index* (ARI) qui s'applique pour la com-

paraison de deux partitions simples, à un indice permettant de comparer les blocs de deux partitions doubles. La mise en place de ce nouvel indice a fait l'objet d'un article soumis que nous avons co-écrit avec V.Robert : [53]. Cet article figure en Annexe C (partie 5) de ce chapitre. Les méthodes de comparaison entre deux doubles classifications y sont passées en revue, l'*ARI* y est détaillé et le nouvel indice en découlant que nous proposons exposé.

Pour résumer, l'*ARI* entre deux partitions z et z' se calcule à partir du tableau de contingence $n^{zz'}$. Dans le cadre de la comparaison entre deux doubles classifications (v, w) et (v', w'), nous pouvons considérer le tableau de contingence $n^{vv'}$ des deux partitons en ligne v et v' et le tableau de contingence $n^{ww'}$ des deux partitions en colonne w et w'. Nous avons démontré que le tableau de contingence $n^{vwv'w'}$ associé aux doubles partitions (v, w) et (v', w') valait $n^{vwv'w'} = n^{vv'} \otimes n^{ww'}$ où $\otimes$ correspond au produit de Kroenecker entre deux matrices. Le *CARI* correspond alors à l'*ARI* appliqué au tableau $n^{vwv'w'}$.

Nous avons également comparé le *CARI* à deux autres indices de référence, l'Erreur de classification (*CE*, [39]) et l'Information mutuelle (*MI*, [65]) étendue et généralisée. Nous avons, entre autres, illustré le fait que le *CARI* pénalisait plus sévèrement les paires de doubles partitions $\big((v, w), (v', w')\big)$, dont les partitions en lignes v et v' ou les partitions en colonne w et w' sont très discordantes, que ces autres deux indices. Ceci représente l'atout majeur du *CARI*. Le détail complet de cette étude est exposé en annexe C.

Illustrons et analysons maitenant l'application du *CARI* aux deux doubles partitions $\big(\widehat{v}^{GLstab}, \widehat{w}^{GLstab}\big)$ et $\big(\widehat{v}^{GLenri}, \widehat{w}^{GLenri}\big)$.

3.2 Résultats

Nous nous sommes dans un premier temps intéressés aux comparaisons entre les partitions en ligne $\widehat{v}^{GLstab}$ et $\widehat{v}^{GLenri}$, puis entre les partitions en colonne $\widehat{w}^{GLstab}$ et $\widehat{w}^{GLenri}$ séparément. Les tableaux de contingence des partitions en ligne, puis des partitions en colonne sont définis dans les tableaux 3.1 et 3.2.

Un coefficient $n_{hh'}^{\widehat{v}\widehat{v}'}$ du tableau 3.1 (resp. $n_{\ell\ell'}^{\widehat{w}\widehat{w}'}$ du tableau 3.2) correspond au nombre de noeuds classés à la fois dans le groupe h de la partition $\widehat{v}^{GLstab}$ et dans le groupe h' de $\widehat{v}^{GLenri}$ (resp. dans le groupe ℓ de $\widehat{w}^{GLstab}$ et ℓ' de $\widehat{w}^{GLenri}$), donc l'ensemble des noeuds j tels que $\widehat{v}_{jh}^{GLstab} = \widehat{v}_{jh'}^{GLenri} = 1$ (resp. des noeuds j' tels que $\widehat{w}_{j'\ell}^{GLstab} = \widehat{w}_{j'\ell'}^{GLenri} = 1$).

Au premier abord, l'allure de ces deux tableaux de contingence est plutôt bonne. Les deux présentent bon nombre de coefficients égaux à 0 ou de faibles valeurs mais aussi quelques coefficients élevés. Ceci irait dans le sens que, globalement, les noeuds d'une classe de la partition $\widehat{v}^{GLstab}$ (resp. $\widehat{w}^{GLstab}$) ne seraient répartis que dans très peu de classes de la partition $\widehat{w}^{GLstab}$ (resp. $\widehat{w}^{GLenri}$) et réciproquement. Notons que ce constat

$\widehat{v}$ \ $\widehat{v'}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	Margin
1	0	5	1	0	59	2	0	0	0	0	0	3	0	0	0	1	0	6	4	11	0	0	5	1	0	0	98
2	0	0	0	11	0	0	1	9	0	24	0	0	3	0	7	0	0	1	0	0	0	0	0	0	0	0	56
3	0	0	17	8	0	2	4	1	0	62	0	0	1	0	35	0	0	1	0	0	0	0	0	0	0	0	131
4	0	12	0	0	3	1	0	0	0	0	1	0	2	2	0	0	0	1	5	8	4	2	6	0	6	4	57
5	0	0	3	1	1	0	0	0	0	2	0	1	0	0	0	0	0	15	0	1	3	0	0	2	2	0	31
6	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	4	0	0	11	0	0	0	2	0	0	0	19
7	0	2	72	8	3	0	0	2	0	13	1	3	11	1	2	0	0	25	0	2	0	0	0	3	1	0	149
8	0	8	1	0	0	0	0	0	0	0	3	1	1	1	0	0	1	2	0	5	3	0	1	0	26	1	54
9	0	0	4	10	0	0	1	38	0	1	17	0	12	1	0	0	0	1	0	1	0	0	0	0	2	0	88
10	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	9	2	0	2	0	0	0	0	0	0	0	30
11	0	2	1	1	0	0	0	7	0	1	36	0	9	1	0	0	0	0	0	2	0	0	0	0	5	0	65
12	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	3	0	0	26	0	0	1	6	0	0	0	43
13	0	0	15	20	0	1	1	11	0	63	1	0	5	0	23	0	0	9	0	0	0	0	0	0	0	0	149
14	0	2	0	0	9	0	0	1	0	0	0	0	1	0	0	1	0	2	0	10	3	0	3	0	0	0	32
15	0	1	0	0	0	0	0	0	2	0	0	0	0	0	0	2	24	0	1	0	2	0	10	0	0	16	58
16	1	0	0	0	0	0	0	0	0	0	0	18	1	4	0	0	0	1	0	0	0	0	0	9	0	0	34
17	0	9	34	2	2	1	0	1	0	2	0	0	6	0	5	1	0	5	0	0	2	0	1	4	1	0	76
18	0	0	2	10	0	0	3	75	0	14	4	0	21	0	10	0	0	1	0	0	0	0	0	0	0	0	140
19	0	6	2	1	1	0	2	0	0	0	11	1	7	1	0	0	0	2	0	3	0	0	0	1	25	0	63
20	0	0	0	0	0	0	0	0	7	0	0	0	0	0	0	15	5	0	1	0	0	0	0	0	0	0	28
21	0	5	1	0	0	0	1	0	0	0	10	0	7	0	0	0	0	1	0	12	9	0	1	0	1	5	53
22	0	1	3	2	0	2	0	55	0	1	13	1	61	3	2	1	0	2	0	1	0	0	0	0	0	0	148
23	0	0	3	0	0	0	1	0	0	0	0	3	2	0	0	0	0	7	0	0	0	0	0	17	0	0	33
24	0	0	0	0	1	0	0	0	0	0	0	7	0	2	0	0	0	1	0	0	0	0	0	3	1	0	15
25	5	0	0	0	0	0	0	0	0	0	1	1	0	21	0	0	0	1	0	0	0	17	0	0	1	0	47
26	24	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	2	0	0	12	0	0	0	0	39
27	1	0	5	1	3	1	0	0	1	0	1	3	0	21	0	0	0	3	0	0	1	4	0	0	0	0	27
28	0	0	0	0	4	2	0	0	0	0	0	0	0	1	0	0	0	0	3	1	0	6	0	0	0	0	17
29	0	3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	6	0	0	2	21	0	28	0	0	18	79
30	0	3	4	0	2	0	0	0	1	0	0	1	1	0	0	0	0	9	0	0	24	0	1	12	0	2	60
Margin	31	59	168	75	96	12	14	200	30	183	99	43	151	60	84	37	38	96	55	59	72	42	64	52	71	46	1937

Table 3.1 – Tableau de contingence $n^{\widehat{v}\widehat{v}'}$ où $\widehat{v} = \widehat{v}^{GLstab}$ et $\widehat{v'} = \widehat{v}^{GLenri}$

est même plus flagrant pour le tableau 3.2 que pour le tableau 3.1 laissant présager que les deux partitions en colonne sont plus en accord que les deux partitions en ligne. Vérifions cette impression à l'aide du calcul de l'ARI et du MI sur nos couples de partitions. Notons que nous ne ferons pas appel au CE car cet indice a pour but de comparer deux partitions possédant un nombre de classes identique, ce qui n'est pas notre cas ici.

	$ARI(.,.)$	$MI(.,.)$
$\left(\widehat{v}^{GLstab}, \widehat{v}^{GLenri}\right)$	0.219	0.451
$\left(\widehat{w}^{GLstab}, \widehat{w}^{GLenri}\right)$	0.408	0.621

Table 3.3 – Résultats des indices de comparaison entre deux simples partitions

Selon le tableau 3.3, les deux indices de comparaison s'accordent à confirmer notre intuition, à savoir une plus forte cohérence entre les partitions en colonne plutôt qu'entre les partitions en ligne. En outre, il est difficile pour un couple de partitions de comparer les valeurs de l'ARI et du MI. En effet, leur construction est différente (voir Annexe 5) et surtout le MI établit une note entre 0 et 1 tandis que l'ARI vaut au maximum 1 lorsque les deux partitions sont identiques à une permutation près, 0 si les noeuds étaient

$\widehat{w}$ \ $\widehat{w'}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	Margin
1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	23	0	0	7	0	0	34	66
2	0	24	2	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	20	0	0	0	0	56
3	0	0	0	0	1	5	0	0	0	0	0	0	0	1	0	0	0	40	0	20	2	0	0	69
4	0	0	9	4	0	0	16	0	0	0	2	0	0	0	65	16	0	0	6	0	0	0	0	118
5	0	0	5	67	0	0	7	0	0	0	0	2	0	0	24	0	0	0	0	0	0	0	0	105
6	0	0	0	0	0	0	8	0	0	0	4	0	2	1	6	80	0	0	7	0	0	0	0	108
7	0	0	0	1	0	0	0	0	3	0	60	0	2	0	0	1	0	0	0	0	0	0	0	67
8	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	4	0	0	1	34	0	41
9	0	24	17	0	0	0	5	0	0	0	0	0	0	0	0	3	0	0	104	0	0	0	0	153
10	0	0	0	0	14	0	0	0	11	0	0	0	0	0	0	0	0	0	0	1	0	0	0	26
11	0	0	1	1	0	0	0	0	0	0	1	1	5	0	26	1	0	0	0	0	0	0	0	36
12	0	0	0	0	0	41	0	0	1	0	0	0	0	0	0	0	24	0	0	24	2	0	1	93
13	0	0	0	0	0	0	0	0	7	0	4	2	0	1	0	0	0	2	0	0	0	0	0	16
14	0	0	0	43	0	0	2	0	0	1	12	14	13	5	9	1	0	0	0	0	1	0	0	101
15	0	0	0	1	0	0	0	0	0	0	4	8	13	6	1	0	0	2	0	0	26	5	0	66
16	0	0	0	0	0	33	0	0	4	0	3	0	0	1	0	0	0	8	0	0	25	2	0	76
17	0	0	0	0	0	0	0	0	0	0	31	1	30	1	0	4	0	1	0	0	5	0	0	73
18	0	3	66	0	0	0	2	0	0	0	0	0	0	0	5	0	0	0	4	0	0	0	0	80
19	0	0	4	4	0	0	112	0	0	0	5	0	0	0	5	15	0	0	5	0	0	0	0	150
20	41	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	42
21	0	0	0	0	0	0	0	0	15	2	3	1	0	0	0	0	0	0	0	0	0	0	0	21
22	0	31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	32
23	0	0	0	1	3	0	0	0	1	0	2	0	0	26	0	0	0	2	0	0	1	2	0	38
24	0	0	0	0	0	0	0	53	0	0	0	0	0	0	0	0	0	0	0	0	0	0	9	62
25	0	0	0	6	0	0	0	0	1	32	0	3	1	1	0	0	0	0	0	0	0	1	0	45
26	2	0	0	0	30	0	0	0	0	0	0	0	0	2	0	0	0	1	0	2	0	0	0	37
27	0	0	0	3	0	0	0	0	0	0	0	32	0	0	0	0	0	0	0	0	0	6	0	41
28	3	0	0	0	0	0	0	16	0	0	0	0	0	0	0	0	0	1	0	44	0	0	0	64
29	0	0	0	0	0	0	0	0	0	0	5	0	1	2	0	0	0	41	0	0	6	0	0	55
Margin	46	82	104	131	49	80	152	71	43	35	136	65	67	47	141	131	47	102	147	98	69	50	44	1937

Table 3.2 – Tableau de contingence $n^{\widehat{w}\widehat{w}'}$ où $\widehat{w} = \widehat{w}^{GLstab}$ et $\widehat{w'} = \widehat{w}^{GLenri}$

répartis aléatoirement dans les classes des deux partitions et est négative lorsque les deux partitions sont très éloignées. On peut néanmoins juger au vu des notes délivrées par ces indices que l'accord entre les classifications établies, sur $\mathcal{G}^{GLstab}$ et $\mathcal{G}^{GLenri}$, des noeuds vus comme contrôleurs est plutôt bonne tandis que les classifications, établies sur ces deux graphes, des noeuds vus comme contrôlés ne sont pas particulièrement proches. Voyons maintenant s'il en est de même pour la comparaison entre les deux couples de partitions.

Posons $\widehat{v} = \widehat{v}^{GLstab}$, $\widehat{v'} = \widehat{v}^{GLenri}$, $\widehat{w} = \widehat{w}^{GLstab}$ et $\widehat{w} = \widehat{w}^{GLenri}$. La matrice de contingence des partitions doubles $n^{\widehat{v}\widehat{w}\widehat{v}'\widehat{w}'}$ correspond au produit $n^{vwv'w'} = n^{vv'} \otimes n^{ww'}$. Cette matrice est de taille $\big((\widehat{H}^{GLstab} \times \widehat{H}^{GLenri}), (\widehat{L}^{GLstab} \times \widehat{L}^{GLenri})\big) = \big((30 \times 26), (29 \times 23)\big) = \big(780, 667\big)$. Elle est donc irréprésentable. La figure 3.15 illustre la répartition de ses coefficients, nous informant ainsi que la majorité d'entre eux sont également nuls voire très faibles. Les valeurs du $CARI$ et du MI étendu et généralisé au cadre des partitions doubles (*Extended MI*) sont affichés dans le tableau 3.4.

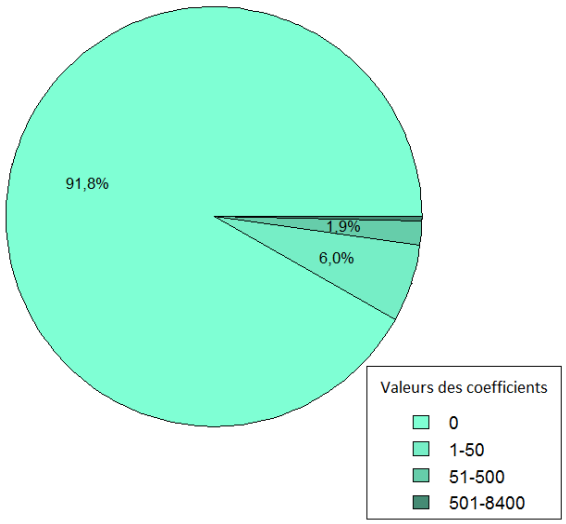

Figure 3.15 – Répartition de la valeur des coefficients de $n^{\widehat{v}\widehat{w}\widehat{v'}\widehat{w'}}$

L'*Extended MI* correspond à la somme de la valeur de l'indice $MI(\widehat{v}, \widehat{v'})$ appliqué aux partitions en ligne et celle de l'indice $MI(\widehat{w}, \widehat{w'})$ appliqué aux partitions en colonne. Sa valeur est donc comprise entre 0 et 2. Divisé par deux, l'*Extended MI* peut être interprété comme étant la moyenne entre $MI(\widehat{v}, \widehat{v'})$ et $MI(\widehat{w}, \widehat{w'})$. Au vu des résultats du MI dans le tableau 3.3, la valeur de l'*Extended MI* divisé par deux dans le tableau 3.4, proche de 0.5 est donc logique. Le $CARI$, basé sur le traitement du tableau de contingence $n^{\widehat{v}\widehat{w}\widehat{v'}\widehat{w'}}$, s'intéresse au contenu des blocs formés par les doubles partitions contrairement à l'*Extended MI*. Comme précisé dans l'annexe 5, le $CARI$ est plus sévère que ce dernier indice surtout lorsque l'un des couples de partitions que ce soit celui en ligne ou celui en colonne, est associé à une faible valeur de l'ARI. C'est ici le cas pour les partitions en ligne $\widehat{v}^{GLstab}$ et $\widehat{v}^{GLenri}$. La valeur du $CARI$ en pâtit ainsi sévèrement puisque celle-ci est proche de 0, appuyant le fait que malgré une bonne cohérence entre les partitions en colonne $\widehat{w}^{GLstab}$ et $\widehat{w}^{GLenri}$, les blocs de la matrice réorganisée B^{GLstab} et ceux de la matrice B^{GLenri} n'ont pas des contenus si proches.

	$CARI\big((.,.),(.,.)\big)$	$\frac{Extended\ MI\big((.,.),(.,.)\big)}{2}$
$\big((\widehat{v}^{GLstab}, \widehat{w}^{GLstab}), (\widehat{v}^{GLenri}, \widehat{w}^{GLenri})\big)$	0.115	0.536

Table 3.4 – Résultats des indices de comparaison entre deux partitions doubles

Nous obtenons donc la réponse à l'interrogation posée à la fin du chapitre précédent. Nous savions que malgré la mise en place de deux procédures de sélection stables, quelques différences subsistaient au niveau des graphes formés par application de ces deux procédures à nos données. Le modèle de classification de graphes orientés utilisé a permis de classer les noeuds, pour chacun de ces graphes, en groupes de noeuds contrôleurs et

groupes de noeuds contrôlés. Les groupes de noeuds contrôleurs établis à partir de chacune des deux procédures de sélection s'accordent bien. Néanmoins les contenus des blocs, à savoir des partitions doubles, formés pour chacune des deux procédures ne sont pas très proches. L'étape de classification n'a donc fait qu'accentuer les différences observées lors de l'étape de sélection.

4 Annexes : algorithmes d'estimation des paramètres

4.1 Annexe A : Algorithme VEM

1. Initialisation de $\theta^{(0)}$, de $r_{jh}^{(0)}$ et de $t_{j'\ell}^{(0)}$.
2. Pour $d = 0 \dots n_{iter}$ (nombre d'itérations choisi) :
   - Étape VE : maximisation alternée de l'énergie libre à $\theta^{(d)}$ fixé avec $t_{j'\ell}^{(t=0)} = t_{j'\ell}^{(d)}$ et $r_{jh}^{(t=0)} = t_{jh}^{(d)}$:
     - calcul de $r_{jh}^{(d+1)}$ à $t_{j'\ell}^{(d)}$ et à $\theta^{(d)}$ fixés :

$$r_{jh}^{(d+1)} = \frac{\rho_h^{(d)} \prod_\ell \left(\left(\alpha_{hl}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)} A_{jj'}} \left(1-\alpha_{hl}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)}(1-A_{jj'})} \right)}{\sum_{h'} \rho_{h'}^{(d)} \prod_\ell \left(\left(\alpha_{h'l}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)} A_{jj'}} \left(1-\alpha_{h'l}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)}(1-A_{jj'})} \right)}$$

     - calcul de $t_{k\ell}^{(d+1)}$ à $r_{jh}^{(d+1)}$ et à $\theta^{(d)}$ fixés :

$$t_{j'l}^{(d+1)} = \frac{\tau_\ell^{(d)} \prod_h \left(\left(\alpha_{hl}^{(d)}\right)^{\sum_j r_{jh}^{(d+1)} A_{jj'}} \left(1-\alpha_{hl}^{(d)}\right)^{\sum_j r_{jh}^{(d+1)}(1-A_{jj'})} \right)}{\sum_{\ell'} \tau_{\ell'}^{(d)} \prod_h \left(\left(\alpha_{h\ell'}^{(d)}\right)^{\sum_{j'} r_{jh}^{(d+1)} A_{jj'}} \left(1-\alpha_{h\ell'}^{(d)}\right)^{\sum_{j'} r_{jh}^{(d+1)}(1-A_{jj'})} \right)}$$

     $\Longrightarrow$ Obtention des probabilités $r_{jh}^{(d+1)}$ et $t_{j'\ell}^{(d+1)}$.

   - Étape M : calcul du paramètre $\theta^{(d+1)}$:

$$\rho_h^{(d+1)} = \frac{\sum_{j=1}^{J} r_{jh}^{(d+1)}}{J}, \quad \tau_l^{(d+1)} = \frac{\sum_{j'=1}^{J'} t_{j'\ell}^{(d+1)}}{J'} \quad , \alpha_{h\ell}^{(d+1)} = \frac{\sum_{j=1}^{J}\sum_{j'=1}^{J'} r_{jh}^{(d+1)} t_{j'\ell}^{(d+1)} A_{jj'}}{\sum_{j=1}^{J} r_{jh}^{(d+1)} \sum_{j'=1}^{J'} t_{j'\ell}^{(d+1)}}.$$

3. Obtention d'un estimateur $\widehat{\theta}^{VEM} = \theta^{(n_{\text{iter}})}$.

Le critère d'arrêt considéré est le minimum entre n_{iter} et l'itération pour laquelle l'énergie libre n'évolue plus à un seuil près.

4.2 Annexe B : Algorithme V-Bayes

1. Initialisation de $\theta^{(0)}$, de $r_{jh}^{(0)}$ et de $t_{j'\ell}^{(0)}$.
2. Pour $d = 0 \dots n_{iter}$ (nombre d'itérations choisi) :
   - Étape VE : maximisation alternée de l'énergie libre à $\theta^{(d)}$ fixé avec $t_{j'\ell}^{(t=0)} = t_{j'\ell}^{(d)}$ et $r_{jh}^{(t=0)} = t_{jh}^{(d)}$:
     - calcul de $r_{jh}^{(d+1)}$ à $t_{j'\ell}^{(d)}$ et à $\theta^{(d)}$ fixés :

$$r_{jh}^{(d+1)} = \frac{\rho_h^{(d)} \prod_\ell \left(\left(\alpha_{hl}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)} A_{jj'}} \left(1 - \left(\alpha_{hl}^{(d)}\right)\right)^{\sum_{j'} t_{j'\ell}^{(d)} (1 - A_{jj'})} \right)}{\sum_{h'} \rho_{h'}^{(d)} \prod_\ell \left(\left(\alpha_{h'l}^{(d)}\right)^{\sum_{j'} t_{j'\ell}^{(d)} A_{jj'}} \left(1 - \left(\alpha_{h'l}^{(d)}\right)\right)^{\sum_{j'} t_{j'\ell}^{(d)} (1 - A_{jj'})} \right)}$$

     - calcul de $t_{k\ell}^{(d+1)}$ à $r_{jh}^{(d+1)}$ et à $\theta^{(d)}$ fixés :

$$t_{j'l}^{(d+1)} = \frac{\tau_\ell^{(d)} \prod_h \left(\left(\alpha_{hl}^{(d)}\right)^{\sum_j r_{jh}^{(d+1)} A_{jj'}} \left(1 - \left(\alpha_{hl}^{(d)}\right)\right)^{\sum_j r_{jh}^{(d+1)} (1 - A_{jj'})} \right)}{\sum_{\ell'} \tau_{\ell'}^{(d)} \prod_h \left(\left(\alpha_{h\ell'}^{(d)}\right)^{\sum_{j'} r_{jh}^{(d+1)} A_{jj'}} \left(1 - \left(\alpha_{h\ell'}^{(d)}\right)\right)^{\sum_{j'} r_{jh}^{(d+1)} (1 - A_{jj'})} \right)}$$

     $\Longrightarrow$ Obtention des probabilités $r_{jh}^{(d+1)}$ et $t_{j'\ell}^{(d+1)}$.

