Développement d'outils statistiques pour l'analyse de données transcriptomiques par les réseaux de co-expression de gènes

Anne-Claire Brunet

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Université Toulouse 3 Paul Sabatier (UT3 Paul Sabatier)*

Présentée et soutenue le *17 juin 2016* **par :**

ANNE-CLAIRE BRUNET

Développement d'outils statistiques pour l'analyse de données transcriptomiques par les réseaux de co-expression de gènes

---

JURY

JACQUES AMAR	CHU Rangueil	Directeur de thèse
JEAN-MARC AZAÏS	Université Paul Sabatier	Directeur de thèse
JEAN-CLAUDE FORT	Université Paris Descartes	Examinateur
MARC LAVIELLE	INRIA Saclay	Rapporteur
JEAN-MICHEL LOUBES	Université Paul Sabatier	Directeur de thèse
JEAN-MICHEL POGGI	Université Paris Descartes	Rapporteur
JOSÉ RAFAEL LEÓN RAMOS	Université centrale du Vénézuela	Examinateur

---

École doctorale et spécialité :
MITT : Domaine Mathématiques : Mathématiques appliquées
Unité de Recherche :
Institut de Mathématiques de Toulouse et VAIOMER
Directeur(s) de Thèse :
Jean-Marc Azaïs, *Jean-Michel Loubes* et *Jacques Amar*
Rapporteurs :
Marc Lavielle et *Jean-Michel Poggi*

Remerciements

Il n'aurait pas été possible de vivre cette aventure d'apprenti chercheur sans soutien. Et du soutien, j'en ai eu beaucoup ! Je suis très heureuse de pouvoir clôturer cette belle aventure par des pensées pleines de reconnaissance pour tous ceux qui y ont contribué.

Jean-Michel Loubes et Jean-Marc Azaïs, mes directeurs de thèse à l'IMT, je vous remercie de m'avoir fait bénéficier, grâce à votre grande complémentarité, de plus d'un double encadrement : d'un encadrement au carré, soit JM^2. Jean-Michel, sans toi je n'aurai pas eu la chance de vivre cette expérience. Laborieuse certe, mais tellement enrichissante. Merci de m'avoir fait confiance ! J'ai toujours pu te parler très librement et tu as su te montrer réconfortant et encourageant. Jean-Marc, je te remercie de m'avoir poussée à être toujours plus rigoureuse en ne laissant jamais de place aux explications approximatives. J'ai beaucoup apprécié ta façon subtile et pleine d'humour de me conseiller.

Je remercie chaleureusement Jacques Amar, mon directeur de thèse chez Vaiomer. Grâce à vous, le vocabulaire de la biologie m'est devenu beaucoup plus familier. Je vous suis reconnaissante d'avoir fait preuve d'une grande patience à mon égard et d'avoir montré un grand intérêt pour mon travail. Je remercie également Michael Courtney d'avoir oeuvré pour mon intégration chez Vaiomer. Je vous remercie de m'avoir fait confiance et de m'avoir aidée à valoriser les résultats de mes recherches. Rémy Burcelin, je vous remercie d'avoir, par votre enthousiasme débordant, su attiser ma curiosité scientifique et me donner l'envie d'aller plus loin dans mes recherches.

Jean-Michel Poggi et Marc Lavielle, je vous remercie d'avoir pris le temps de lire ma thèse en acceptant la fonction de rapporteur. Merci également à Jean-Claude Fort et à Jose Rafael Leòn Ramos d'avoir accepté de faire partie de mon jury.

Une chance d'avoir pu évoluer dans deux univers très différents. À commencer par celui du labo de maths, merci à tous ceux avec qui j'ai eu l'occasion d'échanger.

Je remercie Hélène, Magalie et Claire, avec qui j'ai partagé mon bureau les premières années. Un bureau à l'ambiance studieuse (merci Hélène et Magalie) et parfois à l'ambiance très vivante, voire même explosive (merci Claire). Claire, je te remercie d'avoir continué à m'encourager par écrit une fois partie. Je remercie la seconde fournée très masculine qui est venue vous remplacer. Merci Brendan, Sylvain et Antoine pour les éclats de rire que vous avez pu déclencher. Merci aussi à Clément et Jonathan.

Mélanie, en ayant certains sujets de réflexion en commun, nous avons eu de nombreuses occasions d'en discuter et cela m'a été très bénéfique. Je t'en remercie. Malika, merci de m'avoir donné quelques tuyaux pour la mise en page de mon manuscrit. Stéphane merci de m'avoir

ramené, par ta sympathie, en peu de soleil de mon gard natal. Tibo, Claire et Adil merci pour les petits moments partagés pendant les pauses cigarette.

Merci aussi à Fabrice Gamboa et à Laurent Risser pour leur aide ponctuelle. Je remercie Agnès Requis et Martine Labruyère du pôle administratif, d'avoir été très compréhensives et arrangeantes quand il a pu y avoir quelques petits ratés en fin de parcours.

J'en viens maintenant à remercier toute l'équipe des biologistes. Merci à toute l'équipe de Rémy à l'INSERM. Vous m'avez donné l'occasion de commencer à me familiariser avec la biologie. Et cela n'est pas sans compter sur l'aide de Chantal, François, Lucile et Céline. Merci ! Jason, je te remercie de m'avoir facilité la tâche pour les questions d'ordre informatique. Merci aussi à toute l'équipe Vaiomer. À ceux avec qui j'ai partagé mon bureau : merci Benjamin de n'avoir jamais hésité à te lancer au tableau pour m'enseigner les rudiments de la bio ; merci Florence d'avoir essayé de faire le lien entre mes préoccupations et celles des biologistes ; merci aussi à Karine, Séverine et Jeffrey. Je remercie Jean-Louis, Jérome, Gaëlle, Sandrine, Carine, Bérengère et Céline, en compagnie de qui il a toujours été très agréable de partager une pause café.

Pour mon entourage, cette aventure a pu paraître un peu longue... Soyez-en assurés, j'en ai bel et bien fini avec les études ! Me revoila pleine de fougue !

Mes copinettes, Daïa et Christelle, je ne pourrais jamais trouver assez de mots pour vous dire merci. Le temps passant, ce trio infernal devient toujours plus indestructible et salvateur. Mes copinettes, merci, merci, merci... ! Merci également à vos bien-aimés : à Sojj qui ne cesse de me régaler avec ses numéros de clown empathique, et à Clément pour les très bons moments que nous avons partagés. Sophie, je te remercie d'avoir été très présente et à mon écoute ces derniers mois. Chacune à notre façon nous avons repris du poil de la bête (sans mauvais jeu de mots). Tu y es pour beaucoup ma Zowfi ! Merci Floxou de m'avoir permis de faire le plein de bonne humeur en m'entraînant dans de folles escapades nocturnes. Sab et Raf, je vous remercie de vous être toujours montrés prévenants et de bons conseils. Raf, je ne doute pas que tu sois un danseur hors pair même si certains de tes portés ont pu être destructeurs. Il est grand temps que nous nous remettions au travail... avec Bribri aux platines bien entendu. Merci Flo de m'avoir témoigné beaucoup d'attention et de m'avoir apporté ton soutien presque jusqu'à la fin. Merci également à Ginette et toute ta famille de m'avoir acceptée parmi eux toutes ces longues années. J'en garderai de très bons souvenirs. Mari l'aventurière, merci pour ton affection et les nombreux fous rires que nous avons partagés. Que ce soit au bout du monde, dans un bar à Ginjinha, au coeur du massif pyrénéen... pleins de souvenirs inoubliables qui alimentent mes rêves d'évasion. Vic, merci pour le covoiturage. En bonne compagnie, les trajets en direction de Labège étaient bien plus réjouissants. Elise, je te remercie pour tes encouragements et mes papilles n'oublieront pas les petites douceurs sucrées que tu as pu m'offrir pendant la rédaction de ma thèse. Héléa, merci pour tous ces petits moments simples et toujours très joyeux, autours d'un jeu de société, d'un bon repas, ou bien à crapahuter, à festoyer... JB, je te remercie pour ton énergie et ta bonne humeur très communicatives, ainsi que pour tes encouragements quand je me suis lancée dans des projets d'ébénisterie. Fab, mon vieux poto, merci de n'avoir cesser de m'encanailler depuis tout ce temps. Une mamie veille sur nous... Clém alias tonton, merci pour tes encouragements et tous ces souvenirs de franche rigolade. Quand la classe laisse place au peura, vas-y tonton ! Pierro, merci d'avoir été l'initiateur de nombreuses discussions, parfois enflammées, sur tout type de sujets, et de m'avoir fait connaître quelques-uns de tes coups de coeurs jazz. J'ai appris

qu'on peut rire de tout mais pas avec toi quand tu interprètes du Brassens. Merci Ketsia de me rappeler que la créativité rend la vie tellement pétillante. Merci JJ d'être un ami qui inspire par son courage et sa tenacité. Toujours en pole position ! Shen, je te remercie de m'avoir témoigné ton amitié chaque fois que tu en as eu l'occasion. Eloï, je garde d'excellents souvenirs de nos discussions même si tu prends toujours un malin plaisir à faire le sceptique. Je te remercie de m'avoir fait découvrir le clavicorde et la ville de Montpellier. Kurt où es-tu encore passé ? Je te remercie pour l'ensemble des moments très festifs que nous avons partagés et de m'avoir donné quelques cours de pilotage de drone. Marianne, je te remercie d'avoir voulu me transmettre ta passion de la capoeira et des danses brésiliennes. Du soleil dans les coeurs ! Thomas et Pauline, les rares moments passés ensembles ces dernières années ont toujours été très enthousiasmants. Je vous remercie pour toutes vos petites attentions. Merci Gaël d'avoir grandement participer à rendre mon premier bivouac en montagne totalement inoubliable. Merci Gui de m'avoir fait découvrir la spéléo dans ta grotte préférée qui regorge de paysages à couper le souffle. Merci Nico le petit caribou pour ta visite qui a fait remonter tant de bons souvenirs de la grande époque Ramonville. Merci Fella d'avoir toujours été prévenante et de très bonne compagnie durant nos séjours sur les côtes méditéranéennes. Thomas, Stephane et Laurent, je vous remercie pour ces week-end de folie qui ont pu nous amener sur la route des vins de bourgogne, dans la maison des ours, à nager avec un dragon, à la découverte d'un manège... Pure fantaisie ! Pure féerie ! Le gang des loubards, merci de m'avoir accueillie si chaleureusement dans votre fine équipe. Merci Franck de m'avoir fait profiter de tes talents de cuisto et de chercher à m'aider pour mon avenir pro. Merci Willy de me faire tant rire et d'être toujours partant pour boire la démarrante. Merci Pierro, mon champion, pour toutes ces joutes verbales et parties de ping pong animées. Merci Zaza pour tes mots gentils et réconfortants. Merci Hassan de m'avoir prété une oreille attentive de grand frère. Bernard, je te remercie pour tous les moments exquis que nous avons partagés et pour tous tes petits messages affectueux. Paula, merci d'avoir entre deux coupes de cheveux, pris le temps de m'aider à faire des traductions en anglais. Merci aussi à Margot, Roico, Lila, Torkich, Max, Mathieu et Perrine, Antoine, Gauthier, Marie, Laure, Nico, Adrien, Maëva, Agathe, Mathieu et Fanny, Gaëtan... Je me rattraperai de vive voix pour tous ceux que j'oublie. Merci à mes amis de très longue date que j'ai trop peu vu ces dernière années, Chloé, Emilie et Max. Sylvain, merci de m'avoir rappelé qu'à tout instant il est possible de faire une rencontre inattendue et pleine de tendresse.

Ma petite famille tant aimée, mes parents, mes frères, merci de m'avoir donné bien plus que tout ce que j'aurais pu espérer. Je ne vais pas me lancer dans une énumération sans fin. Vos valeurs et toutes vos différences m'ont toujours inspirées pour rêver d'une vie sur mesure dans laquelle les plus discrets et intimes engagements deviennent essentielles. Pap, mam, François, Philippe, je vous remercie infiniment et vous dédie cette thèse (même si je ne vous conseillerais pas de la lire).

Table des matières

Table des figures

Liste des tableaux

Introduction

Le contexte biologique

L'étude des mécanismes d'une maladie au niveau cellulaire L'objectif de cette thèse était de proposer de nouveaux outils statistiques pour l'identification de biomarqueurs génétiques d'une maladie et pour l'étude des mécanismes génétiques complexes sous-tendant la maladie. Les marqueurs biologiques ou biomarqueurs d'une pathologie sont des caratéristiques biologiques mesurables qui révèlent la pathologie. En quelque sorte, un biomarqueur est une signature biologique de la maladie qui permet de poser un diagnostic. Notre organisme est constitué de différents systèmes biologiques à l'intérieur desquels de multiples entités sont en interaction : ensembles d'organes, de tissus, de cellules, de molécules, de micro-organismes... Avec la découverte de l'ADN et l'apparition des biotechnologies à la fin du 20^{e} siècle, il est désormais possible d'étudier les mécanismes des maladies à l'intérieur des cellules (protéomique, génomique...), ce qui va permettre de proposer des biomarqueurs pour le diagnostic des maladies à des stades précoces, avant même l'apparition de troubles physiologiques. Ces avancées ouvrent de nouvelles perspectives de progrès pour la prévention d'une maladie et pour en améliorer son traitement.

Le génome Le génome est l'ensemble du matériel génétique d'un individu et l'ensemble des gènes, appelé génotype, en est une partie. Les gènes ne constituent qu'une entité du système complexe des organismes, mais ils jouent un rôle crucial dans leur fonctionnement, en permettant la production de protéines qui assurent une multitude de fonctions au sein des cellules et des tissus. Les études à l'échelle du génome (la génomique) ont permis de faire des avancées majeures en biologie. Dans cette thèse, nous allons précisément nous intéresser à cette entité et proposer des solutions statistiques pour étudier les variations d'expression des gènes en lien avec une pathologie donnée.

Du gène à la protéine Chez les eucaryotes (organismes caractérisés par la présence d'un noyau dans ses cellules), un gène correspond à un segment d'ADN placé à un endroit précis sur un chromosome situé dans le noyau des cellules. Il contrôle un caractère (ou phénotype) et constitue l'unité de base de l'hérédité, car sa localisation sur un chromosome reste identique d'une génération à la suivante. L'être humain possède environ 25 000 gènes répartis sur les 23 paires de chromosomes. L'ADN qui porte les gènes est une molécule formée de deux brins complémentaires en forme de double hélices qui sont composés d'une succesion de nucléotides. Un nucléotide peut être une adénine (A), une cytosine (C), une guanine (G) ou une thymine (T). C'est cette suite ou séquence de nucléotides qui détermine l'information. Sur la double hélice, les brins d'ADN sont complémentaires car un nucléotide s'apparie toujours avec le même autre nucléotide. Il existe ainsi quatre combinaisons distinctes : A-T, T-A, C-G, G-C. Lorsqu'un gène s'exprime ou s'active

il code pour une certaine protéine nécessaire à la vie cellulaire. Toutes les cellules possèdent les mêmes gènes dans leur noyau, mais ils ne sont pas tous actifs selon la cellule : une cellule du foie n'a pas les mêmes besoins qu'une cellule du coeur. Un gène s'exprime et permet la synthèse d'une protéine par l'intermédiaire d'une autre molécule appelée ARNm (ARN messager). La structure de l'ARNm est proche de celle de l'ADN, composée de nucléotide, aux différences près, que l'ARNm est constitué d'un unique brin et que le nucléotide de la thymine est remplacé par celui de l'uracile (U). Lorsqu'une cellule a besoin d'une protéine, elle crée, dans un premier temps, un ARNm qui est une copie du gène codant pour la protéine (phase de la transcription), puis dans un deuxième temps, lorsque l'ARNm est sorti du noyau, son message est interprété et la protéine synthétisée (phase de traduction). Nous avons représenté ces différentes étapes sur la Figure 1.

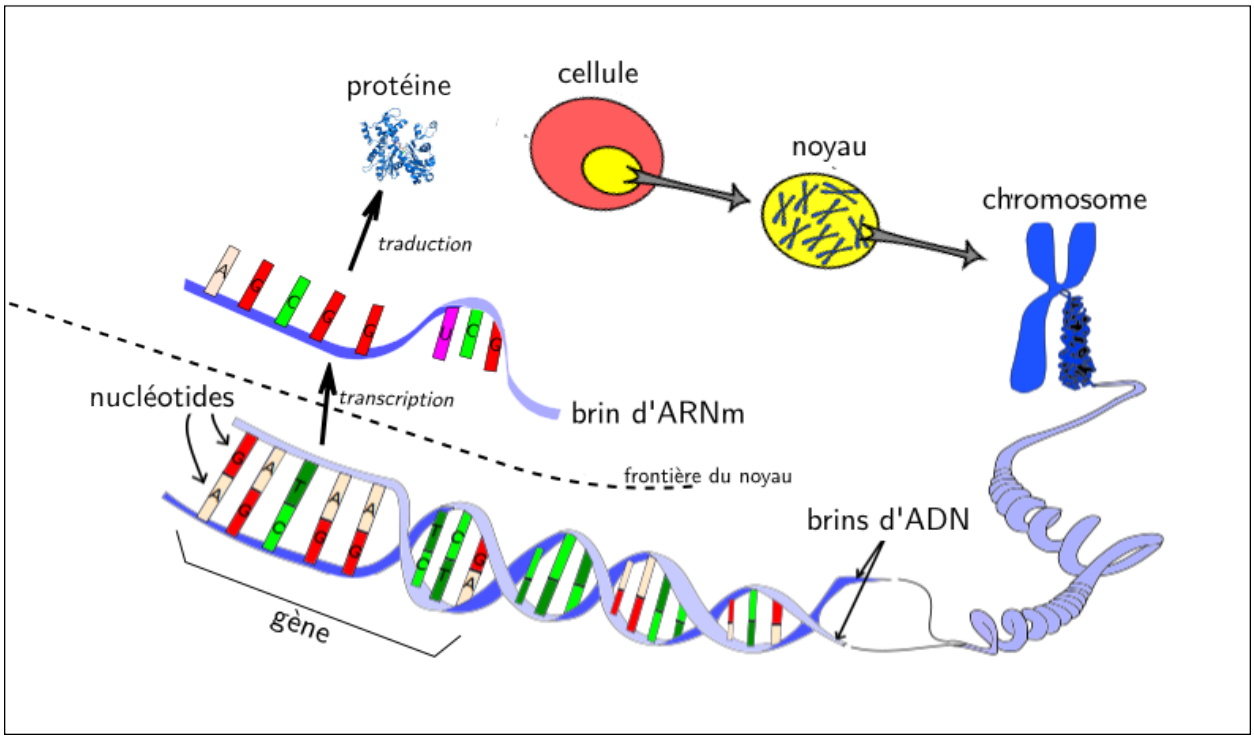


FIGURE 1 – Du gène à la protéine.

Le transcriptome Le transcriptome est l'ensemble des ARN (les messagers en particulier) produits par transcription de l'ADN. La caractérisation et la quantification du transcriptome sur une cellule donnée permet d'identifier l'activité des gènes, ou autrement dit, leur niveau d'expression. Plus un gène est exprimé et plus il crée des ARNm de façon à augmenter la production de la protéine pour laquelle il code.

Les données transcriptomiques

La technique la plus largement utilisée pour mesurer la concentration des différents ARNm est la technique des puces à ADN. Elle offre de larges possiblités du fait de son faible coût et de sa capacité à mesurer simultanément la concentration de milliers d'ARN.

Une puce à ADN est une lame solide sur laquelle sont rangés des fragments d'ADN mono-brin appelés « sondes ». Ces sondes sont choisies et correspondent à des gènes précis. Un emplacement sur la puce appelé spot est reservé à chaque sonde correspondant à un gène donné. A l'intérieur d'un spot, la même sonde (même gène) est présente des milliers de fois. L'ARN total, extrait de la cellule que l'on souhaite étudier, est transformé en ADN complémentaire (par la technique de la rétrotranscription), marqué par un colorant puis déposé sur la puce. L'ADN présent sur la puce va chercher spontanément à reformer sa structure en double hélice (rétrotranscription de l'ARN) et s'apparayer à l'ADN complémentaire quand il le peut. Le niveau de fluorescence des spots est ensuite analysé à l'aide d'un scanner à très haute résolution et c'est ce niveau qui détermine l'expression du gène : plus un spot est fluorescent et plus le nombre de sonde à l'intérieur du spot qui se sont apparayées à l'ADN déposé est important, ce qui reflète la présence au départ d'une plus grande quantité d'ARNm codant pour le gène (gène plus exprimé). À noter que les transcrits ne permettent pas toujours la production d'une protéine. L'information sur la concentration en transcrits ne peut pas directement être mise en parallèle avec celle de la concentration en protéine mais elle en constitue quand même un bon indicateur.

Les données transcriptomiques récoltées par la technologie des puces à ADN caractérisent l'expression de plusieurs dizaines de milliers de gènes sur un tissu donné (le foie, le tissu adipeux, le colon...) et nécessitent des prétraitement bioinformatiques spécifiques que nous ne détaillerons pas : nettoyage, correction de biais, normalisation... Les données « propres » regroupent les observations du logarithme binaire de l'expression (ou quantité d'ARN) de milliers de gènes sur un échantillon donné.

Analyses statistiques du transcriptome

Les objectifs visés par l'étude des données transcriptomiques peuvent être variés : comprendre les interactions et régulations entre les gènes, identifier des biomarqueurs, des cibles thérapeutiques... Les perspectives de recherche en biologie sont immenses mais dépendent en grande partie des solutions proposées sur le plan statistique pour extraire l'information dans un très gros volume de données (big data). Généralement, l'échantillon n'est composé que de quelques dizaines d'individus, et face au très grand nombre de gènes, il devient impossible d'utiliser les outils statistiques classiques (problème de la grande dimension).

Chapitre 1

L'apprentissage supervisé classique : une décomposition du système

Dans ce premier chapitre, nous allons présenter quelques-unes des méthodes pouvant être utilisées pour l'identification de gènes biomarqueurs d'une pathologie. Sur le plan statistique, cette problématique s'inscrit dans le contexte de l'apprentissage supervisé. Il s'agit en grande dimension, de sélectionner un petit nombre de variables (les gènes) qui sont les plus pertinentes pour expliquer ou prévoir une variable d'intérêt (la pathologie). Nous nous sommes intéressés à trois approches très différentes : les tests de permutation (méthode « filtre »), les forêts aléatoires (méthode algorithmique) et les méthodes de régression pénalisées. Nous envisagerons le cas d'une variable à expliquer quantitative (glycémie par exemple) et celui d'une variable à expliquer binaire (malade ou non). Ces méthodes d'apprentissage supervisé classiques procèdent à une décomposition du système à l'intérieur duquel les gènes interagissent en recherchant une somme d'effets individuels pour expliquer la variable d'intérêt. Elles permettent difficilement de comprendre les interactions entre les gènes et l'effet de ces interactions sur la variable à expliquer.

1.1 Un exemple de méthode « filtre » : les tests de permutation

Dans le contexte qui nous intéresse ici, les méthodes « filtre » peuvent être utilisées pour ordonner les gènes selon l'intensité de la relation observée entre leur niveau d'expression et un caractère biologique d'intérêt. Ces méthodes opèrent sur chaque gène indépendamment des autres gènes, et consiste à effectuer un test pour déterminer le degré de significativité du lien observé entre son profil d'expression et le caractère d'intérêt. Ces approches n'intègrent en aucune façon l'idée de sytème et partent du principe que chaque gène peut à lui seul avoir un effet sur le caractère d'intérêt. En grande dimension, il est illusoire de penser réussir, avec ces méthodes, à identifier les quelques gènes les plus pertinents pour expliquer un mécanisme biologique. Elles sont généralement utilisées pour effectuer un premier tri dans les données, mais ne permettent pas de réduire considérablement la taille des données, au risque de conserver de l'information qui soit plutôt redondante que complémentaire. Elles sont très utilisées en biologie en raison notamment de leur efficacité calculatoire. Le test le plus adapté est choisi en fonction des hypothèses que l'on peut faire sur la distribution des données et en fonction de la nature des variables. Par exemple, lorsque l'on souhaite comparer l'expression moyenne d'un gène sur deux échantillons indépendants, on utilise le test de Student ou celui de Wilcoxon-Mann-Whitney (non paramé-

trique) si l'hypothèse de normalité de l'expression du gène dans les deux échantillons n'est pas raisonnable. On fait une analyse de variance si l'on souhaite comparer plus de deux échantillons. Si le caractère d'intérêt est une variable continue, on teste la significativité d'une corrélation...

Un test statistique est une démarche par laquelle on va choisir entre deux possibilités en fonction des observations faites sur un échantillon. Les deux possibilités sont l'hypothèse nulle H_0 et l'hypothèse alternative H_1 (le rejet de l'une implique l'autre). L'objectif est de choisir entre ces deux hypothèses tout en maîtrisant le risque d'erreur associé à ce choix. Deux types d'erreurs sont envisageables : l'erreur de type 1 qui consiste à rejet H_0 (accepter H_1) alors qu'elle est vraie, et l'erreur de type 2, commise en acceptant H_0 à tord (H_1 vraie). Les probabilités des quatres résultats possibles sont notées :

Vérité / *Décision*	H_0	H_1
H_0	$1-\alpha$	β (erreur type 2)
H_1	α (erreur type 1)	$1-\beta$

L'hypothèse principale du test est l'hypothèse nulle H_0. En supposant que la distribution de la statistique de test est connue sous H_0, il devient possible de déterminer la plausibilité de la valeur observée de la statistique de test quand l'hypothèse nulle est vraie. En pratique, on suppose que H_0 est vraie et on calcule la probabilité d'obtenir une statistique de test plus extrême que celle observée (distribution connue). Cette probabilité est appelée la p-valeur du test (p-value en anglais). Pour un risque α fixé, on rejette H_0 si la p-valeur est supérieure à α. Le risque α évolue dans le sens contraire de β. Plus on est exigeant pour le rejet de H_0 (α petit) et plus le risque de l'accepter à tord (β) augmente, entrainant une diminution de la puissance du test $1-\beta$. Il faut être en mesure de calculer β à α fixé pour déterminer le meilleur compromis, ce qui n'est pas toujours possible puisque cela nécessite de connaître la distibution de la statistique de test sous H_1.

Avec les tests de permutation, aucune hypothèse sur la distribution des données n'est nécessaire. Ils pourraient constituer une bonne alternative aux tests classiques pour la recherche en biologie [39], même si ils sont encore assez peu utilisés.

1.1.1 Les tests de permutation

Les tests de permutation ont été proposés par Fisher [18] au début des années 30. Le point de vue adopté est différent de celui de la théorie classique des tests. Au lieu de faire une hypothèse sur la distribution de la statistique de test, celle-ci est estimée empiriquement à partir d'un grand nombre d'échantillons construits par permutations aléatoires des éléments de l'échantillon de départ.

Un test classique est déclaré significatif avec une p-valeur égale à 0.05 si la statistique de test calculée sur les obervations est placée dans la région critique de la distribution supposée de la statistique de test (5% des valeurs les plus extrêmes). Ce qui suggère que si l'on réitère plusieurs

fois l'expérience, en créant d'autres échantillons aléatoires (indépendant du premier) de même taille que le premier, alors on s'attend à avoir un test significatif sur ces nouveaux échantillons dans 5% des cas. Cela revient à postuler la répétabilité de l'expérience à taille d'échantillon identique, et pour des échantillons aléatoires.

Avec les tests de permutation, la vision du problème est tout autre, et après avoir calculé T la statistique du test, la question peut se formuler ainsi : quelle chance avais-je d'observer T sur les données de mon échantillon ? Ainsi, la p-valeur du test qui va être associée à « cette chance » n'a pas vocation à être utilisée pour prévoir l'apparition du phénomène sur de nouveaux échantillons. Cependant, la reproductibilité d'un phénomène mis en évidence sur un échantillon donné n'est pas pour autant exclue, si l'échantillon au départ est jugé suffisamment représentatif et pertinent, en se fondant sur l'expertise (biologique par exemple) plutôt que sur des critères statistiques. Cette démarche est peut être un peu plus honnête dans la mesure où les échantillons sont en pratique très rarement aléatoires comme supposé avec les tests classiques. En ne faisant aucune hypothèse sur la distribution des données les tests de permutation sont applicables pour toutes sortes de statistiques de test et d'hypothèses nulles.

La démarche des tests de permutation est très générale et peut se résumer ainsi lorsque l'on dispose d'une série de couples d'observations $(\mathbf{x}_i, \mathbf{y}_i)_{i=1,...,N}$ pour deux variables X et Y observées sur un échantillon de taille N :

1. On définit l'hypothése H_0 portant sur la variable X, et la statistique de test T correspondante.
2. On génère un grand nombre de données fictives par permutations aléatoires des observations de la variable X : pour tout $l = 1, ..., L$, on définit $\pi_l = \left(\mathbf{x}_{\sigma_l(i)}, \mathbf{y}_i\right)_{i=1,...,N}$, où σ_l est une permutation des entiers $(1, ..., N)$.
3. Pour toutes les données fictives π_l, on calcule la statistique de test T_l (X permuté, Y non permuté). On obtient ainsi une distribution empirique de la statistique de test T.
4. On place sur la distribution empirique la valeur T_0 de la statistique de test calculée sur les observations réelles : si cette valeur fait partie de celles qui sont les plus extrêmes sur cette distribution, alors on rejette H_0.
5. En notant $\pi_0 = (\mathbf{x}_i, \mathbf{y}_i)_i$ les observations réelles, et en supposant que les π_l sont distincts, la p-valeur p du test bilatéral s'écrit :

En ajoutant dans le calcul de la p-valeur une indicatrice sur l'appartenance des observations réelles dans l'ensemble des données fictives, cela évite d'avoir une p-valeur nulle, dans le cas où aucune configuration π_l n'est associée à une statistique de test $|T_l| \geq |T_0|$. En réalité, les observations réelles sont obtenues par la permutation identité, et il est légitime de l'ajouter si elle n'apparaît pas dans les π_l. Dans l'idéal, les échantillons fictifs devraient couvrir l'ensemble des permutations possibles de l'échantillon de départ, ce qui est en pratique irréalisable. Cette difficulté est contournée avec l'échantillonnage aléatoire qui permet de déterminer une distribution empirique approximative de la statistique de test. En ce sens, les tests de permutation tombent sous l'appellation de méthodes de Monte Carlo.

Test sur la corrélation Soient $\mathbf{x} = (\mathbf{x}_1, ..., \mathbf{x}_N)$ et $\mathbf{y} = (\mathbf{y}_1, ..., \mathbf{y}_N)$ les vecteurs des N observations de deux variables quantitatives notées X et Y, et r_0 un coefficient de corrélation entre $\mathbf{x}$ et $\mathbf{y}$. La corrélation linéaire (Pearson) permet d'étudier une relation linéaire entre les variables X

et Y, et lorsque cette relation n'est pas linéaire (monotone) il est possible d'utiliser, par exemple, la corrélation des rangs de Spearman :

— La corrélation linéaire (Pearson)

$$r_0 = \frac{cov(\mathbf{x}, \mathbf{y})}{\sqrt{var(\mathbf{x})var(\mathbf{y})}} = \frac{\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_1 - \bar{\mathbf{x}})(\mathbf{y}_i - \bar{\mathbf{y}})}{\sqrt{\frac{1}{N^2}\sum_{i=1}^{N}(\mathbf{x}_1 - \bar{\mathbf{x}})^2 \sum_{i=1}^{N}(\mathbf{y}_1 - \bar{\mathbf{y}})^2}}$$

— La corrélation des rangs de Spearman si la relation est monotone non linéaire

$$r_0 = \frac{cov(R^{\mathbf{x}}, R^{\mathbf{y}})}{\sqrt{var(R^{\mathbf{x}})var(R^{\mathbf{y}})}},$$

où $R^{\mathbf{x}}$ et $R^{\mathbf{y}}$ sont les vecteurs des rangs des observations $\mathbf{x}$ et $\mathbf{y}$.

On crée L jeux de données fictifs $\pi_l = \left(\mathbf{x}_{\sigma_l(i)}, \mathbf{y}_i\right)_i$ par permutations aléatoires des observations $\mathbf{x}$. Il est possible de construire $N!$ jeux de données fictifs distincts. On définit l'hypothèse nulle $H_0 : \{r_0 = 0\}$ (test bilatéral), où r_0 est le coefficient de corrélation calculé sur les données réelles. Sous l'hypothèse nulle, à une permutation près, on a de grande chance d'obtenir une corrélation très similaire et proche de 0. On calcule le coefficient de corrélation r_l (Pearson ou Sperman) entre $\mathbf{x}_{\sigma_l(i)}$ et $\mathbf{y}_i$ pour tout $l = 1, ..., L$. Finalement, en supposant que les π_l sont distincts, la p-valeur du test s'écrit : Un exemple de test de permutation sur la corrélation de Spearman est représenté sur la Figure 1.1.

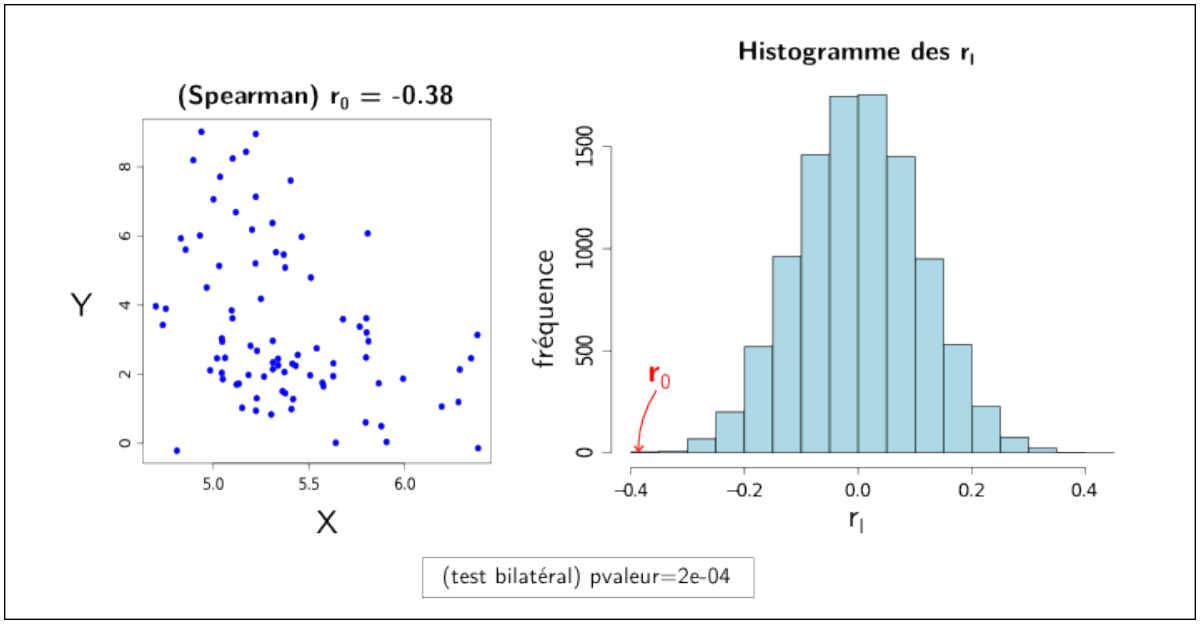


FIGURE 1.1 – Test de permutation sur la corrélation de Spearman (10 000 échantillons fictifs). Représentation du biplot croisant les observations de X et de Y, et de l'histogramme des corrélations calculées sur les échantillons fictifs. Le découpage en tranche pour la représentation de l'histogramme a été effectué par la formule de Sturge.

Si les données de X et Y proviennent d'un couple gaussien, alors la statistique de test $T = \sqrt{N-2}\frac{R}{\sqrt{1-R^2}}$, où R est la variable de la corrélation linéaire (Pearson) sur l'échantillon, suit sous H_0 une loi de Student à $N-2$ degrés de liberté (d'espérance 0 et de variance $\frac{1}{N-1}$). Et elle suit approximativement cette même loi, si R est la variable de la corrélation de Spearman. Sous ces hypothèses, la distribution empirique de la statistique de test déterminée à partir d'un très grand nombre de jeux de données fictifs, est une approximation de la distribution d'une loi de Student à $N-2$ degrés de liberté. On s'attend alors, à obtenir une p-valeur pour un test classique sur la corrélation, très proche de celle obtenue pour un test de permutation (sur notre exemple de la Figure 1.1, on obtient une p-valeur pour le test classique de la corrélation de Spearman de $2.3e-04$, très proche de celle du test de permutation).

Comparaison de deux échantillons indépendants On cherche ici à comparer les valeurs de la variable quantitative X dans deux échantillons indépendants induits par la variable binaire $Y \in \{0,1\}$ (comparaison de l'expression d'un gène, par exemple, chez les diabétiques et chez les non diabétiques). Les vecteurs des N observations de X et Y sont notés $\mathbf{x}$ et $\mathbf{y}$ respectivement. On crée L jeux de données fictifs $\pi_l = \left(\mathbf{x}_{\sigma_l(i)}, \mathbf{y}_i\right)_i$ par permutations aléatoires des observations $\mathbf{x}$. On calcule, pour tout $l = 1, .., L$, la différence m_l des moyennes : où N_1 (resp. N_0) est le nombre d'individus ayant la modalité 1 (resp. 0) pour la variable Y. Il y a $(N_1+N_0)!/(N_1!N_0!)$ permutations possibles. On définit l'hypothèse nulle H_0 : $m_0 = 0$ (test bilatéral), où m_0 est la différences des moyennes sur les données réelles, puis on positionne m_0 sur la distribution empirique des m_l pour déterminer la p-valeur p du test. En supposant que les L jeux de données fictifs sont distincts, elle s'écrit : La Figure 1.2 est un exemple de test de permutation pour la comparaison de deux échantillons indépendants (par les moyennes).

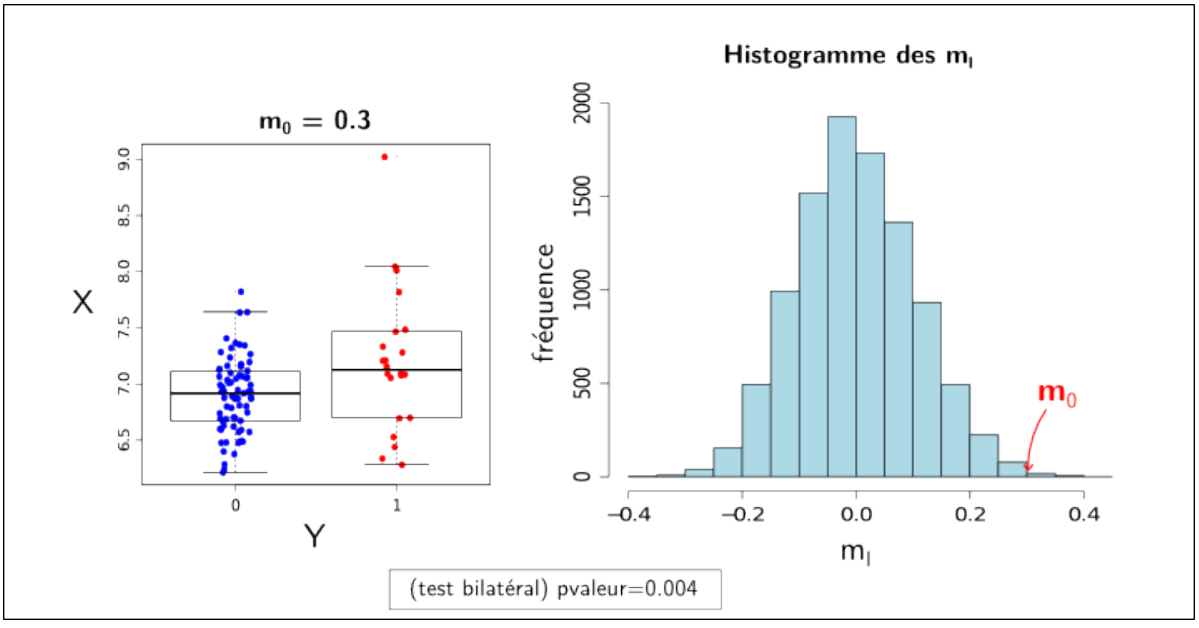


FIGURE 1.2 – Test de permutation sur la différence des moyennes dans deux échantillons indépendants (10 000 échantillons fictifs). Représentation boxplot de la distibution des observations de X dans les deux échantillons, et de l'histogramme des différences des moyennes calculées sur les échantillons fictifs. Le découpage en tranche pour la représentation de l'histogramme a été effectué par la formule de Sturge.

Comme pour la corrélation, on peut faire le parallèle avec un test classique de Student si les hypothèses du test (normalité et égalité des variances dans les deux échantillons) sont respectées sur l'échantillon. La distribution de la statistique de test peut alors être approchée par la distribution empirique déterminée à partir d'un très grand nombre d'échantillons fictifs, et dans ce cas, les p-valeurs pour le test de Student et pour le test de permutation sont proches (sur notre exemple, la p-valeur du test de Student vaut 0.042. Elle est assez éloignée de celle obtenue par les tests de permutation car l'hypothèse de normalité des données dans les deux échantillons n'est pas respectée).

Interprétation de la p-valeur La p-valeur d'un test de permutation est une estimation de la p-valeur du test exact de permutation, du test effectué sur toutes les permutations possibles. Elle correspond à la probabilité d'obtenir une statistique de test plus extrême que celle observée, conditionnellement à la distribution empirique déterminée à partir des données fictives. Cette p-valeur dépend non seulement des jeux de données fictifs qui ont été générés, mais aussi du nombre L de permutations, et elle ne peut être infèrieure à $\frac{1}{L}$.

1.2 Une méthode algorithmique : les forêts aléatoires de Breiman

La méthode algorithmique des forêts aléatoires (random forest en anglais) introduite par Breiman en 2001 [9] est devenue très populaire en biologie pour répondre aux problèmes de classification supervisée, de régression et de sélection de variables. Cette méthode d'apprentissage automatique (« machine learning » en anglais) offre l'opportunité d'identifier des relations non-linéaires entre les données, de construire des modèles complexes aux bonnes capacités prédictives et robustes face au bruit. L'objectif est de remplacer des hypothèses probabilistes (loi de l'échantillon gaussienne par exemple) qui ne sont pas toujours réalistes par de nombreuses simulations. Elle semble particulièrement adaptée pour l'analyse de données transcriptomiques [14]. Cette méthode tire profit de deux sources d'aléa pour améliorer les précisions des prédictions (faible biais et faible variance) : le bagging ou bootstrap averaging ainsi qu'une sélection aléatoire de variables lors de la construction du modèle. C'est une méthode d'ensemble ou un méta-modèle qui repose sur l'aggrégation des predictions obtenues sur un ensemble d'arbres de régression ou de discrimination [10] de façon à en améliorer les performances.

Les arbres de régression et de discrimination Un arbre de régression ou de discrimination est un arbre binaire construit par divisions successives d'un échantillon d'apprentissage à l'aide des prédicteurs (partitionnement récursif de l'échantillon d'apprentissage). Soit $\mathbf{y} = \{y_1, ..., y_N\}$ le vecteur composé de N observations indépendantes d'une variable réponse Y qui peut être aussi bien quantitative (arbre de régression) que qualitative (arbre de discrimination), et $\mathbf{x^1}, ..., \mathbf{x^p}$, les observations sur le même échantillon de p prédicteurs quantitatifs. Chaque sous-échantillon obtenu à la suite d'une division est associé à un noeud de l'arbre. Une division s'effectue à l'aide d'un des prédicteurs (Figure 1.3) qui est choisi pour rendre la division « optimale ». La définition d'un critère permettant de juger de l'optimalité d'une division dépend de la nature de la variable réponse. Partant de l'échantillon complet des N observations, deux noeuds sont créés sur l'arbre par la division de l'échantillon en deux sous-échantillons de tailles N_1 et N_2 respectivement ($N_1 + N_2 = N$).

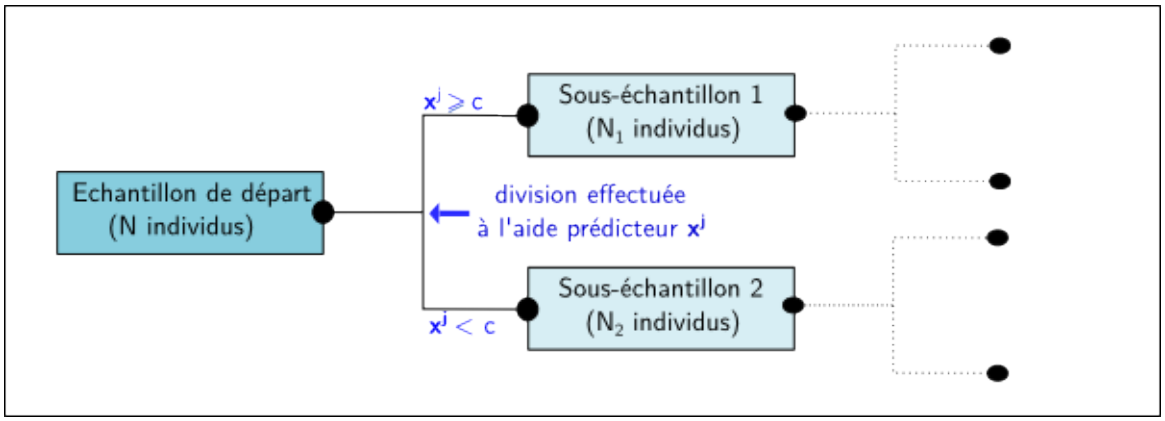


FIGURE 1.3 – Première subdivision de l'échantillon à l'aide du prédicteur $\mathbf{x^j}$ (arbre binaire)

Si $\mathbf{y}$ est quantitative, alors on cherche une division qui minimise la variance intra-groupe, ou de façon équivalente, qui maximise la variance inter-groupe. Dans le cas où $\mathbf{y}$ est une variable qualitative à m modalités, le critère d'hétérogénéité de Gini est utilisé pour mesurer l'impureté d'un ensemble d'observations vis à vis de la réponse. En d'autre termes, cette mesure doit être minimale, c'est à dire nulle, si tous les individus composant le sous-échantillon ont la même modalité. On cherche alors une division qui maximise la diminution de l'impureté. Ces deux critères sont définis dans la Table 1.1.

Division de l'échantillon à partir du prédicteur $\mathbf{x}^j$ pour lequel le critère suivant est maximum :	
$\mathbf{y}$ quantitative : I_{inter}	$\mathbf{y}$ qualitative à m modalités : Δ_{gini}
$I_{inter} = \frac{1}{N}(N_1(\bar{\mathbf{y}}_1 - \bar{\mathbf{y}})^2) + N_2(\bar{\mathbf{y}}_2 - \bar{\mathbf{y}})^2) = \frac{N_1 N_2}{N^2}(\bar{\mathbf{y}}_1 - \bar{\mathbf{y}}_2)^2$ avec $\bar{\mathbf{y}}_1$ (resp. $\bar{\mathbf{y}}_2$) la moyenne de $\mathbf{y}$ dans le sous-échantillon 1 (resp. 2).	$\Delta_{gini} = I^{\mathbf{y}}_{gini} - I^{\mathbf{y}_1}_{gini} - I^{\mathbf{y}_2}_{gini}$ avec $I^{\mathbf{y}}_{gini} = 1 - \sum_{k=1}^{m} p_k^2$, le critère de Gini calculé pour le noeud associé à tous les indi--vidus, où p_k est la proportion d'individus ayant la modalité k. Les critères de Gini $I^{\mathbf{y}_1}_{gini}$ et $I^{\mathbf{y}_2}_{gini}$ sont calculés respectivement pour les noeuds associés aux sous-échantillons 1 et 2.

TABLE 1.1 – Critères pour la subdivision d'un noeud sur un arbre CART.

Ce processus est réitéré sur chaque noeud et chaque sous-échantillon est subdivisé à nouveau. Un noeud est terminal, s'il n'y a plus de division admissible, soit parce que tous les individus dans le sous-échantillon associé au noeud, ont la même valeur pour la variable réponse, soit parce que la taille du sous-échantillon est inférieure à un certain seuil fixé. L'algorithme qui permet de construire un tel arbre est appelé l'algorithme CART (pour « Classification And Regression Trees »).

L'arbre peut ensuite être utilisé pour prédire la variable réponse Y à partir de nouvelles observations des prédicteurs. On parcourt l'arbre depuis le sommet et à chaque division, l'individu est placé dans l'un des deux noeuds en fonction de l'observation de la variable qui sert à la subdivision. Une prédiction de la réponse pour l'individu est déterminée dès que celui-ci se retrouve dans un noeud terminal. Si la réponse est quantitative, la prédiction est la moyenne des observations dans le noeud terminal. Si la réponse est qualitative, la prédiction est la modalité la plus fréquente (règle majoritaire) dans le noeud terminal.

Ces arbres présentent l'avantage de pouvoir être construits quelque soit la nature des variables (nous venons de décrire le procédé pour le cas de prédicteurs quantitatifs mais il se généralise directement au cas qualitatif), et quelque soit la distribution des données. Ils permettent de hiérarchiser les prédicteurs en fonction de la hauteur sur l'arbre à laquelle ils sont utilisés pour les subdivisions, et de faire une sélection de prédicteurs (ceux qui servent à la construction de l'arbre). Ces arbres peuvent être interprétés très facilement. Ils présentent néanmoins un inconvénient majeur qui est celui de l'instabilité des résultats observés pour de légères modifications des données. De petites variations dans les observations peuvent conduire à choisir un prédicteur différent pour une subdivision donnée, et créer un effet boule de neige entrainant la modification de l'ensemble des subsdivisions qui s'en suivent. Il faut disposer d'un échantillon de taille N très importante pour obtenir un arbre qui puisse être suffisamment robuste.

Le bagging ou bootstrap averaging En imaginant que l'ensemble des variables du modèle, $Y, X^1, ..., X^p$, aient pu être observées sur des échantillons différents et de tailles suffisantes (par exemple, en coupant l'échantillon de départ en de multiples sous-échantillons), une solution pour améliorer les performances prédictives des arbres de régression/discrimination, consisterait alors, à constuire un arbre à partir de chaque échantillon, puis à effectuer une prédiction avec chacun de ces arbres, et finalement, à choisir la prédiction qui fait à peu près l'unanimité. C'est l'idée du bagging, procédure par laquelle on va chercher à améliorer la prédiction en se basant sur les prédictions faites par des modèles différents (arbres binaires dans notre cas). Si la réponse Y est qualitative, la prédiction correspond à la modalité qui ressort le plus avec les différents modèles, et quand Y est quantitative, la prédiction correspond à la moyenne des prédictions obtenues avec les différents modèles.

En pratique, cependant, les observations sont faites sur un seul échantillon de taille N qui ne peut être subdivisé en de multiples sous-échantillons de tailles suffisantes. Pour créer ces multiples sous-échantillons « différents », on peut alors avoir recours à la technique du bootstrap, qui consiste à tirer avec remise N individus dans l'échantillon de départ. Un échantillon bootstrap ne contient en général qu'un sous-ensemble d'individus de l'échantillon de départ, et certains d'entre eux apparaissent plusieurs fois, ayant ainsi un poids plus important dans l'échantillon. En multipliant les échantillons bootstrap, on crée de multiples configurations en faisant varier la composition de l'échantillon (individus) et le poids donné aux individus (nombre d'apparition).

En résumé, la technique du bagging appliquée aux arbres de régression ou de discrimination consiste à :

1. créer b échantillons bootstrap $\{\mathbf{z}_1^*, ..., \mathbf{z}_b^*\}$ de taille N à partir d'un échantillon de départ $\mathbf{z} = \{\mathbf{z}_1, ..., \mathbf{z}_N\}$, où $\mathbf{z_i} = (\mathbf{y}_i, \mathbf{x}_i^1, ..., \mathbf{x}_i^p)$ pour tout $i = 1, ..., N$,

2. construire un arbre de régression/discrimination $T_{z_l^*}$ à partir de chaque échantillon bootstrap $\mathbf{z}_l^*$ (algorithme CART),
3. aggréger les prédictions $\hat{\mathbf{y}}_l$ obtenues avec les différents arbres $T_{z_l^*}$. Si la variable réponse Y est :
   — quantitative : $\hat{\mathbf{y}}_{bagging} = \frac{1}{b}\sum_{l=1}^{b} \hat{\mathbf{y}}_l$,
   — qualitative : $\hat{\mathbf{y}}_{bagging} = \arg\max_k \sum_{l=1}^{b} \mathbb{1}_{\{\hat{\mathbf{y}}_l = k\}}$.

Le bagging permet d'améliorer les capacités prédictives du modèle, mais à la fin, on obtient une multitude d'arbres, une forêt en quelque sorte, et il n'est plus possible de déterminer et d'interpréter les règles de division des noeuds comme sur un arbre simple.

Les forêts aléatoires de Breiman Comme nous venons de le décrire, la méthode du bagging permet d'aggréger les résultats obtenus par différents modèles. Les échantillons bootstrap qui servent à construire les arbres ne sont pas indépendants puisqu'ils proviennent du même échantillon de départ, et les similitudes entre ces arbres peuvent être factices. Elles n'auraient pas nécessairement été observées si les arbres avait été établis à partir d'échantillons indépendants. Et pourtant, ces similitudes entre les arbres sont déterminantes pour la prédiction avec le bagging, puisque l'on choisit une prédiction qui fait la quasi-unanimité.

Breiman propose alors d'ajouter une source d'aléa pour rendre les arbres un peu plus « indépendants », qui consiste pour chaque subdivision d'une noeud, à choisir la variable qui crée la division optimale, non plus dans l'ensemble des prédicteurs, mais dans un ensemble réduit de q prédicteurs choisis au hasard (q fixé). Une modification de l'algorithme CART est effectuée :

1. à chaque noeud de l'arbre, q prédicteurs sont tirées aléatoirement sans remise dans l'ensemble des p prédicteurs ($q < p$),
2. la division du noeud est effectuée à partir de l'une de ces q variables, c'est à dire celle qui maximise I_{inter} (resp. Δ_{gini}) pour un arbre de régression (resp. un arbre de discrimination).

L'algorithme des forêts aléatoires est le même que celui du bagging en utilisant la version modifiée de l'algorithme CART. Trois paramètres devront alors être fixés, le nombre q, la taille minimale d'un noeud et le nombre b d'échantillons bootstrap. Liaw et Wiener [38] proposent, en se basant sur des considérations d'ordre empirique, de choisir pour les arbres de régression (resp. les arbres de discrimination), le paramètre $q = \frac{p}{3}$ (resp. $q = \sqrt{p}$) et une taille minimale pour les noeuds de 5 (resp. 1). Le nombre b d'échantillons bootstrap ou d'arbres devra être choisi le plus grand possible pour obtenir des résultats suffisamment stables (aux alentours de 10000).

La validation du modèle L'erreur de prédiction du modèle est calculée à partir des résultats obtenus pour chaque arbre ou échantillon bootstrap, et plus précisément, à partir des erreurs de prédiction faites sur les échantillons OOB (pour « Out-Of-Bag ») composés des individus excluent des échantillons bootstrap :

1. Pour chaque arbre $T_{z_l^*}$, construit à partir d'un échantillon bootstrap z_l^*, on calcule le vecteur $\hat{\mathbf{y}}_l^{oob}$ des prédictions obtenues par $T_{z_l^*}$ pour les individus qui ne sont pas dans z_l^*. L'erreur de prédiction e_l est ensuite définie par :
   — $e_l = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{y}_{l,i}^{oob} - \hat{\mathbf{y}}_{l,i}^{oob})^2$ quand Y est quantitative (erreur quadratique moyenne ou MSE),

— $e_l = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}_{\{\mathbf{y}_{l,i}^{oob} \neq \hat{\mathbf{y}}_{l,i}^{oob}\}}$ quand Y est quantitative (le taux de mal classés).

2. L'erreur globale e du modèle est calculée en faisant la moyenne des erreurs obtenus sur les différents échantillons OOB : $e = \frac{1}{b}\sum_{l=1}^{b} e_l$.

Estimation de l'importance des prédicteurs dans le modèle Sur un arbre de régression ou de discrimination, le rôle des prédicteurs dans le modèle est facilement interprétable et il est possible de les hiérarchiser et de les sélectionner. Dès que l'on aggrège différents arbres comme avec la méthode des forêts aléatoires il n'est plus possible de faire des interprétations aussi directes, et il devient nécessaire de définir des critères pour juger de l'importance des prédicteurs dans le modèle. Deux mesures d'importances ont été proposées :

1. *Mesure définie à partir des échantillons OOB* : pour chaque arbre, l'erreur de prédiction e_l est calculée à partir de l'échantillon OOB, puis les observations sur l'échantillon OOB de chaque prédicteur X^j, sont aléatoirement permutées et l'erreur de prévision à nouveau calculée, notée e_l^j. L'importance d'un prédicteur X^j est alors défini par la moyenne des différences observées entre les erreurs de prédiction e_l et les erreurs de prédiction e_l^j sur les b échantillons bootstrap : $\frac{1}{b}\sum_{l=1}^{b}(e_l - e_l^j)$.
2. *Mesure du gain de pureté aux noeuds* : pour un arbre de discrimination, l'importance d'une variable donnée est définie par la moyenne des gains de pureté Δ_{gini} observés dès que la variable est choisie sur un arbre pour diviser un noeud. Pour un arbre de régression, cette mesure d'importance est calculée par la moyenne des décroissances de la somme des résidus au carrés, observées dès que la variable est choisie sur un arbre pour diviser un noeud.

Ces mesures d'importance peuvent être utilisées pour sélectionner des prédicteurs (mesure d'importance élevée). La première mesure, définie à partir des échantillon OOB, est plus adaptée à un contexte prédictif car elle ne dépend pas de la structure interne des arbres, contrairement à la deuxième, qui se place dans un contexte plutôt explicatif.

1.3 Les modèles de régression classiques et pénalisés

Quelque soit le type de régression, pénalisée ou classique, l'objectif est de modéliser une variable réponse Y en fonction d'un ensemble de prédicteurs supposés quantitatifs et notés $X^1, ..., X^p$. Nous allons nous intéresser au cas où Y est une variable quantitative, et à celui où Y est une variable binaire.

1.3.1 Modèles de régression linéaire et logistique

Le modèle de la régression linéaire Lorsque la réponse Y est quantitative, le modèle linéaire suppose que l'espérance conditionnelle de Y sachant les prédicteurs $X^1, ..., X^p$, est une combinaison linéaire des prédicteurs : $\mathbb{E}(Y|X^1, ..., X^p) = \beta_0 + \sum_{j=1}^{p} \beta_j X^j$. La meilleure approximation de Y s'écrit alors :

$$Y = \beta_0 + \sum_{j=1}^{p} \beta_j X^j + \epsilon,$$

où ϵ est une variable aléatoire d'espérance nulle et indépendante de la loi jointe des prédicteurs, c'est à dire, qu'en notant σ_ϵ la variance de l'erreur ϵ, on a $\text{var}(Y|X^1, ..., X^p) = \sigma_\epsilon$.

Le modèle de la régression logistique Lorsque la réponse $Y \in \{0,1\}$ est une variable binaire, il n'est plus possible de modéliser directement Y en fonction d'une combinaison linéaire des prédicteurs car les valeurs produites ne sont plus dans l'ensemble $\{0,1\}$. L'enjeu est alors de modéliser π, la probabilité a posteriori que la réponse soit positive (égale à 1) sachant les prédicteurs X^j, et par une simple comparaison de $\mathbb{P}(Y=1|X^1,..,X^p)=\pi$ et de $\mathbb{P}(Y=0|X^1,..,X^p)=1-\pi$, on en déduit la valeur de Y :

$$\text{si} \qquad \frac{\pi}{1-\pi} > 1 \qquad \text{alors} \qquad Y=1.$$

Cette quantité, qui définit la règle d'affectation, est appelée le rapport de chances (« odds » en anglais). Le modèle de la régression logistique fait l'hypothèse que c'est la transformation logit de π qui peut être modélisée par une combinaison linéaire des prédicteurs :

$$logit(\pi) = \ln\left[\frac{\pi}{1-\pi}\right] = \beta_0 + \sum_{j=1}^{p} \beta_j X^j.$$

La probabilité π peut alors s'écrire à partir de la réciproque (fonction logistique) de la fonction $logit$: $\pi = logistic(Z) = \frac{e^Z}{1-e^Z} = \frac{1}{1+e^{-Z}}$, où $Z = \beta_0 + \sum_{j=1}^{p} \beta_j X^j$.

Quelque soit le modèle, l'objectif de faire une estimation de ses paramètres à partir d'observations supposées indépendantes. La grande différence entre les méthodes de régression classiques et les méthodes de régression pénalisées, est que ces dernières intègrent une hypothèse supplémentaire sur les coefficients β^j du modèle, de façon à contourner les problématiques rencontrées avec les méthodes classiques dans certains cas pathologiques (grande dimension, multicolinéarité).

1.3.2 Estimation des paramètres dans les modèles de régression classiques

Supposons que l'on dispose de N observations indépendantes du vecteur $(Y, X^1, ..., X^p)$ des variables du modèle. On note $\mathbf{y}$ le vecteur des N observations de la réponse Y, et $\mathbf{X}$ la matrice de dimension $N \times (p+1)$, dont la première colonne contient $\mathbf{1}$, le vecteur à valeurs constantes (égales à 1), et la (j+1)-ième colonne contient le vecteur $\mathbf{x}^j$ des N observations de la variable X^j, pour tout $j = 1, ..., p$.

Estimations dans le modèle de régression linéaire Dans un modèle de régression linéaire classique, le vecteur $\beta \in \mathbb{R}^{p+1}$ des coefficients, et la variance σ_ϵ des erreurs, sont estimés par minimisation du critère des moindres carrés qui s'écrit $||\mathbf{y}-\mathbf{X}\beta||^2 = (\mathbf{y}-\mathbf{X}\beta)'(\mathbf{y}-\mathbf{X}\beta)$. L'estimateur $\hat{\beta}^{MC}$ qui minimise ce critère est donné par :

$$\hat{\beta}^{MC} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y},$$

et la variance σ_ϵ est estimée par :

$$\hat{\sigma}_\epsilon = \frac{1}{N-p-1}||\mathbf{y} - \mathbf{X}\hat{\beta}^{MC}||^2.$$

Cette estimation est réalisable uniquement dans le cas où la matrice carrée $\mathbf{X}'\mathbf{X}$ est inversible (matrice de rang $p+1$). A noter également, que la variance des estimateurs $\hat{\beta}_j^{MC}$ dépend de l'inverse de la matrice $\mathbf{X}'\mathbf{X}$: la matrice de variance-covariance de $\hat{\beta}^{MC}$, s'écrit $\Sigma_{\hat{\beta}^{MC}} = \hat{\sigma}_\epsilon(\mathbf{X}'\mathbf{X})^{-1}$, où $\hat{\sigma}_\epsilon$ est la variance empirique des erreurs du modèle.

Estimations dans le modèle de régression logistique L'estimation du vecteur $\beta \in \mathbb{R}^{p+1}$ des coefficients dans un modèle de régression logistique est effectuée par la méthode du maximum de vraisemblance. La vraisemblance dans le cas de la régression logistique correspond à une vraisemblance conditionnelle puisque l'on ne modélise pas directement Y, mais la probabilité d'avoir $Y = 1$ sachant les prédicteurs. Soit $\pi_i = \mathbb{P}(Y_i = 1|\mathbf{X_i})$, la probabilité que la réponse soit positive pour le i-ème individu sachant $\mathbf{X_i}$, on a alors $\mathbb{P}(Y_i = \mathbf{y}_i|\mathbf{X_i}) = \pi_i^{\mathbf{y}_i}(1-\pi_i)^{1-\mathbf{y}_i}$, et la vraisemblance s'écrit :

$$L(\beta) = \prod_{i=1}^{n} \pi_i^{\mathbf{y}_i}(1-\pi_i)^{1-\mathbf{y}_i} = \prod_{i=1}^{n} \left(\frac{1}{1+e^{-(\beta_0+\sum_{j=1}^{p}\beta_j\mathbf{x}_i^j)}}\right)^{\mathbf{y}_i} \left(1-\frac{1}{1+e^{-(\beta_0+\sum_{j=1}^{p}\beta_j\mathbf{x}_i^j)}}\right)^{1-\mathbf{y}_i}$$

Il n'existe pas de solution analytique au problème de la maximisation de la log-vraisemblance dans ce cas, mais la fonction étant convexe, il est possible d'approcher la solution par un algorithme et d'en déduire un estimateur $\hat{\beta}^{logit}$ de β. L'algorithme de Newton-Raphson est celui qui est le plus utilisé pour maximiser la log-vraisemblance notée LL. Il consiste à choisir un vecteur b^0 quelconque de $\mathbb{R}^{p+1}$ pour l'initialisation, puis à chaque itération t, à poser :

$$b^{t+1} = b^t - \left(\frac{\partial^2 LL(b^t)}{\partial b \partial b'}\right)^{-1} \frac{\partial L(b^t)}{\partial b},$$

et à itérer jusqu'à la converge. Soit $H(\hat{\beta}^{logit}) = \frac{\partial^2 LL(\hat{\beta}^{logit})}{\partial b \partial b'}$ la matrice hessienne des dérivées partielles d'ordre 2, qui s'écrit aussi $H(\hat{\beta}^{logit}) = -\mathbf{X}'V\mathbf{X}$, où V et la matrice diagonale de taille $N \times N$ ayant pour éléments diagonaux les $\hat{\pi}_i(1-\hat{\pi}_i)$. La matrice de variance-covariance asymptotique de l'estimateur $\hat{\beta}^{logit}$ est définie par $\Sigma_{\hat{\beta}^{logit}} = -H(\hat{\beta}^{logit})^{-1} = -(\mathbf{X}'V\mathbf{X})^{-1}$.

En pratique, il est rare que l'échantillon utilisé pour faire les estimations, ait été sélectionné par échantillonnage simple aléatoire, c'est à dire que le nombre de cas observés ($\sum_i \mathbb{1}_{\{\mathbf{y_i}=1\}}$) soit représentatif de la réalité et d'espérance valant Np, où $p = \mathbb{P}(Y = 1)$ (prévalence de la maladie par exemple). En général, quand p est faible, on utilise un échantillonnage stratifié de façon à inclure suffisamment de cas dans le modèle et les cas sont sur-représentés par rapport à la réalité. L'estimation des $\hat{\pi}_i$ est alors incorrecte, et pour la rectifier, il faut ajouter le terme $ln\left(\frac{p}{1-p}\right)$ à la constante $\hat{\beta}_0^{logit}$. Si p ne peut être fixé a priori, alors les $\hat{\pi}_i$ ne peuvent être corrigés et utilisés. Seul le rapport de chance $\frac{\hat{\pi}}{1-\hat{\pi}}$ est interprétable dans ce cas, par exemple, si il vaut deux alors l'individu à deux fois plus de chance d'être un cas ($\mathbf{y}_i = 1$) que de ne pas en être un ($\mathbf{y}_i = 0$), ce qui ne signifie pas qu'il en est un.

1.3.3 Problème de multicolinéarité et « fléau de la dimension »

La multicolinéarité La qualité des estimations des paramètres d'un modèle de régression linéaire ou logistique peut être fortement mise à mal lorsqu'il existe des relations linéaires entre deux ou plusieurs prédicteurs, même imparfaites. Quand ces relations sont parfaites, c'est à dire qu'un ou plusieurs prédicteurs expliquent complètement un autre prédicteur, les estimations dans les modèles deviennent impossibles car la matrice $\mathbf{X}'\mathbf{X}$ (ou $\mathbf{X}'V\mathbf{X}$ pour la régression logistique) n'est plus inversible (de rang $< p+1$). Il est alors nécessaire de faire un choix dans les variables pour supprimer la redondance. Quand il existe des relations linéaires entre certaines variables assez importantes, même si elles restent imparfaites, le phénomène de multicolinéarité peut quand

même engendrer des problèmes importants : variance des estimateurs élevée, coefficients non significatifs dans le modèle même si la qualité globale du modèle est bonne (nous définirons par la suite les critères de qualité d'un modèle)... Les phénomènes de multicolinéarité ne sont pas directement liés à la dimension des données mais ils ont plus de chance d'apparaître lorsque l'on augmente le nombre de prédicteurs.

La grande dimension Un autre cas pathologique qui exclue complètement l'utilisation des modéles de régression classiques, est celui où le nombre de variable excède la taille de l'échantillon ($p > n$). Dans ce cas, les systèmes d'équations qui permettent d'estimer les paramètres du modèle sont sur-déterminés et admettent une infinité de solutions (la matrice $\mathbf{X}'\mathbf{X}$ ou $\mathbf{X}'V\mathbf{X}$ est singulière).

La solution Dans ces deux cas de figure, une régularisation du problème est nécessaire et les méthodes de régressions pénalisées constituent une bonne alternative à celles des régressions classiques.

1.3.4 Les régressions pénalisées : Ridge, Lasso et Elastic-Net

Les méthodes de régression pénalisées consistent à régulariser un problème de régression en introduisant une pénalité sur le vecteur des coefficients. L'idée est de « forcer » un certain nombre de coefficients à être nuls ou proches de zéro dans le modèle. L'ajout d'un biais dans le modèle s'accompagne d'une diminution substentielle de la variance des estimateurs, et permet de lutter contre le sur-ajustement, ou de redimensionner un problème en sélectionnant un nombre raisonnable ($p << N$) de variables. Nous allons commencer par présenter ces méthodes de régression pénalisée dans le cadre d'un modèle de régression linéaire, et nous intéresser ensuite au cas de la régression logistique.

Les pénalités Ridge, Lasso et Elastic-Net pour la régression linéaire Un modèle de régression pénalisée pour une régression linéaire intègre au problème de la minimisation du critère des moindres carrés une pénalité sur le vecteur des coefficients. Le problème s'écrit :

$$\arg\min_{\beta} \left\{ ||\mathbf{y} - \mathbf{X}\beta||_2^2 + g_\lambda(\beta) \right\},$$

où g_λ est la fonction de pénalisation sur $\mathbb{R}^{p+1}$ et à valeur dans $\mathbb{R}$. Les différentes fonctions de pénalisation, Ridge, Lasso et Elastic-Net sont présentées dans la Table 1.2.

Le problème d'optimisation pour l'estimation des paramètres peut se réécrire comme un problème sous contrainte :

$$\arg\min_{\beta} ||\mathbf{y} - \mathbf{X}\beta||^2 \quad \text{s.c. } g_\lambda(\beta) \leq c,$$

où $c \geq 0$ est le paramètre de régularisation. Le paramètre λ de la méthode peut être choisi par validation croisée. Quand $\lambda \to 0$, les estimateurs des coefficients des modèles de la régression Ridge et Lasso tendent vers l'estimateur des moindres carrés. Quand $\lambda \to +\infty$, il ne reste plus que le coefficient β_0 (la constante) dans le modèle. Pour la régression Elastic-Net, si $\lambda_1 \to 0$ alors l'estimateur des coefficients tend vers celui de la regression Ridge, et quand $\lambda_2 \to 0$ il tend vers celui de la régression Lasso.

Type de pénalité	Paramètre	Pénalité	Référence
Ridge	$\lambda \in \mathbb{R}^+$	$g_\lambda(\beta) = \lambda \|\|\beta\|\|_2^2 = \lambda \sum_{j=0}^{p} \beta_j^2$	[29]
Lasso	$\lambda \in \mathbb{R}^+$	$g_\lambda(\beta) = \lambda \|\|\beta\|\|_1 = \lambda \sum_{j=0}^{p} \|\beta_j\|^2$	[61]
Elastic-Net	$\lambda = (\lambda_1, \lambda_2) \in \mathbb{R}^+ \times \mathbb{R}^+$	$g_\lambda(\beta) = \lambda_2 \|\|\beta\|\|_2^2 + \lambda_1 \|\|\beta\|\|_1$ $= \alpha \|\|\beta\|\|_2^2 + (1-\alpha) \|\|\beta\|\|_1,$ en posant $\alpha = \frac{\lambda_2}{\lambda_1 + \lambda_2}$	[73]

TABLE 1.2 – Les pénalités Ridge, Lasso et Elastic-Net

Pour illustrer les effets des différents types de pénalité sur l'estimation des paramètres, nous avons représenté sur la Figure 1.4, l'exemple simple d'un modèle ne contenant que deux prédicteurs. La forme de l'espace des contraintes varie d'un pénalité à l'autre. L'estimateur est déterminé dès qu'une ligne de niveau du critère des moindres carrés rencontre l'espace des contraintes. Avec la pénalité Ridge, l'espace des contraintes (boule) ne présente pas de point anguleux au niveau des axes, contrairement à celui associé à la pénalité Lasso. Ainsi, les ellipses auront beaucoup moins de chance de rencontrer (en premier) les zones sur lesquelles l'un des coefficient est nul. Cette chance est beaucoup plus importante avec la pénalité Lasso, ce qui exlique la plus grande parcimonie de ses estimateurs. Avec la pénalité Elastic-Net, on se place dans l'entre-deux avec une chance non négligeable d'avoir un estimateur parcimonieux.

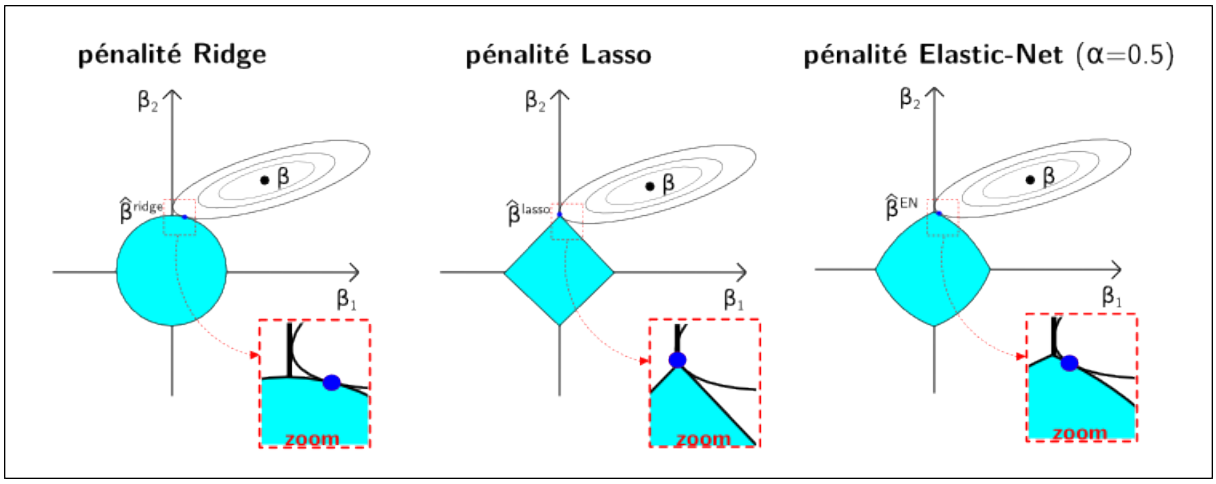


FIGURE 1.4 – Estimation des paramètres du modèle (2 variables explicatives) avec les pénalités Ridge, Lasso et Elastic-Net. Les zones bleues sur les graphiques correspondent à l'espace des contraintes, et les elipses autours de β représentent les lignes de niveau pour le critère des moindres carrés.

Le choix d'une pénalité Le choix d'une pénalité plutôt que d'une autre dépend du phénomène étudié et des contraintes posées par les données. Les regressions Ridge, Lasso et Elastic-Net peuvent toutes trois être utilisées en grande dimension ($p >> N$) et en présence d'un phénomène de multicolinéarité. Cependant, les avantages et les inconvénients des différents types de régressions pénalisées ne portent pas sur les mêmes aspects, et il faut choisir en fonction des

spécificités des données.

La régression Ridge avec sa pénalité en norme $||.||_2$ permet de diminuer la variance des estimateurs en introduisant un léger biais, et a l'avantage de conserver (en général) l'ensemble des prédicteurs dans le modèle : les coefficents sont « rétrécis » vers 0 mais ne sont en général pas complètement nuls. Cette propriété est surtout intéressante quand il existe un phénomène de multicolinéarité, mais que la dimension des données n'est pas trop importante. Des variables très corrélées ne portent pas nécessairement de l'information redondante, et il peut être nécessaire dans certains contextes de conserver l'ensemble des variables pour favoriser les interprétations. C'est le cas notamment pour l'analyse de données transcriptomiques : l'expression de deux gènes peut être très corrélée parce qu'ils sont impliqués dans une même fonction biologique mais leurs rôles peuvent être bien distincts et plus ou moins central dans cette fonction. Si il y avait un choix à faire entre ces deux gènes, il faudrait le faire en se basant sur des connaissances biologiques plutôt que sur des considérations statistiques. Néanmoins, cet avantage fait aussi la limite de la régression Ridge dans le contexte de la grande dimension. En conservant l'ensemble des prédicteurs dans le modèle, quand leur nombre est très important, on se confronte alors au problème du surapprentissage pouvant fortement compromettre les capacités prédictives du modèle, et les résultats sont difficilement interprétables quand le modèle est trop complexe.

La régression Lasso est plus adaptée que la régression Ridge en grande dimension. Sa pénalité en norme l_1 permet de réduire la complexité du modèle en produisant des estimateurs parcimonieux : un certain nombre de coefficients sont mis à zéro ce qui entraine une sélection de variables. La sélection d'un nombre raisonnable de prédicteurs facilite la lisibilité et l'interprétation des résultats tout en diminuant le risque de surapprentissage. La régression Lasso présente quand même l'inconvénient de traiter la question des groupes de prédicteurs fortement corrélés en sélectionnant un représentant au hasard à l'intérieur de chaque groupe. Cette caractéristique peut être limitante en application et notamment pour l'analyse de données transcriptomiques car les gènes sélectionnés à l'intérieur des groupes de gènes aux profils d'expression très corrélés ne sont pas nécessairement ceux qui portent l'information biologique la plus pertinente.

La régression Elastic-Net est un compromis entre la régression Ridge et la régression Lasso. En combinant les deux pénalités, elle permet de sélectionner un nombre raisonnable de prédicteurs tout en essayant de conserver au maximum les structures de groupes de prédicteurs très corrélés : la pénalité l_1 permet d'obtenir un estimateur parcimonieux et la pénalité en norme $||.||_2$ permet de conserver une certaine homogénéité entre les cofficients associés à des prédicteurs très corrélés. Cette méthode de régression semble être plus appropriée pour l'analyse de données transcriptomiques. C'est d'ailleurs en prenant en compte les contraintes inhérentes à ce type d'application que Zou et Hastie [73] ont proposé cette approche.

Afin d'illustrer les différences entre ces trois méthodes, nous avons construit un modèle intégrant quatre groupes de prédicteurs, à l'intérieur desquels les prédicteurs sont fortement corrélés positi- vement, et une variable réponse pouvant être expliquées par les groupes de prédicteurs. Nous avons estimé les coefficients dans les modèles de régression Ridge, Lasso et Elastic-Net (détail sur les méthodes d'estimation dans le paragraphe suivant). Les résultats sont représentés sur la Figure 1.5. Avec la régression Ridge, on identifie clairement les quatres groupes de prédicteurs car les coefficients associés aux prédicteurs dans un même groupe sont très similaires. Pour un paramètre de pénalisation suffisamment grand ($\log(\lambda) > -1$), la régression Lasso sélectionne

quatre prédicteurs, un dans chaque groupe. Quand le paramètre de pénalisation est trop petit, les estimations se mettent à osciller en raison du phénomène de multicolinéarité. La variance des estimations augmente car l'estimateur se rapproche de celui des moindres carrés. Avec la régression Elastic-Net ($\alpha = 0.5$), on réussit à identifier les quatre groupes de prédicteurs même si les coefficients à l'intérieur de ces groupes sont moins homogènes qu'avec la régression Ridge. La parcimonie des estimations avec Elastic-Net n'est pas aussi importante qu'avec le Lasso, mais quelques prédicteurs dans les groupes qui ont le moins d'effet sur la variable réponse sont quand même exclus du modèle.

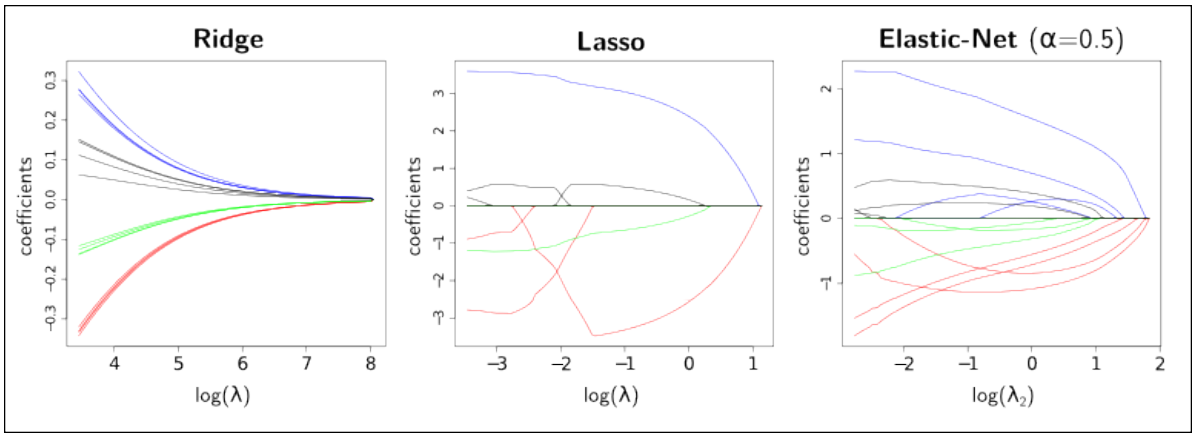


FIGURE 1.5 – Représentation de la trajectoire des coefficients estimés par régression Ridge, Lasso et Elastic-Net, en fonction du paramètre de la pénalisation. Les coefficients associés aux prédicteurs d'un même groupe (fortement corrélés) ont la même couleur sur le graphique.

Les estimations Par convention, les prédicteurs dans la matrice $\mathbf{X}$ sont supposés centrés-réduits, et le vecteur $\mathbf{y}$ centré. Le problème d'optimisation sous contrainte avec la pénalité Ridge a une solution explicite tandis qu'il n'en existe pas avec les pénalités de type Lasso et Elastic-Net. Pour ces dernières, la solution est recherchée à l'aide d'algorithmes.

La solution au problème de l'estimation des coefficients dans un modèle de régression Ridge s'obtient par dérivation matricielle. Elle est unique et s'écrit : .

$$\hat{\beta}_{\lambda}^{ridge} = (\mathbf{X}'\mathbf{X} + \lambda I)^{-1}\mathbf{X}'\mathbf{y},$$

où I est la matrice identitée de dimension $(p+1) \times (p+1)$. L'inversion de la matrice $\mathbf{X}'\mathbf{X} + \lambda I_p$ peut s'avérer être un calcul trop coûteux si le nombre p de prédicteur est élevé. En pratique, le calcul est effectué en utilisant la décomposition en valeurs singulières de la matrice $\mathbf{X}$. En posant $\mathbf{X} = UDV'$, avec U la matrice orthogonale de dimension $N \times p$, V la matrice orthogonale de dimension $p \times p$, et D la matrice diagonale ayant pour éléments diagonaux les p valeurs singulières, l'estimateur de la régression Ridge se réécrit :

$$\hat{\beta}_{\lambda}^{ridge} = V \ \text{diag}\left(\frac{d_j}{d_j^2 + \lambda}\right) U'\mathbf{y}.$$

Pour les estimations dans les modèles de régression Lasso et Elastic-Net différents algorithmes ont été proposés, en particulier l'algorithme LARS [15] pour le Lasso, et l'algorithme de Friedman et al.[20], un algorithme de descente de coordonnées efficace, et adapté pour les trois pénalités (disponible dans le paquet R « glmnet »).

La régression logistique pénalisée Les pénalités Ridge, Lasso et Elastic-Net peuvent aussi bien être utilisées en régression logistique. La pénalisation ne porte plus sur une erreur quadratique comme pour le modèle de régression linéaire, mais sur la log-vraisemblance. Il s'agit alors de résoudre le problème d'optimisation suivant :

$$\arg\max_{\beta} \{LL(\beta) - g_\lambda(\beta)\},$$

où LL est la log-vraisemblance du modèle, et $g_\lambda(\beta)$ peut être la pénalité Ridge, Lasso ou Elastic-Net. Il n'existe pas de solution explicite à ce problème. Plusieurs algorithmes ont été proposés pour le résoudre et en particulier celui de Friedman et al. [20] qui traite le problème quelque soit le type de pénalité.

1.3.5 Évaluation des performances d'un modèle

L'objectif en apprentissage supervisé est de rechercher des modèles parcimonieux c'est à dire des modèles qui intègrent un nombre restreint de variables explicatives. Dans ce contexte, on recherche un modèle qui conduit à des prévisions fiables plutôt qu'un modèle qui explique au mieux la variable d'intérêt. Les estimateurs avec les méthodes de régression pénalisées sont plus ou moins légèrement biaisés pour tenter de faire baisser la variance et donc accroître les capacités prédictives du modèle. L'accroissement de la complexité d'un modèle améliore l'ajustement du modèle sur les données, le biais du modèle diminue, mais en général cela s'accompagne d'une hausse de la variabilité des prédictions, la variance augmente. Ainsi, il s'agit de trouver un compromis biais-variance de façon à obtenir un juste équilibre entre la complexité du modèle et sa capacité à être généralisé sur de nouvelles données.

Choix du paramètre de régularisation en régression pénalisée Le paramètre de régularisation λ pour les régressions Ridge et Lasso, ou α pour la régression Elastic-Net, est associé à la grandeur du biais des estimations, et permet de contrôler la complexité du modèle. Dans le contexte qui nous intéresse ici, la sélection de gènes biomarqueurs, la regression Ridge n'est pas adaptée car elle conduit en général à conserver l'ensemble des gènes dans le modèle. Les pénalités Lasso et Elastic-Net permettent de faire de la sélection de variables (estimations parcimonieuses) et la complexité du modèle, contrôlée par le paramètre de régularisation, est directement liée au nombre de prédicteurs inclus dans le modèle.

Dans l'idéal, il faudrait disposer de deux échantillons pour choisir le paramètre de régularisation associé au modèle qui a les meilleures capacités prédictives : on fait les estimations sur un échantillon d'apprentissage et on teste les capacités prédictives du modèle sur un échantillon de validation. En pratique cependant, les échantillons sont souvent de petites tailles et ne peuvent être scindés en deux sous-échantillons de tailles raisonnables. Une solution très couramment utilisée consiste à utiliser la méthode de la validation croisée. L'échantillon de départ est subdivisé

en K sous-échantillons et chaque sous-échantillon est utilisé successivement pour tester les capacités prédictives du modèle (échantillon de validation) construit sur les $K-1$ sous-échantillons restants (échantillon d'apprentissage). Le modèle retenu est celui qui minimise la moyenne des erreurs de prédiction calculées sur chacun des K sous-échantillons utilisés l'un après l'autre pour la validation.

Évaluation de la prédiction Dans le cas d'un modèle de régression linéaire classique ou pénalisé, la capacité prédictive du modèle peut être évaluée par l'erreur quadratique moyenne ou MSE (pour Mean Square Error) : la moyenne des carrés des différences entre les prévisions et les observations. Dans le cas de la régression logistique, l'aire sous la courbe ROC appelé AUC constitue un bon indicateur de la performance du modèle. La courbe ROC représente la proportion de vrais positifs (observations bien classées) en fonction de la proportion de faux positifs (observations mal classées).

Évaluation de l'ajustement La qualité de l'ajustement d'un modèle de régression linéaire peut être évaluée à l'aide du R^2, défini par le ratio de la variance expliquée par le modèle sur la variance de la variable à expliquer :

$$R^2 = \frac{\sum_i (\mathbf{y}_i - \bar{\mathbf{y}})^2 - \sum_i (\mathbf{y}_i - \hat{\mathbf{y}}_i)^2}{\sum_i (\mathbf{y}_i - \bar{\mathbf{y}})^2},$$

où les $\hat{\mathbf{y}}_i$ sont les valeurs prédites par le modèle. Le R^2 croît avec le nombre de prédicteurs et ne peut être utilisé pour comparer des modèles qui n'intégrent pas le même nombre de prédicteurs. Si tel est le cas on utilise plutôt la version ajustée du R^2 :

$$R^2_{ajust} = \frac{(N-1)R^2 - p}{N - p - 1}.$$

Plus la valeur du R^2 ou du R^2_{ajust} est proche de 1 est meilleur est le modèle.

Pour évaluer la qualité de l'ajustement d'un modèle de régression logistique nous avons utilisé la déviance résiduelle D_r qui compare la déviance du modèle à celle du modèle saturé (qui reconstruit parfaitement les observations) :

$$D_r = -2logLL - (-2LL_s),$$

où LL est la log-vraisemblance du modèle, et $LL_s = \prod_{i=1}^{N} \left(\frac{1}{N_1}\right)^{\mathbf{y}_i} \left(\frac{1}{N - N_1}\right)^{1-\mathbf{y}_i}$ est la log-vraisemblance du modèle saturé, avec N_1, le nombre d'observations qui prennent la valeur 1 pour la variable binaire à expliquer. Plus la déviance résiduelle est proche de 0 est meilleur est l'ajustement du modèle.

1.4 Conclusions

Avec les méthodes d'apprentissage supervisé classiques, on cherche à sélectionner un nombre restreint de gènes pour établir des modèles qui soient prédictifs du caractère biologique d'intérêt.

Mais comment convaincre un biologiste que notre modèle dit vrai ? De sa réelle pertinence ? Un modèle ne doit pas être uniquement satisfaisant pour le statisticien, mais doit « coller » au plus près des contraintes posées par l'application biologique. La recherche de biomarqueurs génétiques ne peut être effectuée uniquement sur des critères statistiques, car la pertinence statistique d'un modèle ne garantit en rien sa pertinence biologique. Il faut trouver des modèles qui aient un fondement biologique solide même si ils ont des performances prédictives moindres, de façon à favoriser la reproductibilité des résultats. Les gènes sélectionnés avec les méthodes d'apprentissage supervisé classiques ont peu de chance d'être ceux qui sont les plus pertinents sur le plan biologique et il est nécessaire de réintégrer de l'information biologique dans les analyses avant d'effectuer des sélections stingentes.

L'une des caractéristiques des données transcriptomiques est qu'il existe des groupes de variables fortement corrélées. L'information redondante sur le plan statistique ne l'est absolument pas sur le plan biologique. En effet, ces corrélations peuvent refleter des interactions entre les gènes et l'information portée par ces corrélations peut s'avérer être déterminante pour la compréhension biologique d'un phénomène. Au lieu de décomposer le système et de rechercher des effets individuels, comme avec les méthodes d'apprentissage supervisé classiques, on va s'intéresser dans le deuxième chapitre à l'analyse de ce système à l'intérieur duquel les gènes interagissent les uns avec les autres.

Chapitre 2

L'analyse des réseaux génétiques : au coeur de la biologie des systèmes

2.1 Introduction

La biologie moléculaire a longtemps adoptée une vision de nature "réductionniste" en expliquant le fonctionnement interne de la cellule comme étant la résultante de la somme des actions individuelles de chacune de ses parties (des gènes et des protéines par exemple). Cette approche a permis de nombreuses avancées dans la compréhension du vivant et des maladies, mais est néanmoins de plus en plus remise en question. En pharmacologie par exemple, la question de savoir quelles peuvent être les conséquences globales d'une action ciblée est primordiale, et le seul moyen d'y répondre (hormis par des essais cliniques) est de replacer les éléments dans un système et d'étudier leurs interactions.

L'idée que les éléments de la cellule forment un système complexe et qu'ils sont complètement interdépendants de ce système ne date pas d'aujourd'hui. Waddington décrit [66], en 1950, le fonctionnement de la cellule à travers le concept de « paysage épigénétique » à l'intérieur duquel les interactions entre les gènes jouent un rôle central. Il imagine un paysage constitué d'un ensemble de monts et de vallées et une bille se déplaçant sur le paysage en fonction des transformations de celui-ci. Les gènes contrôlent la forme du paysage et toute mutation sur un gène entraine une modification de ce paysage (Figure 2.1). La position de la bille dans une certaine vallée détermine quant à elle l'état de la cellule. Si un gène mute, le paysage se modifie, mais quelles sont les conséquences sur le déplacement de la bille d'une vallée à l'autre, c'est à dire sur l'état de la cellule ? En fonction des interactions entre le gène muté et les autres gènes, les modifications du paysage seront plus ou moins importantes et pourront selon la configuration entrainer un déplacement de la bille vers une autre vallée. Cela souligne l'importance de la structure du système et des interactions entre les éléments au sein de ce système.

La biologie des systèmes est aussi séduisante que complexe. Une petite révolution est en marche depuis une dizaine d'années avec l'avènement de nouvelles biotechnologies (la puce à ADN par exemple) offrant la possibilité de caractériser l'ensemble des parties constitutives d'un système (ADN, ARN, protéines... par exemple au niveau de la cellule). L'intégration de ces données engendre de multiples problématiques interdisciplinaires : pour l'extraction de l'information (biologie, physique, chimie, imagerie...), pour le stockage et le traitement des données (bioinformatique)

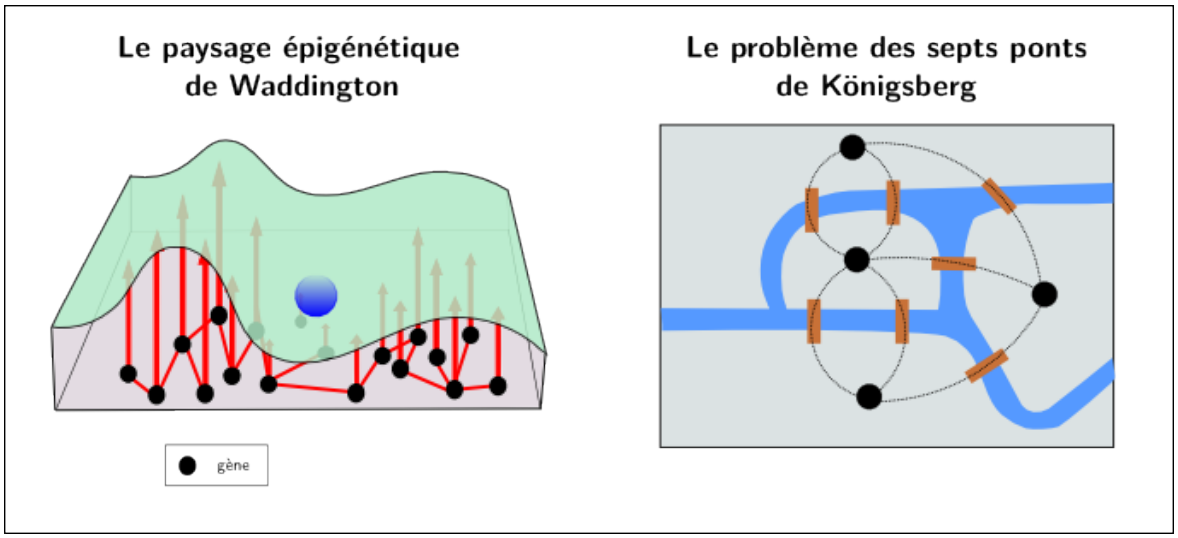


FIGURE 2.1 – Le paysage épigénétique de Waddington et le problème des septs ponts de Königsberg.

et pour la modélisation du système (mathématiques). Jusqu'ici nous avons évoqué la notion de système en faisant référence à la cellule, mais la cellule est elle-même une partie d'un système plus vaste, une population de cellules par exemple, elle même une partie d'un autre système... Le fonctionnement biologique intègre différents niveaux hiérarchisés de systèmes.

L'objectif ultime de la biologie des systèmes est de comprendre les interactions à tous les niveaux et entre tous les éléments en se plaçant le plus haut possible dans la hiérarchie. Les progrès en biotechnologie offre de nouvelles opportunités pour la collecte massive de données mais tout l'intérêt de « ratisser large » est de permettre une meilleure compréhension des phénomènes biologiques dans leur globalité. Ces données sont pourtant encore très largement étudiées en adoptant la vision « réductionniste » faisant abstraction totale des interactions entre les éléments. Un grand nombre de méthodologies statistiques (tests, régression pénalisée, sparse PLS...), développées pour l'analyse de données très volumineuses, ont pour but d'extraire un petit ensemble interprétable d'élements et susceptible d'expliquer un phénomène biologique d'intérêt. Cependant, en sélectionnant quelques « super candidats » pour expliquer une maladie, en se basant par exemple sur une p-valeur comme c'est souvent le cas en biologie, le risque de non reproductibilité des résultats est augmenté [45]. Ne vaut-il pas mieux privilégier une approche cherchant à intégrer la complexité des données plutôt que de l'en dépourvoir, et présenter des résultats paraissant peut être moins significatifs mais plus prometteurs en terme de robustesse ?

L'émergence de la science des réseaux offre une excellente opportunité pour une biologie plus « systémique ». La compréhension des réseaux (ensemble d'éléments interconnectés), ou plus précisément celle de leurs structures intrinsèques, offre de nouvelles perspectives à un plus haut niveau de complexité (intégration de voie métaboliques par exemple ou plus généralement de fonctions biologiques).

L'étude des réseaux s'est développée en mathématique à travers la théorie des graphes. L'origine de la théorie des graphes a été attribuée au mathématicien Euler qui posa le problème en 1736 des sept ponts de Königsberg (Figure 2.1). Le problème consiste à déterminer si il est possible de trouver un circuit, en partance d'un point donné, qui permettrait de traverser les

septs ponts et de revenir à son point de départ sans jamais emprunter deux fois le même pont. La réponse est non, il n'existe pas de tel chemin pour cette configuration (un tel chemin existe si est seulement si tous les sommets du graphe sont de degré pair). Les premières propriétés de la théorie des graphes ont alors vu le jour : un graphe eulérien est un graphe sur lequel il existe un circuit eulérien c'est à dire un chemin passant par l'ensemble des arêtes du graphe une seule fois et se terminant là où il a commencé. Par la suite, nous emploierons le terme réseau pour faire référence à un ensemble d'éléments réels (les gènes par exemple) tandis que le terme graphe désignera l'objet mathématique (abstraction faite à partir du réseau réel).

2.2 Brève introduction à la théorie des graphes

2.2.1 Quelques définitions

Intuitivement, un graphe est constitué d'un ensemble de points représentants des entités (par exemple les gènes pour l'analyse d'un réseau transcriptionnel), et d'un ensemble de liens, chacun reliant deux points distincts du graphe. La définition mathématique d'un graphe peut être fomulée comme suit : un graphe $G = (V, E)$ est défini par un ensemble discret V de sommets ou noeuds (vertices ou nodes en anglais), et un ensemble $E \subset V \times V$ d'arêtes ou liens (edges en anglais). On dit que deux sommets sont connectés, ou liés, ou adjacents, si il existe un lien entre les sommets. Une arête est incidente à un sommet dans un graphe si le sommet est l'une des extrémités de l'arête.

Graphe partiel et sous graphe

- Le graphe $G' = (V, E')$ est un **graphe partiel** du graphe $G = (V, E)$ si E' est inclu dans E. Le graphe G' s'obtient en enlevant une ou plusieurs arêtes sur le graphe G.
- Pour un sous-ensemble de sommets C inclu dans V, le **sous-graphe** de G induit par C est le graphe $G^C = (C, E^C)$ tel que E^C contient tous les liens de G ayant leurs deux extrémités dans C.
- Un graphe partiel d'un sous-graphe de G est un sous-graphe partiel de G.

Différents types de graphes

- Un graphe est dit **non orienté** si E est symétrique c'est à dire que les arêtes sont des paires de sommets non ordonnées, $(i, j) \in E$ si et seulement si $(i, j) \in E$.
- Un graphe est **orienté** si les arêtes sont des paires de sommets ordonnées, $(i, j) \in E \nRightarrow (j, i) \in E$. Une arête orientée est aussi appelées un **arc**.
- Un graphe **simple** est un graphe ne contenant pas de boucle sur un sommet ($(i, i) \notin E$ pour tout i), et si l'ensemble des sommets sont reliés par au plus une arête.
- Un graphe **acyclique** est un graphe ne contenant aucun **cycle**, soit une suite d'arêtes consécutives (**chaine ou chemin**) dont les sommets aux extrémités coïncident.
- Un graphe non orienté et acyclique est une forêt ou ensemble d'arbres, et un graphe acyclique orienté et couramment nommé un **DAG** (Directed Acyclic Graph).
- Un graphe **complet** est un graphe sur lequel il existe une arête entre tous les couples de sommets (pour tout $i, j \in V$ et $i \neq j$ on a $(i, j) \in E$).
- Un graphe est **pondéré** si un poids est associé à chaque arête c'est à dire qu'une fonction de poids $w : E \to [0, \infty[$ est associée au graphe. Un graphe non pondéré peut être vu comme un graphe pondéré auquel est associé la fonction de poids $w \equiv 1$.

- Un graphe est **connexe** si n'importe quel couple de sommets peut être relié par un chemin. Un graphe non connexe peut être découpé en plusieurs composantes connexes (sous-graphe connexe maximal).

Représentation matricielle d'un graphe

- Un graphe simple et fini $G = (V, E)$ peut être représenté par sa **matrice d'adjacence** $A = (a_{i,j})_{1 \leq i,j \leq p}$ à coefficients dans $\{0, 1\}$ et de dimensions $p \times p$, où $p = |V|$ est le nombre (fini) de sommets aussi appelé **ordre** du graphe, et $a_{ij} = 1$ si une arête relie i à j soit $(i, j) \in E$ ($a_{ij} = 0$ sinon). Cette matrice est symétrique dans le cas d'un graphe non orienté.
- Pour un graphe pondéré, la matrice d'adjacence est remplacée par la matrice des poids $W = (w_{ij})_{i,j}$ à coefficients dans $\mathbb{R}^+$ et on parle alors de **matrice d'adjacence pondérée**. Par la suite, nous allons nous intéresser exclusivement à des graphes pondérés et les termes de matrice d'ajacence et de matrice d'adjacence pondérée seront confondus.

Degrés, densité, voisinage

- Le **degré** de connectivité d_i d'un sommet i est le nombre d'arêtes auquelles il est relié (arêtes incidente) dans le cas non pondéré ($d_i = \sum_{j \in V} a_{ij}$). Plus généralement, quand le graphe est représenté par une matrice d'adjacence pondérée, le degré de connectivité est la somme des poids des arêtes incidentes au sommet, $d_i = \sum_{j \in V} w_{ij}$.
- La **densité** $\delta(G)$ du graphe est définie par la somme des poids des liens sur le nombre total de liens, soit $\delta(G) = \frac{\sum_{1 \leq i,j \leq p} w_{ij}}{p(p-1)} = \frac{\sum_{1 \leq i \leq p} d_i}{p(p-1)}$ (pour un graphe non pondéré on a $\delta(G) = \frac{2m}{p(p-1)}$ où $m = |E|$ est le nombre de liens).
- Le **voisinage** $\Gamma(i)$ d'un sommet i est l'ensemble de ses sommets voisins ou sommets avec lesquels il est connecté : $\Gamma(i) = \{j \in V | w_{ij} > 0\}$.

Distance, coefficient de clusterisation

- La **distance géodésique** entre deux sommets d'un graphe est la longueur du plus court chemin les reliant.
- Le **diamètre** d'un graphe est la distance géodésique maximale entre deux de ses sommets.
- Le **coefficient de clusterisation** (clustering coefficient en anglais) est une mesure permettant d'évaluer le niveau de transitivité local ou global d'un graphe. Il définit la probabilité que des sommets soient liés sur le graphe quand ceux-ci ont un voisin commun.
  - On définit le coefficient de clusterisation global C du graphe par le rapport entre 3 fois le nombre de triangles (3 sommets tous connectés ou triade fermée) et le nombre de triades connexes (1 sommet connecte 2 autres sommets).
  - Le coefficient de clusterisation local C_i pour un sommet i donné est le rapport entre le nombre de liens connectant des sommets voisins de i et le nombre possible de lien sur le voisinage de i.

  Le coefficient de clusterisation est élevé (il est compris entre 0 et 1) si la probabilité que deux sommets soient connectés augmente quand ils ont un voisin commun.

2.2.2 Modèles de graphes : du plus simple au plus réaliste

Les modèles de graphes aléatoires, générés selon un processus aléatoire, sont les premiers modèles proposés pour la modélisation des réseaux complexes. Ce type de modèle, très utilisé au départ, a cependant montré ses limites avec les avancées dans la compréhension de la topologies des réseaux complexes réels, dans lesquels les relations entre structure locale et structure globale n'ont rien à voir avec celles des graphes aléatoires. D'autres modèles ont été proposés pour tenter de reproduire les propritétés structurelles des systèmes complexes, en maitrisant chaque étape de leur construction de façon à ajouter des contraintes plus réalistes que celle du pur hasard.

Un modèle de graphe aléatoire Le premier modèle de graphe aléatoire, développé par Erdös et Rényi en 1960 [16], consiste à définir un graphe (simple et non orienté) à p sommets sur lequel chaque sommet est connecté à n'importe quel autre sommet avec la même probabilité θ. La probabilité de construire un graphe avec m liens est $P(m) = C^m_{p(p-1)/2}\theta^m(1-\theta)^{p(p-1)/2-m}$ (loi binomiale). On en déduit qu'un moyenne un graphe contient $\theta p(p-1)/2$ liens. Une caractéristique topologique importante pour l'étude de la stucture d'un graphe est la distribution des degrés de connectivité de ses sommets. Un sommet est de degré k si il est relié à exactement k autres des $p-1$ sommets du graphe. La probabilité p_k qu'un sommet soit de degré k est définie par $p_k = C^k_{p-1}\theta^k(1-\theta)^{p-1-k}$ et vaut $\theta(p-1)$ ($\sim \theta p$ si n grand) en moyenne. Pour un degré moyen fixé $\lambda = \theta p > 0$, dans la limite $p \to \infty$, les degrés sont distribués selon une loi de poisson de paramètre λ d'où $p_k = \lambda^k e^{-k}/k!$.

Une variante de ce modèle consite à paramétrer le modèle par le nombre m de liens, et à placer ces m liens au hasard sur le graphe, c'est à dire à choisir au hasard m liens parmi les $n(n-1)/2$ liens possibles. Les deux modèles coïncident si $p = 2m/p(p-1)$ et $p \to \infty$.