   - Étape M : calcul du paramètre $\theta^{(d+1)}$:

$$\rho_h^{(d+1)} = \frac{a - 1 + \sum_{j=1}^{J} r_{jh}^{(d+1)}}{J + H(a-1)}, \quad \tau_l^{(d+1)} = \frac{a - 1 + \sum_{j'=1}^{J'} t_{j'\ell}^{(d+1)}}{J' + L(a-1)} \quad ,$$

$$\alpha_{h\ell}^{(d+1)} = \frac{b - 1 + \sum_{j=1}^{J} \sum_{j'=1}^{J'} r_{jh}^{(d+1)} t_{j'\ell}^{(d+1)} A_{jj'}}{2(b-1) + \sum_{j=1}^{J} r_{jh}^{(d+1)} \sum_{j'=1}^{J'} t_{j'\ell}^{(d+1)}}.$$

3. Obtention d'un estimateur $\widehat{\theta}^{VB} = \theta^{(n_{iter})}$.

Le critère d'arrêt considéré est le minimum entre n_{iter} et l'itération pour laquelle l'énergie libre n'évolue plus à un seuil près.

Remarques :

1. Dans l'étape VE, [25] ont montré qu'une seule étape alternée suffit à l'algorithme VEM pour se stabiliser. Nous utilisons aussi ce constat pour *V-Bayes*.
2. Les formules de mises à jour dans l'étape M correspondent aux formules de mises à jour de VEM pour les hyperparamètres $a = b = 1$.

5 Annexe C : Comparing high dimensional partitions, with the Coclustering Adjusted Rand Index

Abstract. The popular Adjusted Rand Index (ARI) is extended to the task of simultaneous clustering of the rows and columns of a given matrix. This new index called Coclustering Adjusted Rand Index (CARI) remains convenient and competitive facing other indices. Indeed, partitions with high number of clusters can be considered and it does not require any convention when the numbers of clusters in partitions are different. Experiments on simulated partitions are presented and the performance of this index to measure the agreement between two pairs of partitions is assessed. Comparison with other indices is discussed.

5.1 Introduction

With the advent of large datasets in statistics, coclustering arouses a genuine interest for last years in many fields of applications (text mining with [15], genomics with ([32], [2]), recommendation systems with [57], [65], and so on ...). Initiated by [27], this useful technique aims at reducing the data matrix in a simpler one with the same structure. Indeed, taking profit of the two-dimensional nature of the issue, it enables to provide a simultaneous partition of two sets A (rows, objects, observations, individuals) and B (columns, variables, attributes). To assess the performances of coclustering, partitions obtained by the procedure need to be evaluated. Objective criteria are therefore required to measure how close are these partitions to a reference. On the one hand, [12] suggest a first solution and artificially extend several standard indices from clustering (Dunn index, Baker and Hubert index, Davies and Bouldin index, Calinsky and Harabsz index, Silhouette de Rousseeuw index, Hubert and Levin index, Krzanowski and Lai index and differential method). [65] also extend in the same way, another index relied on the normalized mutual information measure introduced in [62]. However, proceeding in such a manner by just defining a linear combination between the index for row partitions and the index for column partitions, the coclustering structure is not preserved. On the other hand, [39] propose a distance dedicated to coclustering. Nevertheless, the computation of this index is dependent on the number of partition permutations and this property makes it time-consuming so that numbers of clusters can barely exceed nine in each direction. Moreover, no convention is given when the number of clusters of compared partitions is different. The aim of the present paper is to go further and to adapt the very popular and consensual *Adjusted Rand Index* (ARI) developed by [31] from a coclustering point of view. To challenge other indices and tackle the problem of high dimensional partitions with numbers of clusters possibly different, this new index takes into account the coclus-

tering structure while its computation remains time saving. The paper is organized as follows. In the next section, the *Adjusted Rand Index* (ARI) on which our index is based on, is presented. In Section 3, the *Coclustering Adjusted Rand Index* ($CARI$) is detailed and its properties ensuring its efficiency are demonstrated. In Section 4, this new index is exemplified on some partitions. Section 5 is devoted to numerical experiments to illustrate the behaviour of the index and a comparison with other coclustering indices. Finally a conclusion section ends this paper.

5.2 Statistical framework

In order to assess clustering results, objective criteria are required. For this purpose, distances of agreement between two partitions are developed. We will present the popular measure on which we base our new criterion.

5.2.1 Notation

Let two partitions be $\boldsymbol{z} = (z_1, \ldots, z_H)$ and $\boldsymbol{z'} = (z'_1, \ldots, z'_{H'})$ on a set $A = \{O_1, \ldots, O_I\}$, with Card(A)=I. $\boldsymbol{z}$ denotes for example an external reference and $\boldsymbol{z'}$ a clustering result.

5.2.2 The Rand Index and the Adjusted Rand Index

The *Rand Index* (RI) developed by [50], is a measure of the similarity between two data clusterings $\boldsymbol{z}$ and $\boldsymbol{z'}$, and is calculated as follows :

$$\frac{a+d}{a+b+c+d} = \frac{a+d}{\binom{I}{2}}, \tag{3.4}$$

where,

- a denotes the number of pairs of elements that are placed in the same cluster in $\boldsymbol{z}$ and in the same cluster in $\boldsymbol{z'}$,
- b denotes the number of pairs of elements in the same cluster in $\boldsymbol{z}$ but not in the same cluster in $\boldsymbol{z'}$,
- c denotes the number of pairs of elements in the same cluster in $\boldsymbol{z'}$ but not in the same cluster in $\boldsymbol{z}$,
- d denotes the number of pairs of elements in different clusters in both partitions.

The values a and d can be interpreted as agreements, and b and c as disagreements.

To compute all these values, a contingency table can be introduced. Let $\boldsymbol{n^{zz'}} = (n^{zz'}_{h,h'})_{H\times H'}$ be the matrix where $n^{zz'}_{h,h'}$ denotes the number of elements of the set A which belong both the cluster z_h and the cluster $z'_{h'}$. The row and column margins $n^{zz'}_{h,\bullet}$ and $n^{zz'}_{\bullet,h'}$ denote respectively the number of elements in the cluster z_h and $z'_{h'}$. We have the following correspondence [54] :

— $a = \sum_{h}^{H}\sum_{h'}^{H'}\binom{n_{h,h'}^{zz'}}{2} = \frac{\sum_{h}^{H}\sum_{h'}^{H'}\left(n_{h,h'}^{zz'}\right)^2 - I}{2}$,

— $b = \sum_{h}^{H}\binom{n_{h,\cdot}^{zz'}}{2} - a = \frac{\sum_{h}^{H}(n_{h,\cdot}^{zz'})^2 - \sum_{h}^{H}\sum_{h'}^{H'}\left(n_{h,h'}^{zz'}\right)^2}{2}$,

— $c = \sum_{h'}^{H'}\binom{n_{\cdot,h'}^{zz'}}{2} - a = \frac{\sum_{h'}^{H'}(n_{\cdot,h'}^{zz'})^2 - \sum_{h}^{H}\sum_{h'}^{H'}\left(n_{h,h'}^{zz'}\right)^2}{2}$,

— $d = \binom{I}{2} - a - b - c = \sum_{h}^{H}\binom{n_{h,\cdot}^{zz'}}{2} - \sum_{h'}^{H'}\binom{n_{\cdot,h'}^{zz'}}{2} + a = \frac{\sum_{h}^{H}\sum_{h'}^{H'}\left(n_{h,h'}^{zz'}\right)^2 + I^2 - \sum_{h}^{H}\left(n_{h,\cdot}^{zz'}\right)^2 - \sum_{h'}^{H'}\left(n_{\cdot,h'}^{zz'}\right)^2}{2}$.

This symmetric index lies between 0 and 1 and takes the value 1 when the two partitions agree perfectly up to a permutation. Thus, by comparing pairs of elements, this index does not need to review all the permutations of studied partitions and its computation is efficient.

Although, the expected value of the *Rand Index* for two random partitions does not take a constant value and its taken values are concentrated in a small interval close to 1 ([44]). The *Adjusted Rand Index* (ARI) proposed by [31] enables to overcome such drawbacks. This corrected version assumes the generalized hypergeometric distribution as the model of randomness, that is to say partitions are chosen randomly such that the number of elements in the clusters are fixed. The general form of this index which is the normalized difference between the *Rand Index* and its expected value under the generalized hypergeometric distribution assumption, is as follows :

$$\text{ARI} = \frac{\text{Index-Expected Index}}{\text{MaxIndex-Expected Index}}. \tag{3.5}$$

This index is bounded by 1, and takes this value when the two partitions are equal up to a permutation. It can also take negative values, which corresponds to a less agreement than expected by chance.

From Equation 3.5, [31] show the index can be written in this way :

$$\text{ARI}(\boldsymbol{z}, \boldsymbol{z'}) = \frac{\sum_{h,h'}\binom{n_{h,h'}^{zz'}}{2} - \sum_{h}\binom{n_{h,\cdot}^{zz'}}{2}\sum_{h'}\binom{n_{\cdot,h'}^{zz'}}{2}/\binom{I}{2}}{\frac{1}{2}\left[\sum_{h}\binom{n_{h,\cdot}^{zz'}}{2} + \sum_{h'}\binom{n_{\cdot,h'}^{zz'}}{2}\right] - \left[\sum_{h}\binom{n_{h,\cdot}^{zz'}}{2}\sum_{h'}\binom{n_{\cdot,h'}^{zz'}}{2}\right]/\binom{I}{2}} \tag{3.6}$$

$$= \frac{2(ad - bc)}{b^2 + c^2 + 2ad + (a+d)(b+c)}. \tag{3.7}$$

Like the RI, the ARI is symmetric, that is to say $\text{ARI}(\boldsymbol{z}, \boldsymbol{z'})$=$\text{ARI}(\boldsymbol{z'}, \boldsymbol{z})$. Indeed, when the $\text{ARI}(\boldsymbol{z'}, \boldsymbol{z})$ is considered, the associated contingency table is $t(\boldsymbol{n}^{\boldsymbol{zz'}})$, where t denotes the tranpose of a matrix. Besides, in the expression 3.7 of the ARI, the margins of the contingency table work in a symmetric way. That is why, while considering $\boldsymbol{n}^{\boldsymbol{zz'}}$ or its

tranpose matrix $t(\boldsymbol{n}^{\boldsymbol{zz'}})$, the ARI remains unchanged. This remark would be particularly interesting in the next section, when the new index we develop is studied.

5.3 The Coclustering Adjusted Index

We extend the *Adjusted Rand Index* from a coclustering point of view to compare two coclustering partitions which define blocks, and not clusters anymore.

5.3.1 Notation

Let two partitions be $\boldsymbol{z} = (z_1, \ldots, z_h, \ldots z_H)$ and $\boldsymbol{z'} = (z'_1, \ldots, z'_{h'}, \ldots, z'_{H'})$ on a set A and let two partitions be $\boldsymbol{w} = (w_1, \ldots, w_\ell, \ldots, w_L)$ and $\boldsymbol{w'} = (w'_1, \ldots, w_{\ell'}, \ldots, w'_{L'})$ on a set B. $(\boldsymbol{z}, \boldsymbol{w})$ and $(\boldsymbol{z'}, \boldsymbol{w'})$ are two coclustering partitions on the set $A \times B$ where an observation is denoted by $x_{ij}, i = 1, \ldots, I; j = 1, \ldots, J$, with $\text{Card}(A \times B) = I \times J$. Notice that $\boldsymbol{z}$ and $\boldsymbol{z'}$ are called row partitions. Similarly, $\boldsymbol{w}$ and $\boldsymbol{w'}$ are called column partitions.

5.3.2 The Coclustering Adjusted Rand Index

Definition 5.1 *The contingency table* $\boldsymbol{n}^{\boldsymbol{zwz'w'}} = (n_{p,q}^{zwz'w'})_{(H\times L)\times(H'\times L')}$ *is defined such as* $n_{p,q}^{zwz'w'}$ *denotes the number of observations of the set* $A \times B$ *which belongs to the block* p *(related to a pair* (h, ℓ)*) defined by* $(\boldsymbol{z}, \boldsymbol{w})$ *and the block* q *(related to a pair* (h', ℓ') *defined by* $(\boldsymbol{z'}, \boldsymbol{w'})$*).*

The contingency table can be seen as a block matrix which consists of $H \times H'$ blocks of size $L \times L'$ (see Table 3.5).

Notice that a bijection can be defined between the index p of the rows of the contingency table, and the block (h, ℓ) defined by $(\boldsymbol{z}, \boldsymbol{w})$.

An analogous correspondence is defined for the index q and the block (h', ℓ') defined by $(\boldsymbol{z'}, \boldsymbol{w'})$. Thus the notation $(h_p\ \ell_p)$ and $(h'_q\ \ell'_q)$ could be used. We will see afterwards, this trick enables us to describe $\boldsymbol{n}^{\boldsymbol{zwz'w'}}$ in such a convenient way.

Definition 5.2 *Let* $\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{z'}, \boldsymbol{w'}$ *and* $\boldsymbol{n}^{\boldsymbol{zwz'w'}}$ *specified as in Definition 5.1. The Coclustering Adjusted Rand Index (CARI) is defined as follows :*

$$CARI((\boldsymbol{z}, \boldsymbol{w}), (\boldsymbol{z'}, \boldsymbol{w'})) = \frac{\sum_{p,q} \binom{n_{p,q}^{zwz'w'}}{2} - \sum_{p} \binom{n_{p,\bullet}^{zwz'w'}}{2} \sum_{q} \binom{n_{\bullet,q}^{zwz'w'}}{2} / \binom{I\times J}{2}}{\frac{1}{2}\left[\sum_{p} \binom{n_{p,\bullet}^{zwz'w'}}{2} + \sum_{q} \binom{n_{\bullet,q}^{zwz'w'}}{2}\right] - \left[\sum_{p} \binom{n_{p,\bullet}^{zwz'w'}}{2} \sum_{q} \binom{n_{\bullet,q}^{zwz'w'}}{2}\right] / \binom{I\times J}{2}}. \quad (3.8)$$

$$\begin{pmatrix}
n_{1,1}^{zwz'w'} & n_{1,2}^{zwz'w'} & \cdots & n_{1,L'}^{zwz'w'} & \cdots & \cdots & n_{1,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{1,H'L'}^{zwz'w'} \\
n_{2,1}^{zwz'w'} & n_{2,2}^{zwz'w'} & \cdots & n_{2,L'}^{zwz'w'} & \cdots & \cdots & n_{2,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{2,H'L'}^{zwz'w'} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
n_{L,1}^{zwz'w'} & n_{L,2}^{zwz'w'} & \cdots & n_{L,L'}^{zwz'w'} & \cdots & \cdots & n_{L,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{L,H'L'}^{zwz'w'} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & & \vdots \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & & \vdots \\
n_{(H-1)L+1,1}^{zwz'w'} & n_{(H-1)L+1,2}^{zwz'w'} & \cdots & n_{(H-1)L+1,L'}^{zwz'w'} & \cdots & \cdots & n_{(H-1)L+1,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{(H-1)L+1,H'L'}^{zwz'w'} \\
n_{(H-1)L+2,1}^{zwz'w'} & n_{(H-1)L+2,2}^{zwz'w'} & \cdots & n_{(H-1)L+2,L'}^{zwz'w'} & \cdots & \cdots & n_{(H-1)L+2,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{(H-1)L+2,H'L'}^{zwz'w'} \\
\vdots & \vdots & \ddots & \vdots & & & \vdots & \vdots & \ddots & \vdots \\
n_{HL,1}^{zwz'w'} & n_{HL,2}^{zwz'w'} & \cdots & n_{HL,L'}^{zwz'w'} & \cdots & \cdots & n_{HL,(H'-1)L+1}^{zwz'w'} & \cdots & \cdots & n_{HL,H'L'}^{zwz'w'}
\end{pmatrix}$$

Table 3.5 – Contingency table to compare two pairs of coclustering partitions.

Like the ARI, this index is symmetric and takes the value 1 when the couples of partitions agree perfectly up to a permutation. But unlike the index proposed by [39] with which we will compare in Section 5, no convention is needed when the number of clusters is different in partitions. Moreover, it does not rely on the permutations of partitions and can therefore be easily computed even if the number of row clusters or column clusters exceeds nine. Though, the naïve complexity to compute $\boldsymbol{n}^{\boldsymbol{zwz'w'}}$ is still substancial.

Fortunately, we manage to exhib a link between $\boldsymbol{n}^{\boldsymbol{zwz'w'}}$, $\boldsymbol{n}^{\boldsymbol{zz'}}$ and $\boldsymbol{n}^{\boldsymbol{ww'}}$ which makes the computation of the CARI much faster and competitive in a high dimensional setting :

Theorem 5.1 *Let $\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{z'}, \boldsymbol{w'}, \boldsymbol{n}^{\boldsymbol{zwzw'}}, \boldsymbol{n}^{\boldsymbol{zz'}}$ and $\boldsymbol{n}^{\boldsymbol{ww'}}$ be defined as in Definition 5.1. Then we have the following relation,*

$$\boldsymbol{n}^{\boldsymbol{zwzw'}} = \boldsymbol{n}^{\boldsymbol{zz'}} \otimes \boldsymbol{n}^{\boldsymbol{ww'}}, \tag{3.9}$$

where $\otimes$ denotes The Kronecker product between two matrices.

The proof of this theorem is postponed to Appendix A.

Thanks to this property, the contingency table $\boldsymbol{n}^{\boldsymbol{zwzw'}}$ can be computed more efficiently and its complexity is now $\mathcal{O}(HH'+LL'+HH'LL')$. Moreover, even if the Kronecker product is not commutative, it behaves well with both the transpose operator and the margins, and the initial properties of CARI are kept :

Corollary 5.2 *1. $\forall (p,q) \in (H \times L) \times (H' \times L')$, we have the following relations between the margins,*

$$n_{\bullet,q}^{zwzw'} = n_{\bullet h'_q}^{zz'} \otimes n_{\bullet,\ell'_q}^{ww'} \text{ and } n_{p,\bullet}^{zwzw'} = n_{h_p,\bullet}^{zz'} \otimes n_{\ell_p,\bullet}^{ww'}$$

.

2. *The CARI associated with the contingency table* $\boldsymbol{n}^{\boldsymbol{zwzw'}}$ *defined as in Equation 3.9 remains symmetric, that is to say,*

$$CARI((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'})) = CARI((\boldsymbol{z'},\boldsymbol{w'}),(\boldsymbol{z},\boldsymbol{w})).$$

The proof of this corollary is postponed to Appendix B. In the further sections, the contingency table $\boldsymbol{n}^{\boldsymbol{zwzw'}}$ is now defined by Equation 3.9.

5.4 Examples

5.4.1 Comparison of couples of equal partitions up to a permutation.

Let consider the following couples of partitions $(\boldsymbol{z},\boldsymbol{w}) = ((1,1,3,2),(1,2,1,4,3,))$ and $(\boldsymbol{z'},\boldsymbol{w'}) = ((2,2,1,3),(2,1,2,3,4))$ which are equal up to a permutation. The contingency table (see Table 3.6) associated with CARI$((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'}))$ has a size of $(3\times 4, 3\times 4)$.

Thus, the CARI$((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'}))$ behaves well and is equal to $\frac{11-121/190}{1/2\times 22-121/190} = 1$.

Block	$(1,1)$	$(1,2)$	$(1,3)$	$(1,4)$	$(2,1)$	$(2,2)$	$(2,3)$	$(2,4)$	$(3,1)$	$(3,2)$	$(3,3)$	$(3,4)$	Margin
$(1,1)$	0	0	0	0	0	4	0	0	0	0	0	0	4
$(1,2)$	0	0	0	0	2	0	0	0	0	0	0	0	2
$(1,3)$	0	0	0	0	0	0	0	2	0	0	0	0	2
$(1,4)$	0	0	0	0	0	0	2	0	0	0	0	0	2
$(2,1)$	0	0	0	0	0	0	0	0	0	2	0	0	2
$(2,2)$	0	0	0	0	0	0	0	0	1	0	0	0	1
$(2,3)$	0	0	0	0	0	0	0	0	0	0	0	1	1
$(2,4)$	0	0	0	0	0	0	0	0	0	0	1	0	1
$(3,1)$	0	2	0	0	0	0	0	0	0	0	0	0	2
$(3,2)$	1	0	0	0	0	0	0	0	0	0	0	0	1
$(3,3)$	0	0	0	1	0	0	0	0	0	0	0	0	1
$(3,4)$	0	0	1	0	0	0	0	0	0	0	0	0	1
Margin	1	2	1	1	2	4	2	2	1	2	1	1	20

Table 3.6 – Initial contingency table $\boldsymbol{n}^{\boldsymbol{zwz'w'}}$ (see Definition 3.1).

5.4.2 Comparison of couples of partitions with a different number of clusters

Let us now consider the following partitions $(\boldsymbol{z},\boldsymbol{w}) = ((1,2,2,2,1),(1,1,2,1,1,2))$ and $(\boldsymbol{z'},\boldsymbol{w'}) = ((1,1,2,1,1),(1,1,2,1,3,2))$. Remark that partitions $\boldsymbol{w}$ and $\boldsymbol{w'}$ do not have the same number of clusters. The initial contingency tables related to ARI$(\boldsymbol{z},\boldsymbol{z'})$, ARI$(\boldsymbol{w},\boldsymbol{w'})$ and CARI$((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'}))$ are described in Tables 3.7 et 3.8. We observe as announced that

$$\boldsymbol{n}^{\boldsymbol{zwz'w'}} = \boldsymbol{n}^{\boldsymbol{zz'}} \otimes \boldsymbol{n}^{\boldsymbol{ww'}}.$$

The values of the ARIs and the CARI are available in Table 3.9. We notably observe that the ARI's value for rows is negative.

Cluster	1	2	Margin
1	2	0	2
2	2	1	3
Margin	4	1	5

Cluster	1	2	3	Margin
1	3	0	1	4
2	0	2	0	2
Margin	3	2	1	6

Table 3.7 – Contingency table $\boldsymbol{n^{zz'}}$ and $\boldsymbol{n^{ww'}}$ respectively related to ARI$(\boldsymbol{z}, \boldsymbol{z'})$ (at left) and to ARI$(\boldsymbol{w}, \boldsymbol{w'})$ (at right).

Block	$(1,1)$	$(1,2)$	$(1,3)$	$(2,1)$	$(2,2)$	$(2,3)$	Margin
$(1,1)$	6	0	2	0	0	0	8
$(1,2)$	0	4	0	0	0	0	4
$(2,1)$	6	0	2	3	0	1	12
$(2,2)$	0	4	0	0	2	0	6
Margin	12	8	4	3	2	1	30

Table 3.8 – Initial contingency table $\boldsymbol{n^{zz'ww'}}$ (see Definition 3.1).