On peut donc considérer deux cas différents à la limite $p \to \infty$. L'un, à $\lambda = \theta p$ fixé, conduit à construire des graphes de très faible densité ($p \to 0$) avec des degrés distribués suivant une loi de poisson, et l'autre, à $\theta = 2m/p(p-1)$ fixé, conduit à construire des graphes denses contenant m liens en moyenne ($m \to \infty$).

Le modèle d'Erdös-Renyi (distribution binomiale des degrés) ne donne aucun poids à l'espace réel dans lequel peuvent se trouver les éléments du graphe. Il fait l'hypothèse que la structure est complètement aléatoire, qu'elle n'est pas communautaire (pas de groupes de sommets formant des zones plus denses sur le graphe) et que la topologie du graphe est relativement homogène, c'est à dire que tous les sommets ont plus ou moins le même degré.

Des modèles de graphe ayant la propriété d'invariance d'échelle Une propriété plus réaliste observée sur les réseaux complexes réels (en biologie notamment) est celle de l'invariance d'échelle ou absence de degré caractéristique (scale free en anglais) [2, 1]. Elle est caractérisée par une décroissance beaucoup plus lente de la distribution des degrés suivant une loi de puissance, $P_k \sim k^{-\gamma}$ avec $\gamma > 2$, où P_k est la probabilité qu'un sommet soit de degré k. Il n'y a pas de valeur typique ou caractéristique pour les degrés, contrairement au modèle binomial de Erdös-Renyi avec lequel les degrés varient faiblement autours de θp. Cette propriété est justifiée par la présence sur le graphe de quelques sommets très fortement connectés (degré élevé), appelés centres (ou hub en anglais), et d'une très grande majorité de sommets très faiblement connectés. L'invariance d'échelle des réseaux complexes réels sous-entend que le réseau s'est construit selon un dynamique particulière. Le point de vue adopté pour modéliser ce type de réseaux et différent de celui du modèle de Erdös-Renyi. La topologie du graphe n'est pas fixée au départ, elle est

la résultante de la logique ou dynamique de construction du graphe. L'idée est de modéliser le processus dynamique de formation du réseau en reconstruisant pas à pas un graphe, et pour une modélisation suffisamment proche de la réalité, on s'attend à la fin du processus à avoir un graphe invariant d'échelle.

Le modèle de Barabási et Albert [3], proposé en 1999, et couramment appelé modèle de l'attachement préférentiel, repose sur deux hypothèses principales : le graphe est en croissance c'est à dire qu'à chaque instant un sommet vient se connecter au graphe, et les nouveaux sommets se connectent préférentiellement avec les sommets déjà présents sur le graphe qui sont de degrés les plus élevés (les riches deviennent plus riches). Les réalisations de ce modèle sont bien des graphes pour lesquels les degrés sont distribués suivant une loi de puissance.

Le graphe au départ du processus contient un petit nombre m_0 de sommets et à chaque itération un sommet est ajouté au graphe et connecté à $m \leq (m_0)$ sommets déja présents sur le graphe. La probabilité $p_t(i)$ qu'un nouveau sommet se connecte, à l'itération t, à un sommet i déjà présent sur le graphe dépend du degré k_i du sommet i, et est définie par $p_t(i) = k_i / \sum_j k_j$ où $\sum_j k_j$ est la somme des degrés des sommets déjà présents sur le graphe soit le double du nombre de liens. A la fin de l'itération t, le graphe construit contient $t+m_0$ sommets et tm liens. Barabási et Albert ont montré que ce mécanisme conduit à construire des graphes invariants d'échelle, la distribution des degrés correspond à la distribution d'une loi de puissance : $P_k \sim k^{-3}$.

Une autre caractéristique topologique observée sur de nombreux réseaux complexes réels et la caractéristique de « petit-monde » (small-world en anglais) qui reflète une double propriété : celle du faible éloignement entre les sommets i.e. les sommets, même si ils ne sont pas des voisins directs, peuvent être connectés par des chemin de faibles longueurs (faible distance géodésique en moyenne entre les sommets), et celle d'avoir un coefficient de clusterisation assez élevé et indépendant de la taille du graphe i.e. les voisins d'un sommet seront souvent connectés entre eux. Le modèle de Barabási et Albert ne respecte pas cette dernière propriété : le coefficient de clustering décroit quand la taille du graphe augmente ($t \to \infty$) en suivant approximativement une loi de puissance, $C \sim p^{-0.75}$.

D'autres modèles ont été proposés pour contourner ce problème, en particulier celui de Ravasz et al. [52]. L'idée est de construire un graphe hiérarchique en combinant de manière itérative des petites agrégations de motifs. Par exemple, on part d'un ensemble de quatres sommets tous connectés (motif initial). Un sommet du motif représente le centre et les autres sommets sont appelés sommets périphériques. On crée trois répliquats du motif, on connecte les centres de ces répliquats entre eux, et leurs sommets périphériques avec le centre du motif initial. La structure ainsi créée va former un nouveau motif initial, trois répliquats de ce motif vont être créés et des liens sont placés suivant la même règle : on connecte les centres des répliquats entre eux et on connecte les sommets périphériques des répliquats avec le centre du modif initial. On itère le processus autant que nécessaire. Ce modèle génère des graphes qui sont à la fois invariants d'échelle et qui ont un coefficient de clusterisation élevé et indépendant de la taille du graphe.

2.3 Les réseaux de gènes

Choix de l'unité « gène » Les biotechnologies offrent d'immenses possibilités d'analyse. Des gènes, des protéines, des métabolites, des bactéries... Tous ces éléments observés sur différentes

cellules du foie, du tissu adipeux, de l'intestin... Le tout accompagné de données cliniques, pysiologiques... Mais par où commencer ? Comment tenter de répondre efficacement à une question biologique donnée, sans négliger des résultats qui pourraient ouvrir de nouvelles pistes en rapport direct ou indirect avec la question posée ? La tache est colossale et nécessite forcément de faire des simplifications, de restreindre au départ son attention sur un « petit » ensemble d'éléments.

L'unité qui s'impose pour commencer à explorer un phénomène biologique est le gène, en tant qu'acteur central dans le fonctionnement de la cellule. Les interactions génétiques, c'est à dire les modifications dans l'action des gènes induites par l'expression d'autres gènes, sont la conséquences d'une ou plusieurs interactions moléculaires (protéine-ADN, protéine-ARN ou protéine-protéine). Un gène ne contrôle par directement un autre gène et une interaction peut se produire de diverses manières. Un gène peut par exemple coder une certaine protéine qui va réguler l'expression d'un autre gène, ou de façon encore plus indirecte, la protéine codée par le premier gène peut influencer la production d'un certain métabolite qui sera alors le véritable régulateur du second gène. Les systèmes d'interaction génétiques dépendent complètement d'autres systèmes dans la cellule, et constituent une simplification, une image du système complet.

L'étude du rôle du génome dans les processus biologiques (la génomique fonctionnelle) constitue un axe de recherche très en vogue pour plusieurs raisons. La technologie des puces à ADN permet, d'une part, de produire des données relativement fiables et robustes à des coûts abordables. D'autre part, les groupes de gènes en interaction, co-régulés, constituent par leurs associations d'expression l'essentiel de la réponse biologique pour permettre à la cellule de s'adapter face à une extrême variété de situations.

Différents types de réseaux de gènes Le terme de réseau génétique désigne un réseau sur lequel chaque sommet correspond à un gène et chaque arête reflète une interaction entre deux gènes. Les questions relatives à l'analyse des réseaux génétiques peuvent être multiples.

On peut, par exemple, s'intéresser à l'évolution de l'expression des gènes dans le temps, au comportement dynamique des réseaux génétiques. Plusieurs modèles dynamiques ont été considérés pour inférer ce type de réseaux, en particulier à l'aide de réseaux booléens [60] ou de réseaux bayésiens dynamiques [37]. Ces analyses ne sont pas les plus répandues car elles nécessitent des données cinétiques, qui sont plus coûteuses, alors que les analyses de réseaux statiques, c'est à dire à l'instant t, peuvent s'avérer suffisantes et ont déjà donné des résultats prometteurs en génétique.

Si l'on s'intéresse maintenant à un réseau statique, il reste le problème de l'analyse de la structure d'interaction à l'instant t, lui même pouvant être appréhendé sous différents points de vue. L'approche la plus simple consiste à estimer les dépendances ou interactions entre les gènes à partir des corrélations entre les niveaux d'expression des gènes. Il est dans ce cas impossible de distinguer une interaction directe d'une interaction indirecte (au sens statistique du terme car au sens biologique du terme, un gène ne peut avoir une action directe sur un autre gène, il agit nécessairement par le biais de protéines et/ou métabolites). On considère que deux gènes dépendent l'un de l'autre si leurs profils d'expression corrèlent, même si cette corrélation masque l'effet d'un troisième gène expliquant à lui seul la corrélation entre les deux gènes en question. Pour faire une estimation plus précise des dépendances entre les gènes, la corrélation peut être remplacée par la corrélations partielle de façon à exclure de l'analyse les interactions indirectes.

Ce type de réseaux, sur lesquels l'interaction entre les gènes est modélisée par la corrélation ou la corrélation partielle, sont appelés des réseaux de co-expression de gènes.

Une autre approche consiste à identifier des relations causales entre les gènes, c'est à dire à déterminer le sens des interactions : quels sont les gènes responsables des modifications du niveau d'expression d'un gène d'intérêt ? On parle alors de réseaux de régulation de gènes.

Bien évidemment, la complexité des analyses augmente avec le niveau de précision attendu. Il est plus difficile d'estimer des corrélations partielles que des corrélations simples et encore plus délicats d'estimer des interactions causales.

L'apprentissage des réseaux de régulation de gènes à partir de données d'expresssion reste encore à ce jour une question des plus difficiles. Une approche largement étudiée, consiste à modéliser les réseaux de régulation par des réseaux bayésiens [47], lesquels représentent des relations de probabilités conditionnelles entre les éléments. En faisant l'hypothèse que l'organisation structurale des réseaux de régulation peut être décrite par un DAG (graphe orienté acyclique), il est alors possible d'estimer les relations causales entre les gènes directement à partir des données d'expression, sans avoir recours à des pratiques d'intervention comme l'invalidation de gènes (knock-out) par exemple. Friedman et al. [22] sont les premiers à utiliser ce type de modèle pour l'analyse de réseaux de régulation.

La modélisation de réseaux bayésiens est associée à différentes problématiques. En premier lieu se pose le problème de l'estimation des dépendances conditionnelles pour déterminer la structure ou squelette du graphe. Dans le cas gaussien, ce problème rejoint celui de l'estimation des corrélations partielles qui prennent des valeurs nulles en cas d'indépendance conditionnelle. Ce pose ensuite le problème de la détermination des relations de cause à effet entre les variables en orientant les arcs du graphe suivant la configuration la plus probable. Une recherche exhaustive du meilleur graphe, au sens de la causalité, s'avère très vite impossible (problème NP-difficile) car le nombre de graphes candidats augmente de façon super-exponentielle avec le nombre de sommets. Différentes solutions ont été proposées et s'appuyent notamment sur des hypothèses de parcimonie pour limiter le nombre d'interactions entre les sommets : chaque gène est régulé par un petit nombre d'autres gènes.

Parti pris Nous venons de faire la distinction entre deux types de réseaux génétiques, les réseaux de co-expression et les réseaux de régulation. L'analyse des réseaux de régulation de gènes, bien qu'effectuée à partir de données statiques (à l'instant t), met l'accent sur la mise en évidence d'une certaine dynamique dans les relations entre les gènes : quels sont les régulateurs d'un gène donné ? C'est une question de première importance mais qui n'est peut être pas la première à se poser lorsque l'on ne dispose d'aucun a priori biologique sur le contenu des données étudiées. L'estimation de « gros » réseaux de régulation reste une problématique délicate pouvant être très coûteuse en temps de calcul et souffrir d'imprécisions, sans pour autant que l'on soit en mesure d'en quantifier le risque.

L'analyse des réseaux de co-expression ne permet pas de décrire finement les interactions entre les gènes mais offre la possibilité de mettre en évidence des ensembles (ou groupes, ou communautés) de gènes, qui en partageant des profils d'expression similaires, ont de grandes chances d'avoir la même fonction dans la cellule, voir même d'être co-régulés. L'identification de communautés de gènes permet de robustifier les résultats dans la mesure où des variations au

niveau cellulaire, en terme de fonctionnalité, sont plus probables si l'on observe des modifications d'expression portant sur un ensemble de gènes plutôt que sur un gène isolé. Cela permet également d'intégrer des connaissances biologiques qui peuvent favoriser l'interprétation des résultats et la mise en perspective d'un phénomène au niveau cellulaire. Prenons un groupe de gènes co-exprimés et ayant des profils d'expression atypiques. Un certain nombre de gènes composant ce groupe sont connus et ont déjà été identifiés comme étant, par exemple, des acteurs du métabolisme du glucose. Par extrapolation, on pourra penser, d'une part, que les gènes inconnus dans ce groupe sont eux aussi impliqués dans le métabolisme du glucose, et d'autre part, que le profil d'expression anormal de ces gènes est associé à un disfonctionnement au niveau cellulaire dans le métabolisme du glucose.

Un autre aspect que nous avons étudié est celui de l'identification des gènes en position centrale sur le réseau (hub), c'est à dire très connectés avec les autres gènes du réseaux. L'existence de tels gènes peut être justifiée par l'invariance d'échelle des réseaux biologiques. Cela leurs confère la propriété d'être des réseaux très robustes face aux modifications aléatoires des gènes et de leur expression. Une majorité de gènes sont faiblement connectés et des modifications sur ces gènes ont un faible impact sur l'expression de l'ensemble des autres gènes. Par contre, si ces modifications portent sur l'un des quelques gènes centraux, les répercussions sur l'ensemble des autres gènes du réseau peuvent être très importantes. Ces gènes centraux constituent des potentiels régulateurs clés du réseau.

Reste le choix de la corrélation ou de la corrélation partielle pour caractériser les interactions entre gènes. Là encore, nous avons retenu l'approche la plus simple en modélisant les réseaux de co-expression à partir des corrélations entre les profils d'expression des gènes. La corrélation suffit quand il s'agit d'identifier des communautés de gènes co-exprimés et des gènes centraux. L'idée, quand même, n'est pas de se limiter à faire de la classification de gènes, mais d'utiliser cette information pour aller plus loin dans les analyses : pour expliquer un phénotype (caractère observable) ou une maladie, pour analyser plus finement (suppression des interactions indirectes, causalité) la structure d'interaction en se limitant à un sous ensemble de gènes assez petit, ou pour analyser les interactions des gènes avec d'autres éléments (protéines, bactéries)... La classification des gènes permet de réduire la dimension des données, en identifiant dans chaque groupe de gènes co-exprimés un profil moyen d'expression ou un représentant (les gènes centraux par exemple), tout en favorisant les interprétations biologiques qui sont indispensables pour guider le statisticien dans le choix des analyses supplémentaires à mener.

2.4 Modélisation d'un réseau de co-expression de gènes

Rappelons tout d'abord, qu'un réseau complexe réel peut être modélisé par un graphe, sur lequel chaque sommet représente un élément, et deux sommets sont connectés si les élements correspondants sont en relation ou interagissent. La notion de graphe est donc indissociable de celle d'interaction entre éléments. Dans certains contextes, pour les réseaux sociaux par exemple, les interactions entre les éléments s'observent directement : on peut par exemple considérer que deux individus interagissent si ils ont échangés au moins une fois. Ce n'est malheureusement pas le cas en biologie, et même si un a priori peut exister sur certaines interactions, il n'est pas toujours fiable et reste limité. Les interactions entre les molécules biologiques ne sont pas connues à l'avance et doivent être estimées.

L'inférence des réseaux de co-expression de gènes s'effectue à partir de données d'expression, c'est à dire de la quantité d'ARNm produite par chacun des gènes dans la cellule. On suppose que deux gènes interagissent (directement ou indirectement) si ils sont co-exprimés, la co-expression étant caractérisée par le degré de ressemblance des profils d'expression observés sur un échantillon d'individus. Deux gènes co-exprimés s'activent (production d'ARNm) ou se désactivent chez les mêmes individus.

Le cas gaussien La covariance constitue un bon indicateur de l'association (linéaire) entre deux variables. Si deux variables sont indépendantes, la covariance est nulle, et dans le cas gaussien, la réciproque est vraie. En faisant l'hypothèse que les variables d'expression sont gaussiennes, on peut alors considérer que deux gènes interagissent si la covariance entre leurs profils d'expression est non nulle.

On suppose que le vecteur aléatoire $X = {}^t(X^1, ..., X^p)$, constitué des variables d'expression de p gènes, est un vecteur gaussien de $\mathbb{R}^p$. On a $X \sim \mathcal{N}(\mu, \Sigma)$ où $\mu \in \mathbb{R}^p$ est l'espérance, et $\Sigma \in \mathbb{R}^{p\times p}$ est la matrice de variance-covariance définie positive et symétrique.

Avec $\Sigma = (\sigma_{ij})_{i,j}$ connue, le réseau de co-expression peut être modélisé par le graphe non orienté $G = (V, E)$, où V est l'ensemble $\{1, 2, ..., p\}$ des sommets, et E est l'ensemble des liens tel qu'il existe un lien entre deux sommets distincts i et j si la covariance entre l'expression du gène i et celle du gène j est non nulle : $\forall i, j \in V$, avec $i \neq j$, on a $(i, j) \in E \Leftrightarrow \sigma_{ij} \neq 0$.

Les liens du graphe peuvent être pondérés par l'intensité de l'association entre les variables. On définit alors la matrice d'adjacence W du graphe de dimension $p \times p$ ayant pour coefficients les valeurs absolues des corrélations (valeurs sans unité) : $W = (|r_{ij}|)_{i,j}$ où $r_{ij} \in [0, 1]$ est le ij-ème élément de la matrice des corrélations $R = D_\Sigma^{-\frac{1}{2}} \Sigma D_\Sigma^{-\frac{1}{2}}$ et $D_\Sigma \in \mathbb{R}^{p\times p}$ est la matrice diagonale ayant les mêmes éléments diagonaux que Σ (variances).

Toute la question est alors d'estimer Σ et d'identifier ses éléments nuls. Cela n'est pas simple en grande dimension ($p >> N$) car un estimateur classique de Σ, la matrice de covariance empirique $\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (X_i - \bar{X})'(X_i - \bar{X})$, constitue un très mauvais estimateur : elle n'est pas définie positive, elle est fortement biaisée, et en particulier, $\hat{\Sigma}$ n'est pas un estimateur parcimonieux de Σ. L'hypothèse de parcimonie pour un graphe modélisant un réseau complexe réel est pourtant une hypothèse tout à fait réaliste : on s'attend à avoir un grand nombre de sommets très peu connectés. De multiples approches ont été proposées pour améliorer l'estimateur empirique de la covariance, et parmi celles qui reposent sur l'hypothèse de parcimonie, en forçant l'estimateur à être une matrice creuse (beaucoup de zéros), la méthode la plus directe consiste à seuiller les éléments de la matrice de covariance empirique [6].

Le cas non gaussien Plus généralement, si l'on s'extrait du cadre gaussien, la problématique reste la même en grande dimension. L'estimation des associations entre variables doit être suffisament parcimonieuse pour obtenir un graphe sur lequel les différents niveaux d'organisation ressortent clairement. Il faut être en mesure de faire le tri entre les associations qui sont significatives et celles qui ne le sont pas, et toute la difficulté est alors de définir un critère pour évaluer cette significativité.

Définition d'une matrice d'association Soit S une matrice d'association symétrique de dimension $p \times p$ à coefficients dans $\mathbb{R}^+$. Chaque coefficient est une mesure de l'association entre deux variables. Les mesures de corrélation sont très couramment utilisées pour caractériser l'association entre les variables d'expression, c'est à dire la co-expression entre les gènes. Soient $\mathbf{x}^1, \mathbf{x}^2, ..., \mathbf{x}^p$ les vecteurs des N observations relatives à chacune des p variables d'expression.

Prenons par exemple le coefficient de corrélation de Pearson qui mesure l'association linéaire entre les variables. Le coefficient de corrélation de Pearson r_{ij} entre les vecteurs d'observations $\mathbf{x}^i \in \mathbb{R}^n$ et $\mathbf{x}^j \in \mathbb{R}^n$ s'écrit :

$$r_{ij} = \frac{\text{cov}(\mathbf{x}^i, \mathbf{x}^j)}{\sqrt{\text{var}(\mathbf{x}^i)\text{var}(\mathbf{x}^j)}},$$

où $\text{cov}(\mathbf{x}^i, \mathbf{x}^j) = \frac{1}{N}\sum_{k=1}^{N}(\mathbf{x}_k^i - \bar{\mathbf{x}^i})(\mathbf{x}_k^j - \bar{\mathbf{x}^j})$ est la covariance observée (empirique), et $\text{var}(\mathbf{x}^i) = \frac{1}{N}\sum_{k=1}^{N}(\mathbf{x}_k^i - \bar{\mathbf{x}^i})^2$ est la variance observée.

Lorsque les liaisons entre variables ne sont pas nécessairement linéaires, on préfèrera utiliser le coefficient de corrélation de Spearman r_{ij}^{spear}, ou corrélation des rangs, en remplaçant dans le calcul de r_{ij}, le vecteur d'observations $\mathbf{x}^i$ (resp. $\mathbf{x}^j$) par le vecteur des rangs des observations noté R^i (resp. R^j) :

$$r_{ij}^{spear} = \frac{\text{cov}(R^i, R^j)}{\sqrt{\text{var}(R^i)\text{var}(R^j)}}.$$

En posant $l_k = R_k^i - R_k^j$, la différence des rangs d'un même individu selon les classements des deux variables, le coefficient de corrélation de Spearman s'écrit plus simplement :

$$r_{ij}^{spear} = 1 - \frac{6\sum_k l_k^2}{N(N^2-1)},$$

On peut alors définir les mesures d'association s_{ij} entre les variables par $s_{ij} = |r_{ij}|$ ou par $s_{ij} = |r_{ij}^{spear}|$.

En application, nous avons systématiquement utilisé le coefficient de corrélation de Spearman pour l'analyse des réseau de co-expression de gènes car les données peuvent contenir des valeurs extrêmes et les dépendances ne sont pas sytématiquement linéaires.

Définition d'un graphe De façon immédiate, la structure d'interaction entre les gènes peut être représentée à l'aide d'un graphe pondéré de matrice d'adjacence $W \in \mathbb{R}_+^{p\times p}$ ayant pour coefficients les mesures d'association entre profils d'expression, c'est à dire que $w_{ij} = s_{ij}$, $\forall i \neq j$, et $w_{ii} = 0$, $\forall i$. Cela revient à construire un graphe complet (lien entre tout couple de sommets) dans la mesure où il n'y a pas de vrai zéro dans la matrice d'association. Une telle représentation de la stucture d'interaction ne reflète pas l'hypothèse que les gènes interagissent généralement avec un nombre limité d'autre gènes.

Quels sont les éléments de S qui peuvent être considérés comme étant non significatifs et être associés à des valeurs nulles dans la matrice d'adjacence W ? Deux approches sont envisageables, l'une consistant à définir un seuil global λ en dessous duquel les s_{ij} ne sont plus considérés comme étant significatifs, et l'autre, à faire l'hypothèse que chaque sommet du graphe interagit avec un

nombre K limité d'autres sommets, ce qui revient à dire que pour chaque élément i, seules les K valeurs s_{ij} $(j \neq i)$ les plus importantes sont significatives (seuil local).

Dans le premier cas, on construit un graphe de voisinage pour un λ-voisinage de paramètre $\lambda \in \mathbb{R}_+$, et W s'obtient en seuillant la matrice S :

$$w_{ij} = s_{ij} \mathbb{1}_{\{s_{ij} \geq \lambda\}}, \ \forall i \neq j, \text{ et } w_{ii} = 0, \ \forall i.$$

Deux sommets i et j interagissent et sont connectés sur le graphe, si la mesure s_{ij} de l'association entre les deux éléments est suffisante $(s_{ij} \geq \lambda)$.

Dans le deuxième cas, on construit un graphe de voisinage à partir des K-voisinages. Soit $\Gamma_K(i)$ l'ensemble des $K \in \mathbb{N}$ éléments avec lesquels i est le plus associé, $j \in \Gamma_K(i) \Leftrightarrow |\{l \text{ tel que } l \notin \{i, j\} \text{ et } s_{il} > s_{ij}\}| < K$, et les coefficients de W sont définis par :

$$w_{ij} = s_{ij} \mathbb{1}_{\{j \in \Gamma_K(i) \ \cup \ i \in \Gamma_K(j)\}}, \ \forall i \neq j, \text{ et } w_{ii} = 0, \ \forall i.$$

Deux sommets i et j sont connectés sur le graphe si au moins l'un des deux sommets fait parti des K éléments qui sont les plus associés avec l'autre sommet.

Ces deux approches ont l'avantage d'être simples et intuitives mais ont un inconvénient majeur qui est celui de ne pas pouvoir déterminer analytiquement un optimum pour les paramètres λ ou K. Pour choisir le paramètre λ, on peut imaginer de tester la significativité des coefficients de corrélation en utilisant la transformation de Fisher. En grande dimension cependant, cela ne fait que déplacer le problème, car les p-valeurs des tests sont fortement dépendantes de la taille de l'échantillon et le nombre de faux positifs augmente avec le nombre de variables. Il est alors nécessaire d'utiliser une méthode d'ajustement des p-valeurs pour les tests multiples (Bonferroni, FDR,...) et de choisir le taux de faux positifs admis. Si le paramètre λ, pour la construction d'un graphe de λ-voisinage, est choisi trop petit et qu'il existe sur le réseau réel des zones de densités variables, les zones du réseau les plus denses auront tendance à former des sous-graphes complets sur le graphe. Au contraire, si λ est trop grand, les éléments sur les zones les moins denses du réseau vont se retrouvés isolés (aucun voisin) sur le graphe. Le choix d'une unique valeur pour λ ne permet pas de prendre en compte l'hétérogénéité des interactions sur les différentes zones du réseau. Ce problème ne se pose plus si l'on modélise le réseau par un graphe de K-voisinage. Cela revient à seuiller localement autours de chaque élément. Néanmoins, avec cette approche, si K est choisi trop grand, des zones qui ne devraient pas être connectées peuvent l'être, comme par exemple des composantes connexes, et à l'inverse, pour un K trop petit, les différentes zones du réseau n'apparaîtrons pas clairement sur le graphe (autant de connexions à l'intérieur et entre les différentes zones).

En résumé Les coefficients de corrélation de Pearson et de Spearman sont les mesures de co-expression les plus utilisées pour décrire la co-expression entre les gènes, autrement dit, l'association entre profils d'expression des gènes. Un réseau de co-expression de gènes peut être modélisé à partir d'une matrice d'association par construction d'un graphe complet sur lequel les poids des liens correspondent aux valeurs d'association (co-expression) entre les gènes. Pour obtenir une représentation plus lisible de la structure d'interaction, il est possible de construire un graphe de voisinage (λ-voisinage ou K-voisinage) à partir des valeurs d'association. Le paramètre K ou λ doit être choisi avec précaution de façon à ce que les différentes zones du réseau, les groupes de gènes formant des communautés notamment, ressortent assez clairement sur le graphe et puissent être identifiées par une méthode de détection de communautés sur un graphe.

2.5 Détection de communautés

2.5.1 Introduction

Les communautés de gènes ou groupes fonctionnels de gènes La structure des réseaux complexes réels et des réseaux de co-expression de gènes notamment n'est pas aléatoire, les gènes s'organisent et interagissent les uns avec les autres de façon à ce que la cellule puisse répondre aux besoins de l'organisme. Le hasard ne peut être complètement compatible avec la formation de cellules stables et fonctionnelles. Il existe bien des modifications aléatoires de l'expression des gènes mais celles-ci sont maitrisées dans la plupart des cas en raison de la robustesse globale du système d'interaction entre les gènes. Nous avons vu qu'une propriété intrinsèque des réseaux complexes réels est qu'il sont structurés autours de quelques « super » éléments (centres ou hub) très connectés sur le réseau, qui en centralisant les échanges d'information, assurent la stabilité du système (propriété d'invariance d'échelle). Une autre caractéristique des réseaux réels est la présence de zones distinctes (autours des centres), à l'intérieur desquelles les éléments sont en forte interaction, tandis que les interactions aux frontières de ces zones sont assez faibles. Ces groupes d'éléments sont appelés des communautés, et les éléments au sein d'une même communauté ont souvent des propriétés communes, et jouent un rôle similaire ou complémentaire dans la structuration du réseau d'interaction. La ressemblance des profils d'expression des gènes au sein d'une même communauté suggère que les gènes partagent une fonction cellulaire commune [23]. Ainsi, par l'identification de communautés sur un réseau de co-expression de gènes, l'objectif est de mettre en évidence des groupes fonctionnels de gènes.

Comment définir une communauté sur un graphe ? La science des réseaux offre l'opportunité d'intégrer un degré d'information supplémentaire pour l'identification de groupes d'éléments étroitement liés. Au sens classique du terme, deux éléments sont similaires si ils présentent des caratéristiques communes, par exemple des profils d'expression corrélés pour les gènes. Cette notion de similarité doit être étendue de façon à prendre en compte le positionnement des éléments sur le graphe qui modélise le réseau réel. On peut imaginer que deux éléments sont similaires si ils sont proches sur le graphe au sens de la distance géodésique, ou alors qu'ils ont des voisins communs, ou qu'ils peuvent être reliés par une multitude de chemins différents... Les solutions proposées pour caractériser la proximité entre les sommets d'un graphe sont très variées et dépendent des contraintes posées par le domaine d'application.

Pour définir une communauté il n'existe pas non plus de consensus général. Si l'on se base sur une mesure de similarité entre les sommets du graphe, alors intuitivement, une communauté est un groupe de sommets plus fortement similaires entre eux qu'avec les autres sommets du graphe. Selon un autre point de vue, une communauté est considérée comme un sous-graphe ayant des propriétés structurelles particulières : un sous-graphe sur lequel tous les sommets sont adjacents (clique), ou sur lequel tous les sommets sont connectés avec un minimum de k autres sommets du sous-graphe (k-core), ou encore, des sous-graphes sur lesquels la répartition ou l'organisation des liens est trop structurée pour être dûe au hasard (critère de modularité). Intuitivement, en adoptant ce point de vue, une communauté est un sous-graphe dense (beaucoup de liens) et peu connecté au reste du graphe. Cela nécessite cependant de connaître, ou d'estimer assez finement la structure du graphe, afin qu'elle soit suffisament lisible et creuse. Un cas extrême est celui des graphes complets, sur lesquels il devient impossible d'identifier des sous-graphes ayant une structure particulière.

Dans la mesure où les réseaux biologiques sont inconnus au départ et que leur modélisation constitue en elle même une problématique difficile, pouvant conduire à des résultats plus ou moins proches de la réalité, nous nous sommes principalement intéressés aux méthodes de détection de communautés qui reposent sur la caractérisation des proximités ou des similarités entre les sommets du graphe. Ces méthodes restent notamment utilisables quand le graphe est très dense (beaucoup de liens), si celui-ci est pondéré, et même dans le cas extrême d'un graphe complet.

Méthodes de détection de communautés sur un graphe Le problème de la détection de communautés sur un graphe rejoint celui de la classification non supervisée, ou du partitionnement des sommets du graphe si l'ensemble des sommets peut être découpé en communautés disjointes. En réalité, les réseaux réels peuvent contenir des éléments qui n'appartiennent à aucune communauté (bruit), et il se peut également qu'un même élément appartienne à plusieurs commuanutés et provoque un chevauchement de communautés : un gène peut par exemple avoir plusieurs fonctions cellulaires. Le cas le plus étudié reste celui de la détection de communautés disjointes et c'est celui qui va nous intéresser ici.

La littérature dans le domaine de la classification ou du partitionnement des sommets d'un graphe est très abondante [19, 54]. En gardant à l'esprit notre objectif premier, qui est celui de l'analyse des réseaux de co-expression de gènes, nous allons focaliser notre attention sur deux méthodes de détection de communautés qui ont fait leurs preuves dans ce contexte. La plus populaire à ce jour, est l'approche WGCNA (Weighted Gene Co-expression Network Analysis) proposée par Zhang et Horvath [70] qui consiste à mesurer la similarité entre les sommets du graphe en se basant sur la ressemblance de leurs voisinages. Deux sommets sont d'autant plus proches qu'ils partagent les mêmes voisins. La matrice de similarité est ensuite utilisée pour faire de la classification hiérarchique ascendante. Il semblerait que les groupes fonctionnels de gènes, ou plus généralement les éléments dans la cellule, soient eux-même organisés selon une hiérarchie [4], ce qui rend tout à fait naturel l'emploi des méthodes de classification hiérarchiques en biologie. La principale difficulté avec cette approche est de déterminer la partition optimale parmi toutes celles qui sont envisageables, obtenues à chaque étape de regroupement (ou comment couper le dendrogramme).

D'autres méthodes regroupées sous l'appelation de méthodes spectrales [64, 44] consistent à partitionner les sommets du graphe en les plongeant dans l'espace engendré par les vecteurs propres de la matrice décrivant le graphe (matrice d'adjacence ou matrice laplacienne). Ce changement de représentation replace les sommets du graphe dans un espace euclidien, sur lequel on peut utiliser des méthodes de partitionnement classiques du type k-means. Les informations apportées par l'analyse des propriétés spectrales du graphe peuvent s'avérer très utiles pour la caractérisation de la structure communautaire sous-jacente. L'inconvénient majeur de cette méthode est la nécessité de fixer au départ le nombre de classes, sur lequel il est difficile d'avoir un a priori. De nombreuses solutions ont été proposées pour déterminer de façon automatique ce paramètre, mais elles présentent toutes leurs limites, et dans les cas les moins évidents, quand la structure communautaire est peu marquée, elles peuvent se montrer inefficaces.

Nous ne détaillerons pas ici les nombreuses autres méthodes proposées pour la détection de communautés. Certaines d'entre elles reposent sur l'optimisation d'un critère comme celui de la modularité [43, 8]. La modularité compare la proportion d'arêtes qui se trouvent à l'intérieur

des communautés, à la proportion d'arêtes dans ces mêmes communautés, mais sur un graphe construit de façon tout à fait aléatoire (arêtes placées au hasard). Plus la modularité est élevée, et plus les communautés sont marquées, dans le sens où il serait peu probable d'observer une proportion d'arêtes aussi importante à l'intérieur des zones définies par les communautés, si le graphe avait été construit de façon aléatoire (sans structure communautaire). Que ce soit dans son utilisation directe, en tant que fonction à optimiser, ou indirecte, utilisée comme fonction de qualité pour comparer différentes partitions, et choisir une partition optimale, la modularité est très utilisée pour la détection de communautés sur un graphe. Nous pouvons également citer les algorithmes séparatifs, le plus connu et le plus utilisé, étant celui de Girvan et Newman [24, 11], qui proposent de retirer progressivement les arêtes situées entre les communautés (définition d'une mesure de l'intermédiarité des arête, betweenness en anglais) pour qu'elles apparaîssent plus clairement.

Nous avons fait le choix de détailler plus particulièrement les approches WGCNA et spectrales, car elles adoptent des points de vue très éloignés l'un de l'autre, ce qui permet de se faire une idée assez générale sur les problèmatiques rencontrées pour la détection de communautés sur un graphe. Les résultats obtenus avec l'approche WGCNA ont fait l'objet de nombreuses publications en biologie [68, 50, 63, 13]. Un paquet [34] (WGCNA) pour le logiciel R a été développé. Cette approche est séduisante car sa mise en oeuvre est simple et sa documentation très complète. La méthode spectrale suscitent également beaucoup d'intérêt et plus particulièrement dans la communauté mathématique [64, 44, 40, 69, 62, 65]. Elles se fondent sur des concepts mathématiques variés tirés de l'algèbre linéaire, de la théorie des perturbations, des problèmes d'optimisation...

2.5.2 La méthode spectrale

L'appellation « méthode spectrale » pour le partitionnement des sommets d'un graphe désigne l'ensemble des méthodes qui reposent sur l'analyse des propriétés spectrales des matrices représentant le graphe.

Le cas idéal Plaçons nous dans le cas idéal d'un graphe pondéré d'ordre p (nombre de sommets) sur lequel chaque communauté forme une composante connexe du graphe (aucun lien entre les communautés). La matrice d'adjacence pondérée $W \in \mathbb{R}_+^{p\times p}$ associée au graphe est une matrice symétrique, ce qui implique que ses valeurs propres sont réelles et associées à p vecteurs propres indépendants. En effet, d'après le théorème spectrale, toute matrice symétrique peut être décomposée en élément propres dans une base orthonormale de vecteurs propres. De plus, W est diagonale par blocs (un bloc pour chaque communauté d'éléments) donc en principe, pour chacun des blocs, on peut trouver un vecteur propre de W sur lequel, un élément est nécessairement différent de zéro pour un indice pointant sur le bloc, et égal à zéro pour les autres indices. La connaissance de l'ensemble de ces vecteurs propres (un vecteur propre par bloc) répond au problème de la détection des communautés sur le graphe, il suffit de regrouper les éléments qui prennent des valeurs non nulles sur un même vecteur propre de cet ensemble.

Cependant, on ne sait comment identifier ce sous-ensemble de vecteurs propres dans l'ensemble des p vecteurs propres. Ils n'apparaîssent pas nécessairement dans l'ordre des valeurs propres : il se peut que les deux vecteurs propres associés aux deux plus grandes valeurs propres de W pointent tous les deux sur un même bloc de la matrice. C'est la raison pour laquelle le

graphe est représenté par une matrice laplacienne dans la méthode spectrale. A la différence de la matrice d'adjacence, l'ordre des éléments propres d'une matrice laplacienne apporte l'information suffisante pour l'identification du sous-ensemble de vecteurs propres pointant sur chacune des communautés.

Definition La version non normalisée de la matrice laplacienne L, associée à un graphe G de matrice d'adjacence pondérée W est définie par :

$$L = D - W,$$

où D est la matrice diagonale des degrés ayant pour éléments diagonaux les degrés $d_1, d_2, ..., d_p$, avec $d_i = \sum_{j=1}^{p} w_{ij}$ pour tout i.

Les propriétés spectrales de la matrice laplacienne peuvent être utilisées pour répondre au problème de la détection des communautés.

Propriété 1 *La matrice laplacienne L admet p valeurs propres réelles (comptées avec leur ordre de multiplicité), et 0 est la plus petite valeur propre : $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_p$. Le vecteur propre constant $\mathbf{1}_p$ de $\mathbb{R}^p$ est associé à la valeur propre 0.*

La matrice laplacienne L est symétrique (somme de matrices symétriques) et semi-définie positive. En effet, pour tout $u \in \mathbb{R}^p$ on a $u'Lu = \sum_{i,j=1}^{p} w_{ij}(u_i - u_j)^2$, d'où $u'Lu \geq 0$ pour tout $u \in \mathbb{R}^p$. La matrice L admet alors p valeurs propres réelles et positives $\lambda_1, \lambda_2, ..., \lambda_p$ (comptées avec leur ordre de multiplicité), et p vecteurs propres indépendants. Zéro est valeur propre de L et elle est associée au vecteur propre constant $\mathbf{1}_p$ car on a $L\mathbf{1}_p = D\mathbf{1}_p - W\mathbf{1}_p = D - D = 0$.

Propriété 2 *L'ordre de multiplicité de la valeur propre 0 d'une matrice laplacienne L est égal au nombre K de composantes connexes de G. De plus, l'espace propre associé à la valeur propre 0 est engendré par l'ensemble des vecteurs indicatrice $\mathbb{1}_{C_1}, ..., \mathbb{1}_{C_K}$ pointant sur les différentes composantes connexes, où les $C_1, ..., C_K$ sont les ensembles d'indices des sommets dans chacune des composantes connexes de G (les C_k forment une partition de l'ensemble V des sommets du graphe), et les vecteurs indicatrice sont définis pour tout k par : $\mathbb{1}_{C_k,i} = 1$ si le sommet i est dans la k-ième composante connexe du graphe i.e. $i \in C_k$, et $\mathbb{1}_{L_k,i} = 0$ sinon.*

Considérons dans un premier temps un graphe connexe c'est à dire un graphe ne contenant qu'une seule composante connexe, $K = 1$, et montrons que la valeur propre 0 de la matrice laplacienne L est d'ordre de multiplicité 1. Pour tout $u \in \mathbb{R}^p$, on a $u'Lu = \sum_{i,j=1}^{p} w_{ij}(u_i - u_j)^2$. Si u est associé à la valeur propre 0, alors $(\star)$ $\sum_{i,j=1}^{p} w_{ij}(u_i - u_j)^2 = 0$. Si un élément i est connecté à un élément j sur le graphe, alors $w_{ij} > 0$ et $w_{ij} = 0$ sinon. Sur un graphe connexe, pour tout sommet i du graphe, il existe au moins un sommet j différent de i qui soit connecté à i sur le graphe i.e. il existe $i \neq j$ tel que $w_{ij} > 0$, ce qui implique que $w_{ij}(u_i - u_j)^2 = 0$ si est seulement si $u_i = u_j$. On en déduit par $(\star)$ que pour tous sommets i et j différents et connectés sur le graphe, on a nécessairement $u_i = u_j$. De plus, la connexité du graphe implique qu'il existe des chemins permettant de relier tout couple de sommets, donc u est nécessairement un vecteur constant. On peut finalement en conclure, que la valeur propre 0 est de multiplicité 1 et qu'elle est associée au vecteur propre constant $\mathbf{1}_p$.

Si l'on s'intéresse maintenant à un graphe contenant $K > 1$ composantes connexes. La matrice laplacienne L du graphe est alors une matrice diagonale par bloc. Les K blocs de la matrice sont

notés $L_1, ..., L_K$. Le spectre (valeurs propres) de L est l'union des spectres de chaque bloc L_k. Pour une valeur propre λ_l de L provenant du k-ième bloc L_k, le vecteur propre associé à λ_l est défini par les éléments du vecteur propre de L_k associé à λ_l, et complété par des zéros pour les éléments dans les autres blocs. On a montré précédemment que la matrice laplacienne d'un graphe connexe admet 0 pour valeur propre avec un ordre de multiplicité de 1. Cette propriété se vérifie pour toutes les matrices L_k, en tant que matrices laplaciennes de sous-graphes connexes de G. Cela implique que l'ordre de multiplicité de la valeur propre 0 pour L est égal à K (union de K valeurs propres égale à 0), et que les K vecteurs propres associés sont les indicatrices des K blocs de L.

Dans le cas idéal de K communautés formant des composantes connexes, l'identification des K premiers vecteurs propres associés à la valeur propre 0 (d'ordre de multiplicité K) de la matrice laplacienne, suffit à répondre au problème de la détection des communautés. Chaque sommet i du graphe est placé dans la k-ième communauté si et seulement si le i-ème élément du k-ième vecteur propre est différent de zéro, soit égal à 1.

Le cas général Le cas général est celui d'un graphe G sur lequel les communautés ne forment pas nécessairement des composantes connexes, elles peuvent être connectées entre elles. Nous venons de présenter les bonnes propriétés spectrales de la matrice laplacienne du graphe pour permettre la détection de communautés, et fort heureusement, elles ne sont pas complètement remises en question dans le cas général.

Le graphe G peut être vu comme un graphe perturbé ayant la même base qu'un graphe G^* idéal sur lequel les communautés forment des composantes connexes. Les graphes G et G^* ont les mêmes ensembles de sommets, les mêmes liens à l'intérieur des communautés, mais G est perturbé dans le sens où il existe des liens entre les communautés alors qu'il n'en existe pas sur G^*. Si les deux graphes contiennent K communautés, elles peuvent être identifiées sur G^* à partir des K premiers vecteurs propres (associés à la valeur propre 0) de sa matrice laplacienne L^*. Mais qu'en est-il sur G ?

Le théorème de Davis-Kahan, un résultat de la théorie des perturbations, assure que pour une petite perturbation de G^*, autrement dit, que peu de liens (ou poids des liens faibles) apparaîssent entre les communautés sur G, alors les K premiers vecteurs propres de la matrice laplacienne L du graphe G sont très proches des K premiers vecteurs propres de L^*. Plus précisément, la distance entre les espaces engendrés par les K plus petites valeurs propres pour L^* et L (maximum du sinus de l'angle fait par un vecteur propre de L avec sa projection orthogonale sur L^*), est majorée par la taille de la perturbation (norme de Frobenius de la matrice de perturbation ajoutée à L^* pour obtenir L) divisée par le trou spectral, qui est la valeur absolue de la différence entre la K-ième et la (K+1)-ième valeur propre de L^*. Ainsi, plus le trou spectral est grand et la perturbation petite, et plus les K premiers vecteurs propres de L sont proches des indicatrices des communautés (vecteurs propres de L^*).

Pour un graphe G sur lequel la stucture communautaire est assez prononcée (petite perturbation et trou specral assez grand), les K premiers vecteurs propres de L associés aux plus petites valeurs propres, $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_K$, sont proches des vecteurs indicatrice des communautés, $\mathbb{1}_{C_1}, ..., \mathbb{1}_{C_K}$. Les commuanutés peuvent être identifiées par l'algorithme des k-moyennes

(k-means) à partir des coordonnées des sommets dans le sous-espace engendré par les K premiers vecteurs propre de L. Dans ce nouvel espace, la distance entre les sommets est caractérisée par la distance euclidienne.

La détection des communautés à partir des K premier vecteur propres de L a un lien direct avec une version relaxée du problème de la minimisation du ratio de coupe (ratio-cut) pour le partitionnement. Pour un partitionnement en K comunautés, le problème d'optimisation du ratio de coupe consiste à minimiser le coût de la coupe, qui est la somme des poids des liens entre les communautés, et à pondérer par la taille des communautés de façon à limiter la formation de communautés de trop petites tailles (singletons par exemple). On note $cut(C_k, C_l) = \sum_{i \in C_k, j \in C_l} w_{ij}$ le coût de la coupe entre une communauté C_k et une communauté C_l, où les w_{ij} sont les éléments de la matrice d'ajacence pondérée W du graphe à coefficients positifs et de dimensions $p \times p$. Le problème du ratio de coupe pour un partitionnement en K communautés $C_1, ..., C_K$ s'écrit :

$$\arg\min_{C_1,...,C_K} \left\{ \sum_{k=1}^{K} \frac{cut(C_k, \bar{C_k})}{|C_k|} \right\}.$$

En notant H la matrice de dimension $p \times K$ pour laquelle chaque colonne est un vecteur indicatrice de l'une des communautés, pondéré par la taille de la communauté, la k-ième colonne de H s'écrit :

$$H_k = \frac{1}{\sqrt{|C_k|}} \mathbb{1}_{C_k}.$$

On en déduit que $H_k' L H_k = \frac{cut(C_k, \bar{C_k})}{|C_k|} = (H'LH)_{kk}$. et finalement que le problème d'optimisation du ratio de la coupe est équivalent au problème suivant (NP-complet) :

$$\arg\min_{C_1,...,C_K} \left\{ tr(H'LH) \right\} \text{ s.c. } H_k = \frac{1}{\sqrt{|C_k|}} \mathbb{1}_{C_k},$$

Les colonnes de H sont orthogonales d'où $H'H = I$. Une formulation relaxée du problème consiste à rechercher une matrice H orthogonale qui soit solution de :

$$\arg\min_{H \in \mathbb{R}^{p \times K}} \left\{ tr(H'LH) \right\} \text{ s.c. } H'H = I$$

Le théorème de Rayleigh-Ritz montre que la matrice H, solution de ce problème, est la matrice des K premier vecteurs propres de la matrice laplacienne L. La matrice solution du problème est continue et il faut, par l'algorithme des k-moyennes par exemple, la transformer en une partition discrète des sommets du graphe pour obtenir les communautés. Cette version relaxée du problème du ratio de coupe est équivalent à celui du partitionnement des sommets du graphe par la méthode spectrale défnie à partir de la matrice laplacienne L du graphe.

Le partitionnement par la minimisation du ratio de coupe cherche à minimiser le poids des liens entre les communautés mais ne prend pas en compte le poids des liens à l'intérieur des communautés. Un autre problème d'optimisation, appelé problème de la N-coupe (Ncut), a pour objectif de mimiser les liens entre les commmunautés tout en cherchant à maximiser le poids des liens à l'intérieur des communautés. Le problème de la N-coupe s'écrit :

$$\arg\min_{C_1,...,C_K} \left\{ \sum_{k=1}^{K} \frac{cut(C_k, \bar{C_k})}{vol(C_k)} \right\},$$

où $vol(C_k)$ est le volume de la k-ième communauté qui est égal à la somme des degrés dans cette communauté : $vol(C_k) = \sum_{i \in C_k} d_i$. Ce problème d'optimisation intègre un niveau d'information supplémentaire par rapport au précédent et il est plus adapté dans le cas de graphes sur lesquels la densité à l'intérieur des communautés n'est pas homogène. Cette hypothèse est plus réaliste dans le contexte des réseaux réels.

En normalisant la matrice laplacienne du graphe, il est possible de rapprocher le problème de la N-coupe à celui du partitionnement par une méthode spectrale. Deux versions normalisées de la matrice laplacienne ont été proposées :

Definition

$$L_{\text{sym}} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} W D^{-1/2}$$
$$L_{\text{rw}} = D^{-1} L = I - D^{-1} W$$

On définit les colonnes de la matrice H de dimensions $p \times K$ à partir des vecteurs indicatrice des communautés, et on pondère par le volume à l'intérieur des communautés, la k-ième colonne de H s'écrit :

$$H_k = \frac{1}{vol(C_k)} \mathbb{1}_{C_k}.$$

Le problème de la N-coupe est équivalent au problème NP-complet suivant :

$$\arg\min_{C_1,\ldots,C_K} \{tr(H'LH)\} \text{ s.c. } H_k = \frac{1}{vol(C_k)} \mathbb{1}_{C_k}.$$

Avec cette définition de H, on a $H'DH = I$ où D est la matrice diagonale des degrés. Une version relaxée du problème de la N-coupe s'écrit alors :

$$\arg\min_{H \in \mathbb{R}^{p \times K}} \{tr(H'LH)\} \text{ s.c. } H'DH = I,$$

ou de façon équivalente, en remplaçant H par une matrice orthogonale $H^* = D^{1/2}H$:

$$\arg\min_{H \in \mathbb{R}^{p \times K}} \left\{tr(H^{*\prime} D^{-1/2} L D^{-1/2} H^*)\right\} \text{ s.c. } H^{*\prime} H^* = I$$

Les solutions du problème (Rayleigh-Ritz) sont les K premiers vecteurs propres de la matrice laplacienne normalisée $L_{sym} = D^{-1/2} L D^{-1/2}$ ou ceux de la matrice laplacienne normalisée $L_{rw} = D^{-1} L$ (si u est un vecteur propre de L_{sym} associé à la valeur propre λ, alors $D^{-1/2} L D^{-1/2} u = \lambda u$ ce qui équivaut à $D^{-1} L D^{-1/2} u = \lambda D^{-1/2} u$, d'où $v = D^{-1/2} u$ est vecteur propre de L_{rw} associé à la valeur propre λ). Le partitionnement des sommets du graphe à partir des premiers vecteurs propres de l'une des deux matrices laplaciennes normalisées est semblable a une version relaxée du problème de la N-coupe.

Les propriétés spectrales des matrices laplaciennes normalisées L_{sym} et L_{rw} sont semblables à celles de la matrice laplacienne L. Les matrices L_{sym} et L_{rw} sont symétriques et semi-définies positives, elles possèdent p valeurs propres réelles $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_p$, zéro est toujours une valeur propre et peut être associée au vecteur propre constant $\mathbf{1}_p$ pour la matrice L_{rw}, et au vecteur propre $D^{1/2}\mathbf{1}_p$ pour la matrice L_{sym}. La propriété que nous avons montré pour L et qui est au coeur du problème de détection des communauté par une méthode spectrale, peut être montrée (de façon similaire) pour L_{sym} et L_{rw}.

Propriété 3 *L'ordre de multiplicité de la valeur propre* 0 *d'une matrice laplacienne* L_{sym} *ou* L_{rw} *est égal au nombre* K *de composantes connexes de* G*. De plus, l'espace propre associé à la valeur propre* 0 *est engendré, pour* L_{rw}*, par l'ensemble des vecteurs indicatrice* $\mathbb{1}_{C_1}, ..., \mathbb{1}_{C_K}$ *pointant sur les différentes composantes connexes, et pour* L_{sym}*, par les vecteurs indicatrice* $D^{1/2}\mathbb{1}_{C_1}, ..., D^{1/2}\mathbb{1}_{C_K}$*.*

Entre les deux versions de normalisation de la matrice laplacienne, il est conseillé de choisir L_{rw} car les premiers vecteurs propres sont proches des vecteurs indicatrices pointant sur les communautés, tandis qu'avec L_{sym} les vecteurs indicatrices sont multipliés par $D^{1/2}$, ce qui peut introduire un biais et rendre la détection des communautés par l'algorithme des k-moyennes plus délicate.

En résumé, la détection de communautés par une méthode spectrale consiste, à identifier les vecteurs propres $u_1, ..., u_K$ de $\mathbb{R}^p$ associés au plus petites valeurs propres de la matrice laplacienne du graphe, à associer à chaque sommet i le vecteur $y_i = (u_{1,i}, ..., u_{K,i})'$ de ses coordonnées sur l'espace propre engendré par les K valeurs propres, et avec la méthode des k-moyennes, à partitionner en K classes les p vecteurs y_i. Les versions normalisées de la matrice laplacienne sont plus adaptées (car les densités à l'intérieur des communautés ne sont pas nécessairement homogènes), et il est préférable d'utiliser la matrice L_{rw} plutôt que L_{sym} (les premiers vecteurs propres de L_{rw} sont proches des indicatrices des communautés).

Algorithm 1 Normalized spectral clustering algorithm [56]

1: **Input** : weighted adjacency matrix $W \in \mathbb{R}^{p \times p}$, number K of clusters to construct
2: Compute the normalized Laplacian L_{rw}.
3: Compute the first K eigenvectors $u_1, ..., u_K$ of L_{rw} (smallest eigenvalues).
4: For $i = 1, ..., p$, let $y_i = (u_{1,i}, u_{2,i}, ..., u_{K,i})'$ in $\mathbb{R}^p$, be the vector corresponding to the coordinates of i-th nodes in the eigenspace of L_{rw}.
5: Partition the p observations $(y_i)_{i=1,...,p}$ with the k-means algorithm into K clusters $Y_1, ..., Y_K$
6: **Output** : Clusters $C_1, ..., C_K$ with $C_k = \{i \mid y_i \in Y_k\}$

A noter que le point 3 de l'algorithme (calcul des vecteurs propres) peut être remplacé par le problème aux valeurs propres généralisé qui consiste à trouver les vecteurs propres verifiant $Lu = \lambda Du$ où L est la matrice laplacienne non normalisée.

Le nombre K de communautés est un paramètre à fixer au départ et cela peut s'avérer être une tâche difficile. La méthode spectrale a d'autant plus de chances d'être efficace que le trou spectral de la matrice laplacienne est important. Pour fixer K, une approche naturelle dérivant de cette propriété, consiste à parcourir les valeurs propres de la matrice laplacienne dans l'ordre croissant de façon à identifier un saut dans leur évolution. Si les premières valeurs propres $\lambda_1, \lambda_2, ..., \lambda_k$ restent relativement proches et petites et que l'on observe une nette augmentation entre la valeur propre λ_k et λ_{k+1}, il y a fort à penser qu'il y a k communautés sur le graphe. En pratique, il est cependant difficile d'utiliser cette heuristique. car dès que les frontières entre les communautés ne sont pas très franches, le saut entre λ_K et λ_{K+1} ne l'est pas plus. D'autres alternatives reposent sur des critères de qualité d'une partition et sur la comparaison de plusieurs partitions obtenues pour différentes valeurs de K. Nous reviendrons par la suite sur la définition de ces critères.

Il n'y a en théorie pas de présupposé particulier à faire pour garantir l'efficacité de la méthode spectrale, et celle-ci peut aussi bien être appliquée sur un graphe complet que sur un graphe plus creux. En pratique, cependant, le calcul des vecteurs propres d'une matrice représentant un « gros » graphe est trop coûteux en temps, en $O(p^3)$. Il existes des méthodes d'approximation des vecteurs propres pour traiter les matrices creuses de grandes tailles. L'une d'elles est la méthode de Lanczos [25] (projection sur un sous-espace de Krylov), qui a une vitesse de convergence d'autant plus grande que le trou spectral est important. Pour l'analyse d'un très grand réseau, il est alors nécessaire d'estimer un graphe qui soit suffisament creux.

2.5.3 L'approche WGCNA

L'approche WGCNA pour Weighted Gene Co-expression Network Analysis a comme son nom l'indique, été proposée par Zhang et Horvath [70] pour la détection de communautés sur les réseaux de co-expression de gènes spécialement. Elle intègre des solutions pour répondre aux différentes problématiques posées à chaque étape de l'analyse : pour la modélisation du graphe à partir des données d'expression, pour la classification hiérarchique des sommets du graphe et pour l'extraction des communautés sur l'arbre de classification (dendrogramme). Comme précédemment, nous faisons l'hypothèse que les communautés ne se chevauchent pas (disjointes). A la différence de l'approche spectrale, il sera possible avec l'approche WGCNA d'exclure certains éléments si ils sont trop éloignés (faible co-expression) des communautés, et dans ce cas, les communautés ne couvrent pas l'ensemble des éléments.

Modélisation du réseau de co-expression par un graphe invariant d'échelle La démarche pour la modélisation du réseau de co-expression est a peine différente de celle que nous avons présentée dans la section 2.4. La première étape consiste à définir une matrice d'association S entre les p gènes (S est symétrique, de dimensions $p \times p$ et à coefficients dans $\mathbb{R}^+$). Les s_{ij} caractérisent la mesure de la co-expression pour tout couple de gènes, et classiquement, ils sont définis par la valeur absolue du coefficient de corrélation, de Pearson ou de Spearman, entre les profils d'expression des gènes.

Zhang et Horvath propose de modéliser le réseau, non pas à partir d'un graphe de λ-voisinage ou de K-voisinage, mais par un graphe complet sur lequel les poids w_{ij} des arêtes, sont liés aux mesures d'association s_{ij} par une fonction non linéaire, de façon à donner plus de poids aux plus fortes associations, et à très peu considérer celles qui sont les plus faibles. En d'autres termes, il ne s'agit pas de seuiller globalement ou localement les valeurs d'association, mais de les transformer afin d'accentuer l'écart entre les plus grandes et plus petites valeurs. Pour tout $\beta \in \mathbb{N}^*$, les coefficients w_{ij} de la matrice d'adjacence W du graphe sont définis par :

$$w_{ij} = s_{ij}^{\beta}, \forall i \neq j \text{ et } w_{ii} = 0, \forall i.$$

La mesure de co-expression s_{ij} est dans l'intervalle $[0, 1]$ (valeur absolue des corrélations). Ainsi, pour un paramètre $\beta > 1$, les poids w_{ij} des liens du graphe sont dans $[0, 1]$ et convergent plus rapidement vers 0 que les mesures de co-expression s_{ij}.

L'idée est de construire un graphe complet ayant la propriété d'invariance d'échelle, c'est à dire un graphe sur lequel la majorité des sommets ont des degrés faibles et seulement quelques sommets présentent des degrés élevés. Le paramètre β est choisi de façon à obtenir un graphe sur lequel la distribution des degrés est proche de celle d'une loi de puissance : $P_k \sim k^{-\gamma}$ où

P_k est la probabilité qu'un sommet soit de degré k. La validité de cette hypothèse peut être verifiée par la qualité d'ajustement, au sens du R^2 (variance expliquée par le modèle sur variance à expliquer), du modèle de régression linéaire simple qui tente d'expliquer la variable $log(P_k)$ par la variable $log(k)$. En pratique, pour W obtenue pour un β donné, on fixe L ($L = 10$, valeur proposée par les auteurs) intervalles $[a_l, a_{l+1}[$ de même longueur et tels que $\min_{i=1,2,...,p}\{d_i\} \in [a_1, a_2[$ et $\max_{i=1,2,...,p}\{d_i\} \in [a_L, a_{L+1}[$ où les $d_i = \sum_j w_{ij} = \sum_{j\neq i} s_{ij}^{\beta}$ sont les degrés des sommets. On calcule pour tout $l = 1, 2, ..., L$, la probabilité p_l qu'un sommet soit de degré dans l'intervalle $[a_l, a_{l+1}[$ i.e. $p_l = \frac{1}{p}\sum_i \mathbb{1}_{d_i \in [a_l, a_{l+1}[}$, et on ajuste un modèle de régression linéaire simple pour expliquer le vecteur d'observation $\mathbf{y} = (log(p_1), log(p_2), ..., log(p_L))' \in \mathbb{R}^L$ par le vecteur d'observation $\mathbf{x} = \left(log(\frac{a_1+a_2}{2}), log(\frac{a_2+a_3}{2}), ..., log(\frac{a_L+a_{L+1}}{2})\right)' \in \mathbb{R}^L$. Si l'hypothèse d'invariance d'échelle se vérifie sur le graphe, on s'attend à avoir un modèle de qualité avec un $R^2 = \frac{var(\hat{\mathbf{y}})}{var(\mathbf{y})}$ assez proche de 1 ($R^2 \in [0, 1]$). Parmi les valeurs de β qui sont satisfaisantes suivant le critère d'invariance d'échelle, c'est la plus petite d'entre-elles qui est retenue, de façon à conserver un maximum d'information sur le graphe. Il faut trouver un bon compromis entre un R^2 le plus proche de 1 possible et un degré de connectivité moyen le plus élevé possible.

Définition d'une mesure de dissimilarité entre sommets du graphe L'information portée par la matrice d'adjacence W nous renseigne (adjacence et poids des liens) sur les interactions entre les sommets pris deux à deux, c'est à dire sur les interactions directes. L'intérêt de modéliser un graphe pour l'identification de communautés est de replacer les éléments dans un système et de caractériser la proximité ou similarité entre les éléments en se basant non plus uniquement sur les interactions directes, mais en intégrant de l'information plus générale sur le positionnement des éléments dans le système. Par exemple, on peut considérer que deux sommets sont d'autant plus proches ou similaires qu'ils sont en forte interaction et qu'ils interagissent avec les mêmes autres sommets. L'information sur la ressemblance des voisinages des deux sommets est ainsi ajoutée à celle portant sur leur interaction directe. Ce type de mesure de similarité reposant sur la notion de chevauchement des voisinages fait sens dans le contexte des réseaux biologiques dans la mesure où deux éléments (deux gènes par exemple) qui partagent la même fonction au niveau cellulaire vont très certainement interagir avec les mêmes autres éléments [52].

Zhang et Horvath propose d'utiliser la mesure de similarité introduite par Ravasz et al. [52] dans le cas des graphes non pondérés en la généralisant au cas des graphes pondérés. La similarité $s_{i,j}^G \in [0, 1]$ entre deux sommets i et j d'un graphe G de matrice d'adjacence $W = (w_{ij})_{i,j}$ est définie par :

$$s_{i,j}^G = \frac{w_{ij} + \sum_k w_{ik}w_{jk}}{\min(d_i, d_j) + 1 - w_{ij}}, \forall i \neq j \text{ et } s_{ii}^G = 1, \forall i$$

où $d_i = \sum_j w_{ij}$ est le degré du sommet i. Deux sommets sont d'autant plus similaires (s_{ij} proche de 1) qu'ils ont des liens de poids élevés avec les mêmes autres sommets et qu'ils sont eux-mêmes connectés par un lien de poids élevé. Une mesure de dissimilarité d_{ij}^G entre tout couple de sommets du graphe se déduit directement de la mesure de similarité :

$$d_{ij}^G = 1 - s_{i,j}^G.$$

Classification des sommets du graphe La méthode de détection de communautés avec l'approche WGCNA consiste à faire de la classification hiérarchique ascendante (CAH) sur la

matrice des dissimilarités $D^G = (d^G_{ij})_{i,j}$. La CAH est une méthode de classification itérative qui fournit une hiérarchie de partition en créant à chaque étape une nouvelle partition obtenue par agrégation des deux éléments (sommets ou groupes de sommets) les plus proches sur la partition précédente, c'est à dire les deux éléments qui ont la plus faible valeur de dissimilarité. Les objets à partitionner sont ici les sommets du graphe.

Au départ, chaque sommet est isolé dans une classe et à l'étape suivante, les sommets ayant la plus faible dissimilarité sont regroupés dans une même classe. Pour les étapes suivantes, il est nécessaire de définir une méthode d'agrégation afin de pouvoir mesurer la dissimilarité entre un sommet et une classe de sommets. La dissimilarité entre deux classes C_1 et C_2 de sommets (ou entre une classe de sommet C_1 et un sommet isolé $C_2 = \{j\}$, $|C_2| = 1$) est définie dans l'approche WGCNA par la moyenne des dissimilarités entre les éléments des deux classes (average linkage) :

$$d^G_{C_1C_2} = \frac{1}{|C_1||C_2|} \sum_{i \in C_1, j \in C_2} d^G_{ij}.$$

A chaque itération de la CAH, la dissimilarité entre les différents éléments est calculée et les deux éléments (classes ou sommets isolés) qui ont la mesure de dissimilarité la plus faible sont regroupés. A la fin de l'algorithme l'ensemble des sommets sont regroupés dans une seule et même classe. Les partitions obtenues pour les différents niveaux d'agrégation (à chaque itération) peuvent être visualisées sur l'arbre de classification appelé dendrogramme.

Quand le nombre d'objets à partitionner est très important, il n'est pas toujours facile de choisir une partition parmi l'ensemble de celles obtenues car la lisibilité du dendrogramme n'est pas garantie. Choisir une partition revient communément à déterminer une hauteur constante à laquelle le dendrogramme est coupé. Le choix de cette hauteur reste à l'appréciation de l'utilisateur par l'observation de l'allure du dendrogramme. En coupant l'arbre à une hauteur constante, il n'est cependant pas possible d'extraire des classes entre lesquelles et à l'intérieur desquelles les dissimilarités peuvent être très hétérogènes. Un autre inconvénient de cette approche est qu'il n'est pas toujours facile pour l'utilisateur de choisir une hauteur de coupe. Quand le dendrogramme est très grand, il devient difficile d'apprécier correctement l'évolution des partitions sur celui-ci.

L'approche WGCNA intègre deux algorithmes différents pour identifier automatiquement les classes sur le dendrogramme et permettre la détection de classes pouvant apparaître à des hauteurs d'agrégation variables. [35, 36].

L'algorithme appelé « Dynamic Tree » analyse l'évolution de la séquence des hauteurs d'agrégation (dissimilarité), $\mathcal{H} = (h_1, h_2, ..., h_p)$, pour tous les éléments rangés dans le même ordre que celui du dendrogramme. La séquence $\mathcal{H}$ est normalisée par une valeur l : $\mathcal{H}^* = \mathcal{H} - l$. Sur $\mathcal{H}$, toutes les hauteurs sont positives tandis que sur $\mathcal{H}^*$, les h^*_i peuvent être négatifs pour les valeurs les plus faibles de h_i. La première valeur choisie pour le paramètre l est la hauteur moyenne des éléments : $l_m = \frac{1}{p}\sum_{i=1}^{p} h_i$. Si aucune classe n'a pu être détectée pour ce paramètre on augmente la valeur progressivement, $l = \frac{1}{2}[l_m + \min(\mathcal{H})]$, puis $l = \frac{1}{2}[l_m + \max(\mathcal{H})]$. La séquence $\mathcal{H}^*$ est analysée de façon à identifier les transitions entre valeurs positives et négatives. Si l'on observe une suite $h^*_{i_1}, h^*_{i_2}, ..., h^*_{i_k}$ d'éléments consécutifs de $\mathcal{H}^*$ qui prennent tous des valeurs négatives, c'est à dire que les dissimilarités entre éléments sont faibles, et que l'élément suivant $h^*_{i_{k+1}}$ est positif (dissimilarité plus importante), les éléments $h^*_{i_1}, h^*_{i_2}, ..., h^*_{i_k}$ appartiennent à la même classe

et l'élément $h^*_{i_{k+1}}$ est considéré comme le premier élément d'une nouvelle classe. Des contraintes sur la taille des différentes phases (phases positives et négatives) de la séquence $\mathcal{H}^*$ sont ajoutées pour permettre à l'utilisateur de contrôler l'espacement minimum (taille de la phase en début de classe sur laquelle les valeurs sont positives) entre deux classes et la taille minimale des classes. Chaque classe de sommets détectées par l'algorithme est ensuite analysée de façon indépendante. Une classe est associée à un sous-arbre du dendrogramme, et en réitérant l'algorithme sur chacun de ces sous-arbres, les classes peuvent être subdivisées en plusieurs classes. Le processus est réitéré jusqu'à ce que toutes les classes ne puissent plus être subdivisées.

L'autre algorithme, appelé « Dynamic Hybrid », analyse le dendrogramme de bas en haut. Un ensemble d'éléments sur une branche de l'arbre forme une classe si il répond à différentes conditions. Chaque classe doit avoir une taille minimale, contenir un sous-ensemble d'éléments fortement similaires (centres des classes) qui soit de taille suffisante. Les éléments au centre d'une classe doivent être suffisamment éloignés (dissimilaires) de ceux qui sont en périphérie, c'est à dire, de ceux qui ont les plus forts degrés de dissimilarité avec la classe. Ce deuxième algorithme est plus flexible que le premier mais il nécessite de fixer un plus grand nombre de paramètres et il est plus sensible au choix de ces paramètres.

Avec ces deux algorithmes, les éléments qui s'agrègent à des hauteurs excédant un seuil fixé sont considérer comme étant du bruit et sont exclus de l'analyse.

2.5.4 Un exemple d'application

Les contraintes observées Nous avons testé les approches WGCNA et spectrale sur différents jeux de données d'expression réels, ce qui nous a permis d'éprouver les limites de ces deux approches pour l'analyse des réseaux de co-expression de gènes. Les données d'expression sont très volumineuses (plusieurs milliers de gènes) et peuvent contenir une très grande quantité de dimensions non informatives dans le sens où les gènes sur ces dimensions interagissent très peu avec les autres gènes.

Avec l'approche spectrale, le calcul des éléments propres de la matrice laplacienne qui représente un grand graphe (beaucoup de sommets) peut s'avérer être trop coûteux en temps si la matrice, et donc le graphe, ne sont pas suffisamment creux. Dans ce cas, il est possible d'estimer un graphe creux par λ ou K voisinage, mais le problème du choix de la valeur du paramètre reste difficile et peut largement influencer le résultat final (communautés détectées). Un autre inconvénient de l'approche spectrale est sa sensibilité en présence de dimensions non informatives qui ne peuvent être distinguées de celles qui sont informatives. La nécessité de fixer a priori le nombre de communautés recherchées constitue également un inconvénient majeur.

Avec l'approche WGCNA, il est difficile de détecter des communautés, entre lesquelles et à l'intérieur desquelles les densités sont hétérogènes, ce qui est en pratique un cas très courant. Les poids des liens entre les gènes sont estimés par les mesures de co-expression élevées à la puissance β, mais le choix d'une unique valeur pour le paramètre β n'est pas justifié quand il existe des zones de densité hétérogènes sur le graphe. Une autre difficulté est celle de l'identification des communautés à partir de l'arbre de classification (dendrogramme). Des algorithmes ont été proposés pour détecter automatiquement les communautés et exclure de l'analyse les dimensions non informatives, mais ils nécessitent de fixer un certain nombre de paramètres et sont sensibles au choix de ces paramètres.

Un exemple

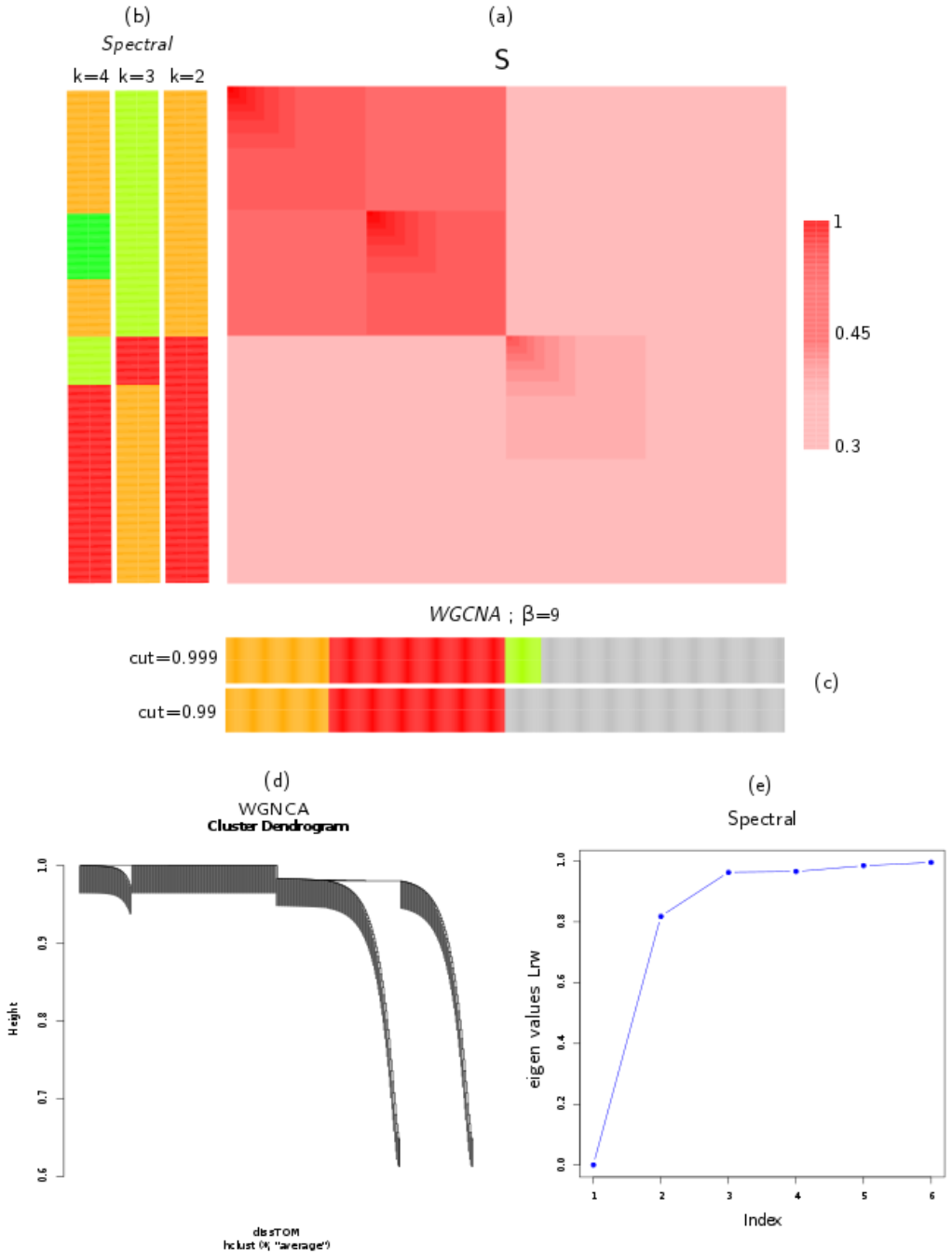


FIGURE 2.2 – (a) Représentation (heatmap) de la matrice S. (b) Représentation des classes formées avec l'approche spectrale pour un partitionnement en 2, 3 et 4 classes. (c) Représentation des classes formées avec l'approche WGCNA en coupant l'arbre avec l'algorithme « Dynamic Tree » pour une hauteur maximale d'agrégation de 0.99 et de 0.999. La couleur grise caractérise les éléments exclus de l'analyse (non informatifs). (d) Dendrograme obtenu avec l'approche WGCNA. (e) Représentation des 6 plus petites valeurs propres de L_{rw}.

Nous avons construit un exemple simple de matrice d'association symétrique et définie par blocs, telle que, les mesures d'associations à l'intérieur et entre les communautés soient hétérogènes et qu'un certain nombre d'éléments soient non informatifs :

$$S = \begin{pmatrix} S_1 & S_{12} & S_{13} & S_{14} \\ S_{21} & S_2 & S_{23} & S_{24} \\ S_{31} & S_{32} & S_3 & S_{34} \\ S_{41} & S_{42} & S_{43} & S_4 \end{pmatrix}.$$

Chaque bloc est de dimensions 100×100. On crée trois communautés composées chacune de 100 éléments et un groupe supplémentaire de 100 éléments non informatifs. Les mesures d'association à l'intérieur des communautés sont représentées dans les blocs diagonaux S_1, S_2 et S_3.

Chacun de ces blocs est défini de façon à faire apparaître un petit groupe d'éléments entre lesquels les mesures d'association sont largement plus elevées que la valeur moyenne des associations dans le bloc, de façon à mimer la propriété d'invariance d'échelle des réseaux d'interaction. Les coefficients $s_{ij}^{(k)}$ à l'intérieur du bloc S_k sont définis pour tout $k = 1, 2, 3$ et pour tout $i, j \in \{1, 2, ..., 100\}$ par $s_{ij}^{(k)} = \min(\alpha_i^{(k)}, \alpha_j^{(k)})$ si $i \neq j$, où $\alpha_i^{(k)} = r_k + 0.4 \times \left(1 - \frac{i}{100}\right)^3$, et par $s_{ij}^{(1)} = 1$ si $i = j$.

En fixant $r_1 = r_2 = 0.61$ et $r_3 = 0.31$, les mesures d'association à l'intérieur du bloc S_3 sont plus faibles que celles dans les blocs S_1 et S_2. Les blocs extra diagonaux sont définis par des matrices constantes : les coefficients des blocs $S_{23} = S_{32}$ sont fixés à 0.6, et ceux des autres blocs extra diagonaux à 0.3. Pour le bloc d'éléments non informatifs S_4 on fixe les coefficients extra diagonaux à 0.3 et ceux sur la diagonale à 1. La matrice S de dimension 400×400 ainsi définie, est représentée sur la Figure 2.2.

Pour tester l'approche spectrale, nous avons construit un graphe complet de matrice d'adjacence W : $w_{ij} = s_{ij}$ pour tout $i \neq j$ et $w_{ii} = 0$ pour tout i, et nous avons partitionné l'ensemble des sommets du graphe en 2, 3 et 4 classes à partir des vecteurs propres de L_{rw}. Pour éviter que l'algorithme des k-moyennes ne converge vers un optimum local, nous l'avons lancé 100 fois avec des initialisations différentes et retenu le meilleur regroupement.

Pour l'approche WGCNA, les coefficients de la matrice d'adjacence du graphe ont été définis par les mesures d'associations élevées à la puissance β : $w_{ij} = s_{ij}^{\beta}$ pour tout $i \neq j$ et $w_{ii} = 0$ pour tout i. Le paramètre $\beta = 9$ a été choisi selon les critères proposés par les auteurs (distribution des degrés proche de celle d'une loi de puissance et degré moyen suffisamment élevé). L'algorithme « Dynamic Tree » a été utilisé pour extraire les classes sur le dendrogramme en testant deux valeurs (0.99 et 0.999) pour le paramètre de la hauteur maximale d'agrégation.

Les résultats obtenus avec les deux méthodes sont représentés sur la Figure 2.2. L'approche spectrale ne permet pas d'identifier directement les éléments non informatifs. Pour un partitionnement en 2 ou 3 classes, les deux premières communautés les plus denses (intra et inter) sont regroupées dans une seule communauté. Il faut partitionner en 4 classes pour avoir une classe d'éléments par communauté, avec quand même des erreurs de classification : les éléments qui ont les mesures d'association les plus faibles dans la deuxième communauté (resp. dans la troisième communauté) sont classés avec les éléments de la première communauté (resp. avec les éléments non informatifs). A noter, également, que l'heuristique du trou spectral suggère de ne pas partitionner l'ensemble des sommets (trou spectral entre la valeur propre 1 et 2), ou alors en 2

classes seulement (petit saut entre les valeurs propres 2 et 3). Avec l'approche WGCNA, les deux première classes les plus denses sont identifiées mais quelques éléments de la première classe sont néanmoins placés dans la deuxième. La méthode exclut les éléments non informatifs (en gris sur le graphique) mais beaucoup d'éléments (ou tous les éléments, en fonction du paramètre de la hauteur maximale d'agrégation choisi) de la communauté la moins dense sont considérés comme étant également non informatifs. Les hauteurs d'agrégations des éléments de la communauté la moins dense sont très proches de celles des éléments non informatifs (effet du choix d'un β unique pour définir le graphe).

2.6 Une nouvelle approche pour la détection de communautés

Motivations Malgré la diversité des solutions proposées pour la détection de communautés sur un graphe, aucune d'entre elles ne nous est apparue complètement satisfaisante pour analyser de façon efficace et suffisamment robuste les réseaux de co-expression de gènes. Pour modéliser ce type de réseau, les interactions entre éléments doivent être estimées, ce qui implique au départ, de faire certaines hypothèses pour obtenir un graphe qui soit une représentation pertinente du réseau réel. En fonction de la représentation choisie, les méthodes de détection de communautés pourront se montrer plus ou moins efficaces, et nécessitent elles aussi de faire des hypothèses pouvant être contraignantes. Nous proposons ici une nouvelle approche pour l'analyse de réseaux réels qui intègre l'étape d'estimation du graphe et celle de la détection des communautés en ne faisant qu'une seule hypothèse sur la taille minimale des communautés.

Idées Quelles peuvent être les caractéristiques de la structure d'interaction au sein d'une communauté de gènes ? Il est intuitif de penser que le niveau de co-expression entre gènes d'une même communauté est globalement plus important que celui entre gènes de communautés différentes. Si l'on replace les gènes sur un réseau de co-expression, on peut alors imaginer qu'une communauté est une région plus dense sur le réseau, qu'il y a plus de liens à l'intérieur des communautés qu'entre les communautés et que les poids des liens à l'intérieur des communautés sont globalement plus élevés. Le réseau peut être modélisé par un graphe de λ-voisinage défini à partir des mesures d'association. Puisqu'une communauté forme une région dense sur le réseau réel, on va chercher sur un graphe de λ-voisinage à faire apparaître la communauté, ou tout du moins certains éléments au coeur de la communauté, sous une forme idéale, c'est à dire comme composante connexe du graphe. Les niveaux de co-expression (ou densité intra) au sein des communautés ne sont pas nécessairement homogènes, et l'idée n'est pas de construire une unique représentation du réseau, mais d'envisager de multiples graphes, obtenus en faisant varier λ, de façon à ce que toutes les communautés puissent apparaître sous leur forme idéale sur au moins l'un de ces graphes. Notre but n'est pas de donner une représentation fidèle du réseau par un unique graphe, mais d'identifier des communautés en intégrant de l'information sur la structure d'interaction. On reconnecte alors les deux problématiques, à la différence des approches classiques, avec lesquelles il est nécessaire d'estimer au préalable un unique graphe ayant certaines propriétés, pour permettre la détection des communautés. On va rechercher sur une collection de graphes, des ensembles de sommets que nous appelerons des noyaux, ayant certaines propriétés structurelles.

Définition d'une collection de graphes On souhaite construire une collection de graphes de λ-voisinage à différentes échelles de représentation en faisant varier le paramètre λ.

Nous disposons au départ d'une matrice d'association S symétrique de dimensions $p \times p$ qui caractérise, par exemple, les mesures de co-expression pour un ensemble de p gènes. Soit $G = (V, E)$ le graphe de matrice d'adjacence $W = (w_{ij})_{1 \leq i,j \leq p}$ avec $w_{ij} = s_{ij}$ pour tout $i \neq j$ et $w_{ii} = 0$ pour tout i. On suppose (sans perte de généralité) que G est complet.

Nous rappelons qu'un graphe partiel d'un graphe G est un graphe qui a les mêmes sommets que G et qui s'obtient en supprimant une plusieurs arêtes sur le graphe G. Le graphe défini à partir des λ-voisinages des sommets est un graphe partiel de G (une ou plusieurs arêtes de G sont supprimées), que nous allons noter $G_\lambda = (V, E_\lambda)$, où E_λ est le sous-ensemble de l'ensemble E qui contient uniquement les liens de poids supérieur ou égal à λ. Plus généralement, nous noterons G_λ, un graphe partiel d'un graphe pondéré G, obtenu par seuillage de la matrice d'adjacence de G, où λ est le paramètre de seuillage. L'opérateur de seuillage T_λ, pour un paramètre de seuillage $\lambda \in \mathbb{R}$ donné, est défini sur l'ensemble des matrices M de $\mathbb{R}^{p \times p}$ par :

$$T_\lambda(M) = \left(m_{ij} \mathbb{1}_{\{m_{ij} \geq \lambda\}} \right)_{1 \leq i,j \leq p}.$$

La matrice d'adjacence W_λ du graphe partiel G_λ s'écrit alors :

$$W_\lambda = T_\lambda(W).$$

En faisant varier le paramètre de seuillage $\lambda_1 < \lambda_2 < \lambda_3 < ...$, on définit une collection $G_{\lambda_1}, G_{\lambda_2}, G_{\lambda_3}, ...$ de graphes partiels de G. Une collection exhaustive s'obtient en définissant tous les graphes partiels pour des valeurs de seuillages dans l'ensemble $\{w_{ij},\ i, j = 1, ..., p\}$ des coefficients de la matrice W. On peut constuire au maximum $p(p-1)$ graphes partiels de G différents, sans compter le graphe vide (sans arête) et en comptant le graphe complet défini pour $\lambda = \min_{i \neq j}\{w_{ij}\}$ de matrice d'adjacence $W_\lambda = W$.

Le graphe complet G contient de l'information plus ou moins significative et redondante sur la structure d'interaction entre les sommets. Une représentation plus synthétique de cette structure peut être construite en ciblant uniquement les chemins les plus fiables sur le graphe.

Definition (capacité d'un chemin) Soit $G = (V, E)$ un graphe simple et pondéré de matrice d'adjacence W. Un chemin p de G est une suite de sommets $(i_1, i_2, ..., i_L) \in V^L$ adjacents, c'est à dire que $(i_k, i_{k+1}) \in E$ pour tout $k = 1, 2, ..., L-1$. La capacité $c(p)$ d'un chemin $p = (i_1, i_2, ..., i_L)$ est le poids minimum des liens traversés par le chemin :

$$c(p) = \min_{k=1,2,...,L-1} \{w_{i_k i_{k+1}}\}.$$

Un chemin est de forte capacité si toutes les mesures d'association (ou d'interaction) entre deux sommets consécutifs sur ce chemin sont élevées. Le problème de l'identification des chemins de capacité maximum sur un graphe a été posé par Pollack en 1960 [49], et Hu [30] a montré que sur un graphe non orienté, la solution à ce problème s'obtient simplement en construisant l'arbre couvrant du graphe de poids maximal (maximum spanning tree en anglais).

Definition Un arbre couvrant $A = (V, E^A)$ d'un graphe $G = (V, E)$ connexe est tel que :

- le graphe A est un arbre c'est à dire un graphe connexe sans cycle.
- le graphe A est un graphe partiel de G ($E^A \subset E$).

Un arbre couvrant de G de poids maximum est un arbre couvrant dont la somme des poids des liens est maximale.

Un arbre couvrant de poids maximum est unique si et seulement si les poids des liens sont deux à deux distincts. L'arbre couvrant de G de poids maximum est la représentation la plus simple des relations entre sommets que l'on puisse obtenir (seulement $p-1$ liens), mais c'est aussi la plus fiable en terme d'information.

Les deux algorithmes les plus célèbres pour trouver un tel arbre sont les algorithmes de Prim [51] et de Kruskal [33]. Le premier consiste à déterminer l'arbre en construisant un sous-graphe partiel connexe qui grossit petit à petit. On part d'un sommet arbritrairement choisi et on le connecte au sommet avec lequel il est le plus fortement associé (poids du lien maximum). Le premier sous-graphe partiel contient alors deux sommets adjacents. A chaque étape, on identifie sur le graphe le lien de poids le plus élevé reliant un sommet qui est sur le sous-graphe partiel à un sommet qui n'est pas sur le sous-graphe partiel, et on ajoute le sommet et le lien correspondant dans le sous-graphe. On réitère jusqu'à ce que l'ensemble des sommets du graphe soient sur le sous-graphe, qui correspond alors à l'arbre couvrant de poids maximum. La complexité en temps pour l'implémentation la plus simple de l'algorithme est en $\mathcal{O}(p^2)$.

Avec l'algorithme de Kruskal, l'arbre couvrant de poids maximum est construit en ajoutant pas à pas des liens sur un graphe partiel de façon à ce qu'il n'y ait pas de cycle sur le graphe partiel. Le premier graphe partiel est le graphe vide composé de tous les sommets du graphe et sur lequel il n'y a aucun lien entre les sommets. Dans un premier temps, on établit une liste ordonnée des liens du graphe, suivant l'ordre croissant des poids des liens. A chaque itération, le premier lien de la liste (de poids maximum dans la liste) est supprimé de la liste et ajouté au graphe partiel uniquement si il ne fait pas apparaître de cycle sur le graphe partiel. Le processus est réitéré jusqu'à ce que la liste des liens soit vide, et le graphe partiel à la fin du processus correspond à l'arbre couvrant de poids maximum. L'algorithme peut être implémenté en $\mathcal{O}(m \times log(p))$.

L'algorithme de Prim est beaucoup plus rapide que celui de Kruskall quand le graphe est très dense et donc plus adapté dans le contexte de l'analyse d'un réseau de co-expression de gènes (le graphe G défini à partir d'une matrice d'association non seuillée est un graphe complet).

Algorithm 2 Prim's algorithm [56]

1: **Input** : weighted undirected graph
2: Choose a single node arbitrarily from the graph and initialize the tree with this vertex
3: Find the maximum weight edge that connect a node in the tree to a node not yet in the tree, and add the corresponding edge and node to the tree (one edge is chosen arbitrarily in cases of equality).
4: Repeat the previous step until all nodes are in the tree
5: **Output** : maximum spanning tree for the graph

Les composantes connexes des graphes de la collection Nous allons par la suite nous intéresser à la détection de composantes connexes dans les graphes G_λ. On rappelle qu'une

composante connexe $G^C = (C, E^C)$ d'un graphe $G = (V, E)$ est un sous-graphe connexe maximal de G. Le problème de l'identification des composantes connexe du graphe est directement lié à celui de la recherche des chemins de capacité maximum. En effet, deux sommets i et j sont dans la même composante connexe de G_λ si et seulement si, il existe sur G un chemin p reliant i et j qui soit de capacité $c(p) \geq \lambda$, ou de façon équivalente, si et seulement si, $\max_{p \in \mathcal{P}_{ij}} c(p) \geq \lambda$, où $\mathcal{P}_{ij}$ est l'ensemble des chemins reliant i et j sur G. On peut alors montrer qu'il suffit d'identifier les composantes connexes de la collection des graphes partielles A_λ de l'arbre couvrant de poids maximum de G, pour en déduire celles de la collection de graphes partielles G_λ de G.

Proposition 4 *Soit $G = (V, E)$ un graphe simple, pondéré et connexe, de matrice d'adjacence W, et $G_\lambda = (V, E_\lambda)$ un graphe partiel de G de matrice d'adjacence $W_\lambda = T_\lambda(W)$. Soit $A = (V, E^A)$ l'arbre couvrant de poids maximum de G, de matrice d'adjacence W^A, et A_λ le graphe partiel de A de matrice d'adjacence $W^A_\lambda = T_\lambda(W^A)$. Pour tout paramètre de seuillage $\lambda \in \mathbb{R}$, les assertions suivantes sont équivalentes :*

1. *$G^C = (C, E^C_\lambda)$ est une composante connexe de G_λ.*
2. *$\max_{p \in \mathcal{P}_{ij}} \{c(p)\} \geq \lambda,\ \forall i, j, \in C$, et $\max_{p \in \mathcal{P}_{ij}} \{c(p)\} < \lambda,\ \forall i \in C, \forall j \in V \setminus C$.*
3. *le sous-graphe de A_λ induit par C est une composante connexe de A_λ.*

La collection exhaustive des graphes partiels A_λ de A, un arbre couvrant de poids maximum de G, contient $p - 1$ graphes partiels (sans compter le graphe vide), et il suffit d'identifier les composantes connexes des A_λ pour en déduire celles des G_λ (induites par les mêmes ensembles de sommets).

Propriétés structurelles d'une communauté, un noyau Une communauté est une zone plus dense (plus de liens et poids des liens plus élevés) sur le réseau réel. Les zones aux frontières des communautés peuvent être plus ou moins marquées et il se peut également que les communautés se chevauchent : un gène peut avoir plusieurs fonctions biologiques. Afin de pouvoir envisager toutes ces possibilités, nous allons dans un premier temps chercher à détecter une zone dense à l'intérieur de chaque communauté, qui soit clairement identifiable sur le réseau, et que nous appellerons un noyau. Nous envisagerons ensuite plusieurs possibilités pour définir les communautés à partir de ces noyaux, la plus simple étant celle qui consistera à considérer qu'une communauté est réduite à son noyau.

La notion de noyau sur un graphe a été introduite par Seidman [55] : un k-noyau, ou noyau d'ordre k, est défini comme un sous-graphe maximal d'un graphe G à l'intérieur duquel l'ensemble des sommets ont des degrés (nombre de voisins dans le cas d'un graphe non orienté et non pondéré) au moins égal à k. Batagelj et Zaversnik [5] généralisent cette définition pour n'importe quelle fonction $f(i, C)$, $i \in V$, $C \subseteq V$, à valeur réelle : un t-noyau, avec $t \in \mathbb{R}$, est un sous-graphe maximal induit par C tel que pour tout élément i de C, on a $f(i, C) \geq t$. Par exemple, en posant $f(i, C) = \max_{j \in C}\{w_{ij}\}$, pour chaque élément à l'intérieur d'un t-noyau, le poids maximal d'un lien connectant l'élément à un autre élément du noyau est supérieur ou égal à t.

Nous allons proposer une nouvelle définition de la notion de noyau, d'une part parce que nous ne souhaitons pas nous limiter à l'analyse d'un seul graphe (on ne cherche pas à modéliser le réseau réel par un graphe optimal), et d'autre part, parce que nous souhaitons prendre en compte une éventuelle hétérogénité des densités sur la structure d'interaction, ce qui ne peut ce faire en

supposant que t est fixe. Avec notre définition, le noyau d'un graphe ne sera pas un sous-graphe mais un ensemble de sommets du graphe de façon à faire coïncider la définition avec celle d'un noyau d'une communauté.

On part du principe que les zones les plus denses sur un graphe G forment des composantes connexes sur certains des graphes partiels G_λ (ou A_λ) de G. Lorsque le paramètre de seuillage λ augmente, le nombre de composantes connexes sur les graphes G_λ augmente aussi, et leurs tailles diminuent (au minimum une composante connexe contient un unique sommet). Seule une composante connexe qui regroupe un minimum de sommet est susceptible d'être associée à un noyau d'une communauté. On définit un n-groupe central comme un ensemble composé d'au moins n sommets qui induit une composante connexe de l'un des graphes partiels G_λ de G.

Definition (n-groupe central) Soit $G = (V, E)$ un graphe pondéré. Pour tout $n \in \mathbb{N}$, un n-groupe central est un ensemble de sommets $C^n \subset V$ qui vérifie les conditions suivantes :

1. $|C^n| \geq n$.
2. il existe un $\lambda \in \mathbb{R}$ tel que le sous graphe de G_λ induit par C^n est une composante connexe de G_λ.

D'après la proposition 4, il est possible de remplacer dans la condition 2. de la définition le graphe G_λ par A_λ, ou de définir la condition à partir de la capacité maximum des chemins.

Une composante connexe d'un graphe G_{λ^*} induite par un n-groupe central C^n est nécessairement subdivisée en plusieurs composantes connexes sur l'un des graphes G_λ avec $\lambda > \lambda^*$. Deux cas de figure peuvent alors être envisagés (représentés sur la Figure 2.3) :

(cas1) La composante connexe se désagrège très progressivement à mesure que λ augmente, c'est à dire que sa taille diminue progressivement et que la composante ne peut se scinder en plusieurs composantes connexes de taille $\geq n$.

(cas2) La composante connexe se scinde au cours du processus de désintégration en plusieurs composantes connexes de taille $\geq n$.

Dans le premier cas, on ne peut pas trouver deux sous-ensembles disjoints de C^n qui sont des n-groupes centraux, tandis qu'on en trouve au moins deux dans le deuxième cas.

A l'intérieur d'une communauté, on s'attend à avoir une structure d'interaction assez homogène pour qu'elle ne puisse être scindée en sous-communautés autonomes. Ainsi, nous allons considérer qu'un n-groupe central C est le noyau d'une communauté si la composante connexe induite par C sur G_{λ^*} se désagrège progressivement sur les graphes G_λ, $\lambda > \lambda^*$ (cas1).

Definition (n-noyau) Soit $G = (V, E)$ un graphe pondéré. Pour tout $n \in \mathbb{N}$, un n-noyau est un sous-ensemble maximal $Q^n \subset V$ qui vérifie les conditions suivantes :

1. Q^n est un n-groupe central de G.
2. Q^n ne peut contenir deux sous-ensembles disjoints qui soient des n-groupes centraux. Autrement dit, si C_1 et C_2 sont des n-groupes centraux et sont inclus dans Q^n, alors $C_1 \subseteq C_2$ ou $C_2 \subseteq C_1$.

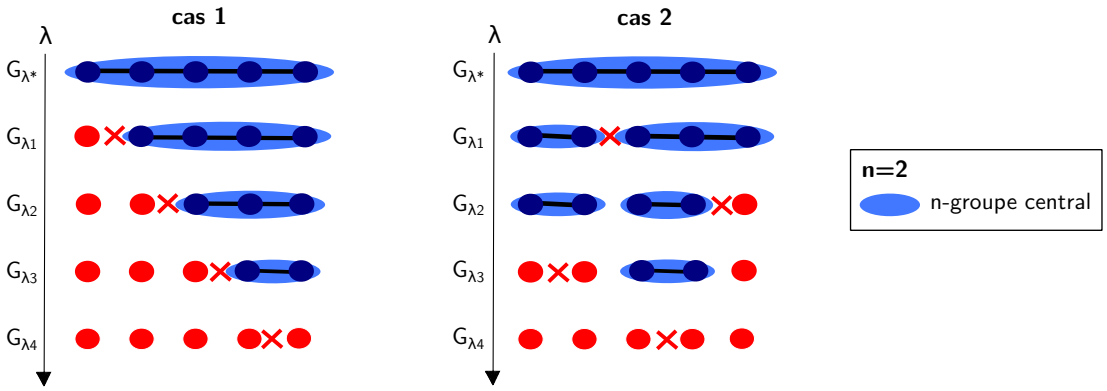


FIGURE 2.3 – Processus de désintégration d'une composante connexe d'un sous-graphe G_{λ^*} induite par un n-groupe central. Cas1 : désintégration progressive avec au maximum une composante connexe de taille $\geq n$. Cas2 : subdivision en plusieurs composantes connexes de taille $\geq n$

Le processus d'identification des n-noyaux d'un graphe (ou d'une communauté) est représenté sur la Figure 2.4.

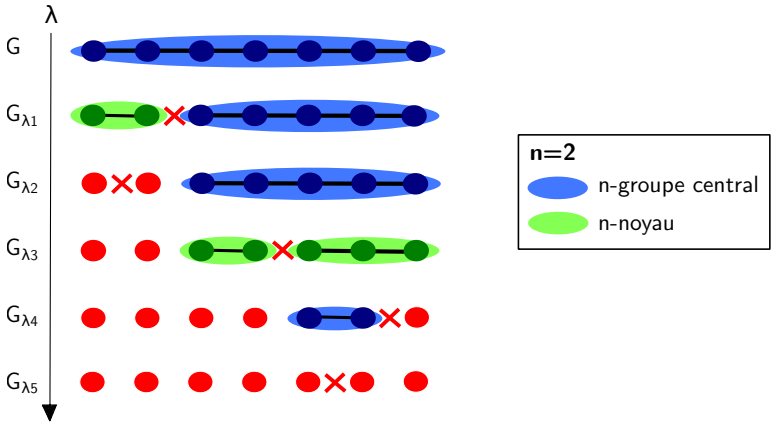


FIGURE 2.4 – Représentation du processus d'identification des noyaux d'un graphe.

L'algorithme qui permet de détecter les noyaux est un algorithme de classification hiérarchique descendante (divisive en anglais). Les liens de poids les plus faibles sont progressivement supprimés sur le graphe (ou sur l'arbre couvrant de poids maximum) et l'ensemble des sommets est subdivisé suivant la répartition des sommets dans les différentes composantes connexes du graphe. Au départ (à t_0), on suppose que le graphe est connexe, et à chaque itération le lien de poids le plus faible est supprimé (ou tous les liens de poids minimum en cas d'égalité), puis l'évolution des composantes connexes est analysée. A t_0, l'unique noyau candidat est celui qui regroupe tous les sommets. Dès que l'on obtient, à l'itération $t > t_0$, un graphe composé d'au moins deux composantes connexes de taille suffisante ($\geq n$), alors le seul noyau candidat (à t_0)

est exclu et chaque composante connexe de taille suffisante regroupe les sommets d'un nouveau noyau candidat. Ce processus est réitéré et l'analyse des subdivisions en composantes connexes est effectuée à l'intérieur de chaque sous-graphe induit par les sommets d'un noyau candidat (les autres éléments sont exclus de l'analyse). Les composantes connexes peuvent être détectées sur un graphe par un algorithme de parcours en profondeur de complexité en temps $\mathcal{O}(p+m)$. La détection des noyaux en remplaçant le graphe par l'arbre couvrant de poids maximum permet de réduire la complexité de l'algorithme de $\mathcal{O}(p^3)$ à $\mathcal{O}(p^2)$.

L'ensemble des sommets contenus dans un noyau peuvent être reliés par des chemins inclus dans le noyau qui ont une capacité plus élevée que celle de n'importe quel chemin qui relie un sommet du noyau à un sommet qui n'est pas dans le noyau (point 2. de la proposition 4). Si l'on définit une mesure de similarité entre deux sommets du graphe par la capacité maximum des chemins qui relient les deux sommets, on peut alors dire que les sommets dans un noyau du graphe sont tous plus similaires ou plus proches entre eux qu'avec n'importe quel autre sommet en dehors du noyau. Pour les sommets qui ne sont pas dans un noyau, il n'est cependant pas possible avec cette définition de la similarité, de déterminer quel est le noyau (ou éléments dans un noyau) qui lui est le plus proche.

Proposition 5 *Soit $G=(V,E)$ un graphe pondéré et connexe, et $Q_1^n,...,Q_K^n$ les différents n-noyaux de G ($K \geq 2$).*

1. *Si un sommet n'est pas dans le noyau Q_k^n, alors la capacité maximum des chemins qui le relient à n'importe quel sommet de Q_k^n est constante : $\forall i \in V \setminus \{Q_k^n\}$, $\forall j \in Q_k^n$, $\max_{p\in\mathcal{P}_{ij}}\{c(p)\} = c_i^k$ (ne dépend pas de j).*
2. *La capacité maximum des chemins qui relient un sommet d'un noyau Q_k^n à un sommet d'un autre noyau Q_l^n est constante : $\forall i \in Q_k^n, \forall j \in Q_l^n$ $(l \neq k)$, $c_i^l = c_j^k = c^{kl}$.*
3. *Si un sommet n'est dans aucun noyau, alors il existe au moins deux noyaux distincts Q_k^n et Q_l^n tels que la capacité maximum des chemins qui relient le sommet à un sommet quelconque de Q_k^n et Q_l^n est constante : si $i \notin Q_k^n, \forall k$, alors il existe $k \neq l$ tel que $\forall j \in Q_k^n \cup Q_l^n$, $c_i^k = c_i^l = c_i^*$, où $c_i^* = \max\limits_{t=1,2,...,K} c_i^t$.*
4. *Soit $i \in V$ un sommet qui n'est dans aucun noyau, et c_i^k la capacité maximum des chemins qui relient i aux sommets de Q_k^n. Pour tout $l \neq k$, si la capacité maximum c^{kl} des chemins qui relient un sommet Q_k^n à un sommet de Q_l^n est telle que $c^{kl} \geq c_i^k$ alors $c_i^l = c_i^k$.*

Preuve Soit G un graphe supposé connexe et $Q_1^n,...,Q_K^n$ la liste des ensembles de sommets qui induisent les différents n-noyau de G avec $K \geq 2$.

1. Soit i un sommet qui n'est pas dans Q_k^n ($i \notin Q_k^n$), et soient deux sommets distincts $j \neq j'$ dans Q_k^n ($j, j' \in Q_k^n$). La capacité maximum d'un chemin qui relie i à j (resp. j') est notée c^1 (resp. c^2). On suppose que $c^2 < c^1$ et l'on montre que l'on obtient une contradiction. L'ensemble Q_k^n induit un noyau de G donc par la proposition 4, on en déduit qu'il existe un $\lambda \in \mathbb{R}$ tel que la capacité maximum d'un chemin qui relie deux sommets dans le noyau est supérieure ou égale à λ, et la capacité maximum d'un chemin qui relie un sommet du noyau à un sommet en dehors du noyau est strictement inférieure à λ, d'où $c^1 < \lambda$ et $c^2 < \lambda$ et $\max_{p\in\mathcal{P}_{jj'}}\{c(p)\} \geq \lambda$. Si l'on considère le chemin de capacité maximum ($\geq \lambda$) qui relie j' à j, et qu'on le prolonge par le chemin de capacité maximum c^1 ($c^1 < \lambda$) qui

relie j à i, on obtient alors un chemin qui relie i à j' de capacité $c^1 > c^2$, ce qui contredit l'hypothèse que c^2 est la capacité maximum d'un chemin reliant i à j'. On en déduit que la capacité maximum des chemins qui relient i à n'importe quel sommet j de Q_k^n est constante.