	ARI$(\boldsymbol{z}, \boldsymbol{z'})$	ARI$(\boldsymbol{w}, \boldsymbol{w'})$	CARI$((\boldsymbol{z}, \boldsymbol{w}), (\boldsymbol{z'}, \boldsymbol{w'}))$
Value	-0.1538	0.5872	0.2501

Table 3.9 – Comparison of the values of ARI$(\boldsymbol{z}, \boldsymbol{z'})$, ARI$(\boldsymbol{w}, \boldsymbol{w'})$ and CARI$((\boldsymbol{z}, \boldsymbol{w}),(\boldsymbol{z'}, \boldsymbol{w'}))$.

5.5 Comparison between different coclustering indices

We will present the indices that we consider in the further simulation study. The notations refer to Section 3.1.

5.5.1 Other coclustering indices

→ *Classification error*

The classification distance presented in ([39]) studies the misclassification rate of the observations in the blocks :

$$\mathrm{dist}_{(I,H)\times(J,L)}((\boldsymbol{z}, \boldsymbol{w}), (\boldsymbol{z'}, \boldsymbol{w'})) = \min_{\sigma\in\mathfrak{S}(\{1,..H\})} \min_{\tau\in\mathfrak{S}(\{1,..L\})} \left(1 - \frac{1}{I\times J} \sum_{i,j,h,\ell} z_{ih} z'_{i\sigma(h)} w_{j\ell} w'_{j\tau(\ell)}\right), \quad (3.10)$$

where $\mathfrak{S}(\{1,..H\})$ denotes the set of permutations on the set $\{1,\ldots,H\}$.

The classification error (CE) is then defined when the cost function measures the difference between the pairs of reference $(\boldsymbol{z^*}, \boldsymbol{w^*})$ partitions and an estimation $(\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{w}})$:

$$\mathrm{CE}((\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{w}}), (\boldsymbol{z^*}, \boldsymbol{w^*})) \quad = \quad \mathrm{dist}_{(I,H)\times(J,L)}((\widehat{\boldsymbol{z}}, \widehat{\boldsymbol{w}}), (\boldsymbol{z^*}, \boldsymbol{w^*})).$$

The classification error is between 0 and 1. Thus, the observation x_{ij} is not in the block (h, ℓ) if the row i is not in the cluster h *or* if the column j is not in the cluster ℓ. When a column is improperly classified, all the cells of this column are penalized, and the classification error is increased by $\frac{1}{J}$.

Furthermore, the distance related to the row partitions can be also defined as follows :

$$\text{dist}_{I,H}(\boldsymbol{z}, \boldsymbol{z'}) = 1 - \max_{\sigma \in \mathfrak{S}(\{1,..H\})} \frac{1}{I} \sum_{i,h} z_{ih} z'_{i\sigma(h)}, \tag{3.11}$$

When the partitions do not include the same number of clusters, a suitable convention we can propose, is to consider H as the maximal number of clusters and the created additional clusters are assumed to be empty. Besides, the computation of this distance when H is higher than nine, remains difficult as the order of the set $\mathfrak{S}(\{1,..H\})$ is $H!$.

In a symmetric way, the distance related to the column partitions is denoted by $\text{dist}_{J,L}$.

[39] show that the classification error could be expressed in terms of the distance related to the row partitions and the distance related to the column partitions :

$$\begin{aligned} \text{dist}_{(I,H)\times(J,L)}((\boldsymbol{z}, \boldsymbol{w}), (\boldsymbol{z'}, \boldsymbol{w'})) &= \text{dist}_{I,H}(\boldsymbol{z}, \boldsymbol{z'}) + \text{dist}_{J,L}(\boldsymbol{w}, \boldsymbol{w'}) \\ &- \text{dist}_{I,H}(\boldsymbol{z}, \boldsymbol{z'}) \times \text{dist}_{J,L}(\boldsymbol{w}, \boldsymbol{w'}). \end{aligned} \tag{3.12}$$

→ *Extended Generalized Mutual Information*

The generalized mutual information introduced by [62] is extended by [65] to compare two coclustering partitions. Originally, the generalized mutual information between two partitions $\boldsymbol{z} = (z_1, \ldots, z_H)$ and $\boldsymbol{z'} = (z'_1, \ldots, z'_{H'})$ on a same set $A = \{O_1, \ldots, O_I\}$ is as follows :

$$\text{MI}(\boldsymbol{z}, \boldsymbol{z'}) = \sum_{h,h'} P_{h,h'} \log\big(\frac{P_{h,h'}}{P_h P_{h'}}\big),$$

$$\text{where,} \quad P_{h,h'} = \frac{1}{I} \sum_{i,i'}^{I} 1_{\{z_i = h, z'_{i'} = h'\}}, \quad P_h = \frac{1}{I} \sum_{i}^{I} 1_{\{z_i = h\}} \quad \text{and} \quad P_{h'} = \frac{1}{I} \sum_{i'}^{I} 1_{\{z'_{i'} = h'\}}.$$

When the two partitions do not present the same number of clusters, the quantity is normalized as follows :

$$\frac{\text{MI}(\boldsymbol{z}, \boldsymbol{z'})}{\max(\mathcal{H}(\boldsymbol{z}), \mathcal{H}(\boldsymbol{z'}))},$$

$$\text{where,} \quad \mathcal{H}(\boldsymbol{z}) = -\sum_{h} P_h \log P_h, \text{ and } \mathcal{H}(z') = -\sum_{h'} P_{h'} \log P_{h'}.$$

Thus, the proposed measure to compare two coclustering partitions ($\boldsymbol{z} = (z_1, \ldots, z_H)$, $\boldsymbol{w} = (w_1, \ldots, w_L)$) and ($\boldsymbol{z'} = (z'_1, \ldots, z'_{H'})$, $\boldsymbol{w'} = (w'_1, \ldots, w'_{L'})$) on a set $A \times B$ is based

on a linear combination of the generalized mutual information of $\boldsymbol{z}$ and $\boldsymbol{z'}$, and the the generalized mutual information of $\boldsymbol{w}$ and $\boldsymbol{w'}$:

$$\mathrm{MI}((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'})) = \mathrm{MI}(\boldsymbol{z},\boldsymbol{z'}) + \mathrm{MI}(\boldsymbol{w},\boldsymbol{w'}).$$

The maximal value of this index is equal to 2 when the partitions perfectly match up to a permutation and is equal to 0 when the correspondence between them is extremely weak. Remark that, by extending this index in this way, the coclustering structure of the problem is not preserved and this major drawback will be tackled in the next section.

5.5.2 Simulation study

To compare the CARI with the other indices, we first propose to test their computation complexity as a function of the number of observations or clusters. Then, we assess their performance to measure how close are two coclustering partitions from a coclustering point of view. Finally, we investigate if there exists any simple link between the indices.

To achieve these objectives, we propose a simulation methodology to generate a set of coclustering partitions more or less close to the considered initial ones. Remark that a simulation approach already exists in the task of clustering in one dimension ([19], [55], [67]), but another point of view is developed here. Our procedure can now be described as follows :

Fix the sizes (I, J) and the number of clusters (H, L) of the coclustering partitions that would be studied. Consider the initial coclustering partitions $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ in the *balanced* or *unbalanced* case, that is to say, where the number of individuals in each cluster is the same or not. For $i = 1, ..., N$ iterations :

1. Choose a coordinate of $\boldsymbol{z^{(i-1)}}$ at random and allocate to it, a new label chosen randomly between 1 and H. The new vector is named $\boldsymbol{z^{(i)}}$.
2. Reproduce the item (1) with the vector $\boldsymbol{w^{(i-1)}}$. The new vector is named $\boldsymbol{w^{(i)}}$.
3. Compute the different indices between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(i)}})$.

Thus, at each iteration i, the coclustering partitions $(\boldsymbol{z^{(i-1)}}, \boldsymbol{w^{(i-1)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(i)}})$ can differ from only one coordinate in each vector. Gradually, the procedure produces a set of coclustering partitions more and more discordant with the initial coclustering partitions $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$. The support of the studied indices from high values to small values, can therefore be well explored if the number of iterations N is high enough.

→ *Time comparison*

The complexity of the three indices related to the number of observations and the number of clusters is assessed. For this purpose, the procedure is run with $N = 10\,000$ iterations considering two situations $(I, J) = (315, 315)$ observations and $(I, J) = (630, 630)$

observations when the number of clusters varies as follows, $(H, L) \in \{(5,5),(7,7),(9,9)\}$. The results are presented in Figures 3.16 and 3.17. We observe that the elapsed time computation in log scale of the MI is the smallest and seems not to be sensitive to the number of clusters or observations. The CARI also behaves well whatever the number of clusters or observations. On the contrary, the time computation of the CE significantly increases with the number of clusters, which illustrates its dependence on the factorial of this quantity.

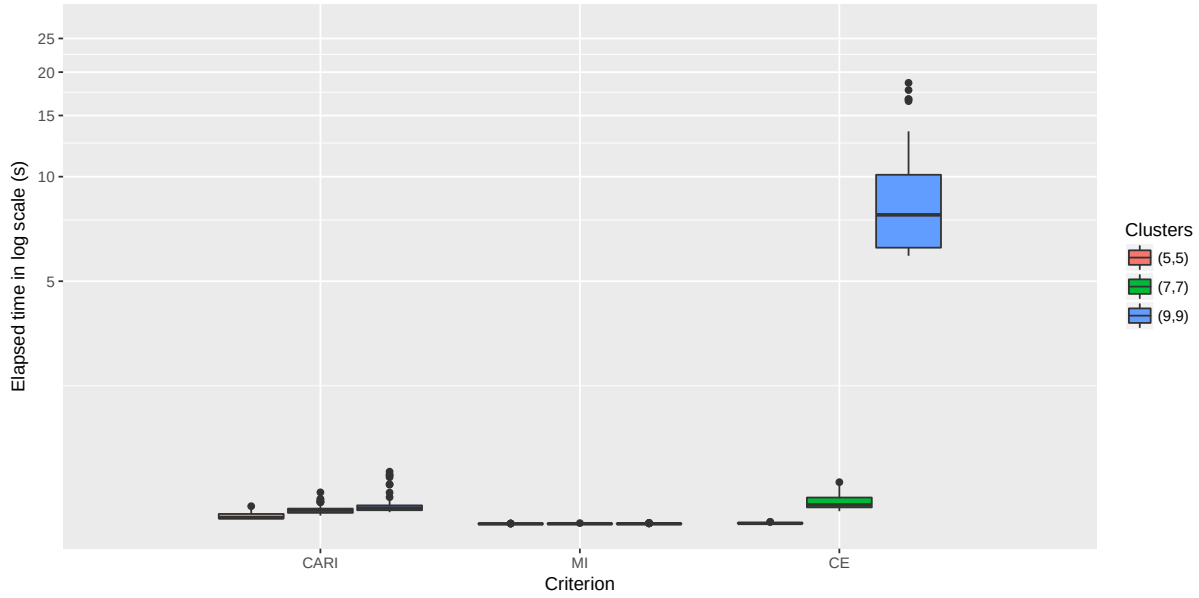

Figure 3.16 – Boxplot of the elapsed time computation in log scale of the CARI, the MI and the CE, for $N = 10\ 000$ iterations of the procedure, with $(I, J) = (315, 315)$ observations and for different number of clusters ; $(H, L) \in \{(5,5),(7,7),(9,9)\}$.

→ *Behaviour comparison*

The first comparison between the three indices is performed by running the procedure with $N = 10\ 000$ iterations, $(H, L) = (5, 5)$ and the following sample sizes $(I, J) = (50, 50)$, $(I, J) = (500, 500)$ and $(I, J) = (1000, 1000)$. The results are presented in Figure 3.18 in the balanced case and in Figure 3.19 in the unbalanced case. In the unbalanced case, the number of observations in each cluster of the initial coclustering partitions is defined in Table 3.10. Remark that we considerer the quantity 1−CE which is more convenient to compare with the CARI. Indeed, a perfect matching between partitions is now corresponding to the value 1 for both indices. First of all, the experiment enables to scan all the supports of the indices, except for the CARI where negative values are not reached. To our knowledge, this phenomenon rather appeared when the agreement of the considered coclustering partitions are very weak and the number of observations is very

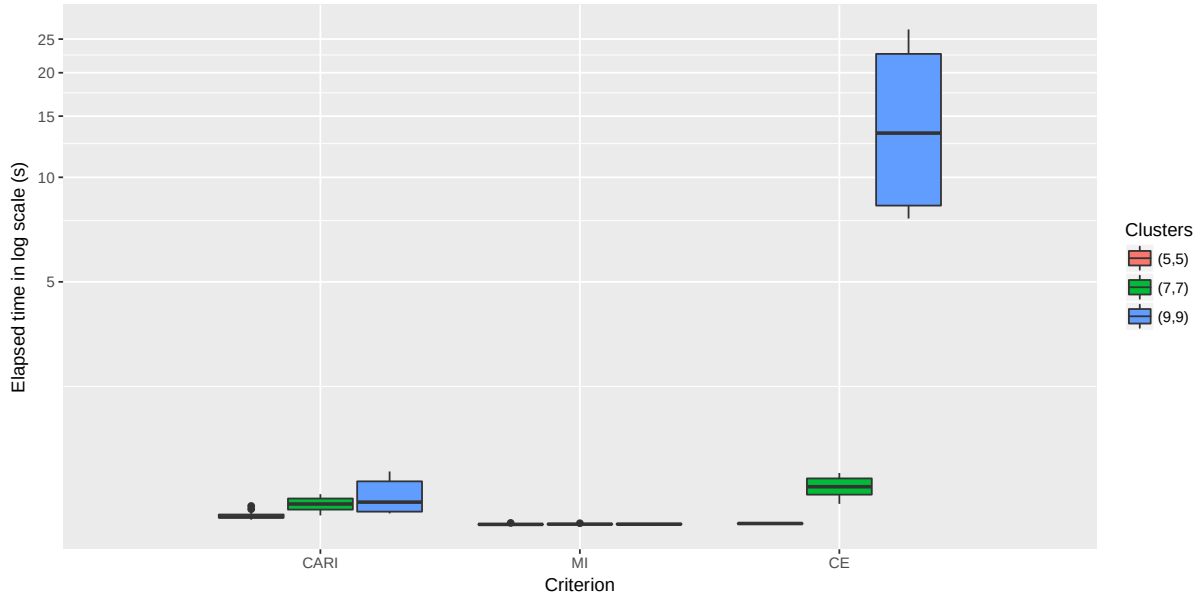

Figure 3.17 – Boxplot of the elapsed time computation in log scale of the CARI, the MI and the CE, for $N = 10\ 000$ iterations of the procedure, with $(I, J) = (630, 630)$ observations and for different number of clusters, $(H, L) \in \{(5, 5), (7, 7), (9, 9)\}$.

small (less than the considered case here, (50,50)).

		cluster number				
		1	2	3	4	5
(I,J)	(50,50)	4	7	10	13	16
	(500,500)	20	35	100	165	180
	(1000,1000)	30	70	200	300	400

Table 3.10 – Repartition of the observations in each cluster of the initial coclustering partitions in the procedure for the unbalanced case.

Then, we observe in Figures 3.18 and 3.19, that the behaviour of the three indices are different enough as all the curves are far from the line bisector, and no simple link, like a linear one for example, can be exhibited. We also notice that in general, the CARI tends to be more demanding and penalizing than the other indices.

Moreover, we notice that when the number of observations is small, the fact that an observation is missclasified, impacts more the values of the three indices as the blue circles are more widely spaced, where the values of the indices are high in Figures 3.18 and 3.19.

In the unbalanced case (see Figure 3.19), the compared behaviour between the CARI and the quantity $1-$CE seems to be globally the same whatever the number of observations. Conversely, we remark a changement in the compared behaviour between the CARI

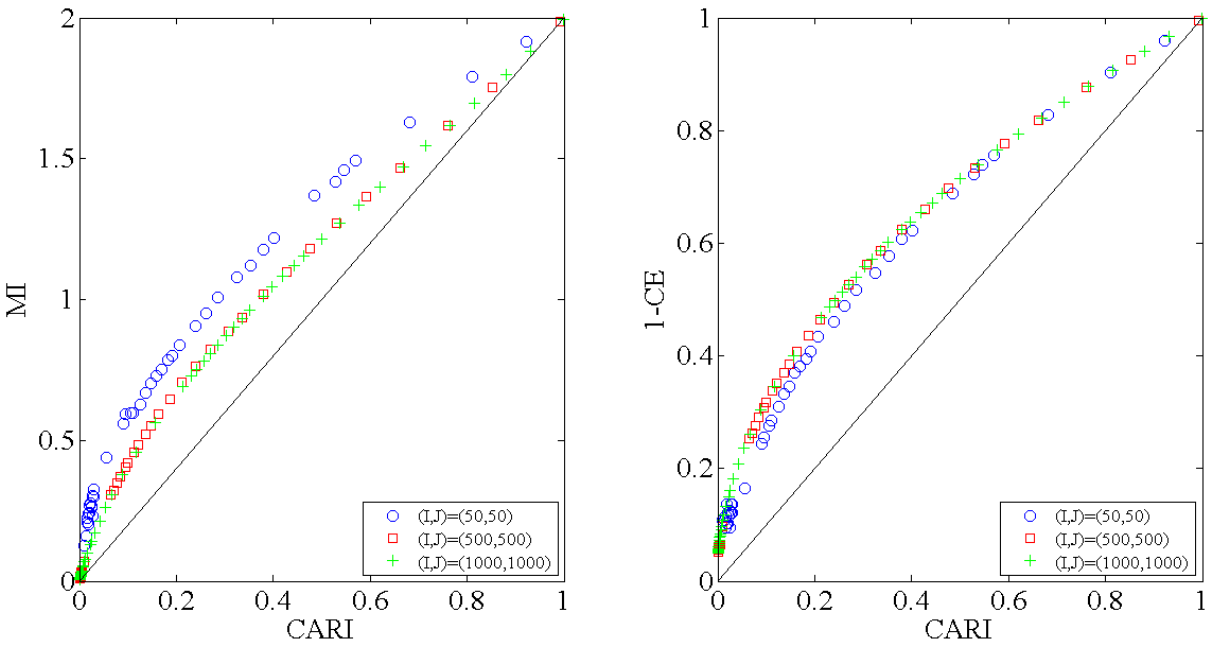

Figure 3.18 – Comparison of the values of the CARI (on the horizontal axis) versus the values of the MI (at the left, on the vertical axis), and versus the values of the 1-CE (at right, on the vertical axis) in the *balanced case*, on a run of the procedure with $N = 10\ 000$, for different sample sizes $(I, J) = (50, 50)$ (blue circle), $(I, J) = (500, 500)$ (red square), $(I, J) = (1000, 1000)$ (green cross).

and the MI. Indeed, when the number of observations is high and the compared colustering partitions differed from few observations (corresponding to the part of red square curves with the highest values for the CARI and the MI in Figure 3.19, at left), the MI and the CARI behave in the same way, whereas the CARI is more demanding when the compared coclustering partitions are very discordant.

The second comparison consists of observing how each criterion behaves when the compared pairs of coclustering partitions have the same row partition or the same column partition. That is why we use again the procedure, presented in Section 5.2 and we complete the step (3) of the procedure for each iteration $i = 1...N$, as follows :

(3) Compute the indices between $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{i})}, \boldsymbol{w}^{(\boldsymbol{i})})$, between $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{i})})$ and between $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{i})}, \boldsymbol{w}^{(\boldsymbol{0})})$.

The results shown in Figures 3.20 and 3.21, illustrate the comparison of the CARI versus the two other indices on a run of the procedure with $N = 10\ 000$, $(H, L) = (7, 5)$, $(I, J) = (630, 630)$ in the balanced case. Each index is computed at each iteration i for the following pairs of coclustering partitions : $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{i})}, \boldsymbol{w}^{(\boldsymbol{i})})$ (blue circle), between $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{i})})$(red square), between $(\boldsymbol{z}^{(\boldsymbol{0})}, \boldsymbol{w}^{(\boldsymbol{0})})$ and $(\boldsymbol{z}^{(\boldsymbol{i})}, \boldsymbol{w}^{(\boldsymbol{0})})$ (green cross).

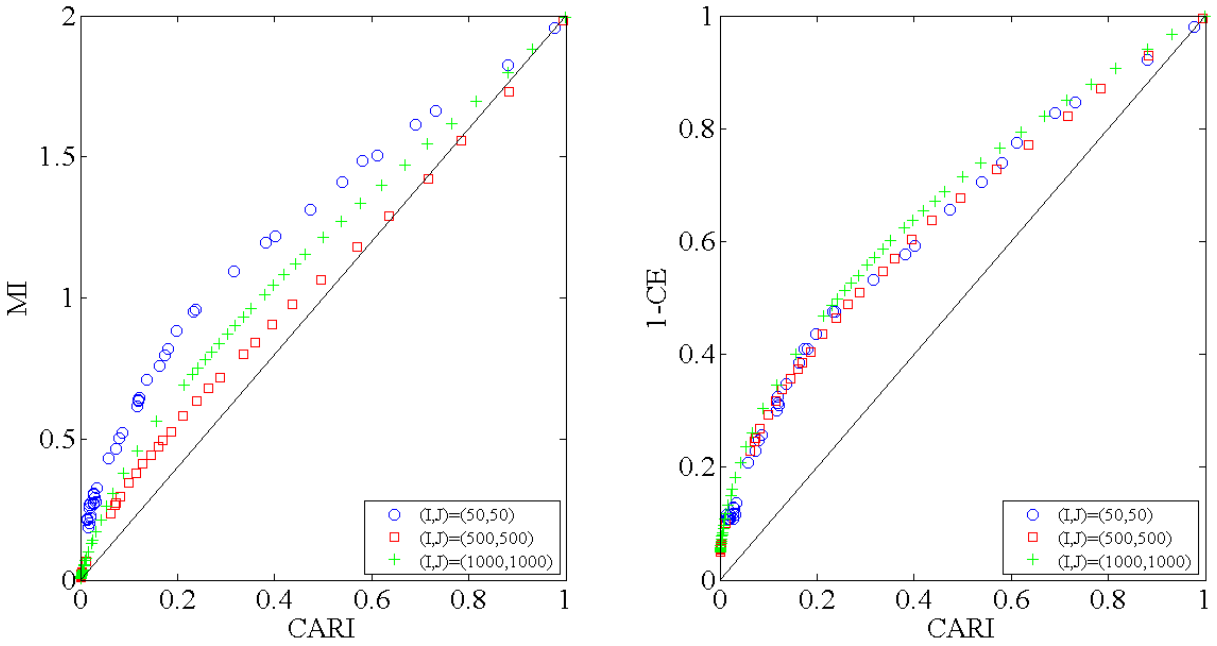

Figure 3.19 – Comparison of the values of the CARI (on the horizontal axis) versus the values of the MI (at the left, on the vertical axis), and versus the values of the 1-CE (at right, on the vertical axis) in the *unbalanced case*, on a run of the procedure with $N = 10\ 000$, for different sample sizes $(I, J) = (50, 50)$ (blue circle), $(I, J) = (500, 500)$ (red square), $(I, J) = (1000, 1000)$ (green cross).

In Figure 3.20 representing the comparison of the CARI versus the MI, we notice that the curves defined by red circles and green crosses are above the curve defined by blue circles. We therefore infer that the CARI is more penalizing than the MI when the compared pairs of coclustering partitions have the same row partition or the same column partition. Besides, in this case, when one partition is fixed (curves defined by red circles and green crosses in Figure 3.20), we observe that the MI, whose maximal value is 2, always remains above 1 even when the partitions $\boldsymbol{w}$ and $\boldsymbol{w'}$ or $\boldsymbol{z}$ and $\boldsymbol{z'}$ are very discordant. From the coclustering point of view, this type of configuration should be very penalised, which does the CARI, but does not the MI due to its construction as a linear combination of a row distance and column distance. Indeed, the CARI takes into account in its construction, the linkage between row partition and column partition, whereas the MI deals with row partition and column partition in a separated way.

5.6 Conclusion

In this article, we introduced a new coclustering index named *Coclustering Adjusted Rand Index* (CARI) and based on the very popular ARI. We prove that, like the classification error proposed by [39] but unlike the criterion developed by [65], the CARI

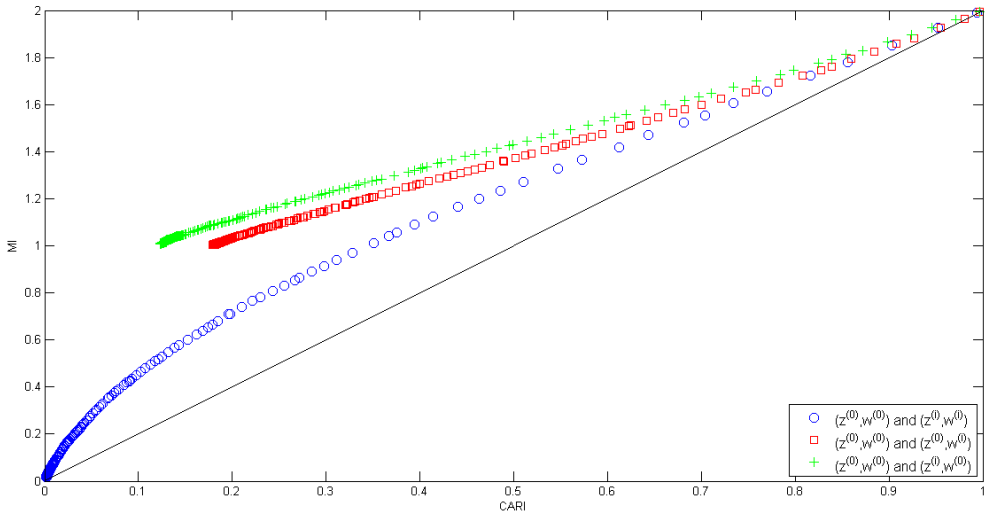

Figure 3.20 – Comparison of the CARI's values (on the horizontal axis) versus the MI's values (on the vertical axis) on a run of the procedure with $N = 10\ 000$, $(H, L) = (7, 5)$, $(I, J) = (630, 630)$ in the balanced case. Each index is computed at each iteration i between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(i)}})$ (blue circle), between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(i)}})$(red square), between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(0)}})$ (green cross).

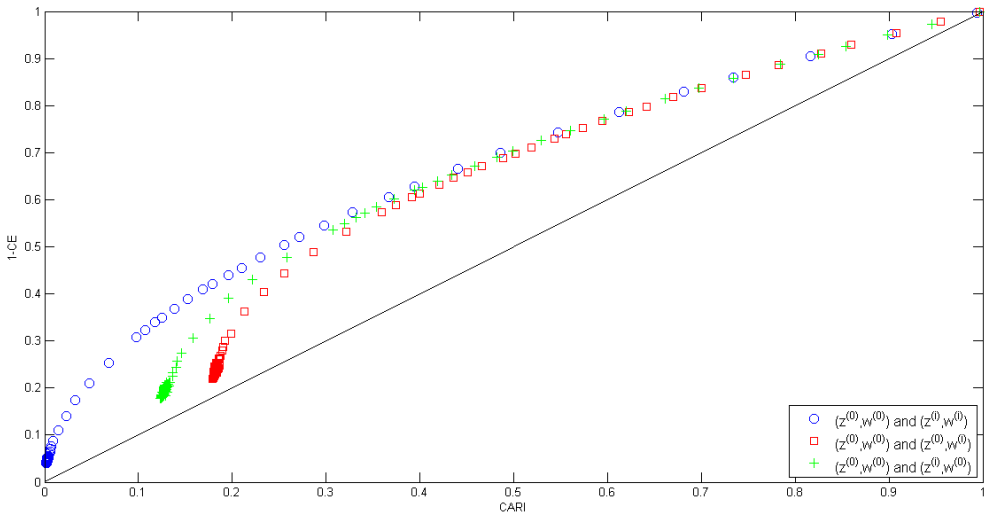

Figure 3.21 – Comparison of the CARI's values (on the horizontal axis) versus the values of the quantity 1-CE (on the vertical axis) on a run of the procedure with $N = 10\ 000$, $(H, L) = (7, 5)$, $(I, J) = (630, 630)$ in the balanced case. Each index is computed at each iteration i between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(i)}})$ (blue circle), between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(i)}})$(red square), between $(\boldsymbol{z^{(0)}}, \boldsymbol{w^{(0)}})$ and $(\boldsymbol{z^{(i)}}, \boldsymbol{w^{(0)}})$ (green cross).

measures the agreement between two pairs of partitions from a coclustering point of view. In addition, we show that the CARI could be computed in an efficient way, whatever the number of clusters or observations, thanks to a simple trick. These good charasteristics makes the CARI convenient and useful in a high dimensional setting, which is a highly topical issue nowadays.

5.7 Appendix A. Proof of Theorem 3.3

Theorem 3.3. *Let* $\boldsymbol{z}, \boldsymbol{w}, \boldsymbol{z'}, \boldsymbol{w'}, \boldsymbol{n^{zwzw'}}, \boldsymbol{n^{zz'}}$ *and* $\boldsymbol{n^{ww'}}$ *be defined as in Definition 5.1. Then we have the following relation,*

$$\boldsymbol{n^{zwzw'}} = \boldsymbol{n^{zz'}} \otimes \boldsymbol{n^{ww'}},$$

where $\otimes$ *denotes The Kronecker product between two matrices.*

Let recall the definition of the Kronecker product. Let $\boldsymbol{A} = (a_{i,j})$ be a matrix of size $H \times H'$ and $\boldsymbol{B}$ be a matrix of size $L \times L'$. The Kronecker product is the matrix $\boldsymbol{A} \otimes \boldsymbol{B}$ of size $H \times L$ by $H' \times L'$, defined by successive blocks of size $L \times L'$. The block of the index i, j is equal to $a_{i,j} \times \boldsymbol{B}$:

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{pmatrix} a_{1,1}\boldsymbol{B} & \dots & a_{1,H'}\boldsymbol{B} \\ \dots & \dots & \dots \\ a_{H,1}\boldsymbol{B} & \dots & a_{H,H'}\boldsymbol{B} \end{pmatrix}.$$

We started by remarking a common trick used in computer science. Indeed, for all $p \in \{1, \dots, HL\}$, the associated pair (h, ℓ) denoting a block of $(\boldsymbol{z}, \boldsymbol{w})$, is respectively the quotient plus 1 and the remainder plus 1 of the Euclidean division of $(p-1)$ by L. In other words, we have :

$$(p-1) = (h-1) \times L + (\ell - 1).$$

We can easily deduce that there is a bijection between each index p and the pairs (h, ℓ). In the same way, the assertion is valid for q and the pairs (h', ℓ').

The next proposition is the last step before proving the final result :

Proposition 5.3 *For all pairs of indices* p *and* q *associated respectively with blocks* (h, ℓ) *and* (h', ℓ'),

$$n_{p,q}^{zwz'w'} = n_{h,h'}^{zz'} n_{\ell,\ell'}^{ww'}.$$

Proof We notice that the observation x_{ij} is in the block (h, ℓ) if and only if the row i is in the cluster h and the column j is in the cluster ℓ. Thanks to this remark, we can easily see that an observation x_{ij} belongs to the block (h, ℓ) and the block (h', ℓ') if and only if the row i belongs at the same time to the cluster h and the cluster h', and the column j belongs at the same time to the cluster ℓ and the cluster ℓ'.

With the previous results, we finally have :

$$
\begin{aligned}
\boldsymbol{n}^{\boldsymbol{zz'ww'}} &= \begin{pmatrix}
n_{1,1}^{zwz'w'} & n_{1,2}^{zwz'w'} & \cdots & n_{1,L'}^{zwz'w'} & n_{1,L'+1}^{zwz'w'} & \cdots & n_{1,H'L'}^{zwz'w'} \\
n_{2,1}^{zwz'w'} & n_{2,2}^{zwz'w'} & \cdots & n_{2,L'}^{zwz'w'} & n_{2,L'+1}^{zwz'w'} & \cdots & n_{2,H'L'}^{zwz'w'} \\
\vdots & \vdots & \ddots & & & & \vdots \\
n_{L,1}^{zwz'w'} & n_{L,2}^{zwz'w'} & \cdots & n_{L,L'}^{zwz'w'} & n_{L,L'+1}^{zwz'w'} & \cdots & n_{L,H'L'}^{zwz'w'} \\
n_{L+1,1}^{zwz'w'} & n_{L+1,2}^{zwz'w'} & \cdots & n_{L+1,L'}^{zwz'w'} & n_{L+1,L'+1}^{zwz'w'} & \cdots & n_{L+1,H'L'}^{zwz'w'} \\
\vdots & \vdots & & & & \ddots & \vdots \\
n_{HL,1}^{zwz'w'} & n_{HL,2}^{zwz'w'} & \cdots & n_{HL,L'}^{zwz'w'} & n_{HL,L'+1}^{zwz'w'} & \cdots & n_{HL,H'L'}^{zwz'w'}
\end{pmatrix} \\
&= \begin{pmatrix}
n_{1,1}^{zz'}n_{1,1}^{ww'} & n_{1,1}^{zz'}n_{1,2}^{ww'} & \cdots & n_{1,1}^{zz'}n_{1,H'}^{ww'} & n_{1,2}^{zz'}n_{1,1}^{ww'} & \cdots & n_{1,L'}^{zz'}n_{1,H'}^{ww'} \\
n_{1,1}^{z'}n_{2,1}^{ww'} & n_{1,1}^{zz'}n_{2,2}^{ww'} & \cdots & n_{1,1}^{zz'}n_{2,H'}^{ww'} & n_{1,2}^{zz'}n_{1,1}^{ww'} & \cdots & n_{1,L'}^{zz'}n_{2,H'}^{ww'} \\
\vdots & \vdots & \ddots & & & & \vdots \\
n_{1,1}^{zz'}n_{H,1}^{ww'} & n_{1,1}^{zz'}n_{H,2}^{ww'} & \cdots & n_{1,1}^{zz'}n_{H,H'}^{ww'} & n_{1,2}^{zz'}n_{1,1}^{ww'} & \cdots & n_{1,L'}^{zz'}n_{H,H'}^{ww'} \\
n_{2,1}^{zz'}n_{1,1}^{ww'} & n_{2,1}^{zz'}n_{1,2}^{ww'} & \cdots & n_{2,1}^{z,z'}n_{1,H'}^{ww'} & n_{2,2}^{zz'}n_{1,1}^{ww'} & \cdots & n_{2,L'}^{zz'}n_{1,H'}^{ww'} \\
\vdots & \vdots & & & & \ddots & \vdots \\
n_{L,1}^{zz'}n_{H,1}^{ww'} & n_{L,1}^{zz'}n_{H,2}^{ww'} & \cdots & n_{L,1}^{zz'}n_{HH'}^{ww'} & n_{L,2}^{zz'}n_{H,1}^{ww'} & \cdots & n_{L,L'}^{zz'}n_{H,H'}^{ww'}
\end{pmatrix} \\
&= \begin{pmatrix}
n_{1,1}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & n_{1,2}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & \cdots & n_{1,L'}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} \\
n_{2,1}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & n_{2,2}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & \cdots & n_{2,L'}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} \\
\vdots & & \ddots & \vdots \\
n_{L,1}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & n_{L,2}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}} & \cdots & n_{L,L'}^{zz'}\boldsymbol{n}^{\boldsymbol{ww'}}
\end{pmatrix} \\
&= \boldsymbol{n}^{\boldsymbol{zz'}} \otimes \boldsymbol{n}^{\boldsymbol{ww'}}.
\end{aligned}
$$

5.8 Appendix B. Proof of Corollary 3.4

Corollary 3.4.

1. $\forall (p,q) \in (H \times L) \times (H' \times L')$, *we have the following relations between the margins,*

$$n_{\bullet,q}^{zwzw'} = n_{\bullet,h'_q}^{zz'} \otimes n_{\bullet,\ell'_q}^{ww'} \text{ and } n_{p,\bullet}^{zwzw'} = n_{h_p,\bullet}^{zz'} \otimes n_{\ell_p,\bullet}^{ww'}$$

.

2. *The CARI associated with the contingency table* $\boldsymbol{n}^{\boldsymbol{zwzw'}}$ *defined as in Equation 3.9 remains symmetric, that is to say,*

$$CARI((\boldsymbol{z},\boldsymbol{w}),(\boldsymbol{z'},\boldsymbol{w'})) = CARI((\boldsymbol{z'},\boldsymbol{w'}),(\boldsymbol{z},\boldsymbol{w})).$$

1. This assertion forms part of the known properties of the Kronecker product.

2. The proof of this result is the direct consequence of the following Lemma :

Lemma 5.4 *Let* $\boldsymbol{z, w, z', w', n^{zz'}}$, *and* $\boldsymbol{n^{ww'}}$ *be defined as in Definition 3.1 and* $\boldsymbol{n^{zwzw'}}$ *be defined according to Theorem 3.3. Then we have,*

$$\boldsymbol{n^{z'w'zw}} = t(\boldsymbol{n^{zwz'w'}}),$$

where t denotes the tranpose of a matrix.

Proof Thanks to the property of the Kronecker product with the transpose, we have,

$$\begin{aligned} \boldsymbol{n^{z'w'zw}} &= \boldsymbol{n^{z'z}} \otimes \boldsymbol{n^{w'w}} \\ &= t(\boldsymbol{n^{zz'}}) \otimes t(\boldsymbol{n^{ww'}}) \\ &= t\left(\boldsymbol{n^{zz'}} \otimes \boldsymbol{n^{ww'}}\right) \\ &= t(\boldsymbol{n^{zwz'w'}}). \end{aligned}$$

Chapitre 4

Procédure de validation du modèle global

Nous avons mis en place une procédure statistique permettant, à partir d'un jeu de données, de classer les noeuds d'un graphe de deux manières (groupes de noeuds contrôleurs et contrôlés) tout en établissant les liens de contrôle entre ces deux types de classification. Cette procédure s'articule en deux étapes successives qui ont fait l'objet des deux chapitres précédents. L'étape de sélection a donné naissance à deux procédures alternatives, Gauss-LASSO stabilisé et enrichi. Calibrées de façon à engendrer des résultats les plus stables possibles, elles ont permis la mise en place de graphes orientés qui ont été ensuite soumis à l'étape de classification. La formation des deux types de groupes de noeuds découle de cette étape. Cette procédure globale forme un modèle complexe composé d'une suite d'opérations dépendantes les unes des autres. Testée sur un jeu de données d'expression réel, elle a permis de classer les facteurs de transcription d'*Arabidopsis thaliana* en groupes de gènes co-régulateurs et de gènes co-régulés. Dans le but d'évaluer la stabilité du modèle complet, nous voulons simuler des jeux de données de même taille que le jeu de données réel et où chaque facteur de transcription s'identifie à la même variable, puis y appliquer la procédure statistique globale afin d'évaluer la proximité des groupes qui en résulteront avec les groupes obtenus sur les données réelles. Le problème réside dans la manière dont nous allons simuler ces données. Nous aspirons à ce que les résultats de la procédure appliquée à un jeu simulé soient proches des résultats du jeu réel. L'idée est, en conséquence, de générer des données selon le modèle obtenu sur les données réelles. Pour ce faire, nous simulons en premier lieu une matrice par blocs de même structure que celle émanant de la procédure globale appliquée au jeu réel, puis nous reconstituons successivement chacune des opérations intervenant dans notre modèle. Une fois de tels jeux simulés et la procédure globale relancée sur ces nouveaux jeux, nous évaluerons la proximité de la classification double avec celle dont nous sommes partis, c'est-à-dire celle obtenue sur le jeu réel, par le biais du *Coclustering Ajusted Rand Index*.

1 Présentation générale

La procédure de validation que nous proposons s'inspire de la philosophie du bootstrap paramétrique ([18]). Nous adaptons cette méthode à notre modèle complexe. Les différentes étapes menant à la création d'un jeu simulé sont les suivantes :

1. Simulation d'une matrice par blocs de même structure que celle issue du jeu réel.
2. Reconstruction de la matrice d'adjacence du graphe modélisant le réseau de FTs.
   $\hookrightarrow$ Obtention des variables explicatives pertinentes de chaque variable.
3. Formation de chaque équation de régression linéaire d'une variable sur ses variables explicatives.
4. Estimation des coefficients de régression via le jeu réel.
5. Simulation pour chaque variable d'un échantillon de la loi jointe ainsi définie à l'aide d'un échantillonneur de Gibbs. Chaque échantillon a pour taille le nombre d'observations du jeu de données réel.
6. Obtention d'un jeu de données de même taille et qui a les mêmes caractéristiques que celles qui ont été identifiées sur le jeu de données réel.

Selon la procédure de sélection (Gauss-LASSO enrichi ou stabilisé) utilisée, la matrice par blocs résultant de la procédure globale diffère. Nous fixons alors l'une de ces deux procédures de sélection et ne nous intéressons dans la suite qu'aux résultats issus de cette dernière.

Une fois un jeu de données simulé formé de la sorte, nous appliquerons le modèle complet sur ces nouvelles données de manière à obtenir de nouveaux groupes de variables sous la forme de partitions doubles. Nous espérons retrouver au mieux les partitions estimés sur le jeu réel. Pour mesurer la proximité entre les groupes obtenus sur données simulées et ceux obtenus sur les données réelles, nous nous appuirons sur la méthode de comparaisons entre deux partitions doubles, à savoir le $CARI$ présenté dans le chapitre précédent.

1.1 Simulation d'une matrice par blocs

Nous voulons simuler une matrice par blocs analogue à celle issue de la procédure statistique appliquée au jeu réel, que l'on appellera abusivement matrice par blocs réelle. Notre volonté de conserver les groupes de co-régulation implique que les blocs de la matrice simulée seront identiques en nombre et en taille à ceux de la matrice réelle. La proportion de coefficients égaux à 1 des blocs sera elle aussi conservée.

Notons $B = B^{GLstab}$ ou B^{GLenri} la matrice par blocs réelle. Elle correspond à la matrice d'adjacence $A = A^{GLstab}$ ou A^{GLenri} du graphe orienté $\mathcal{G} = \mathcal{G}^{GLstab}$ ou $\mathcal{G}^{GLenri}$

issu de l'étape de sélection réorganisée à l'aide d'un LBM. Nous adopterons les notations suivantes, similaires à celles utilisées dans les chapitres précédents :

- n le nombre d'observations du jeu de données réel pour chaque variable.
- $\{1, \dots p\}$ l'ensemble des variables (noeuds). Pour désigner l'une d'entre elles, nous emploierons l'indice j lorsqu'elle est vue comme régressée (noeud contrôlé) et j' comme régresseuse (noeud contrôleur).
- H le nombre de classes en ligne des variables.
- L le nombre de classes en colonne des variables.
- $v = (v_{jh})_{j\in\{1,\dots,p\},h\in\{1,..,H\}}$ la partition en lignes : $\forall j, v_{jh} = 1 \Leftrightarrow$ la variable j appartient à la classe h.
- $w = (w_{j'\ell})_{j'\in\{1,\dots,p\},\ell\in\{1,..,L\}}$ la partition en colonnes : $\forall j', w_{j'\ell} = 1 \Leftrightarrow$ la variable j' appartient à la classe ℓ.
- V_h la h^{eme} classe en ligne (le h^{eme} groupe de FTs co-régulés) : $\forall h \in \{1, .., H\}, V_h = \{j \in J \backslash v_{jh} = 1\}$. On notera ρ_h la proportion de cette classe.
- W_ℓ la ℓ^{eme} classe en colonne (le ℓ^{eme} groupe de FTs co-régulateurs) : $\forall \ell \in \{1, .., L\}, W_\ell = \{j' \in J' \backslash w_{j'\ell} = 1\}$. On notera τ_ℓ la proportion de cette classe.
- $\rho = \big(\rho_h\big)_{h\in\{1,\dots,H\}}$ le vecteur de proportions en ligne et $\tau = \big(\tau_\ell\big)_{\ell\in\{1,\dots,L\}}$ en colonne.
- $\alpha = (\alpha_{h\ell})_{h\in\{1,..,H\},\ell\in\{1,..,L\}}$ la matrice des proportions de coefficients égaux à 1 du bloc croisant la classe h en ligne et la classe ℓ en colonne.

Une matrice par blocs ${}^sB$ simulée comme souhaité présentera exactement ces caractéristiques. Le découpage d'une telle matrice simulée en blocs sera toujours le même. Il reste à simuler chacun des blocs de la matrice.

Notons pour $h \in \{1, .., H\}$ et $\ell \in \{1, .., L\}$, $B^{(h\ell)}$ le bloc associé aux groupes V_h et W_ℓ. La densité $\alpha_{h\ell}$ d'un tel bloc s'écrit :

$$\alpha_{h\ell} = \frac{N_{h\ell}}{|V_h| \times |W_\ell|} \ ,$$

avec $N_{h\ell} = \sum_{j=1}^{|V_h|} \sum_{j'=1}^{|W_\ell|} 1_{\{B_{jj'}^{(h\ell)}=1\}}$, le nombre de coefficients du bloc égaux à 1

L'idée première serait de simuler chaque coefficient du bloc indépendammant les uns des autres selon une loi de Bernoulli $\mathcal{B}(\alpha_{h\ell})$. Notons ${}^sA$ la matrice d'adjacence associée à une matrice par blocs ${}^sB$ simulée. Sachant que le coefficient ${}^sA_{jj'}$ de ${}^sA$ appartient au bloc $h\ell$, il serait donc le résultat d'un tirage d'une variable aléatoire de loi $\mathcal{B}(\alpha_{h\ell})$: $f({}^sA|v, w; \alpha) = \prod_{jj'h\ell} \phi({}^sA_{jj'}; \alpha_{h\ell})^{z_{jh}w_{j'\ell}}$ avec $\phi({}^sA_{jj'}; \alpha_{h\ell}) = {\alpha_{h\ell}}^{{}^sA_{jj'}}(1 - \alpha_{h\ell})^{1-{}^sA_{jj'}}$.