2. Soit i un sommet de Q_k^n et j un sommet de Q_l^n avec $k \neq l$. Le sommet j n'est pas dans Q_k^n donc la capacité maximum des chemins qui relient j à un sommet de Q_k^n (en particulier le sommet i) vaut c_j^k. De la même façon, la capacité maximum des chemins qui relient i à un sommet de Q_l^n (en particulier j) vaut c_i^l, d'où $c_j^k = c_i^l$, $\forall i \in Q_k^n, \forall j \in Q_k^n$.
3. Soit i un sommet qui n'est dans aucun des noyaux du graphe ($i \notin Q_k^n, \forall k$). On peut montrer très simplement, par la proposition 4 et la propriété de maximalité d'un ensemble induisant un noyau du graphe, que si la capacité maximum c_i^k des chemins qui relient i à un sommet de Q_k^n est strictement supérieure à toutes celles des chemins qui relient i aux sommets dans les autres noyaux Q_l^n, c'est à dire $\forall l \neq k$, $c_i^k > c_i^l$, alors le sommet i est nécessairement dans le noyau Q_k^n. Si l'on prend le cas particulier d'un graphe ne contenant que deux noyaux, on ne peut avoir $c_i^1 > c_i^2$ d'où $c_i^1 = c_i^2$. Dans le cas plus général, si $c_i^k = c_i^*$ alors comme on ne peut avoir $c_i^k > c_i^l$ pour tout $l \neq k$, il existe un $l \neq k$ tel que $c_i^k = c_i^l = c_i^*$.
4. Soit i est un sommet qui n'est dans aucun des noyaux du graphe ($i \notin Q_k^n, \forall k$), et c_i^k la capacité maximum des chemins qui relient i à un sommet de Q_k^n. Pour tout $l \neq k$, les sommets de Q_k^n peuvent être reliés à ceux de Q_l^n par des chemins de capacité maximum c^{kl}. Si $c^{kl} \geq c_i^k$, on ne peut avoir $c_i^k < c_i^l$ (resp. $c_i^k > c_i^l$) dans la mesure où pour connecter i à un sommet de Q_k^n (resp. Q_l^n), on peut trouver un chemin qui passe par les sommets de Q_l^n (resp. Q_k^n) qui soit de capacité maximum c_i^l (resp. c_i^k) et le prolonger jusqu'au sommet de Q_k^n (resp. Q_l^n) par un chemin de capacité maximum $c^{kl} \geq c_i^k$ (resp. $c^{kl} > c_i^l$). On obtient alors un chemin de capacité supérieur à c_i^k (resp. c_i^l). De cette contradiction on conclut que $c_i^k = c_i^l$.

Les noyaux sont composés des éléments qui sont les plus stables dans une communauté, dans le sens où la capacité maximum des chemins qui les relient entre eux est nécessairement supérieure à celle des chemins qui les relient à un sommet qui n'est pas dans le noyau. Pour un élément qui n'est pas dans un noyau, il est plus délicat de déterminer le noyau qui lui est le plus proche, ou autrement dit, la communauté à laquelle il appartient. En effet, comme nous venons de le montrer, en définissant une mesure similarité entre les sommets du graphe par la capacité maximum des chemins qui relient les sommets, un sommet qui n'est pas dans un noyau a une similarité constante avec tous les sommets d'un même noyau, et la similarité entre ce sommet et les sommets d'un noyau est maximum pour au moins deux noyaux distincts.

Définition des communautés En fonction des applications, on peut être amené à adopter différents points de vue pour définir une communauté. La question est ici de proposer une solution pour intégrer au sein des communautés les éléments qui ne sont pas dans un noyau. Nous envisageons deux possibilités : la première pour créer des communautés disjointes qui forment une partition de l'ensemble des éléments, et la deuxième, pour créer des communautés chevauchantes. Une autre approche, qui ne pose pas la question de l'intégration des éléments qui ne sont pas dans un noyau, consiste simplement à les exclure de l'analyse.

Avec cette dernière approche, le probléme de la détection des communautés est le même que celui de la détection des noyaux du graphe. A n fixé, chaque ensemble d'élements qui est un n-noyau du graphe est aussi une communauté. Ainsi, une communauté est un ensemble d'éléments préferentiellement connectés sur le graphe par les liens les plus fiables (poids plus élevés). Nous verrons par la suite, que cette approche prend tout son sens pour l'analyse des réseaux de co-expression de gènes. Elle permet de sélectionner les gènes les plus centraux sur le réseau, qui sont indispensables au maintien de l'équilibre de la structure d'interaction et qui jouent un rôle déterminant dans le fonctionnement cellulaire.

Pour créer des communautés disjointes qui forment une partition de l'ensemble des éléments, nous proposons d'utiliser le processus inverse de celui qui permet de détecter les noyaux, en supposant que tous les poids des liens sont deux à deux distincts. Au départ, chaque communauté est un noyau (autant de communautés que de noyaux). Le sommet qui n'est dans aucune communauté et qui est le plus fortement connecté (lien de poids le plus élevé) avec un élément appartenant à l'une de ces communautés est alors ajouté à la communauté correspondante. Cet procédure, de type agglomératif, est réitérée tant qu'il reste des sommets hors des communautés. Elle est très proche de celle proposée par Prim pour constuire un arbre couvrant de poids maximum : une seule étape est ajoutée pour l'actualisation des communautés.

Une formalisation mathématique du procédé peut être donnée en définissant la mesure de similarité s_{iQ} entre un élément extérieur aux noyaux et un noyau Q. Nous avons vu (proposition 5) que la capacité maximum d'un chemin reliant un élément en dehors des noyaux à un élément dans un noyau est constante, quelque soit l'élément du noyau, et qu'il n'y a pas unicité du noyau (au moins deux noyaux). Pour définir une mesure de similarité entre un noyau Q et un élément en dehors des noyaux il est alors nécessaire d'ajouter une contrainte sur les chemins et considérer uniquement ceux qui sont sur l'arbre couvrant de poids maximum et qui ne passent pas par des éléments contenus dans les autres noyaux $Q' \neq Q$. En notant $\mathcal{P}_{iQ}^{A}$, l'ensemble des chemins sur l'arbre couvrant de poids maximum A (unicité si les poids des liens sont deux à deux disctincts), qui relient un sommet i en dehors des noyaux à un sommet de Q, et qui ne passent (ou contiennent) par aucun sommet d'un autre noyau $Q' \neq Q$, la similarité s_{iQ} s'écrit :

$$s_{iQ} = \max_{p \in \mathcal{P}_{iQ}^{A}} \{c(p)\}.$$

Chaque élément i en dehors des noyaux est alors placé dans la communauté qui contient le noyau Q pour lequel la similarité s_{iQ} est maximum. L'ensemble des éléments est ainsi partitionné en K communautés $C_1, C_2, ..., C_K$ engendrées par les noyaux $Q_1, Q_2, ..., Q_K$, en maximisant la similarité entre les éléments dans chaque communauté et le noyau correspondant :

$$\arg \max_{C_1, C_2, ..., C_K} \sum_{k=1}^{K} \sum_{i \in C_k} s_{iQ_k}.$$

Pour définir des communautés chevauchantes, on va considérer que seuls les éléments dans les noyaux sont propres à une seule communauté. Chaque communauté contient un noyau qui lui est propre est l'ensemble des éléments qui ne sont pas dans les noyaux sont considérés comme étant à l'intersection entre plusieurs communautés. On définit la mesure de similarité s_{iQ} entre un élément et un noyau par $s_{iQ} = \max_{p \in \mathcal{P}_{ij}, j \in Q} c(p)$. On en déduit par la proposition 5 que

pour tout élément extérieur aux noyaux, il existe au moins deux noyaux avec lesquels l'élément a une similarité maximum. L'élément est alors placé dans la partie chevauchante de toutes les communautés contenant un noyau avec lequel la similarité est maximum.

Choix du paramètre et amélioration de l'algorithme de détection des noyaux Le paramètre de la taille minimale n des noyaux doit être choisi avec précaution. Pour un n trop petit, les noyaux détectés sont très nombreux et de petites tailles. Cela conduit à créer des noyaux artificiels, car très proches les uns des autres (pouvant être relié par des chemins de capacité maximum élevées) et très dépendants du paramètre n. A l'inverse, si l'on choisit un paramètre n trop élevé, on risque de regrouper au sein d'un même noyau des sommets qui ne sont pas proches, et de regrouper dans un seul noyau, plusieurs noyaux qui sont plus pertinents en eux-mêmes. On va rechercher des noyaux qui soient suffisamment stables, observés sur une plage de valeur de n la plus grande possible.

Nous allons illustrer notre propos à l'aide de résultats obtenus sur des données d'expression réelles que nous étudierons en détail dans le chapitre suivant. Le réseau de co-expression de gènes contient 5405 gènes et nous avons identifié les noyaux pour différentes valeurs $n = 10, 11, ..., 40$. La Figure 2.5 met en évidence l'évolution des noyaux en fonction de n. En analysant l'évolution des noyaux, de la plus petite valeur de n et en remontant jusqu'à la plus grande, on observe une première phase de stabilisation des noyaux à partir de $n = 21$: les 12 noyaux de taille les plus importantes restent inchangés jusqu'à $n = 30$. Plus le paramètre augmente, et plus les noyaux sont stables : à partir de $n = 34$ et jusqu'à $n = 40$ on observe une deuxième phase de stabilisation, mais les noyaux sont de très grande taille et le risque d'avoir fusionné des noyaux qui n'auraient pas dû l'être est plus important.

Pour s'en convaincre, nous avons calculé pour chaque n, la taille moyenne des noyaux ainsi que la densité à l'intérieur des noyaux. La densité δ_{int}^{Q} à l'intérieur des noyaux $Q_1, Q_2, ..., Q_K$ d'un graphe G de matrice d'adjacence $W = S$, où S est la matrice d'association entre les gènes (détaillé dans le dernier chapitre), est définie par :

$$\delta_{int}^{Q} = \frac{1}{p_Q(p_Q - 1)} \sum_{k=1}^{K} \sum_{i,j \in Q_k} w_{ij},$$

où $p_Q = \sum_{k=1}^{K} |Q_k|$ est le nombre de sommets à l'intérieur des noyaux. Sur la Figure 2.6, on observe clairement la première phase de stabilisation des noyaux pour $n \in \{26, 27, .., 30\}$. On va choisir n dans cet intervalle car la densité à l'intérieur des noyaux est assez importante et la taille moyenne des noyaux est suffisante. Pour un n plus élevé, la densité à l'intérieur des noyaux diminue beaucoup trop et la taille moyenne augmente beucoup trop.

Pour rendre l'algorithme de détection des noyaux encore plus robuste face au choix du paramètre de la taille minimale n des noyaux, nous proposons d'agréger les résultats obtenus pour différentes valeurs de n. On observe deux phénomènes distincts dans l'évolution des noyaux : il se peut que deux noyaux, détectés pour un n donné, fusionnent quand n augmente pour n'en former plus qu'un (sur la Figure 2.5, pour $n = 33$ on a les noyaux 2 et 3 et pour $n = 34$ ils sont tous les deux dans le noyau 4), ou qu'un noyau détecté pour un n donné disparaisse complètement,

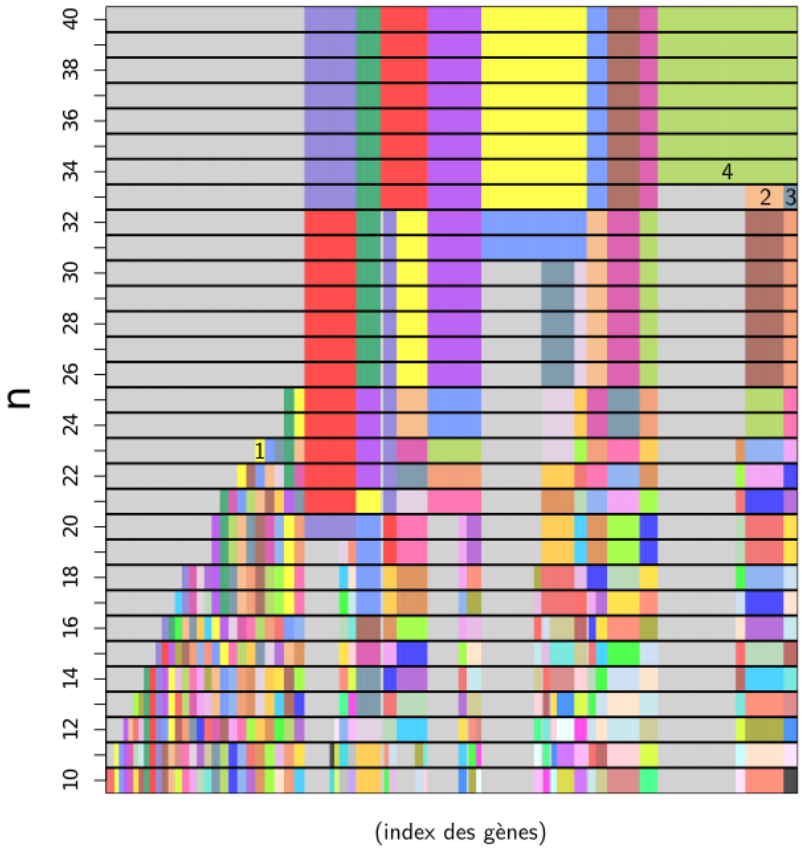


FIGURE 2.5 – Représentation de l'évolution des noyaux en fontion du choix du paramètre n (index des gènes en abscisse et valeur de n en ordonnée). Sur chaque ligne, pour un n fixé, les différents noyaux sont représentés par des couleurs distinctes et la couleur grise est réservée aux gènes qui ne sont pas dans les noyaux.

c'est à dire que les sommets qui le constituent ne soient plus inclus dans un noyau pour un n plus grand (sur la Figure 2.5, le noyau 1 observé pour $n = 23$ disparait pour $n = 24$). Ainsi, l'instabilité des noyaux se caractérise soit par des fusions, soit par des disparitions. Ces deux cas ne reflètent pas la même dynamique et nous proposons de les traiter différemment.

Dans un premier temps, nous allons rechercher des noyaux pour un n donné qui soit suffisamment stables pour « la fusion » : on cherche un bon compromis pour avoir des noyaux stables et de taille suffisante. Sur notre exemple, pour une valeur n dans l'ensemble $\{21, 22, ..., 30\}$, les douzes noyaux les plus grands restent stables. En choisissant la valeur minimum $n = 21$ dans cet intervalle, on perd un grand nombre de noyaux qui ont des tailles infèrieures à 21 (à gauche sur la figure).

On peut alors dans un deuxième temps, choisir de rajouter les noyaux qui ont disparus. Sur notre exemple, l'ensemble des noyaux qui ont disparu à $n = 21$ restent stables tout le temps ou ils apparaissent, ou autrement dit, il n'y a pas de phénomène de fusion sur ces noyaux. On peut alors ajouter chaque noyau si il apparait au moins une fois pour un $n \in \{n_{min} = 10, 11, ..., 20 = n_{max}\}$.

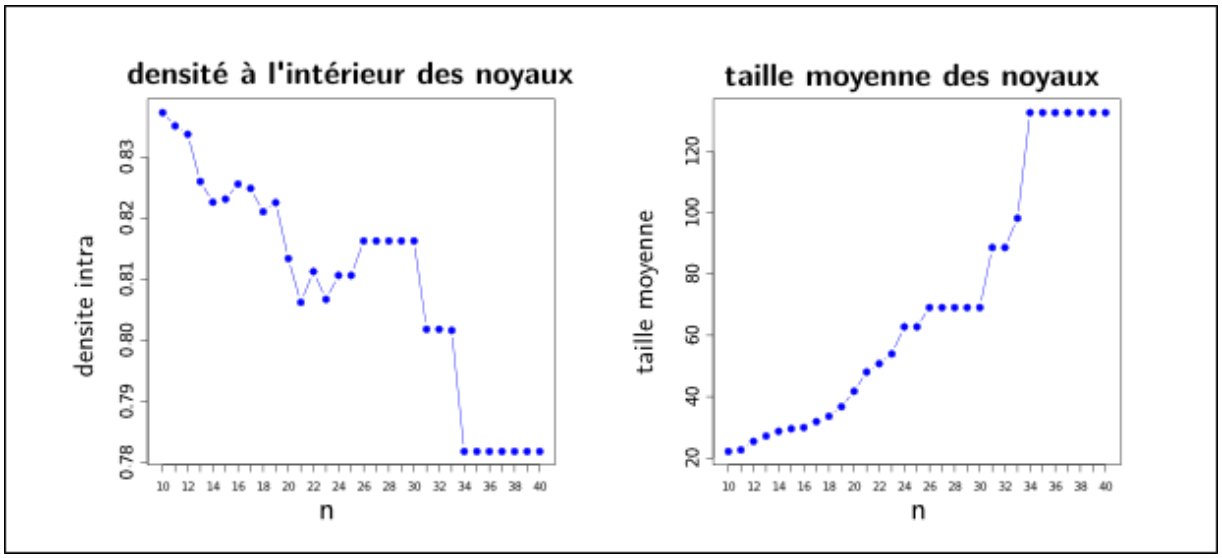


FIGURE 2.6 – Représentation de l'évolution de la taille moyenne des noyaux, et de la densité à l'intérieur des noyaux, en fonction du paramètre n

Dans un cas plus général, il se peut que l'on observe des phénomènes de fusion sur les noyaux qui ont disparus, et dans ce cas, on ajoute un noyau si et seulement si il est détecté pour un paramètre n donné et qu'il a disparu pour le paramèter $n+1$, avec $n \in \{n_{min}, ..., n_{max} - 1\}$, de façon à laisser les noyaux fusionner les uns avec les autres entre n_{min} et n_{max}.

Exemple d'application Nous reprenons l'exemple de la matrice d'association S construite pour mettre en évidence certaines limites des approches spectrale et WGCNA (Figure 2.2). La matrice d'adjacence W de dimension 400×400 du graphe complet G est telle que $w_{ij} = s_{ij}$ si $i \neq j$ et $w_{ii} = 0$ pour tout i.

Dans le bloc S_3 de S, une mesure d'association vaut au minimum 0.31 et les mesures d'association entre un élément de ce bloc et tous les autres éléments en dehors du bloc sont constantes et valent 0.3. Ainsi, les éléments de S_3 forment une composante connexe du graphe partiel G_λ obtenu pour un paramètre de seuillage $\lambda = 0.31$. Chaque élément à l'intérieur du bloc a une mesure d'association constante avec tous les autres éléments du blocs et cette mesure varie d'un élément à l'autre. La composante connexe associée au bloc se désagrége progressivement à mesure que le paramètre de seuillage λ augmente, et les éléments de S_3 engendrent un noyau dès que la taille minimale n des noyaux est fixée dans l'ensemble $\{2, 3, ..., 100\}$.

De la même façon, les blocs S_1 et S_2 sont détectés sur le graphe G_λ pour $\lambda = 0.61$ et engendre un noyau si $n \in \{2, 3, ..., 100\}$.

Sur cet exemple, les communautés sont directement détectées par les noyaux et la méthode proposée pour construire des communautés disjointes qui forment une partition de l'ensemble des éléments n'a pas d'intérêt (d'autant plus qu'elle n'est pas adaptée pour traiter les cas d'égalité). Avec l'approche qui permet de définir des communautés chevauchantes, tous les éléments moins informatifs (bloc S_4) sont placés dans la partie chevauchante des trois communautés.

Cet exemple met également en évidence une certaine robustesse de notre approche face au choix du paramètre n qui peut évoluer dans un ensemble de grande taille sans affecter le résultat.

Nous reviendrons sur cet aspect par la suite en se basant sur des données plus réalistes.

Points forts de notre approche Notre approche présente de nombreux avantages pour l'identification de communautés sur des graphes construits à partir d'une matrice de similarité :

- *Mise en oeuvre facile (un seul paramètre à fixer).*
- *Coût computationnel raisonnable : peut être utilisée sur de très gros graphes.*
- *Possibilité de faire de la sélection de variables : sélection des éléments dans les noyaux.*
- *Flexibilité pour la définition des communautés autours des noyaux : communautés disjointes ou chevauchantes.*

2.7 Utilisation de la structure communautaire

2.7.1 Incorporation de connaissances biologiques

Les différentes études effectuées à partir de données génétiques (transcriptomiques notamment) ont permis d'accroître la connaissance des gènes et celle de leurs fonctions biologiques. Ces connaissances sont organisées et stockées dans des bases de données pour permettre leur diffusion et favoriser les avancées dans la compréhension du génome. En partant du postulat que les gènes co-exprimés ont des fonctions similaires, il devient alors possible d'intégrer cette information a priori sur les fonctions des gènes pour donner une interprétation biologique à un regroupement de gènes en communautés, qui a été déterminé sans a priori à l'issue de l'analyse d'un réseau de co-expression.

Le consortium GO L'étude de la fonction des ARN au niveau cellulaire s'intègre dans la discipline appelée génomique fonctionnelle. Un ARN est constitué d'une suite de bases nucléiques et peut être appréhendé comme un texte qu'il faut déchiffrer pour pouvoir l'expliquer ou l'annoter. L'annotation d'un ARN n'est pas simple, car les mots ou parties de la séquence qui donnent un réel sens au texte sont peu nombreux et mêlés avec de nombreuses séquences qui n'ont aucun sens en elles-mêmes. Les parties d'une séquence d'ARN qui ont un sens sont celles qui correspondent aux gènes. Pour l'annotation des ARN, les deux objectifs principaux sont de définir les séquences des gènes, et dans la mesure du possible, de définir ce qu'ils font, leurs fonctions cellulaire.

Un programme d'annotation génique (`http://www.geneontology.org/`) appelé le consortium de l'ontologie génique (GO pour « Gene Ontology » consortium) a été mis en place à des fins de standardisation et de partage des connaissances. Des milliers de gènes qui ont été identifiés par les projets génomiques sont actuellement annotés. En informatique, une ontologie est un outil qui permet de donner une représentation précise d'un corpus de connaissance. Elle représente un ensemble structuré de concepts qui sont organisés sous forme d'un graphe au sein duquel il existe des relations sémantiques (hiérarchique, associative...) ou d'inclusions (appartient à). Les fonctions des gènes sont classées en trois catégories :

— la fonction cellulaire (BP pour « Biological Function ») qui fait référence à la nature du processus qui est régulé ou modifié, comme la synthèse du glucose par exemple.
— la fonction moléculaire (MF pour « Molecular Function ») qui décrit les activités biochimiques.
— le compartiment cellulaire (CC pour « Cellular Component ») qui renseigne sur l'endroit dans la cellule où le gène remplit sa fonction.

Par exemple, le terme « GO :0006094 » a pour numéro d'accession « 0006094 », il est associé à l'ontologie « BP », il est appelé « gluconeogenesis », et il est defini par « the formation of glucose from noncarbohydrate precursors, such as pyruvate, amino acids and glycerol ». Au sein de chacune des trois ontologies, les termes GO sont organisés sous forme d'un graphe acyclique dirigé (DAG) et liés sur le graphe par des relations : « Is a », « Part of », « Regulates », « Positively Regulated », « Negatively regulates » et « Occurs in ». Le terme de « GO :0006094 (gluconeogenesis) » est (Is a) aussi un processus métabolique (terme « GO :0008152 (metabolic process) »).

L'annotation des ARN par les termes GO est effectuée par des experts qui peuvent notamment utiliser des algorithmes qui recherchent les similarités de structure pour transfèrer les annotations des ARN connus à des ARN similaires et inconnus. Des milliers d'ARN ont ainsi pu être annotés ce qui constitue une source d'information indispensable pour l'interprétation des résultats issus de l'analyse de données transcriptomiques.

Les analyses d'enrichissement L'identification de communautés de gènes co-exprimés a pour vocation de regrouper des gènes potentiellement impliqués dans des fonctions biologiques similaires. La question est alors de savoir quels sont les fonctions biologiques pouvant être attribuées à chacune des communautés. Les fonctions biologiques d'un gène, quand celui-ci a déjà pu être identifié, sont décrites par les termes GO. Pour donner un sens biologique à une communauté, l'approche la plus classique consiste à rechercher les termes GO qui sont surreprésentés ou « enrichis » au sein de la communauté. La surreprésentation d'un terme est testée en comparant la fréquence d'apparition du terme dans la communauté à à celle observée sur l'ensemble des gènes (toutes les communautés).

Le degré d'enrichissement d'une communauté pour un terme GO est évalué en utilisant la loi hypergéométrique. Soit p le nombre total de gènes, θ_T la fréquence d'apparition sur l'ensemble des gènes d'un terme GO noté T ($p\theta_T$ = nombre de gènes parmi les p gènes qui sont associés aux termes T), et n_C la taille d'une communauté C. La loi hypergéométrique de paramètre n_C, θ_T et p, notée $\mathcal{H}(p, n_C, \theta_T)$ décrit l'expérience suivante : on choisit, par un tirage sans remise, n_C gènes parmi les p gènes et on compte le nombre k de gènes qui sont associés au terme T. La variable aléatoire discrète Y associée à une telle expérience (qui donne le nombre $k \in \mathbb{N}$) suit la loi $\mathcal{H}(p, n_C, \theta_T)$:

$$\mathbb{P}(Y = k) = \frac{C^k_{p\theta_T} C^{n_C - k}_{p(1-\theta_T)}}{C^{n_C}_p}.$$

Un test exact de Fisher est utilisé pour tester l'enrichissement du terme T au sein de la communauté C. Cette approche non paramétrique permet de déterminer si deux variables qualitatives distinctes Y^1 et Y^2 à deux modalités sont indépendantes. Elle est une alternative au test du χ^2 qui n'est pas adapté quand les effectifs croisés dans la table de contingence sont trop faibles (< 5). Ici, la variable Y^1 (resp. Y^2) correspond à la variable indicatrice définie sur l'ensemble des gènes qui prend la valeur 1 si le gène appartient à la communauté C (resp. est associé au terme T). La Table 2.1 est la table de contingence qui croise les variables Y^1 et Y^2.

L'hpothèse nulle H_0 du test exact de Fisher est l'indépendance entre Y^1 et Y^2. Sous H_0 et quand les effectifs marginaux sont fixés (à p, θ_T, n_C fixés), la probabilité d'obtenir la configuration décrite par la table de contingence, ou autrement dit, la probabilité d'avoir k, est donnée par la loi hypergéométrique $\mathcal{H}(p, n_C, \theta_T)$. On note k_0 le nombre de gènes dans C qui sont annotés par

	gène dans C	gène en dehors de C	
gène annoté par T	k	$p\theta_T - k$	$p\theta_T$
gène non annoté par T	$n_C - k$	$p(1-\theta_T) - n_C + k$	$p(1-\theta_T)$
	n_C	$p - n_C$	p

TABLE 2.1 – Table de contingence pour les analyses d'enrichissement

T. Si les chances d'obtenir des configurations avec k au moins égal à k_0 sont très faibles, il est alors improbable, en cas d'indépendance de Y^1 et Y^2, de trouver autant de gènes annotés par T dans C, et on peut conclure à un enrichissement significatif de T au sein de C.

Le calcul de la p-valeur est assez laborieux car il faut calculer la probabilité relative à la configuration observée (pour k_0) et aux configurations possibles pour lesquelles k est supérieur à k_0 (test unilatéral). La p-valeur du test vaut $\mathbb{P}(Y = k_0) + \sum_{k=k_0+1}^{p\theta_T} \mathbb{P}(Y = k)$.

Les analyses d'enrichissement permettent d'associer des fonctions aux différentes communautés et éventuellement de proposer des annotions pour les gènes qui sont inconnus, parmi celles qui sont enrichies dans leurs communautés respectives. Afin limiter le risque de faux positifs qui croît avec le nombre de tests réalisés, il est nécessaire d'utiliser une méthode (Bonferroni par exemple) qui permet de diminuer ce risque.

Information sur les voies métaboliques La base de données KEGG (pour Kyoto Encyclopedia of Genes and Genomes) contient de l'information sur l'implication des gènes au sein des voies métaboliques. Une voie métabolique est une série de réactions successives au cours desquelles une même molécule est modifiée jusqu'à l'obtention d'un produit donné. La glycolyse en est un exemple, elle permet d'assimiler le glucose pour produire de l'énergie (pyruvate). La modulation de l'expression des gènes peut participer à la régulation de ces voies métaboliques. Ainsi, en associant aux gènes certaines voies métaboliques au sein desquelles ils ont une action potentielle, il est possible, comme décrit précédemment (analyse d'enrichissement), de rechercher au sein des communautés de gènes les voies métaboliques « enrichies ».

2.7.2 Choix des représentants des communautés

Le choix d'un unique représentant pour chaque communauté permet de réduire considérablement le nombre de variables, de supprimer la redondance, et de remédier au moins partiellement aux problèmes de multi-colinéarité pour les problèmes de modélisation d'un caractère d'intérêt à partir des éléments du réseau.

Sélection des « hubs » dans les noyaux Un « hub » sur un graphe est un sommet qui a un degré de connectivité très élevé. Nous rappelons que le degré de connectivité d_i du sommet i d'un graphe $G = (V, E)$ de matrice d'ajacence W est la somme des poids des arêtes incidentes au sommet, $d_i = \sum_{j \in V} w_{ij}$. Cette notion tient compte du nombre de liens incidents au sommet et du poids de ces liens.

Dans le cas où le graphe G est connu, on peut calculer directement les degrés de connectivité des sommets et sélectionner à l'intérieur de chaque communauté le hub, c'est à dire le sommet de degré de connectivité maximum dans la communauté. Dans le cas de l'analyse des réseaux

de co-expression de gènes, le graphe est inconnu et l'identification des communautés par notre méthode ne se fait pas à partir d'un unique graphe. Nous envisageons de multiples représentations du réseau réel en construisant des graphes de λ-voisinage définis à partir de la matrice d'association entre les gènes. Se pose alors la question du choix d'un graphe pour calculer le degré de connectivité des sommets. Il parait difficile de trouver un critère qui nous permette de faire un tel choix d'autant plus que les communautés identifiées avec notre méthode peuvent avoir des densités hétérogènes. On ne peut en général trouver un unique graphe de λ-voisinage qui reflète de façon satisfaisante les connections entre les sommets à l'intérieur de toutes les communautés. Une solution est alors de choisir le graphe le plus complet, c'est à dire celui pour lequel la matrice d'adjacence est définie directement par la matrice d'association S (pas de seuillage). La densité d'un sommet i sur le graphe s'écrit alors $d_i = \sum_{j \neq i} s_{ij}$. En pratique, ce graphe est complet et le calcul de la connectivité ne tient plus compte du nombre de liens incidents aux sommets mais uniquement du poids des liens. Afin de garantir la « centralité » d'un hub au sein d'une communauté, nous définissons la connectivité d'un sommet i dans une communauté C_k par $d_i^{C_k} = \sum_{j \in C_k} s_{ij}$. La somme des $d_i^{C_k}$ pour tout $i \in C_k$ est une mesure de similarité intra-classe dans la classe C_k. Ainsi, nous ne tenons plus compte de la structure du réseau pour identifier les hubs dans les communautés, mais uniquement des similarités intra-classes définies directement à partir de la matrice d'association S.

Définition des représentants des communautés par l'ACP Une autre possibilité pour définir le(s) représentant(s) d'une communauté consiste à faire une analyse en composantes principales sur les éléments de la communauté, et à choisir autant de composantes principales que nécessaire pour expliquer une part de l'inertie totale suffisante. Cette approche peut être intéressante dans le cas où les similarités entre les éléments à l'intérieur de la communauté sont plus hétérogènes.

2.7.3 Etude des relations causales

L'apprentissage des réseaux de régulation de gènes est une problématique très en vogue ces dernières années qui offre de nombreuses perspectives de recherche en biologie. L'objectif est de donner une représentation de la structure de régulation : l'activité d'un gène va-t-elle influencer l'activité d'un autre gène ? L'activité d'un gène sous-entend ici le niveau d'expression du gène. La modélisation du système par les réseaux bayésiens consiste à proposer, par un raisonnement probabiliste, un modèle graphique qui représente les relations causales entre les variables, et qui repose uniquement sur des observations (données d'expression). La notion de réseau bayésien a été introduite par Pearl [46, 32]. Elle peut prêter à confusion car elle ne fait pas référence à un modèle bayésien (a priori sur la loi des paramètres qui s'estompe à mesure que les observations sont prises en compte), mais au théorème de Bayes qui est utilisé pour définir un modèle graphique. Une approche a priori pour comprendre l'influence d'un gène sur d'autres gènes peut se faire en invalidant ou inactivant complètement le gène (knock-Out ou KO) de façon à observer les conséquences du KO. Ce procédé est malheureusement long et coûteux, ce qui fait tout l'intérêt d'utiliser l'approche sans a priori des réseaux bayésiens. Friedman et al. [22] a introduit ce type de modélisation pour l'inférence de réseaux de régulation de gènes à partir de données d'expression. La plus grande contrainte pour ce type d'analyse est celle du nombre de sommets sur le réseau. Quand celui-ci augmente, il devient très vite impossible de faire une recherche exhaustive de la structure de dépendance causale la plus probable (le nombre de configurations possibles croît

de façon super-exponentielle avec le nombre de sommets). Des solutions ont été proposées, en contraignant l'espace de recherche par exemple, mais nous ne traiterons pas cette problématique dans la mesure où la question que nous allons nous poser est différente de celle qui est posée classiquement.

Au lieu de rechercher la structure de dépendance causale sur un réseau qui contient l'ensemble des gènes, nous allons nous intéresser à l'identification d'effets causaux entre plusieurs communautés de gènes et par extrapolation entre plusieurs fonctions biologiques (on suppose qu'une communauté de gène représente une fonction biologique donnée). En choisissant un représentant par communauté, on peut construire un réseau bayésien de taille très raisonnable pour décrire la structure de dépendance entre les communautés (représentants) et limiter ainsi les problèmes liés à la grande dimension ($p >> N$) et à la taille d'un espace de recherche pour la structure de dépendance causale trop important.

L'hypothèse de suffisance causale pour un modèle de réseau bayésien suppose que l'ensemble des variables qui agissent dans le système sont observées. En pratique cependant, cette hypothèse est trop restrictive et il existe des variables qui n'ont pu être observées, soit parce qu'elle sont inconnues, soit pour des raisons techniques. Ces variables qui n'ont pu être observées mais qui sont impliquées dans le système sont appelées des variables latentes.

Les réseaux bayésiens sans variables latentes Un réseau bayésien est représenté par un graphe sur lequel les sommets correspondent à des variables aléatoires et les arcs (liens orientés) indiquent les influences directes entre les variables. Le graphe est supposé ne pas contenir de cycle : un réseau bayésien est représenté par un graphe orienté acyclique ou DAG (Directed Acyclic Graph en anglais). La modélisation du réseau par un DAG fait l'hypothèse de la suffisance causale des données (pas de variable latente).

Un réseau bayésien $B = (G, P_G)$ qui représente un ensemble de variables aléatoires $V = \{X^1, ..., X^p\}$ est défini par un DAG $G = (V, E)$, où E est l'ensemble des arcs (paires ordonnées de sommets) décrivant les relations directes entre variables, et par un ensemble P_G, des distributions conditionnelles pour chaque variable aléatoire de V, sachant ses parents (prédecesseurs direct) sur G. La loi jointe $P(V)$ sur l'ensemble des variables peut alors être décomposée comme suit :

$$P(V) = \prod_{i=1}^{p} P(X^i | pa(X^i)),$$

où $pa(X^i)$ est l'ensemble des parents de X^i sur G, et $P(X^i|pa(X^i)) \in P_G$.

La représentation des données par un DAG n'a de sens que si la structure du graphe donne une description complète des indépendances conditionnelles utilisées pour la factorisation de la loi jointe. La d-séparation est un critère qui permet de mettre en relation l'information portée par le DAG avec celles des distributions conditionnelles : quelles sont les contraintes d'indépendance représentées par le DAG ?

Definition (d-séparation) Deux variables X^i et X^j sont d-séparées par un ensemble de variables (ou sommets) $\mathcal{Z}$ sur un DAG, si l'une des conditions suivantes est satisfaite :

1. il existe une séquence $X^i \to X^k \to X^j$ (ou $X^i \leftarrow X^k \leftarrow X^j$) ou $X^i \leftarrow X^k \to X^j$ avec $X^k \in \mathcal{Z}$.
2. il existe une séquence $X^i \to X^k \leftarrow X^j$ telle que X^k ainsi que l'ensemble des descendants de X^k ne sont pas dans $\mathcal{Z}$.

Deux ensembles disjoints de sommets, $\mathcal{X}$ et $\mathcal{Y}$, sont d-séparés par un autre ensemble disjoint de sommets $\mathcal{Z}$, si tous les chemins qui relient un sommet de $\mathcal{X}$ à un sommet de $\mathcal{Y}$ sont d-séparés par $\mathcal{Z}$.

La condition 1. du critère de la d-séparation signifie que les variables X^i et X^j sont dépendantes conditionnellement à X^k et qui si l'on connait ou fixe X^k, alors elles deviennent indépendantes. Dès lors que l'on connait X^k, la variable X^i (resp. X^j) ne peut plus avoir d'influence sur X^j (resp. X^i) dans la mesure où X^k reste inchangée. Avec la condition 2. les variables X^i et X^j sont indépendantes mais dès lors que l'on fixe X^k, elles deviennent dépendantes. Ainsi, si l'on ne connait pas X^k ni aucun de ses descendants, alors X^i et X^j sont indépendantes. Le parallèle entre la d-séparation des variables sur le DAG et celui de l'indépendance conditionnelle ne peut être fait que si la condition de Markov est respectée.

Propriété 6 *Un DAG G est un réseau markovien de la loi jointe P si la condition de Markov est satisfaite, c'est à dire que la d-séparation sur G implique l'indépendance conditionnelle pour P : $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subset V$ disjoints,*

$$\mathcal{X} \text{ et } \mathcal{Y} \text{ sont d-séparés par } \mathcal{Z} \text{ sur } G \Longrightarrow \mathcal{X} \text{ et } \mathcal{Y} \text{ sont indépendantes conditionnellement à } \mathcal{Z}.$$

Sur un réseau markovien, chaque variable est indépendante de ses non descendants conditionnellement à ses parents, et cette information permet de factoriser la loi jointe. Pour que les relations d'indépendance conditionnelle soient complètement décrite par le DAG, il faut que la condition de fidélité soit respectée.

Propriété 7 *Une loi de probabilité jointe P et un DAG G sont fidèles (faithful) si et seulement si toutes les indépendances conditionnelles sont identifiables par les d-séparations sur G : $\mathcal{X}, \mathcal{Y}, \mathcal{Z} \subset V$ disjoints*

$$\mathcal{X} \text{ et } \mathcal{Y} \text{ sont d-séparés par } \mathcal{Z} \text{ sur } G \Longleftrightarrow \mathcal{X} \text{ et } \mathcal{Y} \text{ sont indépendantes conditionnellement à } \mathcal{Z}$$

La modélisation des données par un réseau bayésien fait l'hypothèse que la distribution jointe des variables est fidèle à un DAG. Cette hypothèse n'est pas tellement limitante en pratique car il a été montré [41] que la plupart des distributions le sont.

Il se peut que les structures de différents réseaux bayésiens représentent les mêmes relations d'indépendance conditionnelle, autrement dit, qu'elles engendrent la même factorisation pour la loi jointe des variables. Ces réseaux sont dit équivalents au sens de Markov. Le squelette d'un DAG est le graphe non dirigé obtenu en supprimant l'orientation des liens sur le DAG. Une v-structure sur un DAG G est un triplet ordonné (X^i, X^j, X^k) de sommets, tel que G contient les arcs $X^i \to X^j$ et $X^k \to X^j$, et tel que X^i et X^k ne sont pas adjacents sur G. Verma et Pearl [48] ont montrés que deux DAG sont équivalents (Markov) si et seulement si ils ont le même squelette et les mêmes v-structures.

Une classe d'équivalence de Markov, c'est à dire un ensemble de réseaux bayésiens équivalents, est représentée par un DAG partiellement dirigé complété ou CPDAG (pour completed partially directed acyclic graph). Un DAG partiellement dirigé (PDAG) est un graphe qui contient des liens orientés et des liens non orientés, sur lequel il n'y a pas de cycle entre les sommets (suivant n'importe quelle direction pour les liens non orientés). Un PDAG est complété (CPDAG) s'il est le représentant d'une unique classe d'équivalence. Il a les mêmes v-structures que l'ensemble des réseaux dans la classe d'équivalence, et les arcs qui ne sont pas dans une v-structure sont remplacés par des liens non orientés. Autrement dit, un lien non orienté $X^i - X^j$ entre deux sommets sur le CPDAG signifie qu'il existe au moins un DAG dans la classe d'équivalence sur lequel $X^i \rightarrow X^j$, et un autre sur lequel $X^i \leftarrow X^j$. Un exemple de CPDAG est représenté sur la Figure 2.7.

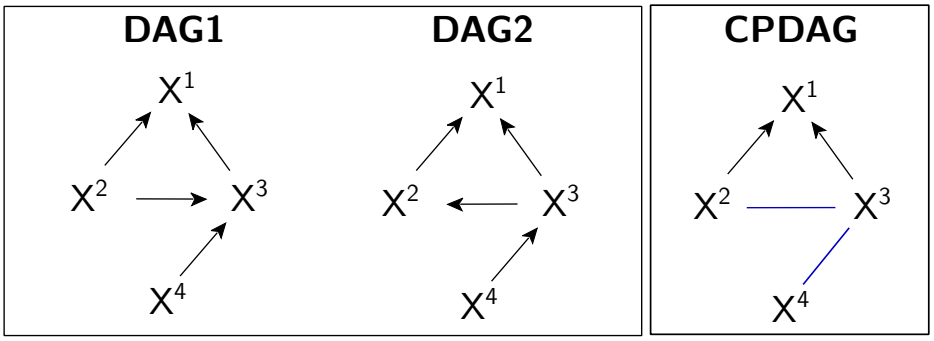


FIGURE 2.7 – Représentation de deux DAG équivalents au sens de Markov (DAG1 et DAG2) et du CPDAG de la classe d'équivalence.

Les réseaux bayésiens avec variables latentes L'hypothèse de suffisance causale est trop restrictive en pratique, dans la mesure où il est généralement impossible d'observer la totalité des variables impliquées dans un système. Par exemple, si le « vrai » DAG contient la configuration $X^1 \leftarrow L \rightarrow X^2$, où X^1 et X^2 sont deux variables observées, et L est une variable latente (non observée) qui agit sur X^1 et sur X^2, l'indépendance entre X_1 et X_2 conditionnellement à L ne peut être détectée par la loi jointe observée, et inversement, on ne peut, à partir des observations, trouver un DAG sur lequel X^1 et X^2 sont d-séparés.

Un autre modèle graphique, introduit par Richardson et Spirtes [53], qui supporte l'existence de variables latentes, est celui des graphes ancestraux maximum ou MAG (pour Maximal Ancestral Graph) [71, 12]. Ce modèle de graphe a été proposé pour représenter les relations entre des variables générées suivant un modèle qui peut contenir des variables latentes et/ou un biais de sélection. Nous nous sommes intéressés uniquement au cas de l'existence de variables latentes, sans biais de sélection.

Un sommet X^i est l'ancêtre d'un sommet X^j s'il existe un chemin dirigé qui part de X^i et qui rejoint X^j : il existe $\{Z^1, ..., Z^k\}$ tel que $Z^1 = X^i$, $Z^k = X^j$ et $Z^k \rightarrow Z^{k+1}$ pour tout $l \in \{1, ..., k-1\}$. Un graphe ancestral (sans biais de sélection) est un graphe qui contient des arcs dirigés ($\leftarrow$ ou $\rightarrow$) et des arcs bidirigés ($\leftrightarrow$) sur lequel on ne peut avoir :

- un cycle dirigé : $X^i \to X^j$ et X^j est un ancêtre de X^i,
- un cycle quasi-dirigé : $X^i \leftrightarrow X^j$ et X^j est un ancêtre de X^i.

La propriété d'absence de cycle sur un DAG est ainsi généralisée pour les graphes ancestraux, qui peuvent contenir des arcs bidirigés. Un DAG est un graphe ancestral particulier (sans arc bidirigés).

Le critère de la d-séparation défini pour les DAG permet de retrouver les indépendances conditionnelles dès que la condition de Markov est satisfaite. Ce critère a été étendu au cas des graphes ancestraux et est appelé le critère de la m-séparation. Soit $p = (X^1, ..., X^k)$ un chemin d'un graphe ancestral. Un sommet $X^j \in p$, $j \in \{2, ..., k-1\}$, crée une collision si les arcs qui le relient avec le sommet précedent $X^{(j-1)} \in p$ et le sommet suivant $X^{(j+1)} \in p$, pointent tout deux en direction de X^j : $X^{(j-1)} *\!\!\to X^j \leftarrow\!\!* X^{(j+1)}$ ($*$ remplace soit une flèche soit un trait).

Definition (m-séparation) Un chemin (non orienté) p est dit bloqué par un ensemble Z si l'une des deux conditions suivantes est satisfaite :

1. p contient une séquence $\{X_{j-1}, X_j, X_{j+1}\}$ telle que $X_j \in Z$ et X_j ne crée pas de collision,
2. p contient une v-structure $X^{(j-1)} *\!\!\to X^j \leftarrow\!\!* X^{(j+1)}$ telle que le sommet qui crée la collision n'est pas dans Z, $X_j \notin Z$, et aucun des descendants de X_j n'est dans Z.

Deux ensembles de sommets disjoints $\mathcal{X}$ et $\mathcal{Y}$ sont m-séparés par un autre ensemble disjoint de sommets $\mathcal{Z}$ si tous les chemins qui relient un sommet de $\mathcal{X}$ à un sommet de $\mathcal{Y}$ sont bloqués par $\mathcal{Z}$.

Pour un graphe ancestral qui ne contient pas d'arc bidirigé (DAG), le critère de la m-séparation est équivalent à celui de la d-séparation. Sur un DAG, si deux sommets ne sont pas adjacents, ils sont nécessairement d-séparés par un ensemble d'autres sommets, ce qui n'est pas le cas avec la m-séparation pour un graphe ancestral : deux sommets qui ne sont pas adjacents ne sont pas nécessairement m-séparés. Dans le cas où un graphe ancestral vérifie cette propriété, il est dit maximal ou MAG. Un MAG est un graphe ancestral qui vérifie la propriété de maximalité :

- si les sommets X^i et X^j ne sont pas adjacents sur le graphe, alors il existe un ensemble de sommets $\mathcal{Z}$ tel que X^i et X^j sont m-séparés par $\mathcal{Z}$.

Le critère de la m-séparation permet de retrouver les indépendances conditionnelles sur un MAG au même titre que celui de la d-séparation sur un DAG. Si le DAG G_D représente les relations entre l'ensemble des variables, celles qui ont été observées et celles qui sont latentes, alors il existe un unique MAG G_M, défini à partir des variables observées, sur lequel deux sommets sont m-séparés par Z si et seulement si ils sont d-séparés par Z sur G_D [53]. Si G_D est fidèle à la loi jointe des variables (observées et latentes), alors deux variables sont conditionnellement indépendantes si et seulement si elle sont d-séparées sur G_D ou (équivalence) m-séparées sur G_M. Il est ainsi possible, sur un MAG, de détecter les indépendances conditionnelles entre les variables observées sans connaître les variables latentes.

Deux MAG différents peuvent décrire les mêmes contraintes d'indépendances conditionnelles en ayant les mêmes ensembles de variables m-séparés. Une classe d'équivalence de Markov pour les MAG peut être représentée par un graphe partiellement ancestral ou PAG (pour partial ancestral graph). Le squelette d'un PAG est le même que celui de l'ensemble des MAG dans la classe d'équivalence. Le symbole supplémentaire « o » permet d'étiqueter l'extrémité d'un arc du PAG,

si l'étiquetage de cette extrémité varie sur les différents MAG de la classe d'équivalence. Sur un PAG, une extrémité d'un arc est étiquetée par une flèche (resp. un trait) si cette même extrémité est étiquetée par une flèche (resp. un trait) sur tous les MAG dans la classe d'équivalence. Un exemple de PAG est représenté sur la Figure 3.6.

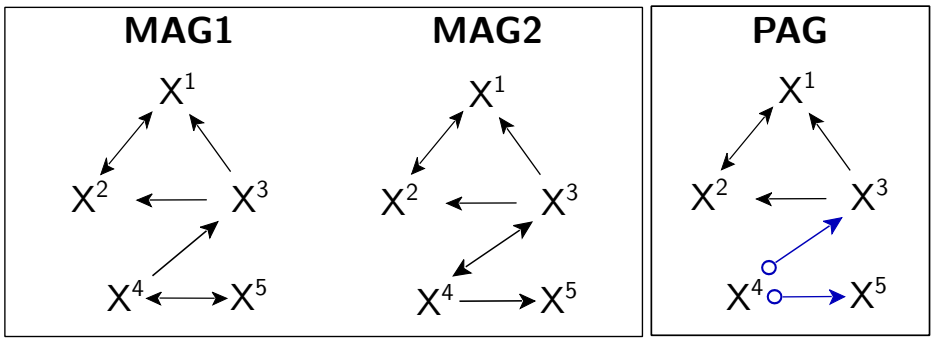


FIGURE 2.8 – Représentation de deux MAG équivalents au sens de Markov (MAG1 et MAG2) et du PAG de la classe d'équivalence.

Interprétation en terme de causalité Il est impossible de découvrir un réseau bayésien complètement causal en prenant en compte uniquement les données observées. La structure du réseau qui est déterminée à partir des observations est un représentant de la classe d'équivalence de Markov qui contient (éventuellement) la « vraie » structure causale. Les résultats sont à interpréter avec précaution et il faut garder à l'esprit les hypothèses qui ont été faites pour déterminer la structure du réseau. Par l'apprentissage de la structure des réseaux bayésiens, on obtient une information sur le sens des relations entre les variables mais la notion de causalité reste très hypothétique. Cette information peut suggérer certains mécanismes au sein du système et cela n'en reste pas moins intéressant.

L'orientation des arcs sur le graphe obtenu par apprentissage pour représenter la structure d'un réseau bayésien suggère la causalité : si $X^i \rightarrow X^j$, cela suggère que X^j ne peut être la cause de X^i et que X^i est éventuellement la cause de X^j ou tout du moins un ancêtre de X^j (impliqué dans la cascade qui a un effet sur X^j). Par abus de langage, nous dirons que X^i est la cause de X^j si l'on a $X^i \rightarrow X^j$ sur le graphe. Sous l'hypothèse de suffisance causale, la structure du réseau bayésien est représentée par un CPDAG sur lequel un arc est dirigé si l'une des variables est la cause de l'autre, et un arc n'est pas dirigé si le sens de la relation ne peut être déterminé. Sous l'hypothèse d'insuffisance causale, on représente la structure du réseau bayésien avec un PAG. Sur ce type de graphe,

- X^i est une « cause véritable » de X^j si $X^i \rightarrow X^j$
- X^i est une « cause artificielle » de X^j si $X^i \leftrightarrow X^j$: X^i (resp. X^j) ne peut être une cause de X^j (resp. X^i), car il existe une variable latente L telle que $X^i \leftarrow L \rightarrow X^j$.
- X^i est une « cause indéterminée » de X^j si X^i o−o X^j : on ne peut savoir si X^i est la cause de X^j ou inversement, ou si les variables ont une cause commune L qui est latente.
- X^i est une « cause potentielle » si X^i o→ X^j : X^i est soit la cause de X^j, soit la conséquence d'une cause latente L.

Apprentissage de la structure d'un réseau bayésien L'apprentissage de la stucture du réseau constitue l'une des problématiques de la modélisation par les réseaux bayésiens. Nous n'aborderons pas celle de l'inférence dans le réseau, c'est à dire de la quantification des dépendances entre les variables. La recherche de la structure d'un réseau bayésien est un problème difficile, du fait principalement, de la taille de l'espace de recherche qui croît de façon super-exponentielle avec le nombre de variables. Les méthodes d'apprentissage peuvent être regroupées en deux grandes familles : les méthodes basées sur la recherche des indépendances conditionnelles pour proposer une structure de réseau en cohérence avec celles-ci, et les méthodes basées sur un score, mesurant l'adéquation de la structure de dépendance décrite par le réseau avec les données, de façon à rechercher la structure qui donne le meilleur score (maximiser la probabilité de la structure sachant les données [27]).

L'algorithme PC de Spirtes, Glymour et Scheiner [57] est l'une des approches les plus populaires dans la première famille de méthode. En faisant l'hypothèse de suffisance causale des données, l'algorithme recherche un CPDAG qui respecte les relations d'indépendances conditionnelles observées sur les données. Une version modifiée de cet algorithme, proposée par Spirtes et al. [59], est l'algorithme FCI qui permet de tenir compte de l'existence de variables latentes (recherche d'un PAG). Avec ces deux approches, les relations d'indépendances conditionnelles sont déterminées à l'aide d'un test statistique. Dans le contexte de l'analyse de réseaux de régulation de gènes, ces approches ne sont pas tellement utilisées car l'estimation des indépendances conditionnelles devient très vite trop coûteuse quand le nombre de variables augmente. Une autre difficulté, liée à l'utilisation d'un test statistique pour l'identification des indépendances conditionnelles, est celle de l'augmentation du risque de faux positifs (dépendance factice entre les variables) avec le nombre de variables. Notre intérêt pour ces approches se justifie dans la mesure où nous ne souhaitons pas inclure la totalité des gènes dans le réseau, mais seulement les quelques gènes choisis comme représentants des communautés de gènes, identifiés à l'issue d'une première analyse de la structure du réseau de co-expression.