Cependant, les coefficients diagonaux de la matrice d'adjacence ${}^sA$ sont automatiquement nuls. Ceci implique que dans chaque bloc, certains coefficients n'ont pas à être

simulés puisqu'ils sont fixés à 0. Le nombre de tels coefficients, pour un bloc ${}^sB^{(h\ell)}$, correspond au nombre de variables que la classe en ligne V_h et la classe en colonne W_ℓ ont en commun. En notant $d_{h\ell} = V_h \cap W_\ell$ le nombre de tels coefficients diagonaux nuls du bloc ${}^sB^{(h\ell)}$, nous obtenons que la densité $\alpha'_{h\ell}$ du bloc privé de ses coefficients nuls fixés vaut :

$$\alpha'_{h\ell} = \frac{N_{h\ell}}{|V_h|\times|W_\ell| - d_{h\ell}}$$

Chaque bloc ${}^sB^{(h\ell)}$ de la matrice simulée possédera ainsi $d_{h\ell}$ coefficients nuls et les $\mid V_h \mid \times \mid W_\ell \mid - d_{h\ell}$ autres seront finalement simulés indépendamment les uns des autres selon une loi de Bernoulli $\mathcal{B}(\alpha'_{h\ell})$.

1.2 Reconstruction de la matrice d'adjacence du graphe

Il s'agit dans cette partie de construire le graphe orienté ${}^s\mathcal{G}$ à partir de la matrice par bloc ${}^sB$ fraichement créée. Pour ce faire, "désorganisons" la matrice ${}^sB$ pour reconstituer la matrice d'adjacence ${}^sA$ de ${}^s\mathcal{G}$.

Soit j et $j' \in \{1, \ldots, p\}$. Le problème consiste en l'attribution à ${}^sA_{jj'}$ d'un coefficient de ${}^sB$. Il faut que chaque coefficient de ${}^sB$ corresponde à un et un seul coefficient de ${}^sA$.

Pour ce faire, on repère le bloc d'appartenance du couple de variables (j, j') dans ${}^sB$. (j, j') appartient au bloc $h\ell$ tel que $v_{jh} \times w_{j'\ell} = 1$. La variable $j \in V_h$, la variable $j' \in W_\ell$ et ${}^sA_{jj'}$ est donc un coefficient du bloc ${}^sB^{(h\ell)}$. Pour choisir un coefficient du bloc en particulier, nous introduisons une relation d'ordre $\leq$ dans les ensembles V_h et W_ℓ :

$$\forall j_1, j_2 \in V_h,\ j_1 \leq j_2 \Leftrightarrow \text{le numéro de } j_1 \text{ est inférieur à celui de } j_2$$
$$\forall j'_1, j'_2 \in W_\ell,\ j'_1 \leq j'_2 \Leftrightarrow \text{le numéro de } j'_1 \text{ est inférieur à celui de } j'_2$$

Notons $n_h(j)$ le classement de la variable j dans V_h selon la relation $\leq$ et $n_\ell(j')$ celui de j' dans W_ℓ. On choisit alors ${}^sA_{j,j'} = {}^sB^{(h\ell)}_{n_h(j),n_\ell(j')}$. En procédant de la sorte pour attribuer une valeur à chaque coefficient de ${}^sA$, nous nous assurons que l'application qui à chaque coefficient de ${}^sA$ associe un coefficient de ${}^sB$ est bijective.

Exemple : On se place dans le cadre suivant :
— $J = \{1, ..., 9\}$, $p = 9$, $H = 2$, $L = 3$.
— $V_1 = \{1, 2, 6, 7\}$, $V_2 = \{3, 4, 5, 8, 9\}$
— $W_1 = \{1, 8\}$, $W_2 = \{2, 4, 6, 9\}$, $W_3 = \{3, 5, 7\}$.

— ${}^sB = \left(\begin{array}{cc|cccc|ccc} \color{red}{0} & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & \color{red}{0} & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & \color{red}{0} & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & \color{green}{0} & 0 & 1 & 1 & \color{red}{0} \\ \hline 1 & 0 & 1 & 1 & 1 & 0 & \color{red}{0} & 0 & \color{blue}{1} \\ 0 & 1 & 0 & \color{red}{0} & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & \color{red}{0} & 1 \\ 0 & \color{red}{0} & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & \color{red}{0} & 0 & 0 & 0 \end{array}\right)$ la matrice par blocs simulée.

Les coefficients rouges correspondent aux coefficients nuls fixés qui se situeront sur la diagonale de la matrice d'adjacence à reformer ${}^sA$.

À l'aide de la méthode mise en place, la matrice d'adjacence associée à ${}^sB$ issue de cette étape sera celle-ci :

$$ {}^sA = \begin{pmatrix} \color{red}{0} & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 1 & \color{red}{0} & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & \color{red}{0} & 1 & 0 & 1 & \color{blue}{1} & 0 & 0 \\ 0 & 0 & 0 & \color{red}{0} & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & \color{red}{0} & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & \color{red}{0} & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & \color{green}{0} & \color{red}{0} & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & \color{red}{0} & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & \color{red}{0} \end{pmatrix} $$

Par exemple, pour retrouver la valeur du coefficient ${}^sA_{3,7}$ coloré en bleu, nous avons repéré le bloc d'appartenance du couple $(3, 7)$. Etant donné que $3 \in V_2$ et $7 \in W_3$, ${}^sA_{3,7}$ est un coefficient du bloc ${}^sB^{(23)}$. Le classement de la variable 3 dans V_2 suivant la relation $\leq$ est $n_2(3) = 1$ et celui de la variable 7 dans W_3 est $n_3(7) = 3$. Ainsi, on attribue à ${}^sA_{3,7}$ le coefficient situé en ligne 1 et colonne 3 du bloc ${}^sB^{(23)}$, coloré en bleu dans la matrice ${}^sB$, à savoir 1.

De même, pour retrouver la valeur du coefficient ${}^sA_{7,6}$ coloré en vert, nous avons repéré le bloc d'appartenance du couple $(7, 6)$. Etant donné que $7 \in V_1$ et $6 \in W_2$, ${}^sA_{7,6}$ est un coefficient du bloc ${}^sB^{(12)}$. Le classement de la variable 7 dans V_1 suivant la relation $\leq$ est $n_1(7) = 4$ et celui de la variable 6 dans W_2 est $n_2(6) = 3$. Ainsi, on attribue à ${}^sA_{7,6}$ le coefficient situé en ligne 4 et colonne 3 du bloc ${}^sB^{(12)}$, coloré en vert dans la matrice ${}^sB$, à savoir 0.

1.3 Formation des équations de régression linéaire

La construction du graphe orienté simulé ${}^s\mathcal{G}$ induit l'obtention pour chaque variable, d'un support ${}^sS_j = \{j' \in \{1,\dots,p\}/{}^sA_{jj'} = 1\}$, à savoir justement l'ensemble de ses variables explicatives. Notons que pour une variable j, le support obtenu sur le jeu réel $\widehat{S}_j$ et celui nouvellement formé ${}^sS_j$ sont deux ensembles dont les contenus sont proches mais néanmoins différents. Nous pouvons reformer les p équations de régression linéaires de chacune des variables sur celles de son support simulé. Ces équations de régression définiront les lois conditionnelles de chaque variable sachant ses variables explicatives. Dans l'optique d'obtenir la loi jointe des variables, puis de simuler un jeu de données selon cette loi, on pourra avoir recours à l'échantillonnage de Gibbs ([23]).

Nous adopterons les notations suivantes :

— ${}^sX$ un tel jeu de données simulé de taille $n \times p$.
— Pour un ensemble $\mathcal{M} \subset \{1,\dots,p\}$, ${}^sX^{\mathcal{M}}$ la matrice ${}^sX$ restreinte aux variables $j \in \mathcal{M}$.
— Si $\mathcal{M} = \{j\}$, on posera ${}^sX^j = {}^sX^{\{j\}}$.On a alors : ${}^sX^j$ est de taille $n \times 1$.
— ${}^sX_j$ la variable aléatoire associée à $j \in \{1,\dots,p\}$ dont ${}^sX^j$ est un échantillon de taille n.

Pour un jeu de données simulé ${}^sX$, l'équation de régression linéaire aux paramètres Θ_j et σ_j de la variable $j \in \{1,\dots,p\}$ sur les variables de son support s'écrit ainsi :

$$ {}^sX^j = {}^sX^{{}^sS_j}\Theta_j + \epsilon_j \text{ avec } \epsilon_j \sim \mathcal{N}(0, \sigma_j^2)$$

Pour avoir recours à l'échantillonneur de Gibbs, les paramètres Θ_j et σ_j doivent être fixés au préalable. Ensuite, une fois les variables ${}^sX^j$ tirées aléatoirement, celles-ci seront remises à jour étape par étape dans l'échantillonneur. Nous décidons d'estimer les deux paramètres à fixer à l'aide du jeu de données réel. En effet, la procédure de simulation dans son ensemble est mise en place de telle sorte que chaque variable j du jeu réel joue le même rôle que la variable j du jeu simulé. Donc estimer les coefficients de régression Θ_j et les variances des résidus ϵ_j d'une variable j à l'aide de sa variable analogue sur le jeu réel paraît être un choix naturel. Nous allons ainsi estimer pour chaque variable j, son vecteur de coefficients de régression Θ_j associé par la méthode des moindres carrés ordinaires. Cette méthode est appliquée à l'équation de régression sur les données réelles X, mais en ne gardant comme variables explicatives que celles du support simulé ${}^sS_j$. On en déduit également une estimation de la variance du résidu ${}^s\epsilon_j$.

Les paramètres ainsi estimés sont notés ${}^m\widehat{\Theta}_j$ et ${}^m\widehat{\sigma}_j$:

$${}^m\widehat{\Theta}_j = \underset{\Theta_j}{\operatorname{argmin}} \left(||X^j - X^{{}^sS_j}\Theta_j||_2^2 \right),\ {}^m\widehat{\epsilon_j} = X^j - X^{{}^sS_j}.{}^m\widehat{\Theta}_j \text{ et } {}^m\widehat{\sigma}_j^2 = \operatorname{Var}({}^m\widehat{\epsilon_j})$$

On en déduit ainsi l'équation de régression linéaire, aux paramètres estimés sur le jeu réel, de la variable j sur celles de son support :

$${}^sX^j = {}^sX^{{}^sS_j}.{}^m\widehat{\Theta}_j + {}^m\widehat{\epsilon_j} \text{ avec } {}^m\widehat{\epsilon_j} \sim \mathcal{N}(0, {}^m\widehat{\sigma}_j^2) \tag{4.1}$$

1.4 Obtention d'un jeu simulé par échantillonneur de Gibbs

À l'aide des équations de régression (4.1), chaque variable aléatoire ${}^sX_j$ suit conditionnellement aux $p-1$ autres une loi normale multivariée :

$$\begin{cases} {}^sX_1 | {}^sX_2, ..., {}^sX_p; {}^m\widehat{\Theta}_1, {}^m\widehat{\sigma}_1 \sim \mathcal{N}\left({}^sX^{{}^sS_1}.{}^m\widehat{\Theta}_1, {}^m\widehat{\sigma}_1^2 I_n\right) \\ \vdots \\ {}^sX_j | {}^sX_1, ..., {}^sX_{j-1}, {}^sX_{j+1}, ..., {}^sX_p; {}^m\widehat{\Theta}_j, {}^m\widehat{\sigma}_j \sim \mathcal{N}\left({}^sX^{{}^sS_j}.{}^m\widehat{\Theta}_j, {}^m\widehat{\sigma}_j^2 I_n\right) \text{ pour tout } j \\ \vdots \\ {}^sX_p | {}^sX_1, ..., {}^sX_{p-1}; {}^m\widehat{\Theta}_p, {}^m\widehat{\sigma}_p \sim \mathcal{N}\left({}^sX^{{}^sS_p}.{}^m\widehat{\Theta}_p, {}^m\widehat{\sigma}_p^2 I_n\right) \end{cases} \tag{4.2}$$

Étant donné que chaque variable X_j du jeu de données réel suit une loi gaussienne $\mathcal{N}(0,1)$, nous tirons aléatoirement, en premier lieu, chacun des vecteurs ${}^sX^j$ comme étant un échantillon de taille $1 \times p$ de cette même loi. Toutes les conditions sont réunies pour appliquer un échantillonneur de Gibbs et obtenir un jeu de données simulées prenant en compte les lois conditionnelles de (4.2). Notons bien, en passant, que les supports ${}^sS_j$ intervenant dans ces équations sont également fixes et ne seront pas réactualisés au fur et à mesure que l'algorithme avance. Posons le vecteur de paramètres ${}^sX^{(t)} = \left({}^sX_1^{(t)}, {}^sX_2^{(t)}, ..., {}^sX_p^{(t)}\right)$ qui correspond au vecteur de variables en sortie de l'étape t de l'algorithme et pour tout $j \in \{1, \ldots, p\}$, f_j la loi conditionnelle de ${}^sX_j$ sachant les autres variables selon (1.2). Celui-ci s'articule ainsi de cette manière :

1. Etape 1 : Initialisation
   $\forall j \in \{1, \ldots, p\}$, génération de ${}^sX_j^{(0)}$ selon une loi $\mathcal{N}(0,1)$.
2. Etape 2 : Hérédité
   (a) génération de ${}^sX_1^{(t+1)}$ selon $f_1\left({}^sX_1 | {}^sX_2^{(t)}, ..., {}^sX_p^{(t)}; {}^m\widehat{\Theta}_1, {}^m\widehat{\sigma}_1\right)$
   (b) génération de ${}^sX_2^{(t+1)}$ selon $f_2\left({}^sX_2 | {}^sX_1^{(t+1)}, {}^sX_3^{(t)}, ..., {}^sX_p^{(t)}; {}^m\widehat{\Theta}_2, {}^m\widehat{\sigma}_2\right)$
   $\vdots$

(c) génération de ${}^sX_j^{(t+1)}$ selon $f_j\left({}^sX_j|{}^sX_1^{(t+1)},...,{}^sX_{j-1}^{(t+1)},{}^sX_{j+1}^{(t)},...,{}^sX_p^{(t)};{}^m\widehat{\Theta}_j,{}^m\widehat{\sigma}_j\right)$

$\vdots$

(d) génération de ${}^sX_p^{(t+1)}$ selon $f_p\left({}^sX_p|{}^sX_1^{(t+1)},...,{}^sX_{p-1}^{(t+1)};{}^m\widehat{\Theta}_p,{}^m\widehat{\sigma}_p\right)$

3. Etape 3 : Arrêt
   Choix d'une étape t d'arrêt de l'algorithme par le biais d'une règle de décision.

Dans un souci de contrôle de la convergence de l'algorithme, nous décidons de regarder, pour une variable j, l'évolution en fonction de t, de la fonction N_j correspondant à la différence en norme 2 entre le vecteur à l'étape $t+1$ et celui à l'étape t :

$$N_j : t \mapsto g(t) = ||{}^sX_j^{(t+1)} - {}^sX_j^{(t)}||_2$$

Comme il est difficile de s'attarder sur ces p telles fonctions, nous décidons de nous restreindre uniquement à l'ensemble des cinq variables typiques que nous avions choisi au chapitre 2. Nous nous restreindrons à un nombre d'étapes t_{iter} dans l'algorithme tel que les normes $N_j(t_{iter})$ se soient stabilisées pour chacune des variables typiques. Par conséquent, nous récoltons le vecteur de variables $\left({}^sX_1^{(t_{iter})},{}^sX_2^{(t_{iter})},...,{}^sX_p^{(t_{iter})}\right)$ que l'on juge stabilisé.

En outre, une fois l'algorithme stabilisé comme souhaité, les variables obtenues sont d'espérance nulle mais les différentes modifications apportées au fur et à mesure de l'algorithme ont modifié sa variance. C'est pourquoi, nous réduisons chacune des variables obtenues de façon à ce que chaque vecteur ${}^sX^j$ soit bien un échantillon de loi gaussienne $\mathcal{N}(0,1)$.

1.5 Traitement du jeu de données

Nous pouvons simuler plusieurs jeux de données suivant la procédure que nous venons de décrire. Une fois différents jeux simulés, il s'agit d'appliquer sur chacun d'entre eux la procédure de sélection que nous avions choisie au préalable, de façon à construire plusieurs graphes puis d'appliquer sur chacun de ces graphes la procédure de classification. Cela nous permettra d'obtenir plusieurs matrices de classification par blocs. L'objectif est alors d'évaluer la proximité de ces dernières avec la matrice par blocs issue du jeu réel de façon à estimer la stabilité de la procédure statistique globale à l'aide du $CARI$. Cela nous permettra notamment de savoir si la procédure globale est plus stable lorsque la procédure de sélection utilisée est Gauss-LASSO stabilisé ou lorsque l'on utilise Gauss-LASSO enrichi. La première de ces deux procédure étant la plus stable, nous pourrons observer si la stabilité recherchée lors de l'étape de sélection a un impact sur la stabilité de la procédure globale.

2 Résultats

Nous avons appliqué cette procédure de validation en simulant dix jeux de données de la sorte pour chacune des deux stratégies de sélection de variables employées (Gauss-LASSO stabilisé et enrichi). Nous analyserons ces résultats en deux parties. La première consistera à examiner le déroulement des étapes successives qui ont mené à la formation des jeux de données, puis la comparaison de ces données ainsi simulées avec les données réelles. La seconde exposera la comparaison des matrices par blocs obtenues après application de la procédure globale sur les jeux simulés avec les matrices par blocs réelles.

2.1 Simulation des jeux

Nous allons exposer les objets obtenus après les trois étapes significatives de la formation des jeux simulés :

1. les matrices d'adjacences reconstruites (étape 2)
2. les paramètres estimés sur les modèles de régression (étape 4)
3. les jeux de données finaux (étape 6)

2.1.1 Comparaison des matrices d'adjacence simulées et réelles

Pour chacune des procédures de sélection, nous avons simulé dix matrices par blocs binaires $\left\{{}^{s_k}B^{GLstab}\right\}_{k\in\{1,\dots,10\}}$ et $\left\{{}^{s_k}B^{GLenri}\right\}_{k\in\{1,\dots,10\}}$, à partir des partitions doubles et des densités des blocs $\alpha'_{h\ell}$ observées sur le jeu réel (voir détail au paragraphe 1.1). La répartition de ces densités est d'ailleurs représentée en figure 4.1. Vingt matrices d'adjacence $\left\{{}^{s_k}A^{GLstab}\right\}_{k\in\{1,\dots,10\}}$ et $\left\{{}^{s_k}A^{GLenri}\right\}_{k\in\{1,\dots,10\}}$ ont été reconstruites à partir de ces matrices par blocs par le procédé détaillé en 1.2. Chacune de ces matrices d'adjacence fait état d'un support pour chacune des p variables. Nous obtenons donc vingt ensembles de p supports :

$$\left\{\left\{{}^{s_k}S_j^{GLstab}\right\}_{j\in\{1,\dots,p\}}\right\}_{k\in\{1,\dots,10\}} \text{ et } \left\{\left\{{}^{s_k}S_j^{GLenri}\right\}_{j\in\{1,\dots,p\}}\right\}_{k\in\{1,\dots,10\}}.$$

Comparons longueurs et contenus des dix premiers ensembles de supports à l'ensemble $\left\{\widehat{S}_j^{GLstab}\right\}_{j\in\{1,\dots,p\}}$ des p supports obtenus par application de Gauss-LASSO stabilisé sur le jeu réel, puis des dix derniers ensembles à $\left\{\widehat{S}_j^{GLenri}\right\}_{j\in\{1,\dots,p\}}$ issus de l'application de Gauss-LASSO enrichi sur le jeu réel.

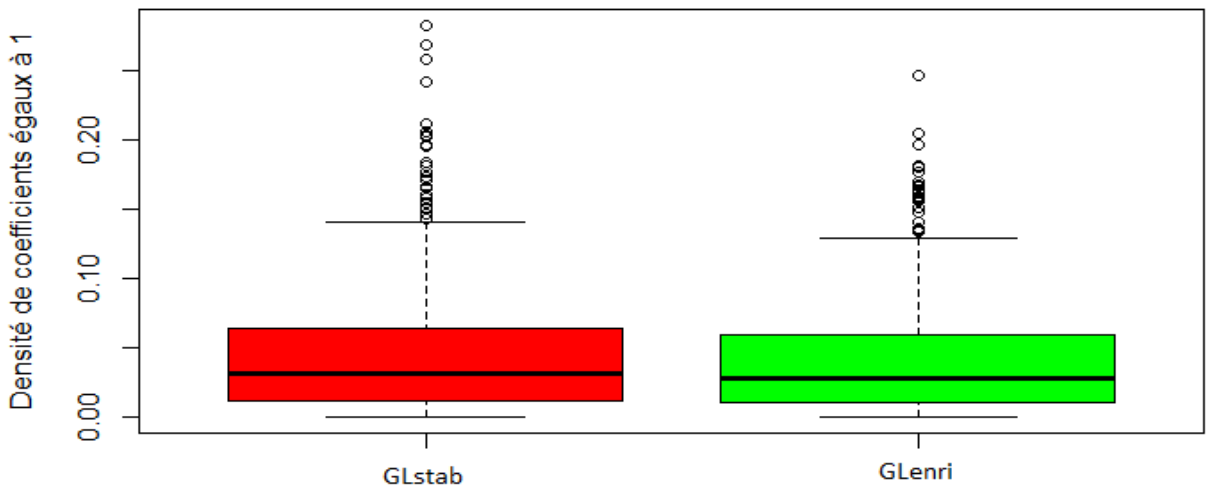

Figure 4.1 – Répartition des densités $\left\{\alpha'_{h\ell}\right\}_{h,l}$ des blocs des matrices B^{GLstab} et B^{GLenri}

Selon la figure 4.2, la distribution de la taille des p supports induits par chacune des matrices d'adjacence simulées dans le cadre de Gauss-LASSO stabilisé ressemble fortement à la distrubution des p supports $\widehat{S}_j^{GLstab}$. Cependant, pour chaque variable j, le nombre de variables communes à l'un de ses supports simulés et à son support réel est plutôt faible (voir figure 4.3). Ces observations ne sont pas étonnantes. En effet, le nombre de coefficents égaux à 1 de chaque bloc sera conservé par simulations puisque celles-ci sont établies à partir des densités des blocs. Ceci explique le constat établi au vu de la figure 4.2. Mais, au vu des faibles densités recensées de la majorité des blocs (voir figure 4.1), les positions de ces quelques coefficients valant 1 tirés aléatoirement dans ces blocs risquent de différer fortement selon le jeu simulé. Ceci explique le constat établi sur la figure 4.3.

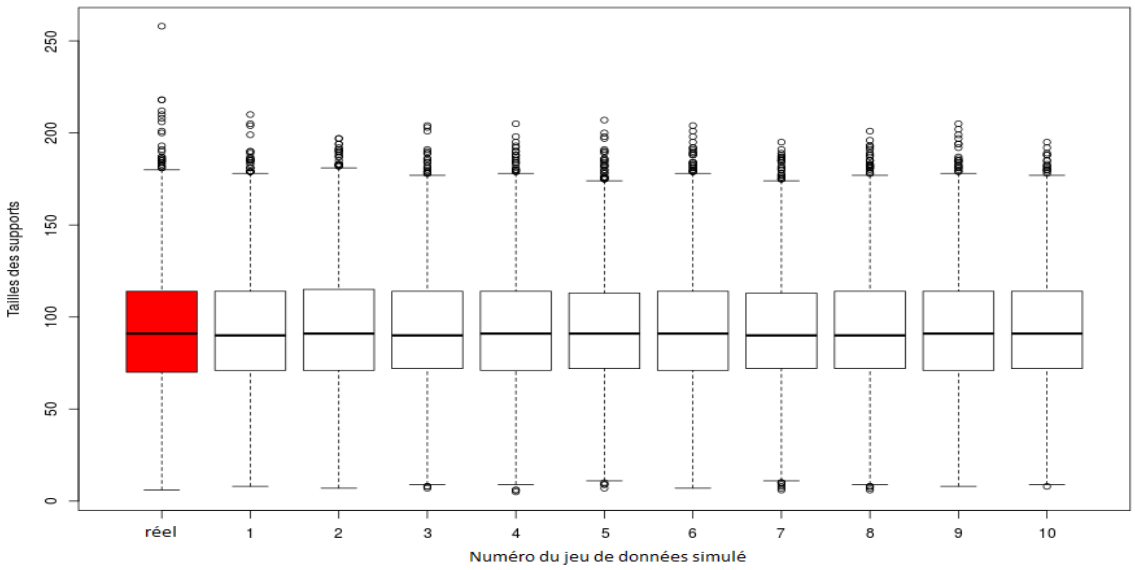

Figure 4.2 – Répartition de la taille des supports induits par les matrices d'ajdacences ${}^{s_k}A^{GLstab}$ comparativement aux supports estimés sur le jeu réel par Gauss-LASSO stabilisé

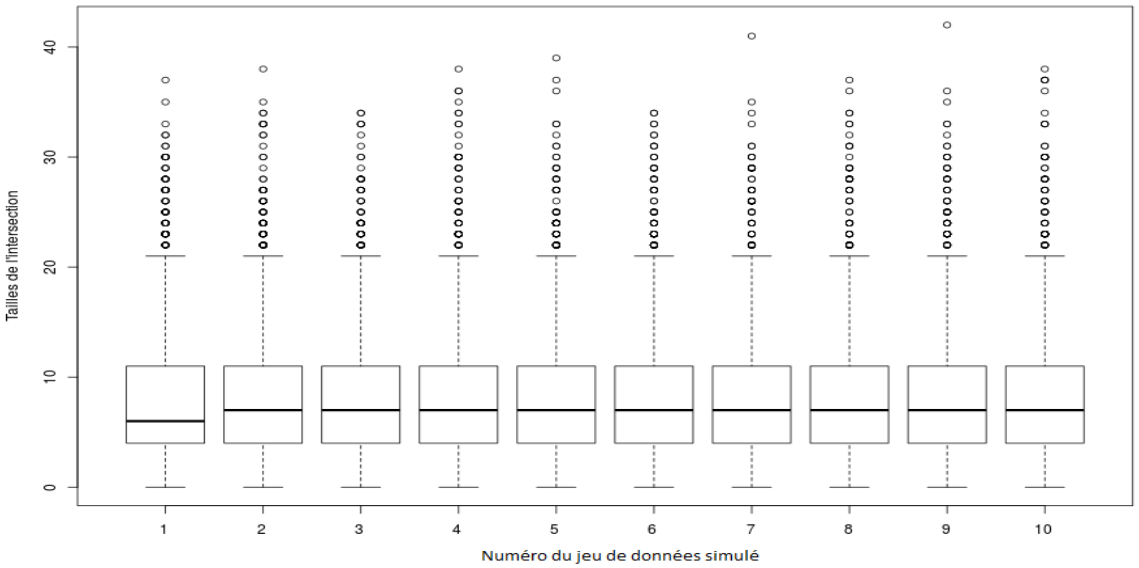

Figure 4.3 – Répartition de la taille de l'intersection, pour chaque jeu simulé s_k, entre les supports $\left\{ {}^{s_k}S_j^{GLstab} \right\}_{j\in\{1,\dots,p\}}$ et les supports $\left\{ \widehat{S}_j^{GLstab} \right\}_{j\in\{1,\dots,p\}}$

Nous observons exactement les mêmes constats pour la comparaison des supports obtenus via les simulations et ceux estimés via le jeu réel dans le cadre de Gauss-LASSO enrichi. (voir figures 4.4 et 4.5). Ces constats appuient le fait que nous recherchons, dans ce chapitre, une stabilité de la procédure dans sa globalité et non une stabilité de l'étape de sélection et de classification séparées. Des matrices d'adjacence issues de l'étape de sélection peuvent être très différentes et fournir des matrices par blocs similaires après classification.

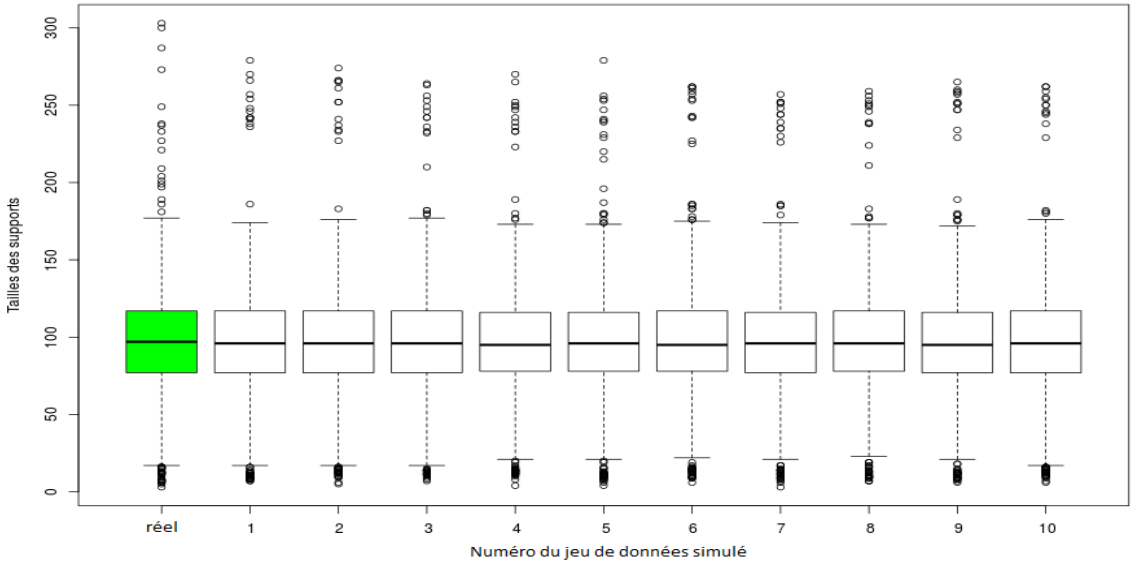

Figure 4.4 – Répartition de la taille des supports induits par les matrices d'ajdacences ${}^{s_k}A^{GLenri}$ comparativement aux supports estimés sur le jeu réel par Gauss-LASSO enrichi

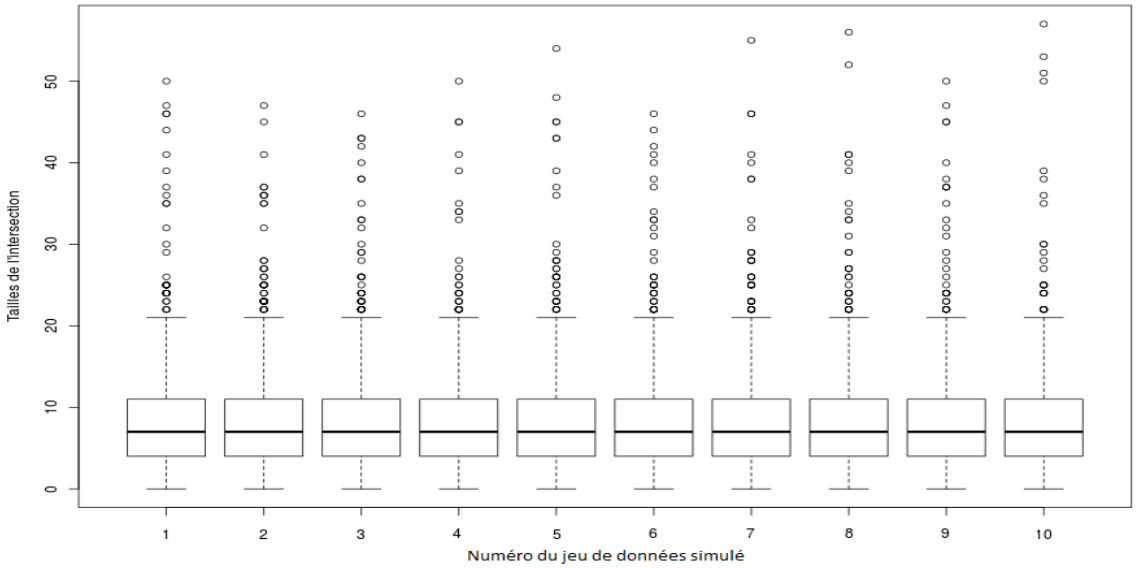

Figure 4.5 – Répartition de la taille de l'intersection, pour chaque jeu simulé s_k, entre les supports $\left\{ {}^{s_k}S_j^{GLenri} \right\}_{j\in\{1,\dots,p\}}$ et les supports $\left\{ \widehat{S}_j^{GLenri} \right\}_{j\in\{1,\dots,p\}}$

2.1.2 Estimation des paramètres des équations de régression

Les équations de régression 4.2 ont leurs paramètres estimés sur les données rélles. Voyons, pour quatre variables typiques, la valeur des coefficients de régression ainsi estimés pour chacune des dix simulations. Leur répartition est exposée en figure 2.1.2 dans le cadre de Gauss-LASSO stabilisé et en figure 2.1.2 dans le cadre de Gauss-LASSO enrichi.

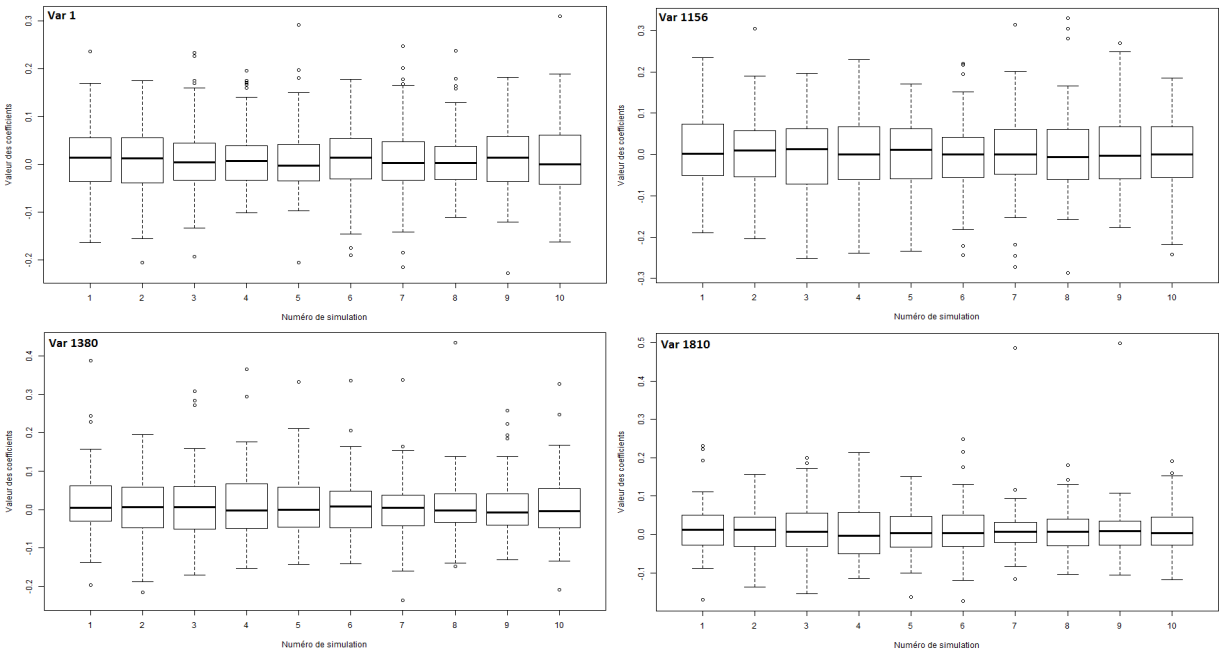

Figure 4.6 – Répartition des coefficients de régression estimés sur le jeu réel pour chaque simulation et pour quatre variables typiques dans le cadre de Gauss-LASSO stabilisé

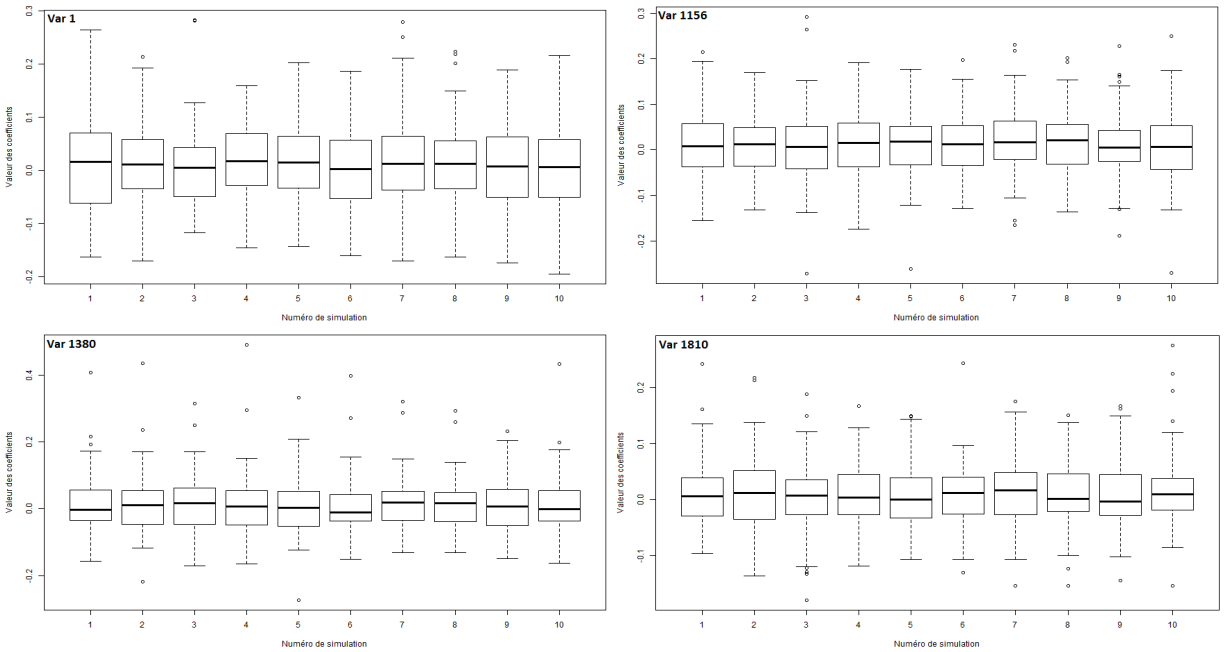

Figure 4.7 – Répartition des coefficients de régression estimés sur le jeu réel pour chaque simulation et pour quatre variables typiques dans le cadre de Gauss-LASSO enrichi

L'allure de ces coefficients ressemble fortement à ceux de l'étude faite dans le chapitre "Procédure de sélection stables". La plupart de ces coefficients ${}^{m}\widehat{\Theta}_j$ ont une valeur faible comprise entre -0.2 et 0.2. Quelques coefficients font exception. Il s'agit de ceux associés aux variables des supports les plus corrélées avec la variable régressée. Dans le cadre de ce précédent chapitre, estimation des supports et des coefficients de régression étaient réalisées toutes deux sur le jeu réel. Dans le cadre du chapitre courant, les supports sont déterminés par simulation et seule l'estimation des coefficients est faite sur le jeu réel.

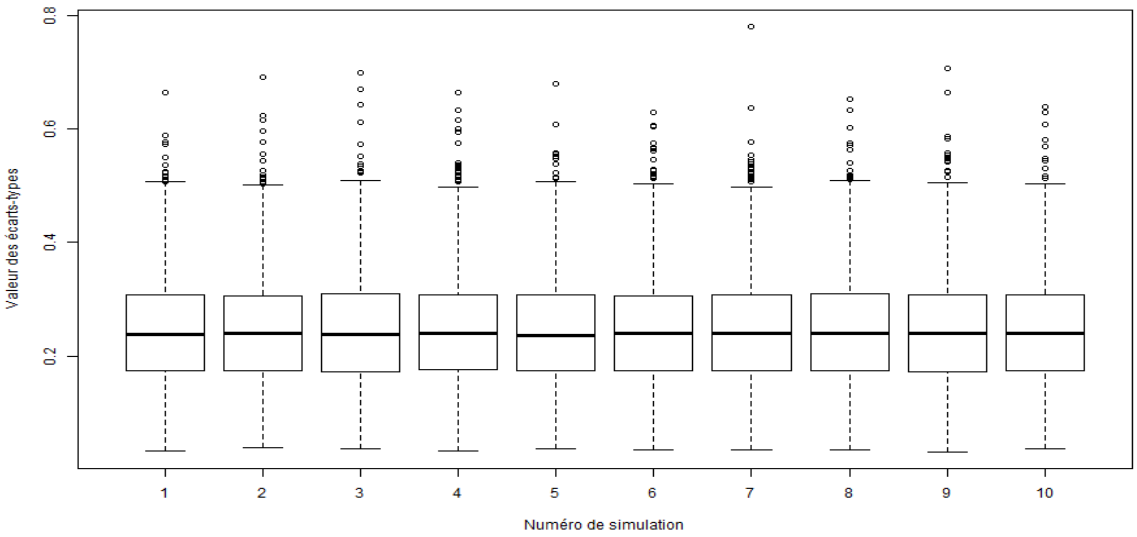

Figure 4.8 – Répartition des écart-types des p résidus estimés sur le jeu réel pour chaque simulation dans le cadre de Gauss-LASSO stabilisé

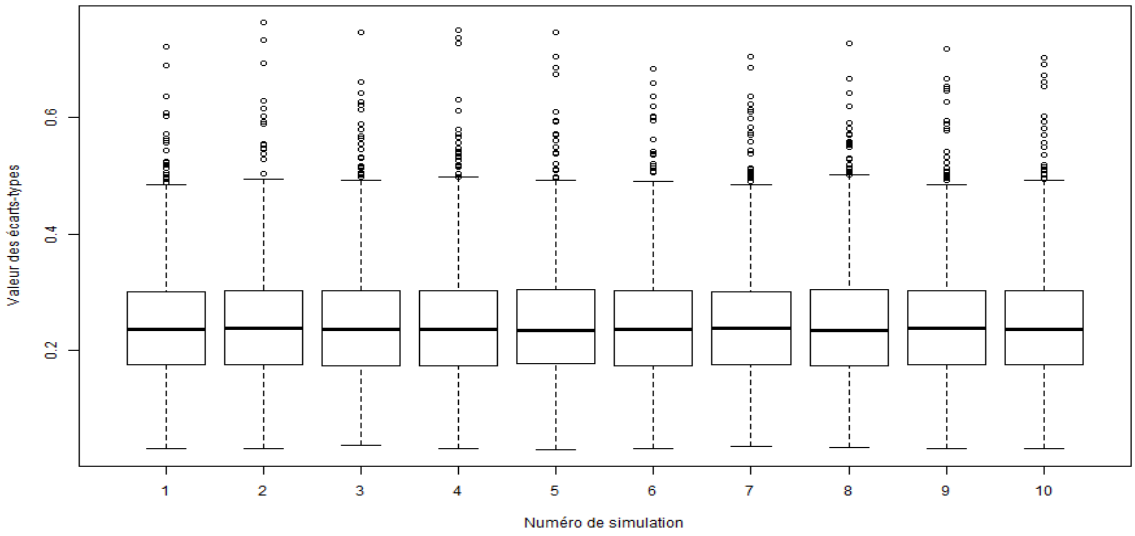

Figure 4.9 – Répartition des écart-types des p résidus estimés sur le jeu réel pour chaque simulation dans le cadre de Gauss-LASSO enrichi

Cette différence tend à expliquer les valeurs des écarts-types ${}^{m}\widehat{\sigma}_j$ des résidus estimés sur le jeu réel pour les p équations de régression établies à partir des simulations. Les figures 4.8 et 4.9 illustrent que ces résidus estimés ont en effet des écart-types relativement élevés. Les supports simulés et estimés sur le jeu réel possédant peu de variables communes, les variables explicatives des équations de régression 4.2 ne sont en majorité pas présentes dans le support réel, entraînant ce constat sur les résidus des équations de régression. Ceci risque d'impacter l'étape d'échantillonnage de Gibbs dans laquelle, à chaque itération des variables normales d'écart-types ${}^{m}\widehat{\sigma}_j$ sont tirées aléatoirement. La stabilisation de l'algorithme risque soit d'être lente si stabilisation il y a.

2.1.3 Etape d'échantillonnage de Gibbs

Les figures 2.1.3 et 2.1.3 illustrent l'évolution de la fonction $N_j : t \mapsto g(t) = ||{}^{s}X_j^{(t+1)} - {}^{s}X_j^{(t)}||_2$ permettant de s'assurer la convergence de l'algorithme pour quatre variables typiques et pour chacune des dix simulations. Contrairement à ce qui a été subodoré précédemment, l'algorithme se stabilise très rapidement quelle que soit la simulation considérée. En général, cinq itérations sont suffisantes pour que l'algorithme se stabilise. En revanche, ces fonctions N_j se stabilisent autour de valeurs relativement éloignées de 0. Ce constat est la conséquence de ce qui avait été constaté au niveau des écart-types des résidus estimés. Nous décidons de conserver comme échantillon final, celui délivré par l'échantillonneur de Gibbs après $t_{iter} = 20$ itérations : $\left({}^{s}X_1^{(20)}, {}^{s}X_2^{(20)}, \ldots, {}^{s}X_p^{(20)}\right)$. Après centrage et réduction de chacune de ces variables, nous obtenons un jeu de données escompté. Nous noterons ces matrices de données $\left\{{}^{s_k}X^{GLstab}\right\}_{k\in\{1,\ldots10\}}$ et $\left\{{}^{s_k}X^{GLenri}\right\}_{k\in\{1,\ldots10\}}$.