Test des indépendances conditionnelles dans le cas gaussien On suppose que l'on observe N réalisations i.i.d. d'un vecteur gaussien X de dimension p, $X = {}^t(X^1, ..., X^p) \sim \mathcal{N}(\mu, \Sigma)$. Dans le cas gaussien, les indépendances conditionnelles peuvent être identifiées à partir des corrélations partielles. La corrélation partielle $r_{X^iX^j|Z}$ caractérise l'association entre les variables X^i et X^j après avoir éliminé l'effet (linéaire) sur ces variables d'une ou plusieurs autres variables $Z = \{X^k; k \neq i, j\}$. Le vecteur X est gaussien donc on a l'équivalence :

$$r_{X^iX^j|Z} = 0 \Longleftrightarrow X^i \text{ et } X^j \text{ sont indépendantes conditionnellement à } Z.$$

L'algorithme PC et FCI utilisent l'approche récursive pour le calcul des corrélations partielles. Pour tout $X^k \in Z$, la corrélation partielle $r_{X^iX^j|Z}$ s'écrit :

$$r_{X^iX^j|Z} = \frac{r_{X^iX^j|Z\setminus\{X^k\}} - r_{X^iX^k|Z\setminus\{X^k\}} r_{X^kX^j|Z\setminus\{X^k\}}}{\sqrt{1 - r^2_{X^iX^k|Z\setminus\{X^k\}}}\sqrt{1 - r^2_{X^kX^j|Z\setminus\{X^k\}}}}.$$

Une corrélation partielle $r_{X^iX^j|\emptyset}$ d'ordre zéro est définie par la corrélation simple $r_{X^iX^j}$. Avec l'approche récursive, la corrélation partielle est calculée via une décomposition en de multiples corrélations simples (Pearson). L'objectif est alors d'identifier quelles sont les corrélations partielles qui prennent des valeurs nulles ou autrement dit, qui ne sont pas significativement différentes de zéro. La significativité d'une corrélation partielle $r_{X^iX^j|Z}$ est testée en utilisant sa

transformation de Fisher :

$$z(r_{X^iX^j|Z}) = \frac{1}{2}\ln\left(\frac{1+r_{X^iX^j|Z}}{1-r_{X^iX^j|Z}}\right).$$

L'hypothèse nulle $H_0 : r_{X^iX^j|Z} = 0$, est rejetée au risque d'erreur α, au profit de l'hypothèse alternative $H_1 : r_{X^iX^j|Z} \neq 0$, si $\sqrt{N-|Z|-3}|z(r_{X^iX^j|Z})| > \Phi^{-1}(1-\alpha/2)$, où Φ est la fonction de répartition d'une $\mathcal{N}(0,1)$.

L'algorithme PC L'algorithme PC utilise un test statistique pour évaluer s'il y a indépendance conditionnelle entre les variables. Lorsque la loi jointe des variables est gaussienne, les tests portent sur les corrélations partielles, comme nous venons de le décrire dans le paragraphe précédent.

La première phase de l'algorithme consiste à rechercher le squelette du réseau, en partant d'un graphe complet (toutes les variables sont connectées), et en supprimant les liens lorsqu'une indépendance conditionnelle est détectée. Une première série de tests sur l'indépendance d'ordre 0 entre les variables est effectuée : X^i et X^j sont-elles indépendantes ? Les liens sur le graphe sont supprimés si l'indépendance est détectée. Pour les liens restants, une deuxième série de tests sur l'indépendance conditionnelle d'ordre 1 est réalisée : X^i et X^j sont-elles indépendantes sachant X^k qui est adjacent à X^i ou X^j ? A l'ordre 2, deux sommets adjacents X^i et X^j sont-ils indépendants conditionnellement à un ensemble $\{X^k, X^l\}$ de sommets qui sont adjacents à X^i ou à X^j ? Et ainsi de suite, on augmente l'ordre jusqu'à ce que l'on ne puisse plus trouver des ensembles de variables connectées de tailles suffisantes pour le conditionnement. Chaque fois qu'un lien est supprimé entre deux variables X^i et X^j sur le graphe, l'ensemble des variables qui crée l'indépendance (conditionnement) est mémorisé dans $sep(X^i, X^j)$ en tant qu'ensemble séparant X^i et X^j (d-séparation).

La deuxième phase de l'algorithme consiste rechercher une orientation pour les liens sur le squelette. Si l'on a deux couples de variables adjacentes (X^i, X^k) et (X^j, X^k), que X^i et X^j ne le sont pas, et que X^k n'est pas dans $sep(X^i, X^j)$, alors le triplet (X^i, X^k, X^j) forme une v-structure sur le DAG et les liens sont orientés : $X^i \rightarrow X^k \leftarrow X^j$. Pour les liens restant, qui ne sont pas dans une v-structure, les deux orientations possibles sont envisagées et si l'une des deux (pas les deux) contredit les hypothèses, c'est à dire qu'un cycle ou une v-structure est créé, alors c'est l'autre qui est choisie. Cette dernière étape est répétée tant qu'il est possible d'orienter de nouveaux liens.

Il n'est généralement pas possible de donner une orientation à tous les liens et l'on obtient un CPDAG qui caractérise la classe d'équivalence des DAG qui sont conformes aux relations d'indépendance conditionnelle décrites par les données. Quand la loi jointe des données est gaussienne et qu'elle est fidèle à un DAG qui ne contient pas de variable latente, Kalisch et Bülhmann [31] ont montré que sous certaines hypothèses (sparsité, régularité), l'algorithme PC est consistant, et le reste en grande dimension ($p >> N$).

L'algorithme FCI L'algorithme FCI est une adaptation de l'algorithme PC pour gérer l'existence de variables latentes. La base de l'algorithme FCI est la même que celle de l'algorithme PC, à laquelle sont ajoutés des tests supplémentaires sur les indépendances conditionnelles entre variables, ainsi que de nouvelles conditions pour l'orientation des arcs. Les conditions initialement proposées par Spirtes et al. pour l'orientation des arcs sur le graphe ont été complétées par

Zhang [72], de façon à ce que tous les aspects de la structure causale puissent être déterminés à partir de l'information sur les relations d'indépendance conditionnelles.

Le squelette du graphe est estimé par la même procédure que celle décrite pour l'algorithme PC, puis tous les liens sont représentés sous la forme o−o. Les v-structures sont ensuite orientées selon la même règle que celle de l'algorithme PC. Une autre série de tests est ensuite effectuée pour supprimer certains liens pour lesquels la présence de variables latentes peut justifier l'indépendance conditionnelle. Pour ce faire, Spirtes et al. [58] étendent la définition d'un ensemble qui sépare deux variables (m-séparation). A ce stade de l'algorithme, une condition nécessaire pour qu'un sommet X^i soit (m-)séparé d'un autre sommet par le sommet X^k est qu'il existe un chemin qui relie X^i à X^k tel que toutes les séquences $\{X_{j-1}, X_j, X_{j+1}\}$ sur le chemin soient une v-structure (X_j crée une collision) ou forment un triangle (ou cycle).

On note $PS(X^i)$ l'ensemble des sommets qui peuvent éventuellement créer une séparation entre X^i et un autre sommet suivant la condition précédente (condition nécessaire mais non suffisante). Ainsi, les étapes suivantes de l'algorithme consistent à caractériser l'ensemble $PS(X^i)$ pour chacun des sommets X^i du graphe, puis, pour tout couple de sommets adjacents (X^i, X^j), à effectuer de nouveaux tests d'indépendance conditionnelle en conditionnant par des sous-ensemble de $PS(X^i)$ ou de $PS(X^j)$, à supprimer le lien entre X^i et X^j si l'indépendance conditionnelle est détectée, et à ajouter dans $sep(X^i, X^j)$ l'ensemble qui crée la séparation. Les tests d'indépendance conditionnelle sont effectués suivant la même procédure que celle employée à la première étape, c'est à dire que l'on augmente progressivement la taille des ensembles qui servent au conditionnement. Tous les arcs du graphe sont ensuite ré-étiqueter pour être de la forme o−o, puis les v-structures sont recherchées et orientées à nouveau en tenant compte de la nouvelle structure du graphe et de l'actualisation des $sep(X^i, X^j)$.

La dernière étape de l'algorithme consiste à tenter de donner une orientation à l'ensemble des arcs qui n'ont encore pu être complètement orientés. Les conditions qui permettent de compléter les orientations des arcs sont décrites et justifiées dans l'article de Zhang [72]. Cette étape est répétée tant qu'il est possible de trouver de nouvelles orientations pour les arcs. Quand la loi jointe des données est gaussienne et qu'elle est fidèle à un MAG, Colombo et al. [12] ont montré que sous certaines hypothèses (sparsité, régularité), l'algorithme FCI est consistant, et le reste en grande dimension ($p >> N$). Cet algorithme est cependant inutilisable quand le graphe contient beaucoup de sommets car il nécessite de faire un grand nombre de tests pour détecter les indépendances conditionnelles.

La corrélation de Spearman pour mesurer les dépendances entre les variables ? Avec les algorithmes PC et FCI, les indépendances conditionnelles sont déterminées en testant la significativité des corrélations partielles. Cette approche est pertinente uniquement dans le cas gaussien et lorsque cette hypothèse est violée, la fiabilité des résultats ne peut être garantie. En particulier, lorsque l'on étudie les relations entre les expressions de gènes, les résultats peuvent être largement impactés par la présence de valeurs extrêmes ou de dépendances non linéaires. Dans ce cas, on est tenté d'utiliser une mesure de dépendance qui soit plus robuste. Une solution simple consiste à remplacer les valeurs d'expression des gènes par leurs rangs sur les séquences ordonnées (corrélation de Spearman), et à utiliser les algorithmes PC ou FCI sur ces nouvelles variables de rang : pour tout gène j on remplace le vecteur des observations $\mathbf{x}^j$ par le vecteur des rangs R^j. Les algorithmes conservent leurs propriétés de consistance sous des hypothèses moins restrictives sur la distribution jointe des variables [26].

Estimation bootstrap de la fiabilité des motifs sur un réseau bayésien Friedman et al. [21] sont les premiers à proposer la technique du rééchantillonnage bootstrap pour analyser la fiabilité de la structure d'un réseau bayésien estimée et représentée par un PDAG. Cette technique se généralise directement au cas d'un PAG estimé avec l'algorithme FCI.

Supposons que les données ont été générées suivant un modèle de réseau bayésien $B = (G, P_G)$, et que l'on observe N réalisations indépendantes $\mathbf{z} = \{\mathbf{x}_1, ..., \mathbf{x}_N\}$ du vecteur aléatoire $(X^1, ..., X^p)$ de l'état de p sommets appartenant à l'ensemble V des sommets du graphe G. Tous les sommets dans V ne sont pas nécessairement observés d'où $p \leq |V|$ (variables latentes). Un graphe peut être décomposé en de multiples sous-graphes que nous appelerons des motifs. Un motif peut par exemple être une relation causale directe entre deux sommets $X^i \rightarrow X^j$, ou un sous-graphe $X^i \leftarrow X^k \rightarrow X^j$ qui met en évidence le fait que X^i et X^j ont une cause commune, ou encore un sous-graphe indiquant que X^i est un ancêtre de X^j... Soit f la fonction indicatrice d'un motif donné sur un graphe. Par exemple, $f(G) = 1$ si X^i est un ancêtre de X^j sur G et $f(G) = 0$ sinon. On s'intéresse alors à la probabilité $p_N(f)$ de trouver un motif donné sur le graphe $\hat{G}_{\mathbf{z}}$ estimé par l'algorithme FCI à partir des observations $\mathbf{z}$:

$$p_N(f) = \mathbb{P}(\ f(\hat{G}_{\mathbf{X}}) = 1 \mid |\mathbf{z}| = N\ ).$$

Sous les mêmes hypothèses que celles qui permettent de montrer la consistance de l'algorithme FCI, on a $p_N(f) \sim f(G)$ pour un N assez grand. La probabilité $p_N(f)$ est proche de 1 si le motif existe sur le graphe G et elle est proche de 0 sinon.

La technique du bootstrap permet de faire une estimation non-paramétrique de cette probabilité à partir d'un unique ensemble d'observations $\mathbf{z}$. Il est intuitif de penser que si l'on perturbe l'échantillon de départ et que l'on estime à nouveau le graphe, les motifs les plus fiables seront retrouvés sur ce graphe. Un échantillon bootstrap $\mathbf{z}^*$de $\mathbf{z}$ est construit par tirage avec remise de N éléments de $\mathbf{z}$. Une estimation de la fiabilité d'un motif observé sur $\hat{G}_{\mathbf{z}}$ s'obtient simplement par la création de b échantillons bootstrap $\mathbf{z}_1^*, ..., \mathbf{z}_b^*$ à partir desquels on construit les b graphes $\hat{G}_{\mathbf{z_1^*}}, ..., \hat{G}_{\mathbf{z_b^*}}$. Le motif d'intérêt est recherché sur les différents graphe $\hat{G}_{\mathbf{z_i^*}}$ et la fiabilité du motif estimée par :

$$\hat{p}_N(f) = \frac{1}{b} \sum_{i=1}^{b} f(\hat{G}_{\mathbf{z_i^*}}).$$

L'estimation de la fiabilité des motifs de $\hat{G}_{\mathbf{z}}$ par la technique du bootstrap est plutôt prudente comme l'a montré Friedman, ce qui renforce le crédit pouvant être accordé aux motifs les plus fiables.

2.7.4 Article méthodologique

Dans cet article, en cours de soumission dans la revue BMC, nous évaluons les performances de notre approche sur des données simulées et réelles, et mettons en évidence ses avantages par rapport aux approches WGCNA et spectrale.

Gene co-expression network analysis based on Core Structure Detection (CSD)

Anne-Claire Brunet, Jean-Michel Loubes, Jean-Marc Azaïs

December 5, 2015

Abstract

We propose a new method, called CSD (for Core Structure Detection), to identify clusters of co-expressed genes, by considering all the possible co-expression graphs obtained by removing stepwise the edges with the lowest weight and identifying the set of nodes which are preferentially connected on this collection of graphs. The first step of the algorithm identifies sets of strongly connected nodes that make good candidates to define the cores of clusters. The second step aggregates the remaining nodes out of the core structures to build the final clusters. The algorithm takes as input the similarity matrix between the variables We introduce the CSD approach before analyzing the performances of the algorithm, and compare its performances with the WGCNA and spectral clustering, two popular graph clustering algorithms. For this purpose, we first use simulated data for explicit estimations of the performances (a priori knowledge of the community structure), and second, we use well-studied transcriptomic data describing gene expression during the cell division cycle of the yeast Saccharomyces cerevisiae [29] to test the ability of the method to identify biologically meaningful clusters, and to highlight the importance of the nodes that compose the core structures based on biological interpretations.

1 Background

The analysis of hidden interaction structures in biological systems is a problem of major importance. High-throughput technologies, and microarray data in particular, have allowed to obtain large quantities of gene expressions. Their structure and relationships may explain the biological properties at stake. Interactions between genes are often described as a gene

network, modeled by an undirected weighted graph. Each node or vertex represents a gene while an edge connects two nodes if the corresponding genes are co-expressed, which means that their expression patterns are similar across the sample in the sense that the transcript levels increase or decrease in a similar way. Weights on the edges characterize the strength of the connections between the nodes. Rather than defining a similarity measure between each pair of variables without taking into account the others variables, the graph structure adds information about adjacency, connectivity, and characterizes similarities within the complex structure of the gene network. The analysis of gene co-expression network made it possible to identify novel disease-related molecular mechanism as in [2], [35].
Identifying the most influential genes or centrally positioned genes in co-expression networks is the next problem. Indeed, the scale-free topological organization in biological networks is considered common property [34], and is characterized by the presence of few nodes or hubs that play a key role in the interaction mechanisms. These hubs may play crucial roles in biological systems and may serve as valuable biomarkers for diseases see for instance in [13], [15]. The hubs make sense as soon as the graph presents a structure of communities around these hubs, hence as soon as the graph can be partitioned into sub structures, each one containing a main hub. So clustering the graph into communities helps to summarize information by reducing the dimensionality from thousands of genes to a small number of cluster, and in this way is an essential component of analysis methods, aiming at highlight the co-expression structure. Finally, clusters or communities of genes exhibit similar expression patterns often related to functional complex [11], as well as highly connected genes (hubs) in the network, may have critical functional roles or associated with key disease-related pathways.
Two problems are thus to be tackled : first the construction of the graph of the genes interactions and second the clustering of the network.

Building a graph to model the co-expression structure is not a trivial task. The starting point is the definition of a co-expression measure or similarity measure between the genes. The most popular measure is the absolute value of the Pearson's correlation coefficient [1], but other measures can be considered. A simple way to construct a graph based on pairwise similarities is to connect all pairs of nodes having non-zero similarity values and to weight edges by the corresponding similarity values. This approach leads to construct a weighted almost fully connected graph with few real zero values in the similarity matrix, which prevents the discovery of particular clusters of interaction. Most of graph clustering methods are usable and efficient

if the graph is sufficiently sparse, such that the organization in clusters is rather pronounced with many edges within each cluster and relatively few between the clusters. Hence, methods that promote the sparsity of the network have received a growing attention over the last decade. One way to estimate a sparse graph is to connect two nodes if their similarity exceeds a certain threshold ϵ, defining an ϵ-neighorhood graph. Another way is to connect each node with all its k-nearest neighbors, a k-nearest neighbor graph. Both approaches are very sensitive to the choice of the parameter ϵ or k. The advantage of a k-nearest neighbor graph, compared to an ϵ-neighborhood graph, is that it can connect nodes using different local similarity thresholds for each nodes (edge connecting nodes in each neighborhood can have highly variable weights), rather than connecting nodes based on a unique threshold. Yet the ϵ-neighborhood graph estimation fails to identify regions or communities of a graph with different levels of density. For a given stringent threshold, many nodes in the weakly dense regions of the graph may be disconnected, while for a low threshold, nodes in the dense regions of the graph may have too many edges, making clustering in these regions difficult. Moreover, the k-nearest neighbor graph also have a tendency to include non significant edges due to the use of a fixed-size neighborhood.

Identifying clusters of similar nodes of graph is also a hard task. The mere concept of graph clustering hides very difficult issues. As a matter of fact, clustering implies splitting the data into groups that share a common behavior. Hence it requires being able to define a notion of similarity or equivalently distance between the different elements.When dealing with non Euclidean structures such as graphs for instance, this notion of similarity is not natural and highly relies on the objective of the study. Actually several authors have proposed different ways of achieving this goal. We refer to[12], [22, 20, 21] or [27] for instance and references therein. A very common method employed for clustering biological networks is the hierarchical clustering. In this class of methods we can cite the WGCNA algorithm [17, 39] widely used for the analysis of co-expression networks [8, 14]. The starting point of hierarchical graph clustering is the definition of a similarity measure between graph objects or nodes, e.g. the topological overlap measure [38]. With the agglomerative (bottom-up) strategy, each node is considered as a separate cluster (singleton) in the starting point of the clustering procedure, and then clusters are iteratively merged if their similarity is sufficiently high. Since clusters are merged based on their mutual similarity, a measure that estimates how two clusters are closed must be defined, for example by computing the average similarity between nodes in one group and nodes in the other (average linkage clustering). The procedure ends up with a unique

cluster that contains all the vertices of the graph. The partition strategy is top-down as it follows the opposite direction, but has been rarely used in practice. Hierarchical clustering has the advantage that it does not require any assumptions on the number and the size of the clusters. However, the results of the method may strongly depend on the similarity measure adopted, while the graph partition relies on the choice of the cut height in the dendrogram which may lead to very different clusters, as in [18]. Spectral clustering [23] [33] is another very popular graph clustering method used in a variety of applications [26]. This method consists in transforming the representation of the graph to make the cluster properties of the initial data sets much more evident. In particular, by replacing the weighted adjacency matrix of the graph with a subset of eigenvectors of the Laplacian matrix, a different representation of the graph is induced and can be useful to extract information on the cluster structure. The problem of clustering nodes is then reduced to a classical clustering problem defined in an Euclidean space. The set of points in the new space (or elements of eigenvectors) can be clustered via standard clustering algorithm, like the k-means clustering. Finally a variety of criteria and methods have been proposed to allow automatic selection of the number of clusters [31], [36]. However, the more noisy or ambiguous the clusters are, the less effective are these methods. In many cases, determining the optimal value for the number of clusters is a nontrivial or even impossible problem.

Hence graph clustering can be achieved in several ways having several advantages and drawbacks. In the analysis of networks, the issue is even more difficult since the induced graph is unknown and has to be estimated, resulting to different possible graph partitions. Since for real data set, there is no ideal underlying graph, the purpose of the clustering procedure and the notion of the estimation of the graph are deeply linked.

In this paper we propose a new method, called CSD (for Core Structure Detection), to simultaneously enhance clusters of co-expressed genes and provide a sparse network of genes. For this we will identify clusters of genes which are strongly connected together. They will be called cluster core structures which bring together centrally positioned genes and build the graph onto such central hubs. Hence the CSD algorithm falls into two steps. The first step identifies sets of strongly connected nodes that make good candidates to define the cores of clusters. The second step aggregates the remaining nodes out of the core structures to build the final clusters. The algorithm takes as input the similarity matrix between the variables (e.g. matrix of absolute correlation coefficients for co-expression network) and

a parameter that characterizes the minimum size of these core structures. More precisely, we consider all the possible graphs obtained by removing stepwise the edges with the lowest weight and identifying the set of nodes which are preferentially connected on this collection of graphs. They will be denoted as the core structures. Then the nodes are connected to these core structures to obtain the clusters of the graph, with each cluster containing a unique core structure and all the nodes that are more strongly connected with nodes in this core structure than with nodes in a different core structure. This process allows the detection of core structures of varying sizes and densities governed by an input parameter that controls the level of granularity of the result partition. Note that the clusters obtained with the CSD algorithm, for different values of the input parameter or the minimum size of the core structures, are nested according to a hierarchical organization. Then a simple observation of the evolution of the partition, as a function of the input parameter, allows to select the best compromise with respect to the number and the size of the obtained clusters.

In the Results and Discussion section, we introduce the CSD approach before analyzing the performances of the algorithm, and compare its performances with the WGCNA and spectral clustering algorithms. For this purpose, we first use simulated data for explicit estimations of the performances (a priori knowledge of the community structure), and second, we use well-studied transcriptomic data describing gene expression during the cell division cycle of the yeast Saccharomyces cerevisiae [29] to test the ability of the method to identify biologically meaningful clusters, and to highlight the importance of the nodes that compose the core structures based on biological interpretations. In the following Methods section, we provide more details on the mathematical formalism of the network clustering algorithms used in the analysis and on the criteria used to analyze and compare the performances of network clustering algorithms.

2 Results and Discussion

2.1 The CSD algorithm

2.1.1 Graph definition

Our starting point is a set of joint expression, or co-expression, of p genes over N individuals. From this co-expression network, a weighted adjacency matrix $W = (w_{ij})_{1 \leq i,j \leq p}$ is constructed, in most of the case using the absolute value of the Pearson correlation coefficient. This weighted adjacency

matrix cab be viewed also as defining a graph $G = (V, E)$ on a set V of p nodes and a set E of bidirectional edges. We assume in addition that no edge connects a node to itself. The weighted adjacency matrix W_0 gives the connection strength between each pair of nodes. So a gene co-expression network cab be modeled as a weighted undirected graph in which each node corresponds to a gene while the edges describe gene co-expression relationships. In some case we need to select a proper adjacency matrix W of the graph sufficiently sparse to provide a good opportunity of highlighting a cluster structure. For this we will threshold the elements in the similarity matrix by setting elements below a given threshold parameter λ to zero in a hard thresholding scheme. The new adjacency matrix W_λ is then defined by

$$w_{ij}^{\lambda} = w_{ij}\mathbf{1}_{w_{ij}\geq\lambda}\ 1 \leq i, j \leq p$$

Note that the graph associated to W_0 is fully connected while as λ tends to 1, the graph G_λ associated to W_λ consists of singletons.

2.1.2 What is a core structure ?

In previous works, the main issue consists in choosing a good threshold parameter to obtain a unique graph G_λ. Here, we emphasize that that genes in different functional pathways may exhibit different levels of co-expression, involving the formation of subgraphs with different levels of density or similarities. Hence, rather than constructing a single sparse graph to highlight the modular structure, we propose to study the evolution of the sparsity of the graphs defined by removing step by step edge with lowest weight, which amounts to threshold the similarity matrix with a growing parameter. Then we identify the groups of vertices that stay connected themselves while they are disconnected in the graphs obtained for larger threshold values.

We first introduce the notion of central group before defining the core structure notion. We call "n-central group" a set containing at least n nodes, which are strongly connected (hence more similar) between themselves than to other nodes out of the set. So there exists a graph G_λ where this group of nodes forms a connected component (any two vertices are connected to each other by a paths and there is no connection with the vertices out of the component). This notion of coreness might have been introduced for the first time in the context of social networks in [28].

Definition 1 Let $W = (w_{ij})_{1\leq i,j\leq p}$ be a weighted adjacency matrix such as $w_{ij} \in [0, 1]$, $\forall i, j \in \{1, 2, ..., p\}$. For any $n \in \mathbb{N}^*$, a n-central group C is a

subset of elements, $C \subset \{1, 2, ..., p\}$, satisfying the following conditions

(i) $|C| \geq n$

(ii) There exists $\lambda \in [0, 1)$ such that C is a connected component of the graph G_λ.

Then we define a "n-core structure" as a maximal subset of elements that have the property to be a n-central group which can not be split into two or more disjoint n-central groups.

Definition 2 Let $W = (w_{ij})_{1 \leq i,j \leq p}$ be a weighted adjacency matrix such as $w_{ij} \in [0, 1]$, $\forall i, j \in \{1, 2, ..., p\}$. For any $n \in \mathbb{N}^*$, a n-core structure Q is a maximal subset of elements, $Q \subset \{1, 2, ..., p\}$ satisfying the following conditions

(i) Q is a n-central group

(ii) For all $\lambda \in [0, 1]$, the elements of Q cannot be split in two or more connected components of G_λ containing at least n nodes (n-central group).

Intuitively, a core structure is a good candidate to be centrally positioned in a cluster of co-expressed genes. It appears, indeed, that the core structures group the most significantly co-expressed genes at all levels of co-expression.

The algorithm for detecting the core structures (presented in the Methods section) just perform hierarchical clustering by removing edges between pairs of vertices with low similarities and then analyze the evolution of the structures get disconnected from each other.

At the beginning of the algorithm, we start with a graph G_{λ_0} where $\lambda_0 = 0$ which is the fully connected graph ($W_{\lambda_0} = S$). At each step t, a new graph G_{λ_t} is defined by removing the edge with lowest weight on $G_{\lambda_{t-1}}$ i.e. $\lambda_t = \min_{\{i,j|w_{ij}^{t-1} \neq 0\}} w_{ij}^{t-1}$. It is not necessary, in fact, to study all possible graphs G_λ obtained with λ in the set of nonzero zero values of the similarity matrix S. We will only consider paths of maximum capacity or paths maximizing the weight of the minimum edge in the path. So two nodes are in the same connected component of the graph G_λ as long as the threshold parameter λ is lower than the capacity of the maximum capacity path between these nodes. The graph G_t has one more connected component than the graph G_{t-1} if λ_t is equal to the minimum capacity of all maximum capacity paths in G_{t-1}. We will then define the maximum spanning tree

(each path between two nodes is the maximum capacity path between these node) of the graph $G_{\lambda_0} = G_0$ and redefine the weighted adjacency matrix W_{λ_0} such that $w_{ij}^{\lambda_0} = 0$ if there is no edge between nodes i and j on the maximum spanning tree. By this process we improve the running time of the algorithm for core structure detection from $O(p^3)$ to $O(p^2)$.

2.1.3 Clustering based on core structures

Once the n-core structures are detected as previously described, we use it to initialize cluster centers and we propose a simple stepwise algorithm for reconnecting nodes around the core structures and then completing clusters. Suppose we have identify K core structures $Q_1, .Q_2, .., Q_K$. Each cluster is at first defined by a unique core structure, $C_{1,0} = Q_1, C_{2,0} = Q_2, ..., C_{K,0} = Q_K$. At each iteration t, we denote by U_t the set of unclustered element i.e. $i \in U_t \ \Rightarrow \ i \notin C_{k,t-1}, \forall k = 1, 2, ..., K$, and we identify the unclustered element $i \in U_t$ and the clustered element $j \notin U_t$ that have the maximum similarity value i.e. $s_{ij} \geq s_{i'j'}, \ \forall \ i' \in U_t, \ \forall \ j' \notin U_t$. We then put the unclustered element i in the cluster $C_{k,t-1}$ including the element i, $C_{k,t} = C_{k,t-1} \cup \{i\}$ (and $C_{l,t} = C_{l,t-1}, \ \forall l \neq k$). The process is repeated until all elements are clustered and ensures that each node and its first nearest neighbor are necessarily placed in the same cluster.

2.2 Performance evaluation using simulated data

To evaluate the ability of our algorithm to identify clusters or communities that are highly intra-connected in correlation graphs, we simulated correlated data such as the correlation matrix displaying community structure, with communities of varying sizes, densities, and level of noise. We also compared the accuracy and robustness of our algorithm with two very popular clustering algorithms, the spectral clustering algorithm and the WGCNA algorithm successfully applied in genes network clustering context.

2.2.1 A model to simulate co-expression matrix

The goal of the analysis is to study the performances of our algorithm in many different situations, so we need to generate correlation graphs may contain communities or clusters of nodes with various size, intra and inter densities and levels of noise. A co-expression gene network may be modeled by a graph which is defined using microarray expression profiles, that are transformed to a co-expression matrix by calculating pairwise correlations. It is more convenient to simulate expression data instead of co-expression

or correlation matrix directly, to create various source of randomness and obtain realistic co-expression matrices. We propose a model inspired by [16] for simulating expression data sourced from different communities of genes, in such a way to define graphs resemble real gene co-expression network. Many complex networks and in particular, gene co-expression networks, are known to be scale free, meaning that the network has few highly connected nodes (hubs) and many poorly connected nodes. Simulation can be seen as a process through which we first generate expression profiles of leader genes or hubs, one hub for each community, and then, expression profiles of other genes are generated in each community on the basis of the corresponding hub, in order to have more or less high correlation with the hub (few high correlations and many moderate or small correlations in each community). Gene expression data is often assumed to be (log)normally distributed and the absolute value of the Pearson correlation coefficient is commonly used to measure the co-expression between genes.

The aim is to generate gene expression profiles where each profile is a vector of size N. Strategy for simulating gene profiles in a single community of size n_C is as following

1. Generate the expression profile $\mathbf{x}^{(1)} = (x_1^{(1)}, x_2^{(1)}, ..., x_N^{(1)})'$ of one gene from a normal distribution, $x_i^{(1)} \sim \mathcal{N}(0,1), \forall i = 1, 2, ..., N$. This profile defines the leader or hub profile.

2. Choose a minimum correlation r_{min} and a maximum correlation r_{max} between the leader gene and other genes in the community.

3. Generate $n_C - 1$ expression profiles such that the correlation of the j-th profile $\mathbf{x}^{(j)}$ with the leader gene profile $\mathbf{x}^{(1)}$ is forall $j = 2, 3, ..., n_C$ close to
$$r_j = r_{min} + \Delta_r \left(1 - \frac{j}{n_C}\right),$$
where $\Delta_r = r_{max} - r_{min}$. The r_2 correlation is close to r_{max} and the n_C-th correlation is equal to $r_{min} + \Delta_r = r_{max}$.

   The j-th profile is generated by adding a Gaussian noise term to the leader profile, to allow a correlation with the leader profile close to the required correlation r_j
$$x_i^{(j)} = x_i^{(1)} + \sqrt{\left(\frac{1}{r_j^2} - 1\right)} \epsilon_i^{(j)}, \forall i = 1, 2, ..., N$$

where $\epsilon_i^{(j)} \sim \mathcal{N}(0,1)$.

We created six particular scenarios for simulated co-expression data. For each scenario we fixed the size of profile $N = 100$ and we simulated gene profiles within $K = 5$ different communities of sizes n_{C_1},n_{C_2},...,n_{C_K} randomly drawn from the set $\{50, 100\}$. A simulated expression data set X is then composed of $p = n_{C1} + n_{C_2} + ... + n_{C_K}$ gene profiles generated for the five communities with the process described above

$$X = \left(\mathbf{x}_{C_1}^{(1)}, ..., \mathbf{x}_{C_1}^{(n_{C_1})}, \mathbf{x}_{C_2}^{(1)}, ..., \mathbf{x}_{C_2}^{(n_{C_2})}, ..., \mathbf{x}_{C_K}^{(1)}, ..., \mathbf{x}_{C_K}^{(n_{C_K})}\right) \in \mathbb{R}^{N \times p}.$$

The coefficients $s_{ij} \in [0, 1]$ of the $p \times p$ co-expression matrix S are defined as the absolute values of Pearson correlation coefficients between gene profiles.

In a first scenario (S-1), we generated well-defined communities inside which correlations between gene profiles decrease linearly from $r_{max} = 1$ to $r_{min} = 0.5$. For the second scenario (S-2), we have allowed communities to have different levels of density by randomly selecting, for each community, the parameter configuration $\{r_{min} = 0.5,\ r_{max} = 1\}$ or the other configuration $\{r_{min} = 0.4,\ r_{max} = 0.7\}$ (smaller density). The third scenario (S-3) products two communities with high intercorrelation. We generated two leader profiles which have a correlation value close to 0.8 (the three other profiles are independently drawn). Genes inside communities are generated with the parameter configuration $\{r_{min} = 0.5,\ r_{max} = 1\}$. The fourth scenario (S-4) consists in creating noisy data. We simulated expression data using $\{r_{min} = 0.5,\ r_{max} = 1\}$, then we have standardized the data and added to each gene profile a gaussian noise term drawn from a normal distribution $\mathcal{N}(0, 1)$. Another scenario (S-5) was designed to study the effects of the presence of irrelevant variables. A total of p gene profiles are generated with parameters $\{r_{min} = 0.5,\ r_{max} = 1\}$ and p irrelevant variables of size $N = 100$, drawn from a $\mathcal{N}(0, 1)$, are added. The last scenario (S-6) is a mix of the five previous scenarios. We generated two leader profiles which have a correlation value close to r which is drawn from the set $\{0.2, 0.4, 0.6\}$ and the three other profiles are independently drawn. For each community, the parameter configuration is randomly selecting from the two configurations $\{r_{min} = 0.5,\ r_{max} = 1\}$ and $\{r_{min} = 0.4,\ r_{max} = 0.7\}$. Simulated data have been stantardized and then a gaussian noise were added to each profile with a standard deviation drawn from the set $\{0.1, 0.5, 1\}$. Additionnaly we added p irrelevant profiles simulated from a $\mathcal{N}(0, 1)$.

We generated 100 expression data sets for each scenario, and calculated corresponding co-expression matrices S (absolute values of Pearson corre-

lations). Examples of co-expression matrices obtained for the six scenarios are shown in Figure 1.

Figure 1: Example of a co-expression matrix produced for the six scenarios.

2.2.2 Results

We studied the performance of the three clustering algorithms (CSD, spectral and WGCNA) on the 600 simulated co-expression matrices. We needed to define a graph and choose appropriate parameter settings for each clustering algorithm. For the spectral clustering and the CSD algorithm we defined a fully connected graph by taking the weighted adjacency matrix of the graph equal to the co-expression matrix, W=S. For the use of WGCNA algorithm we defined the weighted network adjacency matrix by raising the absolute value of the correlation matrix to a power, $w_{ij} = s_{ij}^{\beta}$, and chose the threshold parameter β by applying the approximate scale-free topology criterion [17]. Spectral clustering was run with $K = 5$ (known parameter), and was repeated 50 times for each data set to avoid local minima. More realistically, the number of classes is unknown, so we tried to find it by running the algorithm for a wide range of number of classes (from 2 to 10) and chose clustering result with the highest value of the Dunn cluster validity index

[9]. The WGCNA algorithm identifies classes using hierarchical clustering and the Dynamic cut algorithm [18], is proposed for automatically detecting clusters in a dendrogram based on shape of its branches. We chose the minimum size of a community equal to 40 and the default value (0.99) for the cut height excepted for the scenario S-4 for which it is better to choose a more high cut height value (0.999). Clustering with the CSD algorithm require to select a unique tuning parameter, the minimum core size, and we chose it equal to 10.

We evaluated the quality of the clusters formed by the three clustering algorithms with the adjust Rand index [19], comparing the clustering results with the true class labels. For all the simulated datasets with the six scenarios, the results are shown in Figure 2 as boxplots of the distribution of the adjust rand index. From S-1 simulated data, it is seen that all methods were able to recover the underlying cluster structure well, when classes are well-defined (high intra-cluster density and low inter-cluster density), even if there is many irrelevant genes (S-5). However, the selection of the optimal number of clusters for the spectral clustering may still failed, especially in the presence of irrelevant genes. Because of using a single power level to compute weighted adjacency matrix, the WGCNA algorithm is clearly less adapted to detect clusters with various intra or inter densities (S-2, S-3), that is, however, a realistic case. Moreover, when cluster may have high inter densities (S-3), the choice of optimal number of clusters for spectral clustering is more difficult. The three algorithms are able to detect clusters in presence of a reasonable level of noise (S-4), but for the WGCNA algorithm it was necessary however, to increase cut height level to have good results. Globally, the clustering accuracy is better with our CSD algorithm (S-6), which is able to detect classes of different densities and less sensitive to choice of the parameter in the presence of noise. The CSD algorithm have also the advantage of allowing a selection of more relevant genes composing the core structures. If clusters are very well defined (S-1), the core structures coincide with the clusters (all the genes are in a core structure), and even if not all genes are in core structures (S-2, S-3, S-4, S-5), a minority of them are excluded (values of rand index in (b) boxes remain high). The identification of core structures offer the possibility to exclude irrelevant genes, while many irrelevant genes are not excluded with the WGCNA algorithm (see (f) and (g) boxes for S-5 and S-6 scenarios).

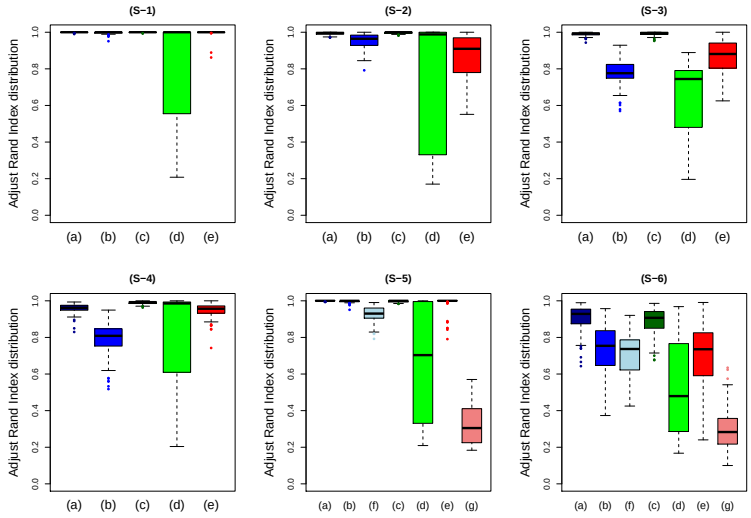


Figure 2: Results on simulated data. Evaluation of the clustering methods (CSD, spectral clustering and WGCNA) on the simulated datasets obtained with the six scenarios. The (a) boxes report adjust rand index for the CSD clustering results. The (b) boxes report adjust rand index for the core structure labels identify with the CSD algorithm (one cluster is composed of the outer genes of the core structures). Evaluations of spectral clustering results correspond to the (c) boxes when the number of clusters is known, and to the (d) boxes when the number of clusters is chosen based on the Dunn's index. The (e) boxes are reserved for WGNCA result evaluations. For the S-5 and S-6 scenarios that contain irrelevant dimensions, we evaluate performances of algorithms only on the basis of relevant gene profiles, and the adjust rand index is computed by removing irrelevant genes for the (a), (b), (c), (d) and (e) boxes. Our CSD algorithm allows to select much relevant genes or the genes in the core stuctures. We then evaluated in (f) boxes the agreement between the irrelevant profiles and the profiles that are out of the core structures. Thus there is one class label reserved to the outer genes of the core structures for the CSD clustering results, one class label reserved to the irrelevant genes for the true cluster labels and the rand index is computed on all genes. The WGCNA algorithm also enables us to exclude irrelevant variables, and as describe above, we evaluated in (g) boxes the ability of the algorithm to detect irrelevant variables

2.3 Performance evaluation on real gene expression data

Using simulated data we have shown that the CSD algorithm has very good performance and tends to outperform the spectral and WGCNA algorithms to cluster high network. Now we are going to look at the performance of CSD algorithm on real gene expression data. For this purpose we use the well-known synchronized yeast cell cycle data set of Spellman et al. [29], which include 77 samples under various times during the cell cycle and a total of 6179 genes, of which 1660 genes are retained for this analysis after preprocessing and filtering. We deleted genes with missing values more than 20% or standard deviation less than 0.4 and missing values were imputed by the KNN algorithm [32]. The goal of cluster analysis is to identify the correlation patterns in the time series of gene expressions of yeast measured along the cell cycle. We defined the elements of the weighted adjacency matrix of the graph as the absolute values of Spearman rank correlations between gene expression levels. We conducted sensitivity analysis, used to test and compare algorithm robustness to parameter settings, and evaluated clustering results based on internal and external criteria.

2.3.1 Parameter sensitivity analysis

The greatest difficulty in the field of data clustering is the need for input parameters which can greatly influence the result. In many applications the optimal values of these parameters are very difficult to determine. To illustrate the robustness of our CSD algorithm, we analysed the effect that changes in parameter settings have on clustering results compared to the input sensitivity of spectral and WGCNA algorithm.

The CSD and spectral algorithms need a unique tuning parameter while WGCNA algorithm needs more parameters. Indeed, the choice of a threshold parameter is first required to define the adjacency matrix and then, more than one other parameters are required to detect clusters in the dendrogram with a cut method less inflexible and more adapted than the use of a constant height cutoff value. For both CSD and WGCNA algorithm, the set of clusters returned are hierarchically structured and the input parameters control the granularity of the clustering results. If the granularity is too fine, there will be too many small clusters with a high correlation among their elements, and conversely, a too low granularity leads to product a few large clusters with a low correlation among their elements. A well-suited parameter setting may be recognizable by visual inspection of the evolution

of the level of granularity at different input parameters. The height and the minimum size of cluster parameters of the dynamic tree cut method (WGNCA) provide improved flexibility for tree cutting but it is not easy to find optimal cutting parameters (Figure 3).

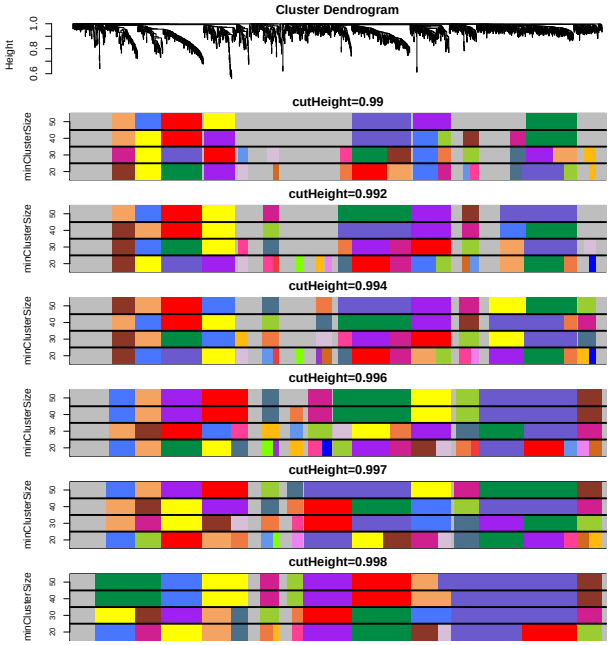


Figure 3: WGCNA algorithm applied to yeast cell-cycle gene expression data set. Representation of the hierarchical cluster tree and representation of the cluster results for various dynamic tree cut input parameters (height cut and minimum size of clusters), which are shown in the color bands. The gray color is reserved for non-clustered genes.

The CSD algorithm leads to embedded clusters and the number of clusters is monotonically decreasing as the parameter n increases, so clustering results with well-fitted levels of granularity can be quickly and easily identify (Figure 4).

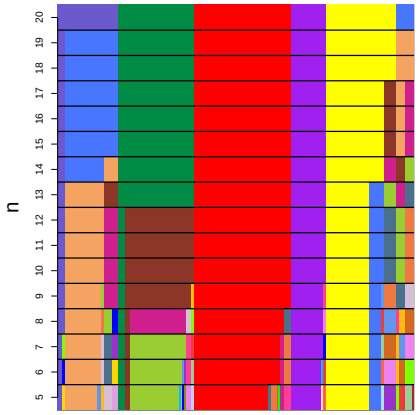


Figure 4: CSD algorithm applied to yeast cell-cycle gene expression data set for various values of the minimum size of the core structures. The color bands on the x-axis shows the cluster membership according to different parameter (n) settings on the y-axis.

The problem of the right choice of the number clusters for spectral clustering algorithm is crucial and still open issue. Even if a variety of methods have been devised for this problem, most of this methods works well if the data contains very well pronounced clusters, and results are more ambiguous in cases of noisy or overlapping clusters. It should also be noted that the spectral clustering result using k-means algorithm depends largely on the initial set of centroids, so many iterations using random initial values are needed to have good performance (we used 500 random initial values).

To illustrate the robustness of the CSD algorithm to changes of the parameter settings, and to compare with the robustness of WGCNA and spectral clustering, we computed the adjust rand index between clustering results obtained for consecutive values of the input parameters (Figure 5). We show that our algorithm is less sensitive to changes in the input parameter and relatively easy to setup, while parameter setting is less obvious with the two other algorithms. Clustering results with spectral clustering are significantly influenced by the clusters number, and the dynamic tree cut method need to specify two input parameters would affect both detection of clusters, should have a wide range of densities, and the number of non-clustered elements.

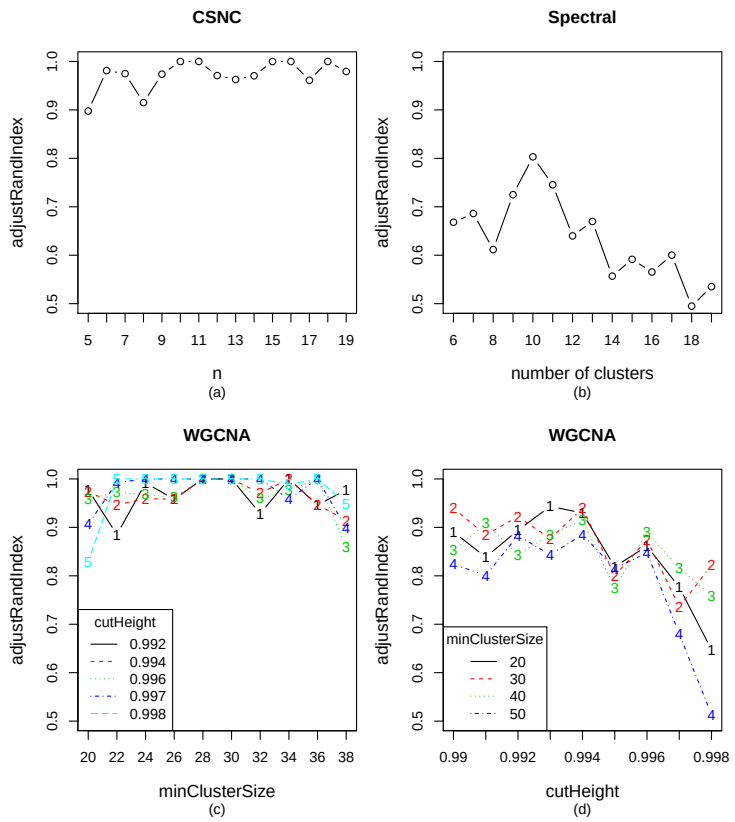


Figure 5: Parameter sensitivity analysis on yeast cell-cycle gene expression dataset. The adjust rand index was computed to compare the clustering result produced for a given input parameter (on the x-axis) with the consecutive result obtained for an increase in the input parameter.

2.3.2 Internal and external evaluations

We are interested here in evaluating the performance of the clustering algorithms using internal and external criteria. Intuitively, based on the data itself (internal clustering), a good clustering will have a high internal density

(clusters having more edges linking their elements among themselves) and sparser external connections. A lot of evaluation metrics have been proposed to quantify the internal quality of a given clustering result. Most of them are however imperfect because they are typically biased toward one algorithm or the other, but may be used as indicative of cluster quality. We compared algorithms results based on four popular quality metrics, the Dunn's index [9], the modularity [22], the Silhouette width [24], and the Figure Of Merit (FOM) [37]. We will have a more rigorous evaluation of the performance of algorithms by using diversified validation measures. We used the adjust Rand index as external validation indices to measure the agreement between clustering results and the biological classes established by Spellman et al. related to five different "phase groups" during the cell cycle

phase group term	**G1**	**G2**	**M**	**M/G1**	**S**
group size	211	63	133	82	42

We compared the results obtained by the three algorithms while varying the number of cluster. There is a monotonic relationship between the input parameter of the CSD algorithm and the number of cluster formed, which permits an easy comparison with spectral clustering results. We used the dynamic tree cut algorithm to cut the hierarchical tree obtained with the WGCNA algorithm. The height parameter of the dynamic tree cut method has been chosen sufficiently high (0.996 or 0.997) to have a maximum number of clustered genes and we controlled the number of clusters by varying the minimum cluster size parameter. The subset of genes (1622 genes) clustered with WGCNA algorithm has been used for internal evaluation of algorithm results. Out of these genes, 531 are found in gold standard classes established by Spellman and have been used for external clustering validation.

Our CSD algorithm achieves the best compromise between all internal validation indices (Figure 6) and gives the most biologically meaningful gene groups consistent with previous knowledge (Figure 7).

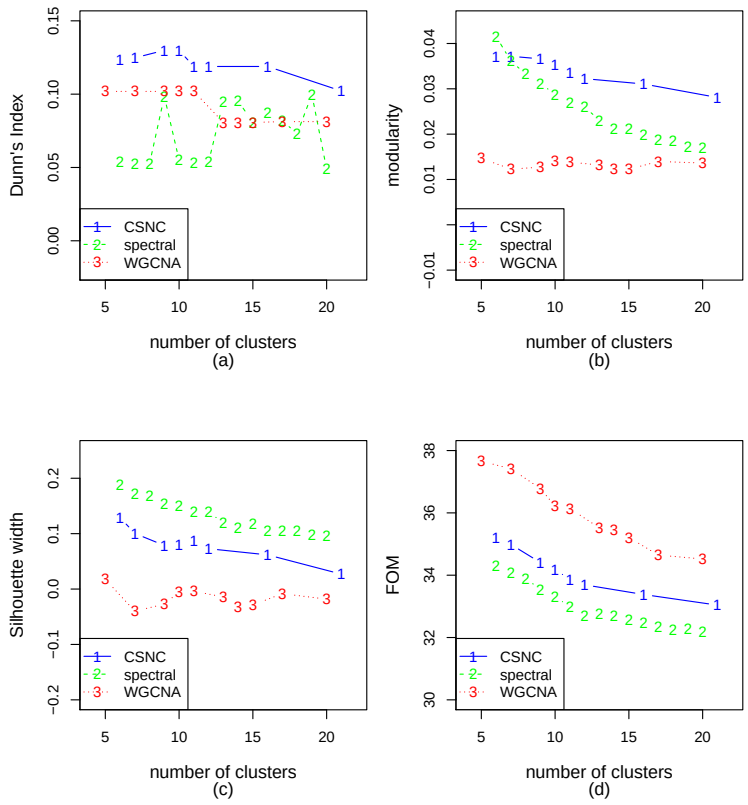


Figure 6: Internal evaluation of clustering algorithms on the yeast cell-cycle gene expression data (1622 genes). (a) Dunn's index. (b) Modularity. (c) Silhouette width. (d) Figure Of Merit.

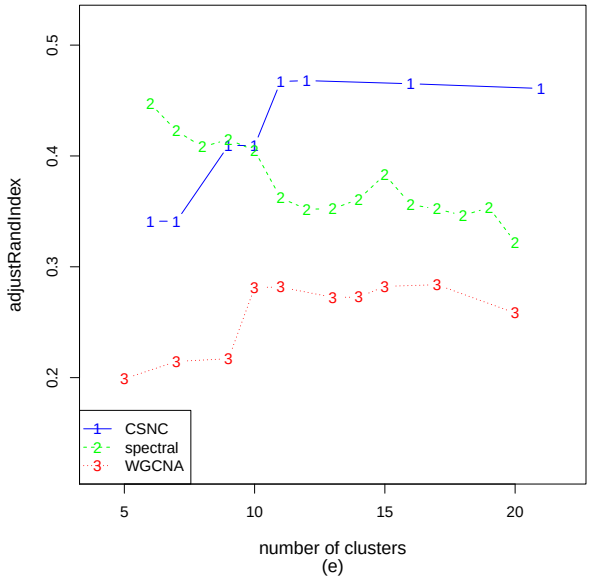


Figure 7: External evaluation of clustering algorithms on the yeast cell-cycle gene expression data (531 genes). The adjust rand index to compare clustering results with the gold standard classes established by Spellman.