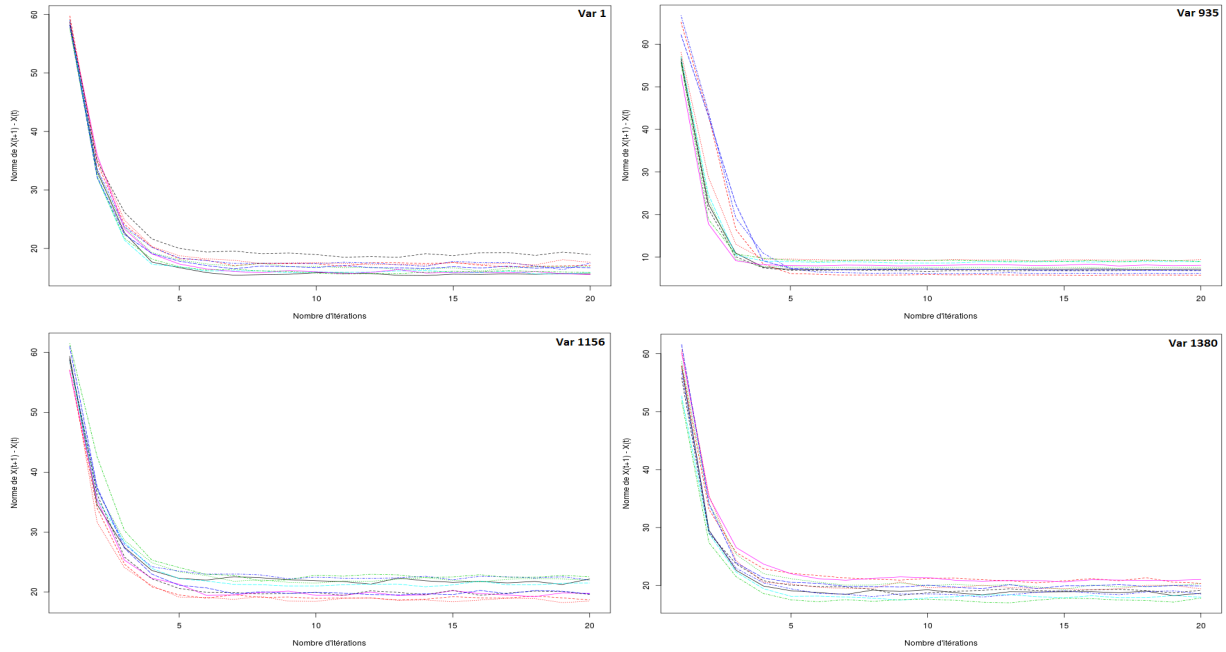

Figure 4.10 – Représentation des courbes $||{}^{s}X_j^{(t+1)} - {}^{s}X_j^{(t)}||_2$ en fonction de t pour chacune des 10 simulations et pour chaque pour quatre variables typiques dans le cadre de Gauss-LASSO stabilisé

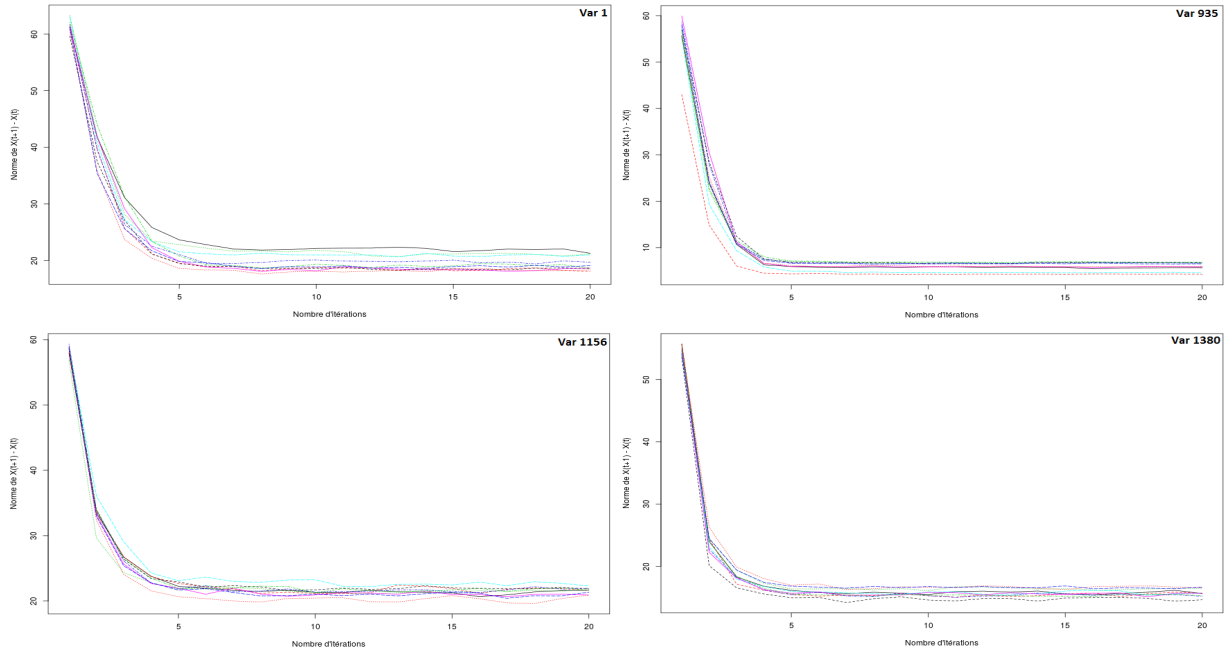

Figure 4.11 – Représentation des courbes $||{}^{s}X_j^{(t+1)} - {}^{s}X_j^{(t)}||_2$ en fonction de t pour chacune des 10 simulations et pour chaque pour quatre variables typiques dans le cadre de Gauss-LASSO enrichi

Les figures 4.12 et 4.13 établissent la comparaison entre chacune des dix matrices de données ainsi fraîchement simulées, dans le cadre d'une sélection par Gauss-LASSO stabilisé ou enrichi, avec la matrice de données réelles X. Les répartitions des coefficients

de chaque matrice $D_k^{GLstab} = {}^{s_k}X^{GLstab} - X$ (respectivement $D_k^{GLenri} = {}^{s_k}X^{GLenri} - X$) y sont en effet exposées.

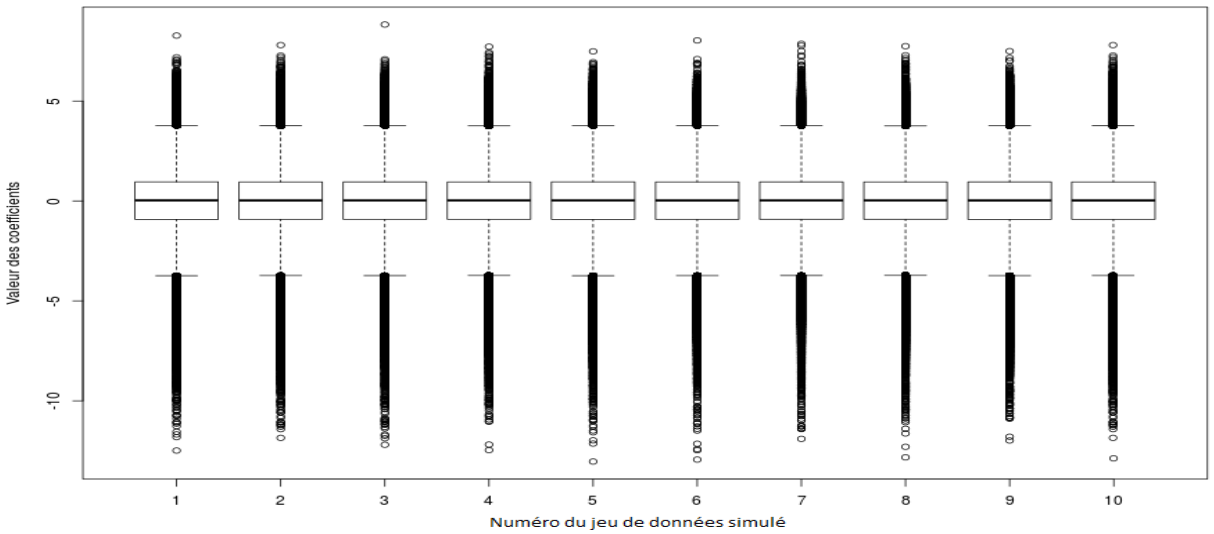

Figure 4.12 – Répartition des coefficients de chacune des matrices D_k^{GLstab}

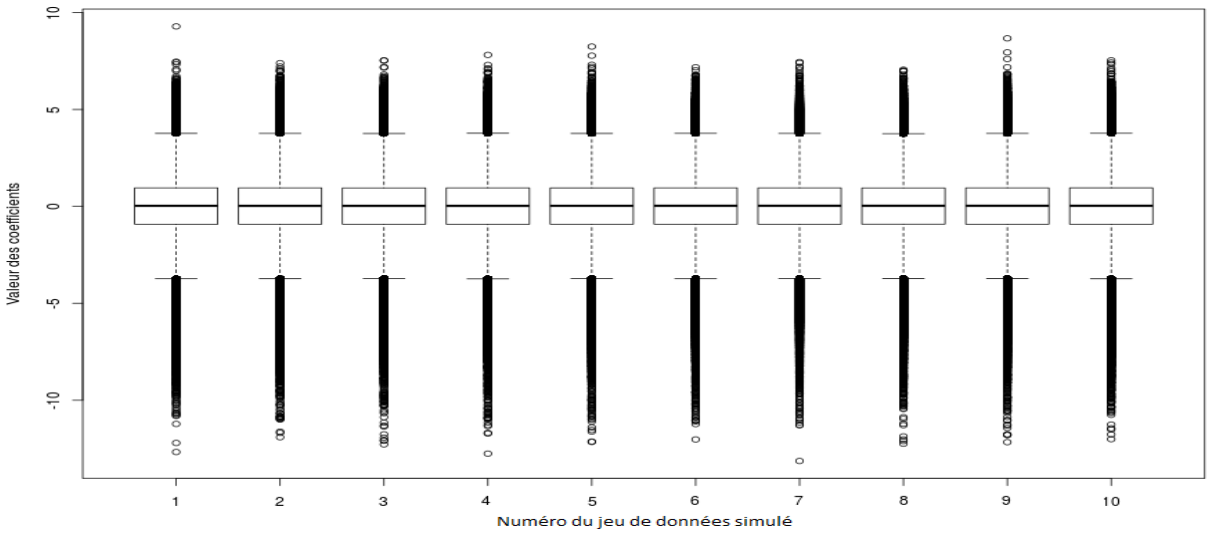

Figure 4.13 – Répartition des coefficients de chacune des matrices D_k^{GLenri}

Il semble clair que les matrices de données finales ainsi simulées et la matrice réelle sont éloignées. Au vu des différences observées dans les étapes détaillées précédemment, ce constat n'est guère surprenant. Néanmoins si notre procédure statistique globale est stable, l'appliquer sur chacun des jeux de données ainsi créés par notre méthode de simulation doit nous livrer des classifications doubles proches de celles obtenues sur le jeu de données réel. Ceci constitue l'indicateur qui permettra de valider ou non cette procédure globale.

2.2 Application de la procédure globale

Nous avons relancé la procédure globale sur les dix jeux ${}^{s_k}X^{GLstab}$ avec l'étape de sélection réalisée par Gauss-LASSO stabilisé et sur les dix jeux ${}^{s_k}X^{GLenri}$ avec l'étape de sélection réalisée par Gauss-LASSO enrichi. Nous avons ainsi obtenu dix matrices par blocs que l'on notera ${}^{s_k}\widehat{B}^{GLstab}$ puis dix matrices par blocs que l'on notera ${}^{s_k}\widehat{B}^{GLenri}$ pour $k \in \{1, \ldots, 10\}$. Une de ces matrices par blocs est représentée en figure 4.14. Notons les couples de partitions doubles associées à chacune de ces matrices $\left({}^{s_k}\widehat{v}^{GLstab}, {}^{s_k}\widehat{w}^{GLstab}\right)$ possédant un couple de nombre de classes $\left({}^{s_k}\widehat{H}^{GLstab}, {}^{s_k}\widehat{L}^{GLstab}\right)$ dans le cadre de Gauss-LASSO stabilisé et $\left({}^{s_k}\widehat{H}^{GLenri}, {}^{s_k}\widehat{L}^{GLenri}\right)$ dans le cadre du Gauss-LASSO enrichi.

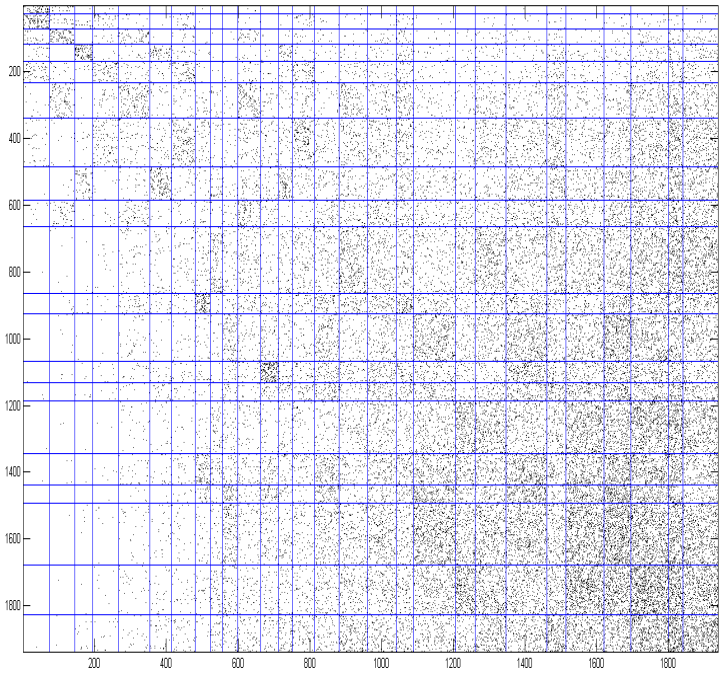

Figure 4.14 – Matrice par blocs ${}^{s7}\widehat{B}^{GLstab}$

Nous avons d'abord recensé le nombre de classes en ligne et en colonne de chacune de ces matrices par blocs. Ces résultats sont affichés dans le tableau 4.1. On peut noter que la plupart des estimations réalisées sur les jeux de données simulés munissent les matrices par blocs de partitions doubles aux nombres de classes inférieurs à ceux estimés sur le jeu de données réel. Il est également notable que globalement le nombre moyen de partitions doubles estimées dans le cadre de Gauss-LASSO stabilisé est plus important que celui estimé dans le cadre de Gauss-LASSO enrichi. Pour comparer ces classifications dans le détail, nous faisons appel au $CARI$, qui va nous permettre d'évaluer la stabilité de la procédure globale. Les valeurs de l'indice exprimant le lien entre chaque partition double estimée sur les jeux simulés et celle estimée sur le jeu réel sont recensées dans les tableaux 4.2 et 4.3.

Simulation k	$\left({}_{s_k}\widehat{H}^{GLstab}, {}_{s_k}\widehat{L}^{GLstab}\right)$	$\left({}_{s_k}\widehat{H}^{GLenri}, {}_{s_k}\widehat{L}^{GLenri}\right)$
1	$(22, 25)$	$(23, 27)$
2	$(28, 20)$	$(26, 26)$
3	$(23, 21)$	$(19, 23)$
4	$(26, 22)$	$(20, 19)$
5	$(31, 28)$	$(24, 21)$
6	$(18, 25)$	$(17, 26)$
7	$(20, 28)$	$(21, 18)$
8	$(25, 19)$	$(17, 19)$
9	$(28, 26)$	$(22, 23)$
10	$(29, 30)$	$(23, 23)$
Jeu réel	$(30, 29)$	$(26, 23)$

Table 4.1 – Nombres de classes en ligne et en colonnes estimés sur chaque jeu simulé

Les résultats délivrés par cette étude de comparaison des partitions doubles sont, au premier abord, bons et plus spécifiquement encore dans le cadre de Gauss-LASSO stabilisé. Le $CARI$ est l'indice de comparaison de partitions doubles le plus sévère (comparativement à l'$Extended\ MI$ et le CE selon l'étude faite au chapitre précédent). Cependant, les valeurs du $CARI$, excepté pour la simulation s_3 sont toutes supérieures à 0.3 lorsque l'étape de sélection est effectuée par Gauss-LASSO stabilisé. Lorsque l'on sait que le $CARI$ prend des valeurs négatives lorsque le lien entre les partitions est très mauvais, ce résultat indique la stabilité de la procédure globale via une sélection réalisée par Gauss-LASSO stabilisé. Dans le cadre de Gauss-LASSO enrichi, les résultats de stabilité sont moins bons mais restent tout à fait corrects. Toutes les simulations, exceptées la s_8, fournissent des partitions doubles dont la valeur du $CARI$ avec la partition double réelle est supérieure à 0.2. Les valeurs du $CARI$ sont plus faibles que dans le cadre de Gauss-LASSO stabilisé mais mettent en valeur le fait que la procédure globale via sélection par Gauss-LASSO enrichi est plutôt stable également.

Pour appuyer la qualité de ces résultats, nous comparons les valeurs du $CARI$ ainsi obtenues à des valeurs du $CARI$ entre la partition double obtenue sur le jeu réel et des partitions doubles simulées aléatoirement. Nous avons ainsi simulé, dans le cadre de Gauss-LASSO stabilisé (respectivement Gauss-LASSO enrichi) cent partitions doubles aux paramètres "nombre de classe" $(H, L) = \left(\widehat{H}^{GLstab}, \widehat{L}^{GLstab}\right)$ (respectivement $(H, L) = \left(\widehat{H}^{GLenri}, \widehat{L}^{GLenri}\right)$) et "proportions des classes" $(\rho, \tau) = \left(\widehat{\rho}^{GLstab}, \widehat{\tau}^{GLstab}\right)$ (respectivement $(\rho, \tau) = \left(\widehat{\rho}^{GLenri}, \widehat{\tau}^{GLenri}\right)$) fixés comme étant ceux de la partition double $\left(\widehat{v}^{GLstab}, \widehat{w}^{GLstab}\right)$ (respectivement $\left(\widehat{v}^{GLenri}, \widehat{w}^{GLenri}\right)$) obtenue sur le jeu réel. Pour chacune de ces cent partitions doubles, la partition ligne correspond au tirage

aléatoire d'un p-échantillon de loi multinomiale $\mathcal{M}\big(\widehat{\rho}_1^{GLstab}, \ldots, \widehat{\rho}_{\widehat{H}^{GLstab}}^{GLstab}\big)$ (respectivement $\mathcal{M}\big(\widehat{\rho}_1^{GLenri}, \ldots, \widehat{\rho}_{\widehat{H}^{GLenri}}^{GLenri}\big)$) et la partition colonne à celui d'un p-échantillon de loi multinomiale $\mathcal{M}\big(\widehat{\tau}_1^{GLstab}, \ldots, \widehat{\tau}_{\widehat{L}^{GLstab}}^{GLstab}\big)$ (respectivement $\mathcal{M}\big(\widehat{\tau}_1^{GLenri}, \ldots, \widehat{\tau}_{\widehat{L}^{GLenri}}^{GLenri}\big)$).

Les valeurs du $CARI$ entre chacune de ces cent partitions doubles simulées aléatoirement et la partition double du jeu réél sont toutes comprises entre 0 et 0.0012, que ce soit dans le cadre de Gauss-LASSO stabilisé ou enrichi. Ces résultats sont éloquents : le critère d'accord entre partitions CARI ne fournit des valeurs notablement différentes de 0 que si les partitions ont des liaisons réelles. Les partitions doubles estimées sur le jeu de données réelles sont donc bien plus proches des partitions doubles formées par notre méthode de simulation globale que de partitions doubles tirées au hasard.

Simulation k	$CARI\big((^{s_k}\widehat{v}^{GLstab}, {}^{s_k}\widehat{w}^{GLstab}), (\widehat{v}^{GLstab}, \widehat{w}^{GLstab})\big)$
1	0.325
2	0.528
3	0.146
4	0.476
5	0.602
6	0.336
7	0.385
8	0.387
9	0.532
10	0.579

Table 4.2 – Comparaison de chaque partition double simulée et la réelle pour Gauss-LASSO stabilisé

Simulation k	$CARI\big((^{s_k}\widehat{v}^{GLenri}, {}^{s_k}\widehat{w}^{GLenri}), (\widehat{v}^{GLenri}, \widehat{w}^{GLenri})\big)$
1	0.324
2	0.452
3	0.256
4	0.316
5	0.404
6	0.203
7	0.275
8	0.142
9	0.408
10	0.507

Table 4.3 – Comparaison de chaque partition double simulée et la réelle pour Gauss-LASSO enrichi

Nous avons donc mis en valeur la stabilité de nos procédures globales via cette procédure de validation des résultats. Les résultats de classification que nous obtenons sur le jeu de données réel sont donc fiables. Un traitement biologique de ces résultats en aval de ce traitement statistique des données peut alors être effectué en toute confiance. En étudiant les caractéristiques biologiques des groupes de régressés et régresseurs obtenus, nous espérons pouvoir trouver de l'information biologique sur les facteurs de transcription.

Chapitre 5

Conclusion et perspectives

Le travail réalisé durant ma thèse est une contribution méthodologique pour la classification de variables dans un cadre de haute dimension statistique. Sous hypothèse d'une hiérarchie composée de groupes de contrôle au sein de l'ensemble des variables, elle permet de regrouper ces variables en classes de variables contrôleuses et en classes de variables contrôlées. Appliquée à un jeu de données transcriptomiques dans le cadre de la régulation des gènes d'un organisme, la méthode a pu être conçue et perfectionnée de façon à former des groupes de variables stables. Dans ce chapitre final, nous effectuons dans un premier temps un récapitulatif du travail réalisé en vue de la mise en place de cette méthode, puis dressons un bilan des résultats obtenus et des premiers enseignements que l'on peut en tirer. Nous émettrons notamment un avis préférentiel pour l'une des procédures de sélection de variables mises en places. Dans un second temps, nous émettrons des pistes de travaux à réaliser pour consolider les résultats statistiques obtenus et des propositions d'autres méthodologies fondées notamment sur la classification de graphes à partir de matrices d'adjacence non binaires. La validation statistique définitive de la méthodologie passera en grande partie par une validation biologique des groupes de facteurs de transcription obtenus.

1 Conclusion

1.1 Résumé de la démarche adoptée

Le premier chapitre a consisté à présenter le problème biologique qui nous a amené à nous poser cette question statistique méthodologique. Les biologistes essaient de détecter l'activité des facteurs de transcription à l'aide de protocoles expérimentaux permettant de les étudier un par un. Or de par leur faculté à travailler en module, nous avons supposé la présence d'une hiérarchie au sein de l'ensemble des facteurs de transcription structurée en groupes de gènes régulateurs et en groupes de gènes régulés. Avoir une vision globale de la structure régissant la régulation au sein des facteurs de transcription de la plante modèle *Arabidopsis thaliana* a été l'objectif biologique qui nous a poussé à élaborer et étudier notre question statistique. Pour traiter cette dernière, nous avons proposé une méthodologie scindée en une phase de construction d'un graphe orienté, dont les noeuds étaient les variables, représentant l'ensemble des liens de contrôle existant entre ces variables, puis d'une phase de classification des nœuds du graphe en les groupes attendus. Le gage de fiabilité de ce modèle global était sa capacité à produire des groupes de variables stables. La volonté d'obtenir des résultats stables nous a d'ailleurs servi de fil conducteur tout au long de la construction de la méthodologie.

Le second chapitre était le traitement de la première phase de la méthodologie. Pour espérer obtenir des groupes de variables stables en aval de l'ensemble de la méthodologie statistique, nous avons cherché à mettre en place, dans un premier temps, une procédure de sélection de variables visant à fournir, pour chacune des variables, un ensemble de variables pertinentes stables la contrôlant. Comme préconisé dans [45], nous avons mené ce problème à l'aide des modèles de régression linéaire pénalisée (LASSO) d'une variable sur les autres traités de manière indépendante. Pour chaque modèle de régression, la calibration de la pénalité LASSO via des critères tels que la validation croisée (ou même AIC) a résulté en des sélections trop peu sévères. Sa calibration par le critère BIC nous a fourni, quant à elle, des ensembles de variables de tailles plus petites. Néanmoins, une étude de la stabilité de ces ensembles réalisée à l'aide de jeux simulés à partir de notre modèle graphique a révélé l'instabilité de ces ensembles de variables. Pour pallier ce problème, nous avons eu recours au rééchantillonnage dans le but de disposer de plusieurs chemins de régularisation. Nous avons mis en place une méthode de rééchantillonnage inspirée de [46], [3] et [28], puis avons combiné les chemins de régularisation de deux manières différentes pour n'en former qu'un seul. Ce procédé a donné naissance à deux procédures de sélection, Gauss-LASSO stabilisé et Gauss-LASSO enrichi. Pour chacune de ces procédures, c'est l'heuristique de pente qui nous a fourni l'ensemble de variables du chemin de régularisation établi le plus stable, après comparaison avec plusieurs autres critères de sélection. Un graphe orienté

par procédure de sélection a ainsi été construit. Nous avons en outre remarqué que ces deux graphes étaient à environ 97% concordants.