2.3.3 Functional validation based on gene ontology

One of the common assertion in expression analysis is that genes sharing a similar pattern of expression are more likely to be involved in the same regulatory processes [10]. Prior knowledge about the biological functions of some genes are available in the Gene Ontology (GO) database which is organized in three levels of detail for the decription of the biological process, molecular function, and cellular component of gene products. We have limited our analysis to the GO terms which describe biological processes. Each gene may be associated to one or more biological process GO term. We performed enrichment analysis to identify biological processes over-represented in each community of genes [7]. The enrichment test is based on the hypergeometric test or, equivalently, the one-tailed Fisher Exact Test, indicates if a given GO term appears in a sublist of genes (genes within a given community) higher than expected by chance [30].

Table 1 presents the three more enriched biological process that have a Bonferroni corrected p-value ≤ 0.05 within each cluster produced by the CSD algorithm for a minimum core size parameter equal to 10 (12 clusters). We confirmed that the results produced by the proposed method are reliable according to the Spellman et al. results. The cluster 5 of genes involved in DNA repair contains 70% of genes in group phase G1 and all the genes group in the CLN2 cluster by Spellman and al. The small cluster 10 contains all genes placed in the Y cluster by Spellman and al. and 15% of genes in group phase G1. Following G1 phase during S phase, nucleosomes are assembled, using histones. The cluster 6 is composed by genes involved in this phase (contains 81% of genes in group S and all genes identified by Spellman in the histone cluster). The cluster 11 groups genes involved in the tricarboxylic acid cycle, an essential cycle in aerobic growth, and contains 33% of genes group phase G2. The cluster 4 contains genes involved in mitosis (71% of genes in group M). We found in this cluster the genes in the Spellman's cluster CBL2 and in the cluster MCM which is induced by CLB2. The cluster 12 contains genes involved in cytokinesis and which reach peak in late M or at the M/G1 boundary. All genes in the Spellman's cluster SIC1 and 30% of genes in M/G1 are placed in this cluster.

cluster	size	GO Term	description	count	count in cluster	pvalue (Bonferroni)
1	36	GO:0002181	cytoplasmic translation	55	16	1.719e-13
1	36	GO:0006412	translation	63	16	1.927e-12
1	36	GO:0006414	translational elongation	12	5	0.0004
2	33	GO:0006457	protein folding	28	16	3.928e-20
2	33	GO:0006950	response to stress	53	16	1.496e-14
2	33	GO:0042026	protein refolding	10	9	2.713e-13
3	450	GO:0030476	ascospore wall assembly	23	18	7.182e-05
3	450	GO:0007126	meiotic nuclear division	48	29	0.0002
3	450	GO:0008150	biological_process	250	99	0.0004
4	163	GO:0007049	cell cycle	104	29	8.249e-06
4	163	GO:0006811	ion transport	33	15	1.625e-05
4	163	GO:0055072	iron ion homeostasis	12	9	2.155e-05
5	203	GO:0006281	DNA repair	40	24	6.995e-11
5	203	GO:0006260	DNA replication	29	20	1.596e-10
5	203	GO:0007064	mitotic sister chromatid cohesion	15	13	1.491e-08
6	70	GO:0006334	nucleosome assembly	12	9	9.215e-09
6	70	GO:0009086	methionine biosynthetic process	9	8	1.074e-08
6	70	GO:0006333	chromatin assembly or disassembly	10	8	5.194e-08
7	184	GO:0042254	ribosome biogenesis	80	67	8.431e-54
7	184	GO:0006364	rRNA processing	63	53	1.339e-41
7	184	GO:0042273	ribosomal large subunit biogenesis	21	21	5.542e-19
8	321	GO:0055114	oxidation-reduction process	111	49	8.714e-08
8	321	GO:0032543	mitochondrial translation	12	11	2.245e-05
8	321	GO:0009060	aerobic respiration	18	11	0.019
9	64	GO:0002181	cytoplasmic translation	55	32	1.296e-32
9	64	GO:0006412	translation	63	30	1.119e-26
9	64	GO:0006407	rRNA export from nucleus	8	6	1.261e-05
10	43	GO:0008150	biological_process	250	25	8.347e-09
10	43	GO:0000722	telomere maintenance via recombination	13	7	1.332e-06
11	53	GO:0006099	tricarboxylic acid cycle	16	5	0.016
12	40	GO:0007109	cytokinesis, completion of separation	8	5	6.156e-05

Table 1: Enrichment analysis of the 12 clusters produced by the CSD algorithm.

2.4 The core structures include the most biologically informative genes

We have shown previously that our CSD algorithm is able to identify biologically meaningful gene clusters from a gene co-expression network. Now, we will show the biological relevance of the core structures. It is well accepted that genes with critical functional roles are centrally positioned on the gene co-expression network and have high connectivity (hubs). The core structures by definition contain highly connected or co-expressed genes, and so have the best chance of containing potential biologically meaningful genes.

We studied the core structures constructed by the CSCN algorithm on the

yeast cell-cycle gene expression data set of Spellman et al., and compared the biological interpretation of the core structures with interpretations based on clustering analysis of all the genes presented in the previous section. Within the core structure we found 472 genes out of a total of 1660 genes. We will make comparisons based on two levels of information, enriched GO terms and detection of cell cycle-regulated genes identified by Spellman. We show in Table 2 the three more enriched biological process that have a Bonferroni corrected p-value ≤ 0.05 within each core stucture produced by the CSD algorithm for a minimum core size parameter equal to 10 (12 clusters). Annotated biological process for core structures (Table 2) and for entire clusters (Table 1) are very similar, demonstrating that the core structures hold the most meaningful information. In addition to that, the core structures highlight most of the genes identified by Spellman to be cell cycle-regulated. Among the 472 genes located in the core structures, we found 85% of the 156 genes identified in the previous analysis on the 1660 genes and clustered in the CLN2, Y, histone, CBL2, MCM and SIC1 by Spellman.

core structure	size	GO Term	description	count	count in cluster	pvalue (Bonferroni)
1	19	GO:0002181	cytoplasmic translation	55	15	5.383e-18
1	19	GO:0006412	translation	63	15	5.423e-17
1	19	GO:0006414	translational elongation	12	5	1.544e-05
2	12	GO:0006457	protein folding	28	7	2.397e-08
2	12	GO:0006950	response to stress	53	7	2.918e-06
2	12	GO:0042026	protein refolding	10	3	0.0062
3	106	GO:0030435	sporulation resulting in formation of a cellular spore	36	10	0.0085
4	62	GO:0007049	cell cycle	104	22	3.116e-10
4	62	GO:0051301	cell division	62	13	2.85e-05
4	62	GO:0007067	mitotic nuclear division	38	9	0.001
5	61	GO:0007049	cell cycle	104	16	6.417e-05
5	61	GO:0006260	DNA replication	29	9	6.957e-05
5	61	GO:0006974	cellular response to DNA damage stimulus	46	10	0.0005
6	13	GO:0006334	nucleosome assembly	12	9	1.114e-16
6	13	GO:0006333	chromatin assembly or disassembly	10	8	7.538e-15
7	101	GO:0042254	ribosome biogenesis	80	57	1.779e-56
7	101	GO:0006364	rRNA processing	63	45	7.685e-43
7	101	GO:0042273	ribosomal large subunit biogenesis	21	18	6.041e-18
9	14	GO:0002181	cytoplasmic translation	55	13	3.277e-17
9	14	GO:0006412	translation	63	10	4.906e-10
10	31	GO:0008150	biological_process	250	22	3.714e-10
10	31	GO:0000722	telomere maintenance via recombination	13	6	7.367e-06
11	17	GO:0031505	fungal-type cell wall organization	33	4	0.047
12	17	GO:0007109	cytokinesis, completion of separation	8	5	5.996e-07

Table 2: Gene Ontology enrichment analysis for the 12 core structures produced by the CSD algorithm applied on the yeast cell-cycle gene expression data set of Spellman et al.

In addition, we have represented on Figure 8 the induced subgraph of the graph containing nodes included in the core structures, by keeping only edges with a weight larger than 0.75, to have a sparse representation of this subgraph. This simple representation gives us an idea of the interactions between the different core structures, in line with biological knowledge. Indeed, it is very interesting to observe a ring describing the phases of the cell cycle. Core structures have been previously associated with one phase of the cell cycle and are placed on the ring in the order of the cycle that suggests that genes in a given core structure interact strongly with genes in other core structures associated to the same phase or consecutive phases of the cell cyle, which makes sense.

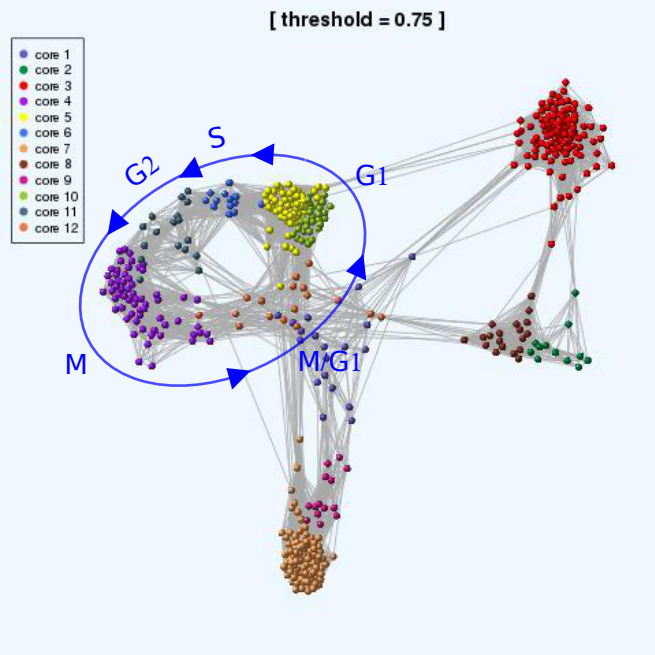


Figure 8: Gene co-expression network of genes included in the core structures produced by the CSD algorithm applied on the yeast cell-cycle gene expression data set of Spellman et al. Each core structure is shown with a different color and an edge connect two nodes if its weight (similarity between the two genes) is greater than 0.75.

3 Conclusions

Investigation of complex interaction patterns among genes is an effective way for understanding complex molecular process, predicting gene functions, finding genes with critical functional roles.The discovery of clusters of co-expressed genes and centrally positioned genes is of fundamental and practical interest. In this paper, we presented a promising algorithm for both the analysis of the community structure in large interaction networks and the selection of subgroups of genes centrally positioned in the network. Several method for network clustering exist. However, they have their limitations and require to make assumptions can be restrictive in practice. For example, the WGCNA algorithm (hierarchical clustering) needs a sparse estimation of the graph by thresholding the input similarity matrix (e.g. absolute value of Pearson correlations), needs to choose an appropriate similarity measure between nodes of the graph and needs to select other parameters for tree cutting. These choices can have a large impact on the results. Another common method for graph clustering is the spectral clustering (partitional clustering). The main limitation of this method is the difficult problem of determining the number of clusters. Another drawback of this algorithm is that it tends to form clusters of homogeneous sizes.

Our Core Structures based Network Clustering (CSD) algorithm combines advantages of existing methods and reduce the bias introduced by limiting the number of input parameter. Initially, the variables (e.g. gene expressions) are organized in a similarity matrix (e.g. absolute value of Pearson correlations). The first step of the algorithm is to identify group of variables preferentially connected, called core structures, on the collection of all neighborhood graphs obtained by similarity thresholding. The algorithm doesn't need to determine an unique threshold parameter for graph estimation by studying all possible graphs. A unique input parameter is required which is the minimum size of core structures. This parameter controls the level of granularity of the results (embedded results) and this is easy to choose it in order to have not too many small clusters and not too large clusters. The second step of the algorithm consists in clustering all the variables based on the core structures knowledge. Each cluster is composed of a unique core structure and nodes associated with this core structure in the iterative process consists of selecting the edge of maximum weight between one clustered element and one unclustered element and attributing the cluster label of the clustered element to the unclustered element.

We evaluated performances of our CSD algorithm and compared with WGCNA and spectral clustering algorithms using simulated data and public gene expression data of yeast cell cycle have been studied by Spellman [29]. We showed that our algorithm outperforms other algorithms based of internal and external criteria, is less sensitive to the input parameter, is easy to use and is able to select groups of centrally positioned genes in the network or core structures contain informative genes.

4 Methods

4.1 Network clustering

4.1.1 CSD algorithm

Let $S = (s_{ij})_{1 \leq i,j \leq p}$ be a similarity matrix which defines a fully-connected graph G whose weighted adjacency matrix is $W = S$. The input parameter or the minimum size of a core structure is denoted by n. Our CSD algorithm is described in algorithm 1. There are three main steps for CSD algorithm, identification of the maximum spanning tree of G (e.g. Prim's algorithm), identification of the core structures (algorithm 2) and clustering based on the core structures knowledge (algorithm 3).

Algorithm 1 CSD algorithm

1: **procedure** CSD(W,n) ▷ *W is the weighted adjacency matrix ; n is the minimum size of a core structure*
2: $T \leftarrow$ MaximumSpanningTree(W) ▷ *algorithm (Prim's algorithm) returns the weighted adjacency matrix of the maximum spanning tree*
3: $Q \leftarrow$ CORE(T,N)
4: $\{C_1, ..., C_K\} \leftarrow$ CLUST(Q,T)
5: **return** $\{C_1, ..., C_K\}$ ▷ *The family of clusters*
6: **end procedure**

4.1.2 Spectral clustering

Spectral clustering [23, 33] consists in changing the representation of the initial data set to enhance the clustering structure and make it easy to detect via standard algorithms, like k-means clustering, in the new representation. The main tool for spectral clustering is the graph Laplacian matrix. Let $W = (w_{ij})_{1 \leq i,j \leq p}$ be the weighted adjacency matrix of the graph. The degree matrix D is a diagonal matrix, whose diagonal is the degree for each node,

Algorithm 2 Core structure algorithm

procedure CORE(W,n) ▷ $W \in \mathbb{R}^{p \times p}$
  $Q \leftarrow \{1, ..., p\}$ ▷ *Q is the family of core structures*
  while $|\{(i,j)\ ;\ i,j \in \{1,...,p\},\ w_{ij} > 0\}| > 2(N-1)$ **do**
    Selecting the lowest cost edge (i,j) of the graph defined by W
    $w_{ij} \leftarrow 0$
    $V_i \leftarrow \text{DFS}(W,i)$ ▷ *Depth First Search algorithm (DFS) returns nodes in connected component of the graph defined by W containing i*
    $V_j \leftarrow \text{DFS}(W,j)$
    if $|V_i| \geq n$ and $|V_j| \geq n$ **then**
      Selecting the core structure $S \subset Q$ containing $V_i \cup V_j$
      $Q \leftarrow Q \setminus S$
      $Q \leftarrow \{Q, V_i, V_j\}$
    else
      if $|V_i| \leq n$ **then**
        for all $u, v \in V_i$ **do** $w_{uv} \leftarrow 0$
        end for
      end if
      if $|V_j| \leq n$ **then**
        for all $u, v \in V_j$ **do** $w_{uv} \leftarrow 0$
        end for
      end if
    end if
  end while
  return Q ▷ *The family of core structures*
end procedure

Algorithm 3 Clustering algorithm

procedure CLUST(Q,W) ▷ $Q = \{Q_k; k = 1, ..., K\}$
  for all $k = 1, ..., K$ **do**
    $C_k \leftarrow Q_k$
  end for
  $U \leftarrow \{i \ ; \ i \in \{1, ..., p\}, \ i \notin Q_k, \forall k = 1, ...K\}$
  while $U \neq \emptyset$ **do**
    Selecting the greatest cost edge (i, j) of G such that one node is in U and the other node is in C_k $(k = 1, ..., K)$. Suppose that $i \in U$ and $j \in C_k$.
    $C_k \leftarrow C_k \cup \{i\}$
    $U \leftarrow U \setminus \{i\}$
  end while
  return $\{C_1, ..., C_K\}$ ▷ *The family of clusters*
end procedure

the sum of the weights of edges are incident with the node, $d_{ii} = \sum_{j \neq i} w_{ij}$. We applied spectral clustering method by using the normalized Laplacian L_{rw} [6] defined as $L_{rw} = D^{-1}L$ where $L = D - W$ is the unnormalized Laplacian. The change of the representation is induced by the eigenvectors of L_{rw}. In order to partition a graph into K classes, we compute with the first K eigenvectors (corresponding to the K smallest eigenvalues) of the normalized Laplacian L_{rw} and cluster the points in $\mathbb{R}^K$ whose coordinates are elements of eigenvectors, with the k-means algorithm into K clusters (R eigen() and kmeans() functions).

4.1.3 WGCNA

All the needed functions for network clustering in WGCNA procedure [39] are available in the WGCNA R software package [17]. The weighted adjacency matrix $W = (w_{ij})_{1 \leq i,j \leq p}$ of the graph is defined by transforming the initial similarity matrix $S = (s_{ij})_{1 \leq i,j \leq p}$ using a procedure consists in raising the similarity to a power, $w_{ij} = s_{ij}^{\beta}$. To choose the threshold parameters β we applied the proposed approximate scale-free topology criterion. As suggested by Langfelder and Horvath, we performed hierarchical clustering using the topological overlap measure (TOM) as input similarity, and used the Dynamic Tree cut algorithm to extract clusters from the dendrogram.

4.2 Cluster evaluation

4.2.1 Internal evaluation

The Dunn's index [9] quantifies the goodness of a clustering, by measuring the maximal diameter of clusters and relating it to the minimal distance between clusters. The Dunn's index D_K for a given clustering partition, $C_1, ..., C_K$ is defined as follows

$$D_K = \frac{\min\limits_{1 \leq i < j \leq K} \delta(C_i, C_j)}{\max\limits_{1 \leq k \leq K} \Delta_k}, \tag{1}$$

where K is the number of clusters, $\delta(C_i, C_j)$ is the intercluster dissimilarity between C_i and C_j defined as $\delta(C_i, C_j) = \min\limits_{x \in C_i, y \in C_j} d(x, y)$, and Δ_k is the diameter of the cluster C_k defined as $\Delta_k = \max\limits_{x,y \in C_k} d(x, y)$. Thus, high Dunn's index values indicate the presence of clusters such that the distances between the clusters are high and the diameter of the clusters are small. The Dunn's index does not exhibit any trend with respect to number of clusters and can be used as an indication to help the choice of the number of clusters for a clustering algorithm needs to tune this parameter.

The Silhouette width [24] is another index that combine the compactness and the separation measures. A silhouette value is measured for each node of the graph to quantify the degree of confidence in the clustering assignment of the nodes. For one node i, the silhouette is defined as follows

$$S(i) = \frac{b_i - a_i}{\max(b_i, a_i)}, \tag{2}$$

where a_i is the average dissimilarity between i and all the other nodes in the same cluster, and b_i is the average dissimilarity between i and the observations in the nearest cluster

$$b_i = \min_{\{C_k; i \notin C_k\}} \sum_{j \in C_k} \frac{d(i, j)}{|C_k|},$$

where $d(i, j)$ is the dissimilarity between nodes i and j. The Silhouette width S is then the average value of each node's silhouette value, should be maximized and lies in the interval $[-1, 1]$.

A figure of merit (FOM) [37] is an estimate of the predictive power of

a clustering algorithm. A typical gene expression data set contains measurements of expression levels of p genes in n samples. The gene expression data matrix is denoted by $X \in \mathbb{R}^{n \times p}$. Suppose there are K clusters, $C_1, C_2, ..., C_K$. Let x_{ij} be the expression level of the gene j in the sample i, and $\mu_i^{C_k} = \frac{1}{|C_k|} \sum_{j \in C_k} x_{ij}$ be the average expression level in the sample i of genes in cluster C_k. This mean expression $\mu_i^{C_k}$ is used as if it was a prediction of expression level in the sample i for each gene in the cluster C_k and is compared with the known values of expression in this sample for all these genes. The figure of merit FOM is defined as follows

$$FOM = \sum_{i=1}^{n} \sqrt{\frac{1}{p} \sum_{k=1}^{K} \sum_{j \in C_k} (x_{ij} - \mu_i^{C_k})^2}. \tag{3}$$

The lower the FOM value is, the higher the predictive power of the algorithm is.

We used the dissimilarity measure $d(i, j) = 1 - x_{ij}$ for both the dunn's index and the silhouette width, where $w_{ij} \in [0, 1]$ is element of the weighted adjacency matrix (absolute values of Pearson or Spearman correlations). The Dunn's index, the silhouette width and the FOM are implemented in the R package clValid [3].

The modularity of Newman and Girvan [22] is the most popular quality function for graph clustering that measures the non-randomness of a graph partition. This index has been used in many algorithms as a quality function or as an objective function [21, 25, 26, 36]. The idea behind the modularity index is that a graph have a communitity structure if it is different from a random graph or a null model is not expected to have communitity structure. The null model proposed by Newman and Girvan consists of a randomized version of the original graph, where edges are placed at random, under the constraint that the expected degree of each node matches the degree of the node in the original graph. The modularity index Q is defined as follows

$$Q = \sum_{i=1}^{K} \left[\frac{l_i}{m} - \left(\frac{d_i}{2m} \right)^2 \right], \tag{4}$$

where K is the number of clusters, and for the weighted version of the modularity, m is the sum of weights of all edges of the graph, l_i is the sum of weights of edges connecting vertices of module C_i and d_i is the sum of the

degrees of the vertices of C_i. A high modularity means that there are more edges within the clusters that you expect by chance. Modularity is always smaller than one and can be negative.

4.2.2 External evaluation

To compare the clusters identified by an algorithm to the gold standard clusters, we computed the adjusted Rand index [19]. Suppose that $A_1, A_2, ..., A_{K_A}$ and $B_1, B_2, ..., B_{K_B}$ represent two different partitions of the nodes of the graph. Let $n = |V|$ be the number of nodes in the graph and n_{ij} be the number of nodes that are both in cluster A_i and in cluster B_j. Let $n_{i.} = |A_i|$ and $n_{.j} = |B_j|$ be the number of nodes in cluster A_i and in cluster B_j respectively. The adjusted Rand index is defined as follows

$$\frac{\sum_{i,j}\binom{n_{ij}}{2} - \left[\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}\right] - \left[\sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}\right]/\binom{n}{2}} \tag{5}$$

The adjusted Rand index lies in the interval $[0, 1]$ and is equal to 1 when the two partitions agree perfectly.

4.2.3 Functional evaluation

To assess the functional significance of gene clusters, we analysed the enrichment of GO terms for the genes within each cluster. We used the R packages org.Sc.sgd.db [5] and GO.db [4] R package in Bioconductor to identify GO terms associated with the genes. We measured the statistical significance of the GO term enrichment by an hypergeometric test (R phyper() function). The p-values are adjusted by Bonferroni corrections for multiple testing problems.

References

[1] Dominic J Allocco, Isaac S Kohane, and Atul J Butte. Quantifying the relationship between co-expression, co-regulation and gene function. *BMC bioinformatics*, 5(1):18, 2004.

[2] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics*, 5(2):101–113, 2004.

[3] Guy Brock, Vasyl Pihur, Susmita Datta, and Somnath Datta. clvalid, an r package for cluster validation. *Journal of Statistical Software (Brock et al., March 2008)*, 2011.

[4] Marc Carlson. *GO.db: A set of annotation maps describing the entire Gene Ontology*. R package version 3.0.0.

[5] Marc Carlson. *org.Sc.sgd.db: Genome wide annotation for Yeast*. R package version 3.0.0.

[6] Fan RK Chung. *Spectral graph theory*, volume 92. American Mathematical Soc., 1997.

[7] Gene Ontology Consortium et al. The gene ontology (go) database and informatics resource. *Nucleic acids research*, 32(suppl 1):D258–D261, 2004.

[8] Matthew V DiLeo, Gary D Strahan, Meghan den Bakker, and Owen A Hoekenga. Weighted correlation network analysis (wgcna) applied to the tomato fruit metabolome. *PLoS One*, 6(10):e26683, 2011.

[9] Joseph C Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. 1973.

[10] Patrik D'haeseleer, Shoudan Liang, and Roland Somogyi. Genetic network inference: from co-expression clustering to reverse engineering. *Bioinformatics*, 16(8):707–726, 2000.

[11] Michael B Eisen, Paul T Spellman, Patrick O Brown, and David Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95(25):14863–14868, 1998.

[12] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[13] Xionglei He and Jianzhi Zhang. Why do hubs tend to be essential in protein networks. *PLoS Genet*, 2(6):e88, 2006.

[14] Alexander E Ivliev, Peter AC't Hoen, and Marina G Sergeeva. Coexpression network analysis identifies transcriptional modules related to proastrocytic differentiation and sprouty signaling in glioma. *Cancer Research*, 70(24):10060–10070, 2010.

[15] Hawoong Jeong, Sean P Mason, A-L Barabási, and Zoltan N Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, 2001.

[16] Peter Langfelder and Steve Horvath. Eigengene networks for studying the relationships between co-expression modules. *BMC systems biology*, 1(1):54, 2007.

[17] Peter Langfelder and Steve Horvath. Wgcna: an r package for weighted correlation network analysis. *BMC bioinformatics*, 9(1):559, 2008.

[18] Peter Langfelder, Bin Zhang, and Steve Horvath. Defining clusters from a hierarchical cluster tree: the dynamic tree cut package for r. *Bioinformatics*, 24(5):719–720, 2008.

[19] Glenn W Milligan and Martha C Cooper. A study of the comparability of external criteria for hierarchical cluster analysis. *Multivariate Behavioral Research*, 21(4):441–458, 1986.

[20] Mark EJ Newman. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330, 2004.

[21] Mark EJ Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6):066133, 2004.

[22] Mark EJ Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[23] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 2:849–856, 2002.

[24] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65, 1987.

[25] Jianhua Ruan, Angela K Dean, and Weixiong Zhang. A general co-expression network-based approach to gene expression analysis: comparison and applications. *BMC systems biology*, 4(1):8, 2010.

[26] Jianhua Ruan and Weixiong Zhang. An efficient spectral algorithm for network community discovery and its applications to biological and social networks. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 643–648. IEEE, 2007.

[27] Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1):27–64, 2007.

[28] Stephen B. Seidman. Network structure and minimum degree. *Social Networks*, 5(3):269–287, 1983.

[29] Paul T Spellman, Gavin Sherlock, Michael Q Zhang, Vishwanath R Iyer, Kirk Anders, Michael B Eisen, Patrick O Brown, David Botstein, and Bruce Futcher. Comprehensive identification of cell cycle–regulated genes of the yeast saccharomyces cerevisiae by microarray hybridization. *Molecular biology of the cell*, 9(12):3273–3297, 1998.

[30] Aravind Subramanian, Pablo Tamayo, Vamsi K Mootha, Sayan Mukherjee, Benjamin L Ebert, Michael A Gillette, Amanda Paulovich, Scott L Pomeroy, Todd R Golub, Eric S Lander, et al. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[31] Robert Tibshirani, Guenther Walther, and Trevor Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2):411–423, 2001.

[32] Olga Troyanskaya, Michael Cantor, Gavin Sherlock, Pat Brown, Trevor Hastie, Robert Tibshirani, David Botstein, and Russ B Altman. Missing value estimation methods for dna microarrays. *Bioinformatics*, 17(6):520–525, 2001.

[33] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[34] Xiao Fan Wang and Guanrong Chen. Complex networks: small-world, scale-free and beyond. *Circuits and Systems Magazine, IEEE*, 3(1):6–20, 2003.

[35] Matthew T Weirauch. Gene coexpression networks for the analysis of dna microarray data. *Applied Statistics for Network Biology: Methods in Systems Biology*, pages 215–250, 2011.

[36] Scott White and Padhraic Smyth. A spectral clustering approach to finding communities in graph. In *SDM*, volume 5, pages 76–84. SIAM, 2005.

[37] Ka Yee Yeung, David R. Haynor, and Walter L. Ruzzo. Validating clustering for gene expression data. *Bioinformatics*, 17(4):309–318, 2001.

[38] Andy M Yip and Steve Horvath. Gene network interconnectedness and the generalized topological overlap measure. *BMC bioinformatics*, 8(1):22, 2007.

[39] Bin Zhang and Steve Horvath. A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, 4(1), 2005.

2.7.5 Brevet

L'approche que nous avons proposée pour l'identification de communautés sur un réseau de co-expression a donné lieu au dépôt d'un brevet.

(19) RÉPUBLIQUE FRANÇAISE

INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

COURBEVOIE

(11) **N° de publication :** ***3 021 776***

(à n'utiliser que pour les commandes de reproduction)

(21) **N° d'enregistrement national :** **14 54889**

(51) Int Cl[8] : ***G 06 F 19/10*** *(2013.01)*

(12) DEMANDE DE BREVET D'INVENTION A1

(22) **Date de dépôt :** 28.05.14.

(30) **Priorité :**

(43) **Date de mise à la disposition du public de la demande :** 04.12.15 Bulletin 15/49.

(56) **Liste des documents cités dans le rapport de recherche préliminaire :** *Se reporter à la fin du présent fascicule*

(60) **Références à d'autres documents nationaux apparentés :**

( ) **Demande(s) d'extension :**

(71) **Demandeur(s) :** *VAIOMER Société par actions simplifiée* — FR, *UNIVERSITE PAUL SABATIER TOULOUSE III Etablissement public* — FR et *CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE (C.N.R.S) Etablissement public* — FR.

(72) **Inventeur(s) :** BRUNET ANNE-CLAIRE, LOUBES JEAN-MICHEL, AZAIS JEAN-MARC et COURTNEY MICHAEL.

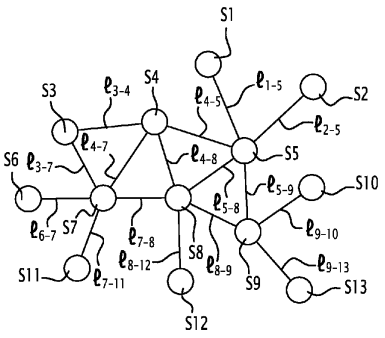
(73) **Titulaire(s) :** VAIOMER Société par actions simplifiée,UNIVERSITE PAUL SABATIER TOULOUSE III Etablissement public, CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE (C.N.R.S) Etablissement public.

(74) **Mandataire(s) :** CABINET LAVOIX Société par actions simplifiée.

(54) PROCEDE D'IDENTIFICATION D'UNE RELATION ENTRE DES ELEMENTS PHYSIQUES.

(57) La présente invention concerne un procédé d'identification d'une relation entre des éléments physiques, lesdits éléments présentant éventuellement une activité mesurable, le procédé comprenant les étapes suivantes :

- définir des graphes candidats, chaque graphe candidat étant un graphe associé à une des valeurs de seuillage de la pluralité de valeurs de seuillage,
- pour chaque valeur de seuillage, obtenir une répartition associée par optimisation de la répartition en classes des sommets du graphe associé à la valeur de seuillage considérée, l'optimisation partant d'une répartition initiale dans laquelle à chaque coeur est associé une classe pour obtenir une répartition finale dans laquelle chaque sommet d'une classe partage plus de liens avec les autres sommets de la même classe qu'avec les sommets d'une autre classe,
- sélectionner un graphe optimal parmi la pluralité de graphes candidats selon au moins un critère.



FR 3 021 776 - A1

2.7.6 Conclusion

Nous avons développé une approche originale pour l'analyse de données transcriptomiques présentant de nombreux avantages par rapport à celles existantes. À partir d'une estimation de la co-expression entre les gènes, couramment effectuée par le calcul d'un coefficient de corrélation (Pearson ou Spearman), nous proposons d'analyser les structures d'une collection exhaustive de graphes de λ-voisinage construits par seuillage des coefficients de la matrice de co-expression. Ce procédé nous permet d'identifier des ensembles de gènes, appelés des noyaux, qui restent préférentiellement connectés sur la collection de graphe. Ainsi, nous contournons la problématique de l'estimation d'un unique graphe pour modéliser le réseau de co-expression de gènes tout en intégrant de l'information sur la structure du réseau pour l'identification des communautés. Contrairement aux autres approches, qui nécessitent de fixer un certain nombre de paramètres pour l'estimation d'un unique graphe et pour l'identification des communautés sur le graphe, avec notre approche, un seul paramètre doit être choisi, celui de la taille minimale des noyaux.

L'identification de communautés de gènes co-exprimés favorise les interprétations biologiques car les communautés ont en général un rôle fonctionnel dans la cellule pouvant être déterminé par des analyses d'enrichissement en annotations GO ou KEGG. Il est également possible de résumer l'information contenue dans les données transcriptomiques en sélectionnant les hubs dans les communautés, c'est à dire les gènes qui ont une position centrale sur le réseau de co-expression.

Nous proposons également d'étudier les relations de causalité pouvant exister entre les différents hubs, de façon à explorer les mécanismes cellulaires impliquant les différentes fonctions biologiques portées par les communautés. L'analyse des relations de causalité peut aussi porter sur les gènes qui composent une communauté de façon à caractériser les co-régulations entre les gènes à l'intérieur de la communauté.

En application sur des données réelles, nous allons montrer dans le chapitre suivant que notre approche permet d'obtenir des résultats convaincants sur le plan biologique, et plus complets et robustes que ceux obtenus avec les méthodes d'apprentissage classiques présentées dans le premier chapitre.

Chapitre 3

Application : obésité, diabète et fibrose hépatique

3.1 Introduction

Le nombre de personnes souffrant d'obésité dans le monde est en augmentation perpétuelle à tel point que l'organisation mondiale de la santé (OMS) considère l'obésité comme l'épidémie du siècle. En 2014, plus de 600 millions d'adultes sont obèses, soit environ 13% de la population adulte mondiale. La cause principale de cette épidémie est liée à un mode de vie sédentaire, et à la consommation d'aliments qui sont peu nutritifs (teneur en nutriments faible devant le nombre de calories) trop riches en gras, en sel et en sucre.

Le diabète de type 2 est l'une des conséquences très fréquente de l'obésité. L'insuline, une hormone sécrétée par le pancréas, permet l'entrée du sucre dans les cellules pour qu'il soit utilisé comme carburant, particulièrement dans les muscles et le foie. Chez les diabétiques de type 2 (insulino-résistants), les cellules et les tissus deviennent moins sensibles à l'insuline et le taux de glucose dans le sang (glycémie) n'est plus contrôlé comme à la normale, ce qui provoque une hyperglycémie chronique, soit un taux trop élevé de glucose dans le sang. L'OMS prévoit qu'en 2030, le diabète de type 2 sera la septième cause de décès dans le monde. Le risque de décès chez les diabétiques est au moins deux fois plus élevé que chez les non diabétiques, et en 2014, plus de 300 millions de personnes sont diabétiques dans le monde.

Les maladies cardio-vasculaires sont les premières causes de décès chez les diabétiques et les autres complications liées au diabète peuvent être très handicapantes : amputation, cécité, insuffisance rénale...

Les atteintes hépatiques ou NAFLD (pour non-alcoholic fatty liver disease) constituent un autre facteur majeur de morbidité chez les diabétiques. La plus répandue étant la stéato-hépatique non alcoolique, caractérisée par une accumulation de graisse (stéatose) au niveau du foie accompagnée de lésions cellulaires inflammatoires (hépatites). L'évolution de la maladie peut conduire dans un faible nombre de cas à un processus de cicatrisation exagéré qui à la base est déclenché pour remplacer les cellules hépatiques endommagées. Cette formation de cicatrices fibreuses étendues sur le foie caractérise la fibrose hépatique et au stade le plus avancé de la maladie (stade de la cirrhose), quand les cicatrices sont trop nombreuses, le foie ne remplit plus ses fonctions et les risques de cancers sont accrus.

On évalue mal le nombre de personnes souffrant de maladies chroniques du foie, notamment

parce que ces maladies restent silencieuses (asymptomatiques) tout au long de leur évolution et avant d'atteindre un stade très avancé. Les causes de la maladies ne sont pas uniques mais l'obésité et la résistance à l'insuline (diabète de type 2) sont des facteurs de risques bien connus. Ainsi, la forte augmentation du nombre de personnes souffrant d'obésité et de diabète de type 2 laisse présager une hausse sensible du nombre de personnes atteintes de maladies chroniques du foie.

La fibrose peut dans certains rares cas être réversible si les lésions ne sont pas trop sévères et que la cause est traitée. Il est important de pouvoir diagnostiquer le degré de fibrose et de pouvoir suivre l'évolution de la maladie pour en assurer le traitement approprié. Les solutions proposées pour le diagnostic de la maladie reste à ce jour insuffisantes et de nombreuses équipes de recherche travaillent à l'identification de biomarqueurs non invasifs (sans prélèvement d'une partie du foie) ayant de bonnes performances prédictives.

3.2 Le projet « Florinash »

3.2.1 Introduction

Le projet « Florinash » (`http://www.florinash.org`) est un projet européen qui a pour objectif d'étudier les mécanismes biologiques responsables des maladies chroniques du foie en lien avec les problèmes d'obésité et de diabète. Ces recherches visent notamment à améliorer le diagnostic de la maladie par la recherche de biomarqueurs non invasifs, ainsi qu'à en améliorer son traitement, par l'identification de cibles thérapeutiques.

Pour obtenir un diagnostic fiable du stade de fibrose, il est encore à ce jour nécessaire d'examiner un fragment de foie recueilli par une biopsie. Cet examen invasif comporte certains risques et ne peut être renouvelé facilement. Des marqueurs non invasifs utilisant le sérum notamment ou des techniques d'imagerie ont été proposés, mais leurs performances restent insuffisantes. Ainsi, l'un des objectifs du projet « Florinash » est d'améliorer ces marqueurs, ou d'en proposer de nouveaux, afin d'éviter la biopsie pour déterminer le stade et l'évolution de la maladie. Concernant les cibles thérapeutiques, les pistes privilégiées par l'équipe « Florinash » sont celles qui pourraient agir sur l'adaptation tissulaire aux stress environnemental (plus précisément le stress du réticulum endoplasmique), sur l'inflammation causée par les facteurs de nécrose tumoral (le TNF pour tumor necrosis factor est une cytokine, une molécule synthétisée et sécrétée par les globules blanc), et sur la lipogenèse, c'est à dire la synthèse des lipides (des acides gras en notamment). Un autre axe de recherche se focalise sur l'implication des bactéries dans le développement de la maladie. La translocation bactérienne, c'est à dire la migration des bactéries intestinales vers d'autres tissus (foie, sang...) a déjà été mise en cause dans les problèmes de diabète et d'obésité, ainsi que dans la progression des NAFLD.

Que ce soit pour la découverte de biomarqueurs ou de cibles thérapeutiques, il est important de comprendre les mécanismes biologiques impliqués dans la maladie. En travaillant complètement à l'aveugle, c'est à dire en sélectionnant par des méthodes statistiques, quelques cibles ou biomarqueurs qui paraissent avoir les meilleurs capacité prédictives, la reproductibilité des résultats ne peut être garantie. Une collaboration étroite entre biologiste et statisticien est en ce sens indispensable. Au niveau transcriptomique notamment, l'approche qui consiste à replacer les gènes sur un réseau pour identifier des communautés de gènes co-exprimés favorise l'interprétation des résultats (par exemple en faisant des analyses d'enrichissement) et garantie une meilleure reproductibilité des résultats (le hasard est moins probable lorsque les variations d'expression

sont observées sur tout une communauté de gènes).

3.2.2 Ce que l'on connaît des mécanismes de la fibrose hépatique

La fibrose hépatique se caractérise par l'apparition de cicatrices fibreuses sur le tissu hépatique, en réponse à une agression tissulaire qui laisse des lésions. L'inflammation chronique du foie provoquée, soit par des facteurs intra-hépatique (inflammation, mort cellulaire), soit par des facteurs extra-hépatiques (translocation bactérienne), joue un rôle clé dans l'évolution des NAFLD vers la fibrose notamment. Même si l'obésité et le diabète de type 2 sont des causes reconnues de la fibrose, elles ne sont pas les seules et les mécanismes moléculaires responsables de son apparition sont toujours imparfaitement connus. La constitution d'une fibrose se caractérise au niveau moléculaire par la rupture de l'équilibre de la matrice extracellulaire (MEC) qui conduit à une accumulation de constituants de la MEC (les collagènes sont les principaux composants de la MEC) : les processus de synthèse et de dépôt des constituants de la MEC l'emportent sur la dégradation (fibrose extensive). En d'autres termes, le processus de cicatrisation qui se met en place pour réparer la lésion « s'emballe », et la cicatrice qui en résulte s'étend bien au-delà de la lésion. La stéatose (« foie gras ») pourrait être favorisée par l'arrivée massive d'acides gras en provenance du tissu adipeux, ainsi que par une agmentation de la lipogénèse sur le foie, une suite de réaction effectuée à partir du glucose et des acides gras qui conduit à la synthèse des lipides. Les cellules du foie ou hépatocytes chargées de gras deviendrait alors plus vulnérables à différents types d'agression : l'inflammation causée par l'augmentation des cytokines pro-inflammatoires (telles que TNF-α), stress oxydant (stimule l'activation des cellules fibrogéniques), stress du réticulum endoplasmique (déclenche une apoptose des hépatocytes, une mort cellulaire programmée)...

3.2.3 Les données

La base de données « florinash » a été constituée à partir de deux cohortes de plusieurs centaines de patients obèses sur le point de subir une chirurgie bariatrique. L'une des deux cohortes regroupe des patients espagnols et l'autre des patients italiens. Un grand nombre de données métaboliques, cardiovasculaires et cliniques ont été collectées et les données transcriptomiques ont été générées pour un certain nombre de patients sélectionnés selon des critères d'âge (entre 40 et 60 ans), de corpulence (BMI > 30), de comportement (consommation d'alcool raisonnable), ne présentant pas d'autres maladies que celle étudiée (pas de maladie cardiaque, pas de cirrhose, de maladie autoimmune...), et ne prenant pas de médicament pouvant altérer l'action de l'insuline. Nous avons retenu pour l'analyse les 97 patients pour lesquels les données transcriptomiques et le résultat de la biopsie d'un fragment de foie sont disponibles.

Toutes les analyses ont été réalisées avec le logiciel R (version 3.2.2).

Les variables d'obésité, de résistance à l'insuline, de la translocation bactérienne et de la fibrose Notre objectif étant d'analyser les effets de l'obésité et de la résistance à l'insuline sur la fibrose, nous avons choisi une variable pour chaque pathologie dans l'ensemble des variables disponibles, de façon à ce qu'elle contienne le moins possible de valeurs aberrantes et manquantes.

Pour caractériser la résistance à l'insuline, nous avons retenu la variable du « clamp » (euglycémique hyperinsulinémique). Elle constitue une mesure très fiable de la résistance à l'insuline. Elle s'obtient en réalisant une perfusion simultanée de glucose et d'insuline, de façon à maintenir constantes les concentrations en insuline et en glucose, à un taux élevé pour l'insuline et à un taux normal pour le glucose. Cela permet de déterminer la quantité de glucose nécessaire pour compenser le niveau d'insuline anormalement élevé. Si cette quantité est faible, alors l'action de l'insuline est limitée, ce qui reflète un problème de résistance à l'insuline. Le clamp mesure le miligramme de glucose nécessaire par minute (mg/min). Au dessus de 7.5 mg/min, le patient ne souffre pas de résistance à l'insuline, en dessous de 4 mg/min, la résistance à l'insuline est avérée, et entre les deux, le diagnostic ne peut être définitif (premiers signes de résistance à l'insuline).

Le degré d'obésité est estimé à partir de l'indice de masse corporelle ou BMI (pour Body Mass Index). Cet indice se calcule en fonction de la taille et de la masse du patient : $BMI = \frac{masse}{taille^2}$. Une personne est obèse si son BMI est au moins égal à 30 : obésité modérée entre 30 et 35, obésité sévère entre 35 et 40, et obésité morbide au delà.

Une piste explorée par l'équipe « florinash » pour éventuellement proposer un biomarqueur de la fibrose est celle de la translocation bactérienne dans le sang. Le microbiote est l'ensemble des micro-organismes (bactéries en particulier) qui colonisent notre corps. Dans l'intestin, on estime qu'il y a environ 100 000 miliards de bactéries, soit au moins 150 fois plus de bactéries que de cellules humaines. Le microbiote intestinal, aussi appelé flore intestinale, agit en étroite collaboration avec son hôte humain. Il est considéré comme un organe à part entière et joue un rôle clé dans les réactions du métabolisme, du système immunitaire, et autres... La translocation bactérienne est un phénomène par lequel les bactéries intestinales migrent vers d'autres tissus (foie, sang, tissu adipeux...). Les facteurs qui peuvent la favoriser incluent une multiplication d'un certain type de bactéries (gram-négatif), une diminution de la réponse immunitaire et une perméabilité de la paroie intestinale. Elle a notamment été mise en cause dans les problèmes d'obésité et de diabète.

La quantification de la charge bactérienne sur un tissu donné peut être réalisée par la méthode de la PCR (pour Polymerase Chain Reaction) quantitative. La charge bactérienne correspond a la quantité d'ARN ribosomique 16S (ARNr 16S) qui est l'un des constituants spécifique des bactéries (ou procaryotes). Par la technique de la PCR quantitative, les séquences d'ARNr 16S sur un échantillon de tissu sont amplifiées, c'est à dire dupliquées en grand nombre, par la répétition de cycles de réactions. Le processus de duplication est suivi en continu à l'aide de molécules fluorescentes qui viennent se fixer sur les ARN recherchés. La fluorescence de l'échantillon augmente proportionnellement au nombre de molécules d'ARN, et la quantité d'ARN est déterminée en fonction du profil d'augmentation de cette fluorescence au cours du temps : plus il y a de molécule d'ARN à l'origine et plus le nombre de cycles d'amplification nécessaire pour atteindre un nombre déterminé de molécules est réduit.

Afin de pouvoir éventuellement mettre en cause l'augmentation de la translocation bactérienne dans les problèmes de fibrose, et de pouvoir détecter cette translocation bactérienne chez les patients par un examen non invasif, la quantification de la charge bactérienne a été effectuée sur le tissu sanguin des patients (simple prise de sang). Le prélèvement n'a cependant pas été fait de la même façon chez les patients dans les deux cohortes (espagnole et italienne). Pour les patients espagnols, la charge bactérienne a été quantifiée à partir d'un prélèvement de « buffy

coat » c'est à dire d'une fraction d'un échantillon de sang contenant la plupart des globules blancs ainsi que les plaquettes. Pour les Italiens, c'est le sang total qui a été analysé, composé non seulement des globules blancs et des plaquettes, mais également du plasma et des globules rouges. Ainsi, les données de la charge bactérienne ne sont pas directement comparables pour les patients des deux cohortes et il est difficile de proposer une normalisation qui permettrait de corriger cette différence. On peut difficilement connaitre a priori l'impact de la différence de prélèvement sur la quantification de la charge bactérienne. Au risque de perdre en puissance statistique dans les analyses, mais pour garantir une plus grande fiabilité des résultats, nous avons fait le choix d'étudier les effets de la charge bactérienne indépendamment dans chaque cohorte. L'unité de mesure de la charge bactérienne, variable que nous appelerons « qPCR16S », est le nombre de copies d'ARNr 16S par volume d'extrait.

Sur les 97 patients composant l'échantillon ayant servi pour les analyses, 22 d'entre eux présentent une fibrose du foie. Les variables de la résistance à l'insuline (clamp) et de la charge bactérienne (qPCR16S) contiennent des valeurs manquantes. Cela pose un problème (perte de puissance statistique) pour l'étude de l'effet de la qPCR16S sur le risque de fibrose, effectué indépendamment chez les Italiens et les Espagnols : la donnée de qPCR16S est disponible pour seulement cinq des patients italiens qui ont une fibrose. La répartition des cas de fibrose en fonction de la nationalité (cohorte), et celle du nombre d'observations disponibles pour les variables clamp et qPCR16S sont décrites dans les Table 3.1 et 3.2.

	Foie sain	**Fibrose**
Italiens	39	10
Espagnols	36	12
Total	75	22

TABLE 3.1 – Répartition des cas de fibrose en fonction de la nationalité des patients

	clamp	**qPCR16S**
Italiens avec fibrose	10	5
Italiens sans fibrose	38	32
Total Italiens	**48**	**37**
Espagnols avec fibrose	9	11
Espagnols sans fibrose	28	25
Total Espagnols	**37**	**36**
Total	**85**	**73**

TABLE 3.2 – Nombre d'observations disponibles pour les variables clamp et qPCR16S

Une première analyse exploratoire met en évidence un effet de la résistance à l'insuline et de la charge bactérienne sur la fibrose du foie (Figure 3.1).

Les mesure du clamp sont significativement (test de Student) plus faibles chez les patients avec fibrose : la résistance à l'insuline est globalement plus importante chez les patients avec une fibrose. À noter qu'une grande majorité de patients ont un problème de résistance à l'insuline : seulement 7 patients sont clairement non résistants à l'insuline (clamp$>$ 7.5) et 60 patients ont un problème de résistance à l'insuline avéré (clamp$<$ 4).

Chez les Espagnols, la qPCR16S est significativement plus élevée chez les patients avec une fibrose. Chez les Italiens, on observe la même tendance mais on ne peut conclure à la significativité de cette tendance à cause du trop faible nombre de cas ou de la différence technique de prélèvement (effet masqué sur le sang total ?).

Le BMI ne permet pas de discriminer les patients qui ont une fibrose de ceux qui n'en ont

pas. Tous les patients sont obèses (BMI minimum de 33.5) et à ce niveau de corpulence le risque de fibrose est comparable. L'effet du poids sur la fibrose ne pourra être mis en évidence à partir de cet échantillon qui ne contient pas de patient ayant une corpulence modérée ou faible. Nous ne nous intéresserons pas à cette variable par la suite.

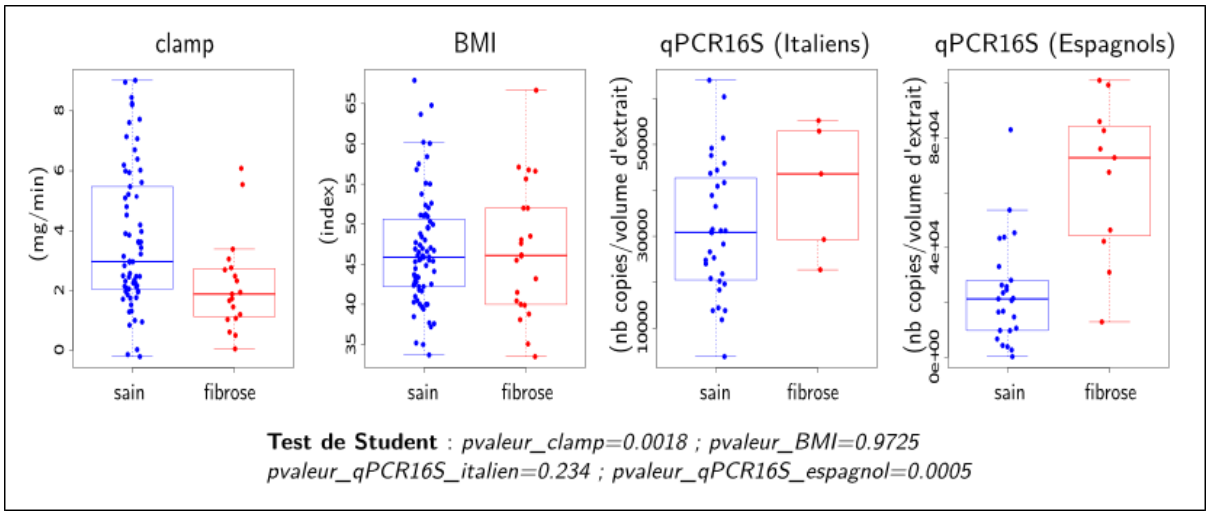


FIGURE 3.1 – Représentation « boxplot » de la distibution des valeurs de la résistance à l'insuline (clamp), de la concentration en ADN bactérien dans le sang (qPCR16S) et de l'index de masse corporelle (BMI) chez les patients avec une fibrose du foie et chez les patients sans fibrose. La médiane est représentée par un trait épais à l'intérieur de la boîte qui s'étend du premier quartile au troisième quartile. Par défaut, avec la fonction « boxplot » de R, les moustaches de la boite n'excèdent pas 1.5 fois la distance interquartile.

La transcriptomique Le niveau d'expression des gènes (quantité d'ARN produite) sur le fragment de foie (biopsie) des 97 patients de l'échantillon a été analysé par la technologie des puces à ADN. Après traitement bioinformatique (nettoyage, correction de biais, normalisation en logarithme base 2) des données brutes, la base de donnée contient les valeurs d'expression de 31476 gènes et ne contient pas de valeurs manquantes. Quand l'expression d'un gène varie en rapport avec un caractère d'intérêt, plus les différences observées entre les différents niveaux d'expression du gène sont importantes, et plus le gène est susceptible d'être informatif pour le caractère d'intérêt. Ainsi, nous avons sélectionné les 5405 gènes pour lesquels la variabilité des niveaux d'expression observée chez les 97 patients est la plus importante : $log(\sigma^2) > -2$ où σ^2 est la variance empirique de la distribution du niveau d'expression du gène calculée sur l'échantillon des 97 patients. Comme illustré sur la Figure 3.2, ce seuil a été choisi de façon à sélectionner les gènes avec des σ^2 placés sur la queue de la distribution (découpage en tranche par la formule de Sturge).