Une fois ces graphes stables formés par la première phase de la méthodologie, nous les avons soumis, dans le troisième chapitre, aux modèles à blocs latents dont nous avons montré la pertinence en tant que modèles de classification non supervisée de graphes orientés. Nous avons présenté les méthodes de calibration des paramètres du modèle que nous avons employées, à savoir l'algorithme bayésien $V - Bayes$ d'estimation des paramètres du modèle pour un nombre de classes de variables contrôleuses et contrôlées fixées, le critère ICL pour calibrer ces nombres de classes, ainsi que la stratégie $Bi - KM1$ de parcours de couples de classes à étudier. Nous avons obtenu les classifications en blocs des variables attendues pour chacun des deux graphes traités. Pour évaluer la proximité de ces classifications, nous étions en quête d'un juge de paix. Les mesures de comparaison entre deux couples de partitions déjà existantes étaient soit inadaptées aux partitions présentant des nombres de classes différents, soit coûteuses, soit peu sévères pour pénaliser une mauvaise adéquation entre l'une des deux partitions simples. Nous avons créé, en réponse à ces défauts, le Coclustering Ajusted Rand Index. L'utilisation de ce critère pour comparer les couples de partitions émanant des deux graphes a souligné leurs différences malgré la proximité affichée de ces graphes en amont de l'étape de classification. Pour valider la méthodologie statistique s'appuyant sur une phase de sélection via Gauss-LASSO stabilisé et celle s'appuyant sur Gauss-LASSO enrichi, nous avons étudié la stabilité de leurs résultats à l'aide de jeux de données simulés.

Le quatrième chapitre a justement consisté en la méthode de formation de ces jeux de données simulés. La méthodologie de formation des groupes de variables à partir d'observations que nous avons mise en place est un mécanisme complexe composé d'une succession d'étapes dépendantes les unes des autres. Pour chacune des deux procédures de sélection, une méthode simulant une matrice de même structure que la matrice de classification en blocs obtenue en aval de la méthodologie statistique appliquée sur le jeu de données réél puis remontant étape par étape son mécanisme jusqu'à la formation d'un jeu de données simulé a été mise en place. L'échantillonnage de Gibbs y a notamment été employé pour reformer les équations de régression d'une variable sur les autres, aux paramètres estimés sur le jeu réel. Nous avons créé dix jeux de données simulés pour chacune des procédures de sélection et appliqué la méthodologie statistique sur chacun d'eux. La comparaison des couples de partitions obtenus sur chacun des jeux simulés et le jeu réel à l'aide du Coclustering Ajusted Rand Index a donné des résultats de stabilité satisfaisants permettant de valider la méthodologie statistique s'appuyant sur Gauss-LASSO stabilisé ou sur Gauss-LASSO enrichi.

1.2 Quelques enseignements des résultats obtenus

Un premier enseignement que l'on peut tirer de ce travail méthodologique est que même si Gauss-LASSO stabilisé et enrichi présentent tous deux des résultats de stabilité satisfaisants, la palme revient tout de même à Gauss-LASSO stabilisé. En effet, le chapitre 2 atteste du fait que cette procédure de sélection appliquée à des jeux de données simulés en adéquation avec notre modèle graphique orienté estime des ensembles de variables aux contenus plus proches des ensembles estimés sur le jeu réel que Gauss-LASSO enrichi. En outre, les résultats du chapitre 4 démontrent que la méthodologie statistique globale s'appuyant sur Gauss-LASSO stabilisé forme des groupes de variables contrôleuses et contrôlées encore plus stables que celle s'appuyant sur Gauss-LASSO enrichi. Nous avons donc tendance à préférer la version stabilisée de Gauss-LASSO plutôt que la version enrichie.

En outre, l'étude de l'impact de la forte réduction du nombre d'observations de n à $n/4$ sur les résultats de Gauss-LASSO stabilisé, a exposé les limites de l'utilisation du rééchantillonnage dans un cadre statistique moins avantageux que le nôtre. Tout d'abord, ce nouveau cadre contraint les procédures de rééchantillonnage utilisées à ne pas estimer des ensembles de variables de taille plus grande que 45 pour ne pas travailler en ultra haute dimension, selon [61]. Les ensembles de variables estimés sur le jeu de données entier étant de taille majoritement plus grande que cette limite, les résultats de sélection ne pourraient être validés sous un tel cadre statistique à p variables et $n/4$ observations. De plus, les résultats du chapitre 2 exposaient une répartition de l'ensemble des scores des variables explicatives potentielles aux variables typiques beaucoup plus homogène que dans le cadre à p variables et n observations. Ceci témoignait du mauvais comportement du rééchantillonnage dans ce cadre moins avantageux et du manque de stabilité des ensembles de variables qui y seraient sélectionnées.

Enfin, la comparaison entre les couples de partitions formées par la méthodologie statistique s'appuyant sur Gauss-LASSO stabilisé et celle s'appuyant sur Gauss-LASSO enrichi, effectuée au chapitre 3, a témoigné du manque de proximité entre les partitions des variables vues comme contrôlées, au vu du résultat de l'ARI. Les partitions des variables vues comme contrôleuses concordaient, quant à elles, beaucoup plus. Nous aurions ainsi tendance à accorder plus de confiance aux groupes de variables contrôleuses qu'aux groupes de variables contrôlées émanant de la méthodologie statistique. Nous conforterons ce sentiment par la prise en compte de la construction même du modèle global s'appuyant sur les régressions linéaires d'une variable sur les autres pour directement détecter les variables contrôleuses de chaque variable et non les variables contrôlées par chaque variable.

2 Perspectives

Dans cette thèse, nous avons construit des méthodologies stastiques à l'aide d'un modèle ne concordant pas tout à fait avec les données transcriptomiques que nous avons traitées, comme en témoigne l'étude préliminaire de la normalité des variables au chapitre 1. Il n'a pas été question dans notre travail de chercher à modifier le modèle de manière à miniser l'erreur d'approximation de ce dernier vis-à-vis du jeu de données réel. Il s'agissait plutôt d'estimer aux mieux des groupes de variables stables à partir de ce modèle. Parvenir à évaluer cette erreur d'approximation et faire évoluer le modèle en conséquence pourrait constituer un travail ultérieur à cette thèse visant à solidifier la méthodologie statistique que nous avons employée.

En outre, en conservant le modèle utilisé, il est déjà envisageable de mettre en place d'autres méthodologies de formation de groupes de variables contrôleuses et contrôlées notamment de par la variété des données surlesquelles peuvent être appliqué les modèles à blocs latents. Comme détaillé au chapitre 3, il existe des versions des modèles à blocs latents traitant des données de comptage ([26]) ou des données continues ([39]). Il est donc possible d'apporter de l'information supplémentaire aux matrices d'adjacence binaires que nous avons traitées par les modèles à blocs latents. Nous pourrions envisager, par exemple :

1. La construction de matrices d'adjacence trinaires possédant en position j, j' un coefficient valant :
   — 0 si les variables j et j' ne présentent pas lien de régulation entre elles.
   — -1 si la variable j' régule la variable j en l'inhibant.
   — 1 si la variable j' régule la variable j en l'activant.
   La détection de l'inhibation ou de l'activation des variables explicatives vis-à-vis de la variable expliquée peut se faire par le signe des coefficients de régression de chaque variable pertinente pour la régression de la variable expliquée.
2. La construction de matrices d'adjacence continues possédant en position j, j' un coefficient valant :
   — 0 si les variables j et j' ne présentent pas lien de régulation entre elles.
   — $\widehat{\theta}_{j,j'}$, le coefficient de régression de j' pour la régression de j si j' régule j.

Il serait intéressant de comparer les couples de partitions formés dans notre cadre d'un modèle à blocs latents pour données binaires avec ceux formés dans le cadre de données trinaires ou continues, à l'aide du Coclustering Ajusted Rand Index. Ceci permettrait d'évaluer si l'information ajoutée modifie fortement les groupes de variables établies en aval de la méthodologie statistique.

Il est donc possible d'améliorer notre méthodologie statistique de différentes manières. Cependant, la meilleure procédure de validation possible de cette méthodologie resterait la

validation biologique des groupes de facteurs de transcription co-régulateurs et co-régulés formés. Évaluer la pertinence biologique des groupes formés par le biais d'expériences réalisées à la paillasse est le moyen le plus fiable qui nous permettrait de nous conforter ou de nous détourner de l'approche statistique employée dans ma thèse. Cependant, à l'heure actuelle, de telles vérifications expérimentales sont impossibles. Une première étude sur l'enrichissement biologique des classes de facteurs de transcription fournies par la méthodologie s'appuyant sur notre procédure de sélection favorite, Gauss-LASSO stabilisé, a cependant pu être réalisée. Celle-ci a été effectuée par le biais d'analyses biologiques fondées sur des tests hypergéométriques permettant de savoir si la représentativité d'un groupe de facteurs de transcription d'*Arabidopsis thaliana* au niveau structural ou au niveau fonctionnel est différente de celle de l'ensemble global des facteurs de transcription de la plante. Nous avons testé cet enrichissement des groupes de facteurs de transcription sur la connaissance structurale regroupée dans la base de données TAIR ([58]) pour savoir si certains groupes de facteurs de transcription étaient spécifiquement enrichis en certaines des 79 familles structurales. Des bases de données telles que GO Slim and subset (www.geneontology.org/page/go-slim-and-subset-guide) ou encore la base DAVID (Database for Annotation, Visualization and Integrated Discovery, [30], http ://david.abcc.ncifcrf.gov) constituée de cinq niveaux d'ontologies et plus précise que GO Slim, permettent de tester l'enrichissement de groupes de gènes selon leurs fonctions. C'est sur cette dernière base de données que nous sommes appuyés pour l'analyse de l'enrichissement fonctionnel.

Tout d'abord, les résultats semblent attester d'enrichissements plus importants dans les groupes de facteurs de transcription co-régulateurs formés que dans les groupes de facteurs de transcription co-régulés. Ce constat va dans le sens de notre remarque précédente concernant l'accord d'un crédit plus important aux groupes de variables contrôleuses que contrôlées. De plus, quelques groupes de facteurs de transcription seraient enrichis en une famille structurale au vu de cette analyse. Une mention spéciale revient au groupe de facteurs de transcription co-régulateur numéro 22 qui est enrichi en trois familles structurales. Au niveau des enrichissements fonctionnels, les groupes de facteurs de transcription numéros 3 et 22 sont enrichis en David 5 de par leur lien avec les organites, ce qui correspond au niveau d'ontologie maximum de la base de données. Le groupe de facteurs de transcription co-régulé numéro 25 présente quant à lui un enrichissement David 4 de par sa réponse au stimulus de carbohydrate. Des liens entre groupes de gènes de co-régulation et groupes de gènes aux fonctions proches existeraient donc selon ces premières analyses. Ces résultats sont encourageants et renforcent l'intérêt d'une validation des groupes de facteurs de transcription formés par le biais de moyens expérimentaux.

Bibliographie

[1] Hirotogu Akaike. Information theory as an extension of the maximum likelihood principle. In *2nd International Symposium on Information Theory*, pages 267–281. Springer, 1973.

[2] Julie Aubert, Trung Ha, and Tristan MaryHuard. Modele à blocs latents pour l'analyse de données métagénomiques. *46ème journées de Statistiques de la SFdS*, 2014.

[3] Francis R Bach. Bolasso : model consistent lasso estimation through the bootstrap. In *Proceedings of the 25th international conference on Machine learning*, pages 33–40. ACM, 2008.

[4] Yannick Baraud, Christophe Giraud, Sylvie Huet, et al. Estimator selection in the gaussian setting. In *Annales de l'Institut Henri Poincaré, Probabilités et Statistiques*, volume 50, pages 1092–1119. Institut Henri Poincaré, 2014.

[5] Jean-Patrick Baudry and Gilles Celeux. Em for mixtures. *Statistics and Computing*, 25(4) :713–726, 2015.

[6] Christophe Biernacki, Gilles Celeux, and Gérard Govaert. Assessing a mixture model for clustering with the integrated completed likelihood. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(7) :719–725, 2000.

[7] Christophe Biernacki and Julien Jacques. Model-based clustering of multivariate ordinal data relying on a stochastic binary search algorithm. *Statistics and Computing*, pages 1–15, 2015.

[8] Lucien Birgé and Pascal Massart. Gaussian model selection. *Journal of the European Mathematical Society*, 3(3) :203–268, 2001.

[9] Lucien Birgé and Pascal Massart. Minimal penalties for gaussian model selection. *Probability theory and related fields*, 138(1-2) :33–73, 2007.

[10] Vincent Brault. *Éstimation et sélection de modèle pour le modèle des blocs latents*. PhD thesis, Université Paris Sud 11, 2014.

[11] Gabriel Castrillo, Franziska Turck, Magalie Leveugle, Alain Lecharny, Pilar Carbonero, George Coupland, Javier Paz-Ares, and Luis Oñate-Sánchez. Speeding cis-trans regulation discovery by phylogenomic analyses coupled with screenings of an arrayed library of arabidopsis transcription factors. *PloS one*, 6(6) :e21524, 2011.

[12] M. Charrad, Y. Lechevallier, G. Saporta, and M. Ben Ahmed. Détermination du nombre de classes dans les méthodes de bipartitionnement. In *17ème Rencontres de la Société Francophone de Classification*, pages 119–122, Saint-Denis de la Réunion, June 2010.

[13] Yongjun Chu and David R Corey. Rna sequencing : platform selection, experimental design, and data interpretation. *Nucleic acid therapeutics*, 22(4) :271–274, 2012.

[14] Arthur P Dempster, Nan M Laird, Donald B Rubin, et al. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal statistical Society*, 39(1) :1–38, 1977.

[15] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM, 2003.

[16] Adrian Dobra, Chris Hans, Beatrix Jones, Joseph R Nevins, Guang Yao, and Mike West. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, 90(1) :196–212, 2004.

[17] Bradley Efron, Trevor Hastie, Iain Johnstone, Robert Tibshirani, et al. Least angle regression. *The Annals of statistics*, 32(2) :407–499, 2004.

[18] Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[19] Edward B Fowlkes and Colin L Mallows. A method for comparing two hierarchical clusterings. *Journal of the American statistical association*, 78(383) :553–569, 1983.

[20] Ove Frank and Frank Harary. Cluster inference by using transitivity indices in empirical graphs. *Journal of the American Statistical Association*, 77(380) :835–840, 1982.

[21] Séverine Gagnot, Jean-Philippe Tamby, Marie-Laure Martin-Magniette, Frédérique Bitton, Ludivine Taconnat, Sandrine Balzergue, Sébastien Aubourg, Jean-Pierre Renou, Alain Lecharny, and Veronique Brunaud. Catdb : a public access to arabidopsis transcriptome data from the urgv-catma platform. *Nucleic Acids Research*, 36(suppl_1) :D986–D990, 2007.

[22] Xiangchao Gan, Oliver Stegle, Jonas Behr, Joshua G Steffen, Philipp Drewe, Katie L Hildebrand, Rune Lyngsoe, Sebastian J Schultheiss, Edward J Osborne, Vipin T Sreedharan, et al. Multiple reference genomes and transcriptomes for arabidopsis thaliana. *Nature*, 477(7365) :419, 2011.

[23] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6) :721–741, 1984.

[24] G. Govaert and M. Nadif. Clustering of contingency table and mixture model. *European Journal of Operational Research*, 183 :1055–1066, 2007.

[25] G. Govaert and M. Nadif. Block clustering with Bernoulli mixture models : Comparison of different approaches. *Computational Statistics and Data Analysis*, 52 :3233–3245, 2008.

[26] Gérard Govaert and Mohamed Nadif. *Co-Clustering*. ISTE Ltd and John Wiley & Sons, Inc, 2013.

[27] John A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, 99th edition, 1975.

[28] Anne-Claire Haury, Fantine Mordelet, Paola Vera-Licona, and Jean-Philippe Vert. Tigress : trustful inference of gene regulation using stability selection. *BMC systems biology*, 6(1) :145, 2012.

[29] Paul W Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels : First steps. *Social networks*, 5(2) :109–137, 1983.

[30] Da Wei Huang, Brad T Sherman, Qina Tan, Jack R Collins, W Gregory Alvord, Jean Roayaei, Robert Stephens, Michael W Baseler, H Clifford Lane, and Richard A Lempicki. The david gene functional classification tool : a novel biological module-centric algorithm to functionally analyze large gene lists. *Genome biology*, 8(9) :R183, 2007.

[31] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification*, 2(1) :193–218, 1985.

[32] M. Jagalur, C. Pal, E. Learned-Miller, R. T. Zoeller, and D. Kulp. Analyzing in situ gene expression in the mouse brain with image registration, feature extraction and block clustering. *BMC Bioinformatics*, 8(Suppl 10) :S5, 2007.

[33] C. Keribin. Consistent estimation of the order of mixture models. *Sankhya Series A*, 62 :49–66, 2000.

[34] C. Keribin, G. Govaert, and G. Celeux. Estimation d'un modèle à blocs latent par l'algorithme SEM. In *42e Journées de Statistique, SFdS*, Marseille, France, May 2010.

[35] Christine Keribin, Vincent Brault, Gilles Celeux, and Gérard Govaert. Estimation and selection for the latent block model on categorical data. *Statistics and Computing*, 25(6) :1201–1216, 2015.

[36] Hirohisa Kishino and Peter J Waddell. Correspondence analysis of genes and tissue types and finding genetic links from microarray data. *Genome Informatics*, 11 :83–95, 2000.

[37] Tim Lenoir and Eric Giannella. The emergence and diffusion of dna microarray technology. *Journal of biomedical discovery and collaboration*, 1(1) :11, 2006.

[38] Hubert W Lilliefors. On the kolmogorov-smirnov test for normality with mean and variance unknown. *Journal of the American statistical Association*, 62(318) :399–402, 1967.

[39] Aurore Lomet. *Sélection de modèle pour la classification croisée de données continues.* Thèse, Université de Technologie de Compiègne, Décembre 2012.

[40] Colin L Mallows. Some comments on cp. *Technometrics*, 42(1) :87–94, 2000.

[41] M. Mariadassou, S. Robin, and C. Vacher. Uncovering latent structure in valued graphs : a variational approach. *The Annals of Applied Statistics*, 4(2) :715–742, 2010.

[42] Catherine Matias and Stéphane Robin. Modeling heterogeneity in random graphs through latent space models : a selective review. *ESAIM : Proceedings and Surveys*, 47 :55–74, 2014.

[43] Cathy Maugis. *Sélection de variables pour la classification non supervisée par mélanges gaussiens. Application à l'étude de données transcriptomes.* PhD thesis, Université Paris Sud-Paris XI, 2008.

[44] Marina Meilă. Comparing clusterings—an information based distance. *Journal of Multivariate Analysis*, 98(5) :873–895, 2007.

[45] Nicolai Meinshausen and Peter Bühlmann. High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, pages 1436–1462, 2006.

[46] Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 72(4) :417–473, 2010.

[47] Nobutaka Mitsuda and Masaru Ohme-Takagi. Functional analysis of transcription factors in arabidopsis. *Plant and Cell Physiology*, 50(7) :1232–1248, 2009.

[48] Ryan D Morin, Matthew Bainbridge, Anthony Fejes, Martin Hirst, Martin Krzywinski, Trevor J Pugh, Helen McDonald, Richard Varhol, Steven JM Jones, and Marco A Marra. Profiling the hela s3 transcriptome using randomly primed cdna and massively parallel short-read sequencing. *Biotechniques*, 45(1) :81, 2008.

[49] H Parkinson, Misha Kapushesky, Mohammadreza Shojatalab, Niran Abeygunawardena, Richard Coulson, Anna Farne, Ele Holloway, N Kolesnykov, P Lilja, Margus Lukk, et al. Arrayexpress—a public database of microarray experiments and gene expression profiles. *Nucleic acids research*, 35(suppl_1) :D747–D750, 2006.

[50] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336) :846–850, 1971.

[51] Payam Refaeilzadeh, Lei Tang, and Huan Liu. Cross-validation. In *Encyclopedia of database systems*, pages 532–538. Springer, 2009.

[52] Valérie Robert. *Classification croisée pour l'analyse de bases de données de grandes dimensions de pharmacovigilance*. PhD thesis, Paris Saclay, 2017.

[53] Valérie Robert and Yann Vasseur. Comparing high dimensional partitions, with the coclustering adjusted rand index. *arXiv preprint arXiv :1705.06760*, 2017.

[54] Jorge Santos and Mark Embrechts. On the use of the adjusted rand index as a metric for evaluating supervised classification. *Artificial neural networks*, pages 175–184, 2009.

[55] Gilbert Saporta and Genane Youness. Comparing two partitions : Some proposals and experiments. In *Compstat*, pages 243–248. Springer, 2002.

[56] Gideon Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2) :461–464, 1978.

[57] H. Shan and A. Banerjee. Bayesian co-clustering. In *Eighth IEEE International Conference on Data Mining, 2008. ICDM'08*, pages 530–539, 2008.

[58] David Swarbreck, Christopher Wilks, Philippe Lamesch, Tanya Z Berardini, Margarita Garcia-Hernandez, Hartmut Foerster, Donghui Li, Tom Meyer, Robert Muller,

Larry Ploetz, et al. The arabidopsis information resource (tair) : gene structure and function annotation. *Nucleic acids research*, 36(suppl_1) :D1009–D1014, 2007.

[59] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[60] Sara van de Geer, Peter Bühlmann, Shuheng Zhou, et al. The adaptive and the thresholded lasso for potentially misspecified models (and a lower bound for the lasso). *Electronic Journal of Statistics*, 5 :688–749, 2011.

[61] Nicolas Verzelen et al. Minimax risks for sparse regressions : Ultra-high dimensional phenomenons. *Electronic Journal of Statistics*, 6 :38–90, 2012.

[62] Nguyen Xuan Vinh, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison : Variants, properties, normalization and correction for chance. *Journal of Machine Learning Research*, 11(Oct) :2837–2854, 2010.

[63] Lewis Wolpert. Positional information and pattern formation. *Current topics in developmental biology*, 6 :183–224, 1971.

[64] Xintao Wu and Yong Ye. Exploring gene causal interactions using an enhanced constraint-based method. *Pattern Recognition*, 39(12) :2439–2449, 2006.

[65] J. Wyse, P. Latouche, and N. Friel. Inferring structure in bipartite networks using the latent block model and exact ICL . *Network Science*, 2016.

[66] Yee Hwa Yang and Terry Speed. Design issues for cdna microarray experiments. *Nature reviews. Genetics*, 3(8) :579, 2002.

[67] Genane Youness and Gilbert Saporta. Une méthodologie pour la comparaison de partitions. *Revue de statistique appliquée*, 52(1) :97–120, 2004.

[68] Peng Zhao and Bin Yu. On model selection consistency of lasso. *Journal of Machine learning research*, 7(Nov) :2541–2563, 2006.

[69] Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American statistical association*, 101(476) :1418–1429, 2006.