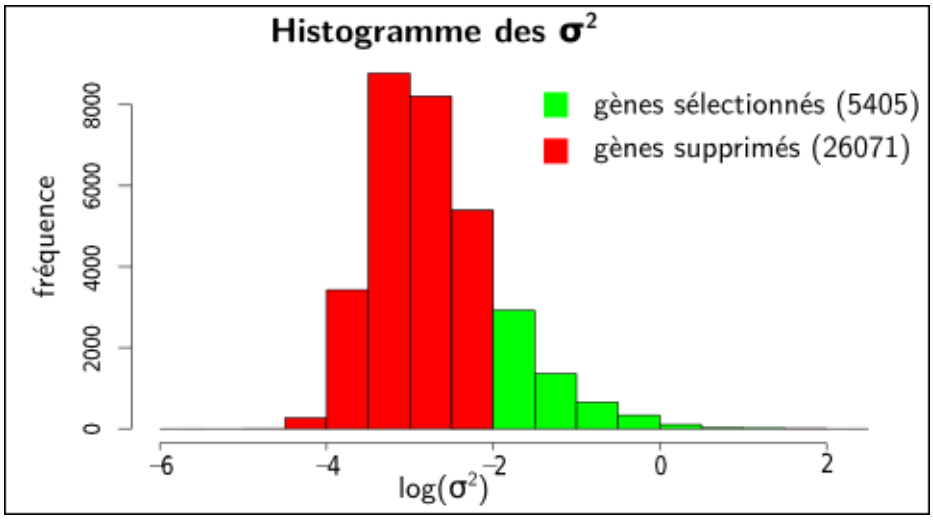


FIGURE 3.2 – Représentation de la distribution des σ^2 (les variances empiriques de la distribution du niveau d'expression des gènes)

Tableau récapitulatif des variables

Les variables biologiques et cliniques				
variable	**nature**	**taille échantillon**	**description**	**valeurs manquantes**
Fibrose	binaire	97	22 patients avec une fibrose	0
Clamp	quantitative	97	$min = -0.21$, $max = 9.02$ $moy = 3.37$	12
qPCR16S Italiens	quantitative	49	$min = 3770$, $max = 64000$ $moy = 32734$	12
(sang total)				
qPCR16S Espagnols (buffy coat)	quantitative	48	$min = 237$, $max = 101000$ $moy = 36330$	12
La transcriptomique				
variable	**nature**	**taille échantillon**	**description**	**valeurs manquantes**
Expression du gène (foie)	quantitative	97	5405 gènes	0

TABLE 3.3 – Tableau récapitulatif des variables pour les analyses « Florinash »

3.3 Analyse du réseau de co-expression de gènes

Le principal objectif de l'analyse d'un réseau de co-expression de gènes est l'identification de communautés de gènes co-exprimés. Nous avons utilisé notre algorithme, basé sur la recherche

de noyaux, pour identifier des communautés de gènes co-exprimés sur le foie des patients « florinash ». La co-expression s_{ij} entre deux gènes i et j, avec $i, j \in \{1, 2, ..., p = 5405\}$, a été mesurée par $s_{ij} = (1 + r_{ij}^{spear}) \times 0.5$, où r_{ij}^{spear} est le coefficient de corrélation de Spearman entre l'expression des gènes i et j sur l'échantillon de patients ($N = 97$).

Identification des communautés

L'unique paramètre à fixer avec notre méthode est la taille minimale n d'un noyau. Nous avons choisi le paramètre $n = 26$ comme décrit dans le chapitre précédent (Figure 2.5 et 2.6), et ajouté l'ensemble des noyaux identifiés pour $n = 10$ qui disparaissent pour $n = 26$ (qui ne sont pas inclus dans un noyau).

Nous avons défini les communautés en les réduisant à leur noyau : les gènes en dehors des noyaux ont été exclus de l'analyse et les 1351 gènes restants sont répartis dans 42 noyaux ou communautés distinctes.

Les communautés identifiées contiennent en moyenne 32 gènes avec un minimum de 10 gènes et un maximum de 130 gènes.

A l'intérieur de chaque communauté, nous avons sélectionné le « hub » qui est le gène qui maximise la somme des mesures d'association avec les autres gènes de la communauté.

La matrice des corrélations de Spearman entre ces hubs ainsi que la taille des différentes communautés sont représentées sur la Figure 3.3. On observe quelques corrélations importantes entre les hubs, mais globalement ces corrélations sont assez faibles.

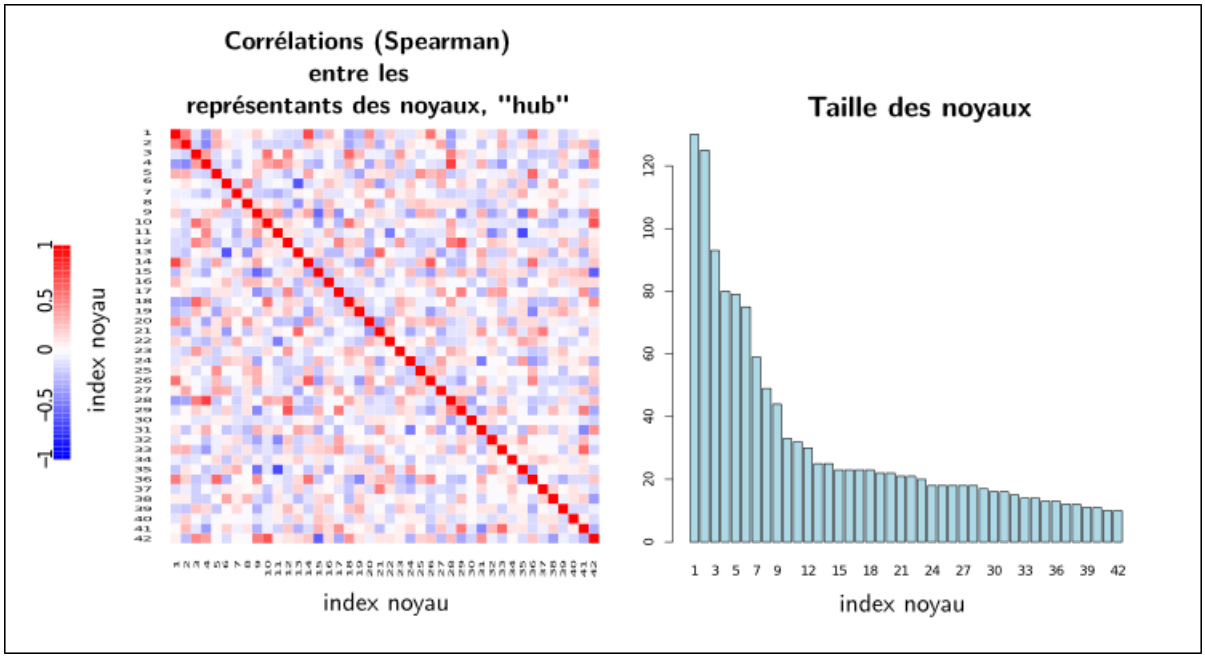


FIGURE 3.3 – Représentation de la matrice des corrélations de Spearman entre les hubs des communautés et de la taille des communautés.

Analyses d'enrichissement dans les communautés Les annotations GO des processus biologiques associés aux 1351 gènes de nos communautés ont été recherchées dans la base de données « GO.db », et celles des annotations KEGG pour les voies métaboliques ont été recherchées dans le base de donnée « KEGG.db ». Ces deux bases de données sont disponibles à l'adresse `www.bioconductor.org` et peuvent être directement chargée dans R. Plus de la moitié des gènes (Table 3.4) ont pu être associés à au moins un processus biologique (GO) et un peu moins d'un tiers des gènes ont été identifiés comme étant impliqués dans une voie métabolique (KEGG).

	nombre de gènes annotés
GO	711
KEGG	369

TABLE 3.4 – Nombre d'annotations GO et KEGG associées aux gènes

Aucune annotation GO n'a été trouvée dans les communauté numérotées 30 et 42, et aucune annotation KEGG dans les communautés 29, 30, 32, 38, 39, 42.

Nous avons recherché à l'intérieur de chacune des communautés les annotations sur-représentées à l'aide du test exact de Fisher unilatéral, puis ajusté les pvaleur des tests par la méthode des FDR (False discovery rate) de façon à contrôler le taux d'erreur de première espèce. Nous utiliserons ces résultats par la suite pour donner un sens fonctionnel au communautés de gènes.

Diagnostic pour la multi-colinéarité L'apprentissage supervisé pour expliquer un caractère d'intérêt (la fibrose par exemple) à partir des données transcriptomiques est d'autant plus délicat quand il y a de la redondance, des données non informatives (bruitées) et un très grand nombre de gènes. La sélection des hubs permet de réduire considérablement la taille des données (42 hubs pour 5404 gènes au départ), de supprimer la redondance, ainsi que les gènes non informatifs aux profils atypiques et faiblement associés avec les profils des gènes dans les communautés. Nous allons comparer par la suite notre méthode de sélection par les hubs avec celle de la régression parcimonieuse pour mettre en évidence les points forts de notre approche. En grande dimension ($p >> N$) les méthodes de régression et de sélection classiques sont inenvisageables. Il est nécessaire de sélectionner un nombre raisonnable de variables explicatives avant d'estimer leurs effets sur la réponse. Le conditionnement de la matrice des corrélations entre les variables sélectionnées va déterminer la précision des estimations. Un mauvais conditionnement conduit à des estimateurs de fortes variances.

Pour l'exemple, nous avons sélectionné par régression lasso les 41 gènes les plus pertinents pour expliquer la variable « clamp » et nous avons calculé l'indice de conditionnement $I_c = \frac{\lambda_{max}}{\lambda_{min}}$, où λ_{max} et λ_{min} sont respectivement la plus grande et la plus petite valeur propre de la matrice des corrélations entre les gènes sélectionnés. Un mauvais conditionnement de la matrice est associé à une valeur I_c très élevée (très problématique pour $I_c > 1000$). On obtient la valeur $I_c^{lasso} = 135$. L'indice de conditionnement pour la matrice des corrélations entre les hubs vaut quant à lui $I_c^{hub} = 307$ sachant qu'on a 42 hubs (une variable de plus que pour la sélection Lasso). Il est supérieur à celui obtenu avec le lasso mais il reste raisonnable avec un nombre de prédicteurs supérieur (une variable en plus dans le modèle).

Pour un modèle de régression linéaire $y = X\beta + \epsilon$, avec X la matrice des prédicteurs de dimension $N \times p$ suposée centrée et réduite, la matrice de variance-covariance Σ_β des coefficients du modèle s'écrit :

$$\Sigma_\beta = \sigma_\epsilon^2 (X'X)^{-1} = \frac{\sigma_\epsilon^2}{N} R^{-1},$$

où σ_ϵ^2 est la variance des résidus ϵ du modèle, et $R = \frac{1}{N} X'X$ la matrice des corrélations empiriques (données centrées-réduites). Ainsi, les variances $\sigma_{\beta_j}^2$ des coefficients du modèle (diagonale de Σ_β) sont proportionnelles au éléments diagonaux $(R^{-1})_{jj}$ qui ne sont autre que le carré d'un coefficient de corrélation multiple :

$$(R^{-1})_{jj} = \frac{1}{1 - R_j^2},$$

où R_j^2 est le carré du coefficient de corrélation multiple de X^j avec les $p-1$ autres prédicteurs, également appelé coefficient de détermination de la régression linéaire de la variable X^j sur les autres variables. Cette valeur $(R^{-1})_{jj}$ est appelé le facteur d'inflation de la variance (VIF) de la variable X^j. Sa valeur augmente avec la variance de l'estimateur $\hat{\beta}_j$ et quand elle est très élevée, cela traduit un phénomène de multicolinéarité. Nous avons calculé ce VIF pour les données des hubs et les avons représentés sur la Figure 3.4.

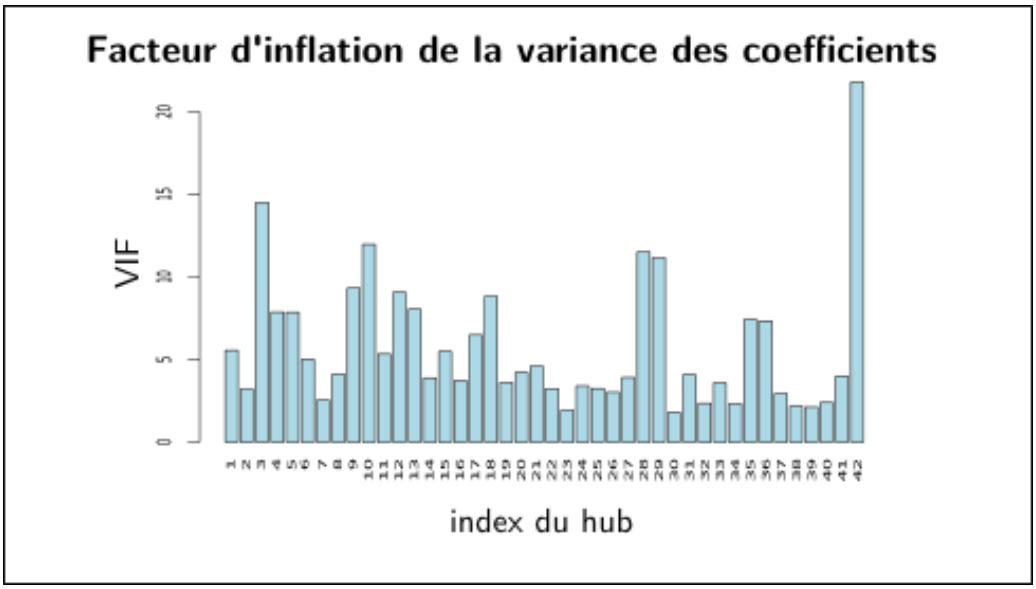


FIGURE 3.4 – Représentation « barplot » des facteurs d'inflation de la variance des coefficients associés aux hubs dans un modèle de régression.)

Pour le hub de la communauté numéro 42, le VIF est particulièrement élevé (égal à 22). En supprimant cette variable, on obtient un indice de conditionnement, pour la matrice des corrélations entre les hubs restants, $I_c^{hub} = 197$. A nombre de variable égal, on est très proche de la valeur obtenue pour les variables sélectionnées par le Lasso. Le conditionnement de la matrice des corrélations entre les hubs ne sera pas problématique pour la modélisation d'un caractère d'intérêt à partir des hubs.

3.4 Recherche de marqueurs génétiques et exploration des mécanismes de la fibrose

L'objectif de notre analyse est de trouver des marqueurs génétiques fonctionnels de la fibrose et de tenter de comprendre les mécansimes biologiques qui la favorise. Pour cela nous allons intégrer l'information obtenue sur les communautés de gènes afin de chercher à expliquer la fibrose hépatique, non pas par des gènes isolés, mais par des communautés de gènes.

3.4.1 Détail des analyses statistiques réalisées

Tests de permutations Pour chacun des 1351 gènes, nous avons évalué par des tests de permutation, la significativité des corrélations de Spearman entre l'expression du gène et la résistance à l'insuline (clamp), en supprimant de l'échantillon (97 patients) les 12 patients pour lesquels la donnée du clamp est manquante. De la même façon nous avons testé la corrélation entre l'expression des gènes et la charge bactérienne (qPCR16S), indépendamment chez les Espagnols (36 patients pour lesquels les données sont complètes) et chez les Italiens (37 patients). Un test de permutation a également été effectué pour tester l'égalité de la moyenne des expressions de gènes chez les 22 patients atteints de fibrose du foie, avec celle observée chez les 75 patients sans fibrose du foie. Nous avons ensuite sélectionné les communautés de gènes les plus significativement associées à nos trois variables d'intérêt (fibrose, clamp et qPCR16S) : une communauté a été sélectionnée pour une variable donnée si la médiane de la distribution des pvaleurs des tests à l'intérieur de cette communauté est supérieure à 0.05 ou si la pvaleur associée au représentant de la communauté (hub) est supérieure à 0.05. Ces résultats sont représentés sur la Figure 3.5.

En majorité, les communautés sélectionnées par ce procédé le sont aussi bien en fixant le seuil de significativité (0.05) sur la médiane de la distribution que sur la pvaleur associé au hub. Cela corrobore l'hypothèse que le hub est un représentant de sa communauté. Ce procédé nous permet néanmoins de « rattraper » certaines communautés pour lesquelles les associations avec les variables d'intérêt sont un peu plus mitigées. Il se peut que les données d'expression à l'intérieur de certaines communautés soient moins homogènes, mais elles n'en restent pas moins intéressantes, si un ensemble non négligeable de leurs gènes est explicatif du caratère d'intérêt. Par exemple, pour la sélection des communautés à partir de la qPCR16S chez les Espagnols, la communauté numéro 1 n'est pas sélectionnée avec la médiane et elle l'est avec le hub, et à l'inverse, la communauté numéro 36 est sélectionnée avec la médiane mais pas avec le hub.

A seuil de significativité égal et à taille d'échantillon comparable, la puissance des tests effectués pour la variable qPCR16S sur les deux sous-échantillons est comparable, et pourtant aucune communauté de gènes n'est sélectionnée sur le sous-échantillon des Italiens. Or, nous avons précédemment observé (Figure 3.1) sur le sous-échantillon des Espagnols, une augmentation significative de la charge bactérienne chez les patients atteints de fibrose par rapport à ceux sans fibrose. Cette tendance semblait se confirmer chez les Italiens sans pour autant être significative en raison d'un nombre de cas trop faible (5 patients atteints de fibrose) ou de la différence technique de prélèvement. Cela suggère que l'effet de la charge bactérienne sur l'expression des gènes du foie est détectable lorsque celle-ci augmente significativement, en lien peut-être avec la fibrose hépatique. Les communautés numérotées 1, 36 et 37 sélectionnées chez les Espagnols font aussi parti des communautés de gènes à l'intérieur desquelles les pvaleurs sont les plus faibles chez les Italiens (tendances identiques mais non significatives chez les Italiens pour les mêmes raisons que celles qui masquent le lien entre la charge bactérienne et la fibrose ?).

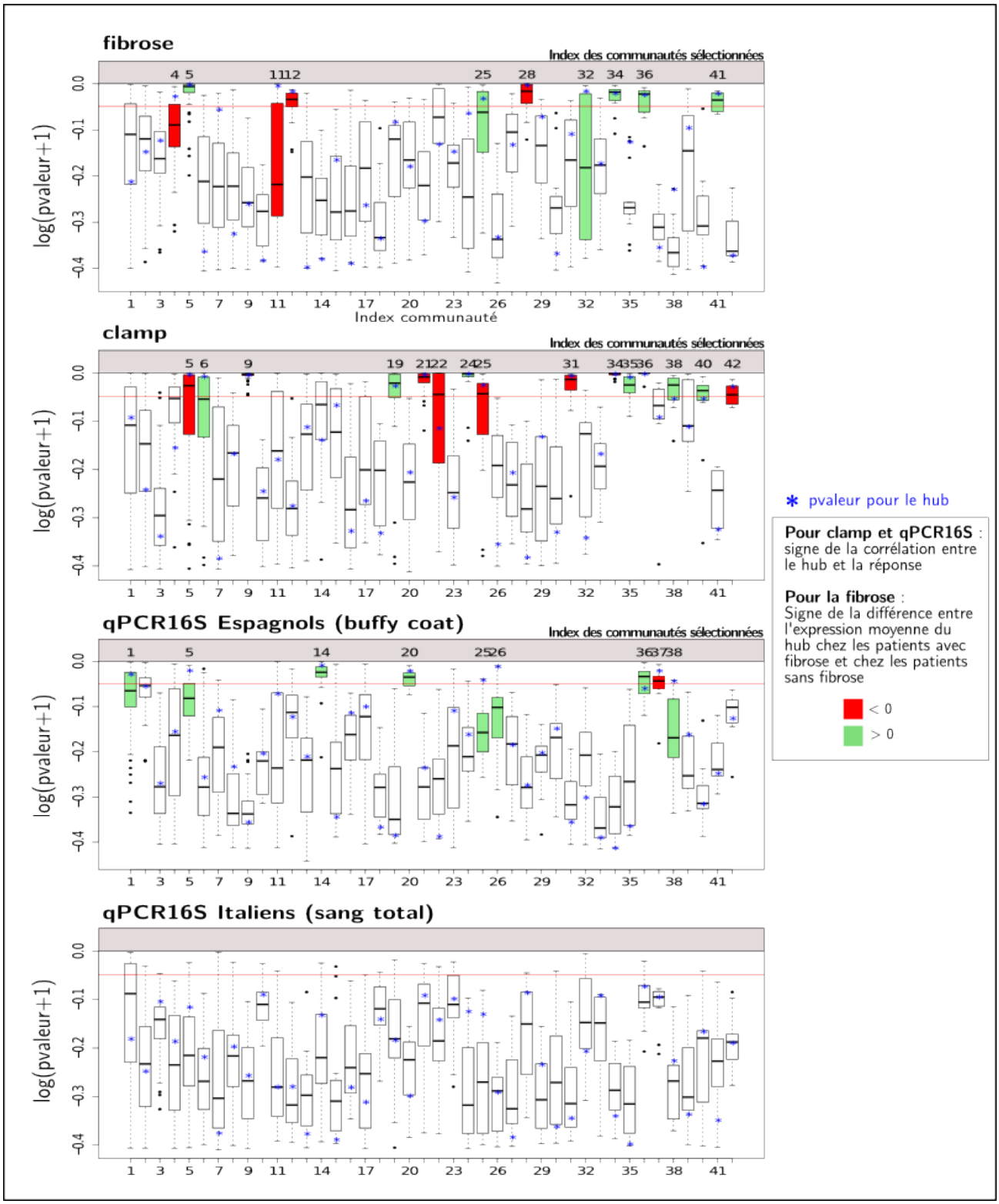


FIGURE 3.5 – Représentation « boxplot » des distributions des pvaleurs au sein des communautés pour les différentes variables réponses : fibrose, clamp, qPCR16S chez les Espagnols et chez les Italiens. Sélection de communautés si la médiane des pvaleur dans la communauté ou si la pvaleur associé au hub de la communauté est < 0.05 (ligne rouge).

variable	index des communautés sélectionnées	nombre de communautés
fibrose	4, 5***, 11, 12, 25*** 26, 32, 34**, 36***, 41	10
clamp	5***, 6, 9, 19, 21 22, 24, 25***, 31, 34** 35, 36***, 38**, 40, 42	15
qPCR16S Espagnols	3, 5***, 14, 20, 25*** 26, 36***, 37, 38**	9
communauté sélectionnée deux fois (∗∗), trois fois (∗ ∗ ∗)		

TABLE 3.5 – Tableau récapitulatif des communautés sélectionnées (26 au total)

Etude des relations causales Nous avons étudié les liens de causalité pouvant exister entre les différentes fonctions biologiques portées par les communautés de gènes. L'analyse statistique a été éffectuée à partir des 42 représentants de communauté, c'est à dire des hubs. Nous avons également ajouté la variable du clamp dans l'analyse des relations de causalité. Les 12 valeurs manquantes de la variable clamp ont été imputées par la moyenne des valeurs observées et l'analyse a été effectuée sur les 97 patients de l'échantillon. La variable qPCR16S n'a pas été intégrée à l'analyse (échantillon scindé entre Espagnols et Italiens et beaucoup de valeurs manquantes).

Nous avons estimé la structure du réseau bayésien avec l'algorithme FCI, disponible dans le paquet « pcalg » de R, en fixant un risque d'erreur $\alpha = 0.05$ pour les tests d'indépendance conditionnelle. Nous avons travaillé sur les rangs des variables plutôt que sur leurs valeurs pour assurer une plus grande robustesse des résultats. Le résultat est un graphe partiellement ancestral (PAG) qui tient compte de la présence éventuelle de variables latentes (variables non observées). Nous ne reviendrons pas sur le contexte méthodologique de l'analyse qui a été présenté dans le chapitre précédemment, mais rappelons quand même l'interprétation à donner aux différents types de liens sur le graphe : soit H^1 et H^2 deux sommets distincts,

Relation	H^1 **est une (*) de** H^2
$H^1 \longrightarrow H^2$	(*) « cause véritable »
$H^1 \longleftrightarrow H^2$	(*) « cause artificielle »
H^1 $o\!\!-\!\!o$ H^2	(*) « cause indéterminée »
H^1 $o\!\!\longrightarrow$ H^2	(*) « cause potentielle »

TABLE 3.6 – Interprétation des arcs sur un PAG

Le PAG est représenté sur la Figure 3.6. Nous avons estimé la fiabilité de chaque arc du graphe (lien et orientation) par la fréquence d'apparition de l'arc sur les PAGs construits à partir de 1000 échantillons bootstrap (le détail de cette procédure est donné dans le chapitre précédent).

La recherche de la causalité pour donner une orientation aux liens du graphe est une problématique très difficile et elle peut être dissociée de celle de l'estimation des dépendances conditionnelles (liens sans leur orientation), au moins en partie puisqu'avec l'algorithme FCI la présence de variables latentes peut dans certains cas justifier l'indépendance conditionnelle.

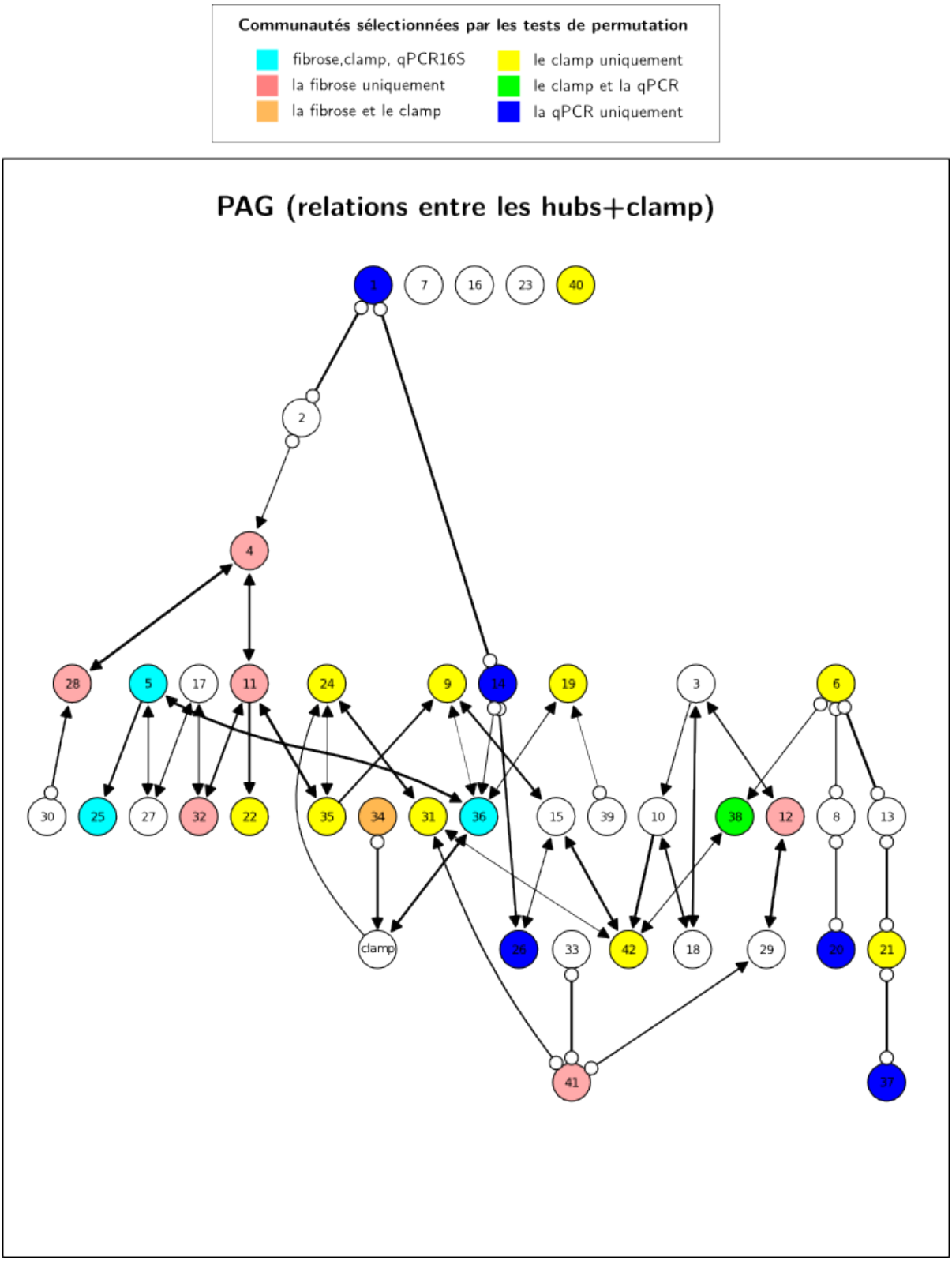


FIGURE 3.6 – Représentation du graphe partiellement ancestral (PAG) estimé par l'algorithme FCI (risque d'erreur $\alpha = 0.05$) sur les hubs des communautés et de la variable clamp. L'épaisseur d'un lien est proportionnelle à sa fréquence d'apparition (sans son orientation) sur les PAGs estimés à partir de 1000 échantillons bootstrap.

Si l'on estime (bootstrap) la fiabilité d'un lien sans son orientation sur le PAG, on obtient alors une fréquence d'apparition minimum de 0.27 (entre le hub 9 et 36), maximum de 0.74 (entre le hub 4 et le 28), et une fréquence d'apparition moyenne de 0.71. Ainsi, la structure des dépendances conditionnelles représentée sur le graphe est très stable même si celle de la causalité reste plus discutable (fiabilité moyenne de 0.22).

Les arcs (lien orientés) qui sont les plus fiables sur le graphe représentent des « causes artificielles » c'est à dire des relations entre couples de variables qui pourraient être expliquées par d'autres variables qui ne sont pas sur le graphe (variables latentes). Ce résultat est peu étonnant dans la mesure où nous avons effectué une sélection drastique parmi l'ensemble des gènes (à la nuance près que les variables latentes ne sont pas nécessairement des gènes).

L'organisation des dépendances entre les hubs sur le graphe, respecte la logique de sélection des communautés effectuée par les tests de permutation : pour chacune des variables fibrose, clamp, et qPCR16S, on identifie un groupe de hubs connectés sur le graphe qui sont les représentants de communautés sélectionnées pour la variable. Les trois hubs sélectionnés pour les trois variables (5, 36, 25) sont eux-mêmes connectés sur le graphe, et le hub numéro 36 est à la croisée des chemins, il fait le lien entre les différents groupes de hubs.

Quel gain en robustesse en travaillant sur les rangs plutôt que sur les valeurs des variables ? Nous avons comparé la fiabilité (estimée à partir de 1000 échantillons bootstrap) des liens du PAG estimant les relations entre les variables de rang (appelé PAG1) avec la fiabilité des liens du PAG estimant les relations entre les variables non transformées (appelé PAG2). Cette comparaison a été effectuée pour les liens non orientés (dépendance conditionnelle) et pour les liens orientés (causalité).

Le nombre de liens sur les deux PAGs est comparable (43 liens sur le PAG1 et 47 liens sur le PAG2). Les résultats sont présentés sur la Figure 3.7). Nous n'observons pas de différence notable dans les distributions des mesures de fiabilité pour les liens orientés entre les deux graphe. Par contre, pour les liens non orienté, la variabilité de la distibution des mesures de fiabilité est plus faible sur le PAG1 et la distribution est plus concentrée vers des valeurs de fiabilité élevées que sur le PAG2.

La structure non orientée du PAG paraît plus robuste face à des perturbations de l'échantillon si l'on estime le graphe à partir des variables de rang plutôt qu'à partir des variables non transformées. Ce résultat est donné à titre indicatif, mais il faudrait également comparer les structures des deux PAGs pour apprécier la qualité des résultats. Cette démarche est difficile à entreprendre dans la mesure où nous ne disposons pas d'information a priori sur les relations entre les sommets. Pour se donner une idée des différences entre les structures des deux graphes sans tenir compte de l'orientation des liens, nous avons calculé la distance de Levenshtein entre la structure du PAG1 et celle du PAG2, c'est à dire le nombre d'opérations (ajout/suppression de liens) nécessaires pour reconstruire l'un des deux graphes à partir de l'autre. Par exemple, si un lien est sur le PAG1 et n'est pas sur le PAG2, il faut supprimer le lien pour reconstuire le PAG2 à partir du PAG1. A l'inverse, si un lien du PAG2 n'apparait pas sur le PAG1, alors il faudra l'ajouter. Le PAG1 et le PAG2 que nous avons estimé ont des structures assez différentes puisqu'il est nécessaire d'effectuer 28 opérations d'ajout/suppression de liens pour « passer » d'un graphe à l'autre.

Nous verrons par la suite que les interprétations biologiques rendues possibles avec le PAG1 nous encouragent à penser que nous avons fait le bon choix et qu'il est en général plus pertinent pour l'analyse des données d'expression (la linéarité des relations n'est pas garantie).

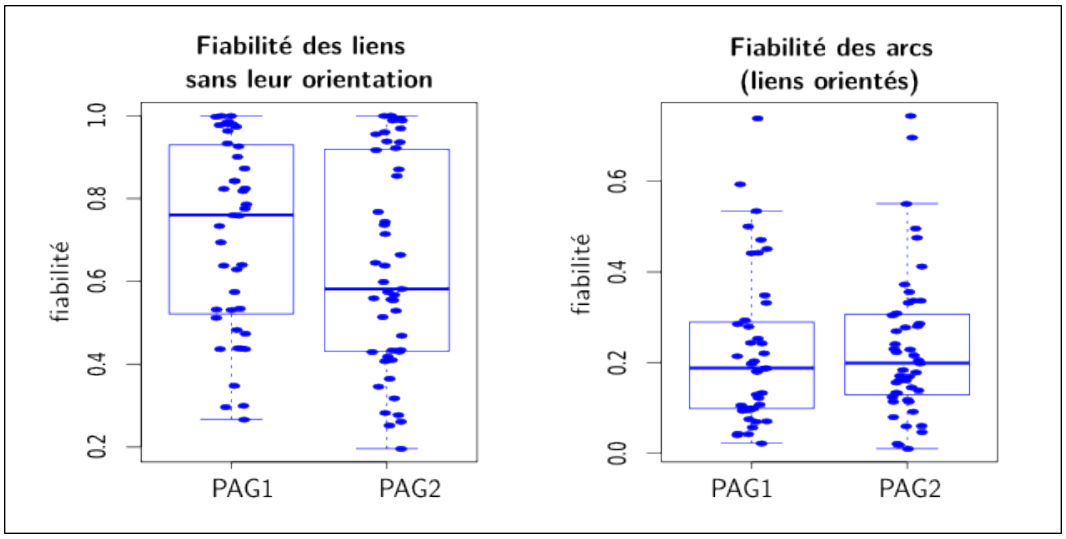


FIGURE 3.7 – Représentations boxplot pour la comparaison de la distribution des mesures de fiabilité (des liens non orientés et des liens orientés), estimées par bootstrap (fréquence d'apparition sur 1000 PAGs construits à partir d'échantillons bootstrap), entre le PAG1 et le PAG2.

3.4.2 Interprétation des résultats

Dans un premier temps, nous avons recherché des marqueurs fonctionnels de la fibrose, de la résistance à l'insuline et de la translocation bactérienne. Nous appelons marqueur fonctionnel, une communauté de gène à l'intérieur de laquelle une grande majorité de gènes sont significativement associés à la réponse (résultats des tests de permutation sur la Figure 3.5). La fonction de cette communauté a ensuite été étudiée à l'aide des résultats obtenus par les analyses d'enrichissement.

Dans un deuxième temps, nous nous sommes intéressés au mécanismes de la fibrose, en se basant entre autre sur l'analyse des relations entre les communautés (PAG).

Identification de marqueurs fonctionnels de la fibrose La communauté numéro 5 est celle qui est la plus significativement associée à la fibrose pour les tests de permutation : cette communauté regroupe 79 gènes, la variabilité de la distibution des pvaleurs est très faible et la médiane est à 0.0059 (seulement 6 gènes associés à une pvaleur > 0.05). Les résultats des analyses d'enrichissement pour cette communauté sont les suivants (les deux plus significatives pour les annotations GO et KEGG) :

index	type	annotation	pvaleur (FDR)	description
5	GO	GO :0030198	1.750e-16	extracellular matrix organization
		GO :0030199	1.651e-13	collagen fibril organization
	KEGG	04974	7.845e-06	Protein digestion and absorption
		04512	7.845e-06	ECM-receptor interaction

TABLE 3.7 – Résultats des analyses d'enrichissement effectuées sur les gènes de la communauté 5

Les gènes « de la fibrose » : la communauté 5
Les gènes de la communauté 5 sont impliqués dans l'organisation de la matrice extracellulaire et ils sont sur-exprimés chez les patients atteint de fibrose (Figure 3.8). La fibrose hépatique peut être considérée comme un processus de réparation du tissu hépatique faisant suite à une lésion initiale. Ce processus de réparation est cependant exagéré, en raison notamment d'une inflammation chronique, et il s'accompagne d'une accumulation d'éléments de la matrice extracellulaire (fibre et collagène) qui s'étend progressivement au delà de la lésion (large cicatrice). Lorsque le tissu hépatique se couvre de cicatrices, les cellules ne peuvent plus « communiquer » entre elles et remplir leurs fonctions de métabolisme et de détoxification, vitales pour l'organisme (stade de la cirrhose). Cela confirme la validité de nos observations : une sur-activité des gènes de la matrice extracellulaire chez les patients atteints de fibrose (du gène COL1A1 notamment, un gène du collagène de type 1 qui est le composant majoritaire de la fibrose hépatique ; ce gène a un profil d'expression similaire à celui du hub).

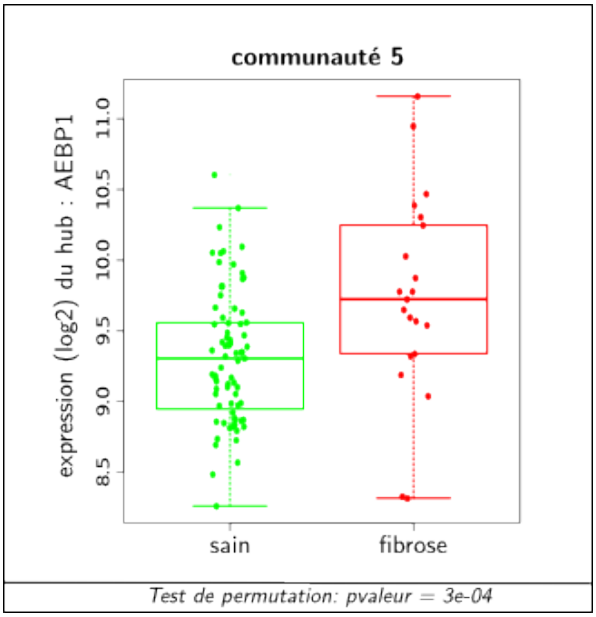


FIGURE 3.8 – Représentation de la distribution de l'expression du hub (AEBP1) de la communauté 5, chez les patients atteints de fibrose et chez les patients sans fibrose.

Identification de marqueurs fonctionnels de la résistance à l'insuline Nous avons sélectionné les quatres communautés, numérotées 9, 24, 34 et 36, qui sont les plus significativement corrélées avec le clamp (tests de permutation). La variabilité de la distribution des pvaleurs au sein de ces communautés est très faible et la pvaleur médiane est inférieure à 0.002. Les annotations GO et KEGG sur-représentées dans ces communautés sont les suivantes (les deux plus significatives pour chaque type d'annotation) :

index	type	annotation	pvaleur (FDR)	description
9	GO	GO :0097284	2.014e-06	hepatocyte apoptotic process
	GO	GO :0033209	0.0001	tumor necrosis factor-mediated signaling pathway
	KEGG	05130	0.0006	Pathogenic Escherichia coli infection
24	GO	GO :0070885	0.0002	negative regulation of calcineurin-NFAT signaling cascade
	GO	GO :0043567	0.0208	regulation of insulin-like growth factor receptor signaling pathway
	KEGG	pas d'enrichissement KEGG		
34	GO	GO :0006412	0.0311	translation
	GO	GO :0034720	0.0349	histone H3-K4 demethylation
	KEGG	04622	0.0019	RIG-I-like receptor signaling pathway
	KEGG	03013	0.0357	RNA transport
36	GO	GO :0006656	8.657e-05	phosphatidylcholine biosynthetic process
	GO	GO :0015758	0.0003	glucose transport
	KEGG	03320	4.544e-06	PPAR signaling pathway

TABLE 3.8 – Résultats des analyses d'enrichissement effectuées sur les gènes des communautés 9, 24, 34 et 36

Métabolisme des lipides et inflammation : la communauté 36 (13 gènes)
La communauté 36 est de très petite taille mais on retrouve des annotations (GO et KEGG) très significativement sur-représentées. Ces annotations sont associées aux gènes FABP5 et LPL qui sont tous deux impliqués dans le métabolisme des lipides. Nous reviendrons par la suite sur la fonction de cette communauté qui est particulièrement intéressante, car elle a été sélectionnée pour les trois variables d'intérêt (Figure 3.5), et se place en position centrale sur le PAG (Figure 3.6). Les gènes de la communauté sont faiblement exprimés chez les patients ne présentant pas de problème de résistance à l'insuline (clamp>7.5), et chez les autres, l'expression des gènes corrèle négativement avec la valeur du clamp (Figure 3.9). On identifie en petit groupe de patients qui sont quasiment tous atteints de fibrose et chez lesquels l'expression des gènes est très élevée, et les problèmes de résistance à l'insuline très sévères (groupe entouré sur le biplot et composé de patients qui se retrouve dans le groupe identifié pour le hub 9). Les gènes FABP5 et LPL ont déjà été identifiés comme étant sur-exprimés sur le foie des patients souffrant d'une maladie inflammatoire chronique [67]. Le hub de la communauté, le gène TREM2, est un gène qui participe à la réponse immunitaire et qui pourrait être impliqué dans l'inflammation chronique.

Un antidiabétique dans la communauté 24 (18 gènes)
La communauté 24 est de petite taille et peu d'annotations ont pu être identifiées comme étant sur-représentées (avec un niveau de significativité assez faible). Il est difficile de donner une interprétation fonctionnelle à cette communauté. Cependant, on retrouve un gène particulièrement intéressant, le gène IGFBP2, qui aurait un effet antidiabétique [28]. Nos observations (Figure 3.9) viennent corroborer cette hypothèse : l'expression du gène est d'autant plus faible que la résistance à l'insuline est importante (même profil que le hub de la communauté). Ce gène pourrait être utilisé comme antidiabétique car il est un possible médiateur de l'effet de la leptine, une hormone impliquée dans la régulation de la glycémie (« hormone de la satiété »).

Les gènes de la mort cellulaire : la communauté 9 (44 gènes)
Pour le hub 9, on voit apparaître sur un sous-groupe de patients ayant des problèmes de résistance à l'insuline (clamp<4), une relation (linéaire) entre l'expression du gène et le clamp, et particulièrement chez ceux avec une fibrose (Figure 3.9). L'expression du gène augmente avec la résistance à l'insuline. Chez les autres patients qui ne présentent pas un problème avéré de résistance à l'insuline, on observe une tendance similaire mais moins claire (seulement deux patients avec une fibrose). A l'intérieur de la communauté représentée par le hub 9, on retrouve le gène (et de nombreux pseudo-gènes) de la cytokératine 18 (KRT18, cytokératine de type 1), qui est sur-exprimé chez quelques patients qui ont un problème avéré de résistance à l'insuline (profils d'expression similaires à celui du hub). Les anntotations GO et KEGG sur-représentées dans la communauté sont associées à ce gène. Il a déjà fait l'objet de nombreuses publications, et en particulier, il est utilisé comme biomarqueur des NAFLD (maladie chroniques du foie) et du niveau de sévérité de la maladie. L'inflammation chronique, caractérisée en cas de NAFLD, s'accompagne d'un processus d'apoptose hépatocytaire (mort programmée des cellules), dans lequel la cytokératine 18 est impliqué. Il est possible de détecter la cytokératine 18 dans le sérum, même après la mort des cellules, car au cours de l'apoptose, une réaction aboutit à la libération de fragments de cytokératine 18 en dehors de la cellule. Il suffit alors de quantifier la cytokératine 18 dans le sérum pour obtenir une information sur les phénomènes d'apoptose : l'activation de l'apoptose s'accompagne d'une augmentation de la concentration en cytokératine 18 dans le sérum qui peut aider à déterminer le degré d'inflammation et les dommages hépatiques [17]. L'expression du gène KRT18 ne nous permet pas de discriminer le groupe des patients avec fibrose de celui des patients sans fibrose, mais peut être est-elle associée au degré de sévérité de la maladie (nous ne disposons pas de cette information) ? Nos observations mettent en évidence un lien entre les phénomènes d'apoptose et la résistance à l'insuline. Nous observons également que les patients (entourés sur le biplot) pour lesquels les gènes de l'apoptose sont le plus exprimés et qui ont la plus forte résistance à l'insuline ont presque tous une fibrose : l'aptoptose (inflammation) augmente avec la résistance à l'insuline et quand elle est trop importante, le risque de fibrose est accru.

Les gènes du chromosome Y : la communauté 34 (14 gènes)
Le hub 34 est particulièrement atypique puisque le gène est exprimé chez seulement quelques patients (Figure 3.9). En réalité, ce groupe de patients est constitué des 20 patients de sexe masculin de notre échantillon. Le gène ne s'exprime pas chez les 77 patientes car il fait parti des gènes du chromosome Y, ainsi que l'ensemble des gènes de cette communauté. La quasi-totalité des hommes de l'échantillon présentent un problème de résistance à l'insuline et 40% d'entre-eux sont atteints de fibrose (points rouges sur le biplot) contre seulement 18% chez les femmes. Cette particularité n'est certainement pas complètement indépendante de la sélection des patients effectuée pour constituer l'échantillon, mais elle est pourrait aussi être liée à une prévalence de la maladie plus élevée chez les hommes que chez les femmes [7].

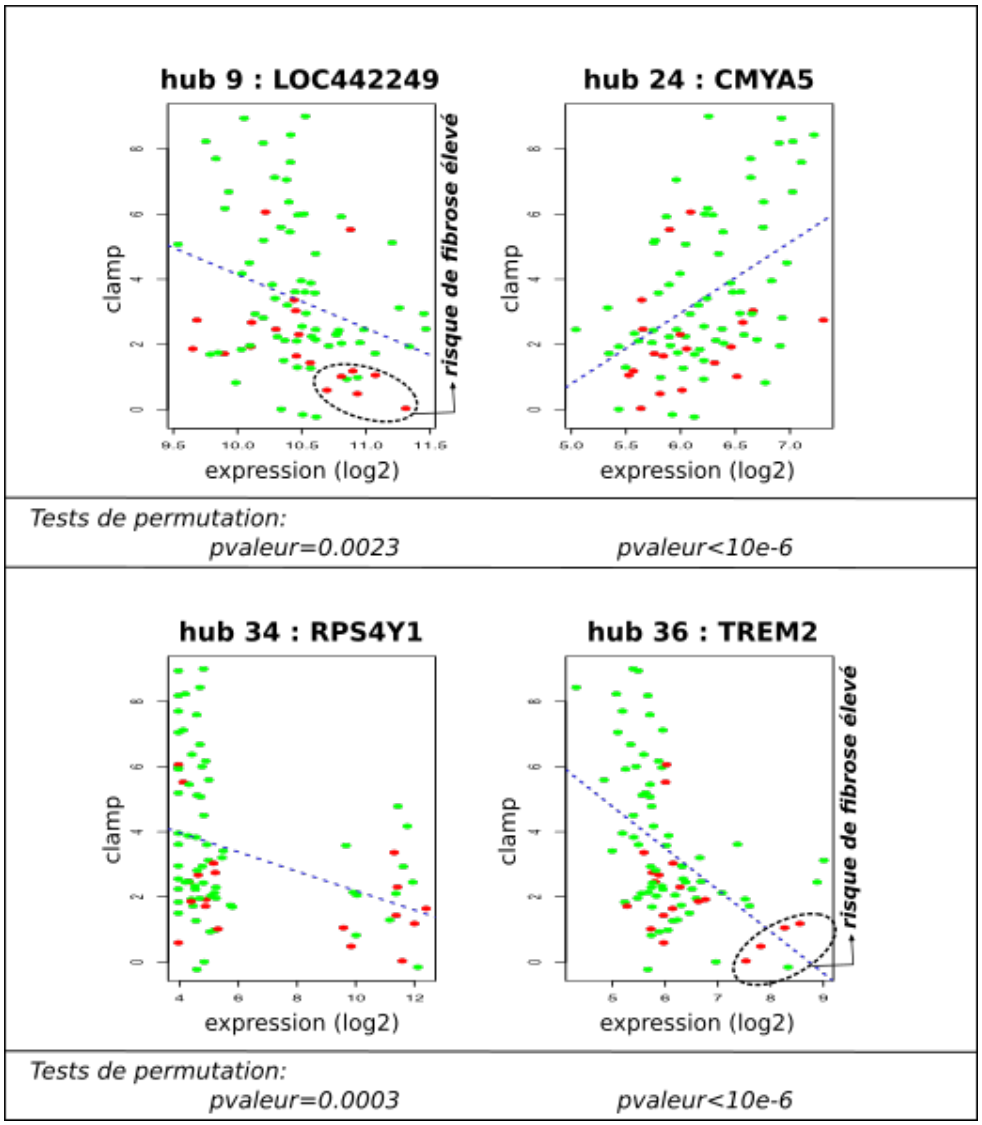


FIGURE 3.9 – Représentation pour les communautés 9, 24, 34 et 36, du nuage de points, avec en abscisse l'expression du hub de la communauté et en ordonnée la valeur du clamp. La droite bleue en pointillés correspond à la droite de la régression linéaire. Les points rouges sont associés aux patients avec fibrose et les points verts aux patients sans fibrose.

Identification de marqueurs fonctionnels de la translocation bactérienne chez les Espagnols La communauté numéro 14 est celle qui est la plus significativement (tests de permutation) corrélée avec la tranlocation bactérienne (qPCR16S) chez les Espagnols (médiane des pvaleurs égale à 0.0231). Cette communauté regroupe 25 gènes et certaines annotations GO et KEGG sont fortement sur-représentées dans la communauté (les deux plus significatives pour chaque type d'annotation) :

index	type	annotation	pvaleur (FDR)	description
14	GO	GO :0001916	6.693e-30	positive regulation of T cell mediated cytotoxicity
	GO	GO :0002474	3.721e-22	antigen processing and presentation of exogenous peptide antigen via MHC class I
	KEGG	04612	1.694e-17	Antigen processing and presentation
	KEGG	04650	4.906e-16	Natural killer cell mediated cytotoxicity

TABLE 3.9 – Résultats des analyses d'enrichissement effectuées sur les gènes de la communauté 14

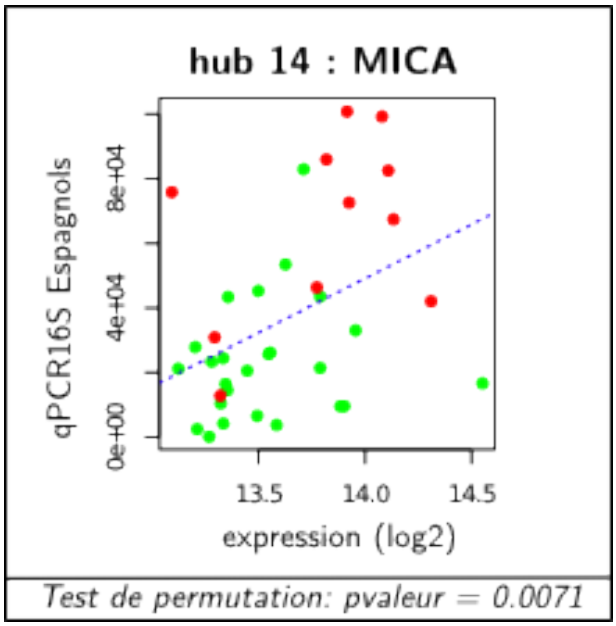


FIGURE 3.10 – Représentation du nuage de points, pour les patients Espagnols uniquement, avec en abscisse l'expression du hub de la communauté 14 et en ordonnée la valeur de la qPCR16S. La droite bleue en pointillés correspond à la droite de la régression linéaire. Les points rouges sont associés aux patients avec fibrose et les points verts aux patients sans fibrose.

Le CMH de classe 1 : la communauté 14 (25 gènes)
La communauté 14 est composée principalement de gènes du complexe majeur d'histocompatibilité de classe I (CMH 1). Ces gènes sont impliqués dans la réponse immunitaire en assurant la présentation de l'antigène aux lymphocyte T (CD8). Nous observons (Figure 3.10) une forte corrélation positive de l'expression de ces gènes avec la qPCR16S, laissant penser que les (certaines) bactéries présentes dans le sang déclenchent la réponse immunitaire. Les patients atteints de fibrose sont caractérisés par l'expression très importante de ces gènes, et une concentration en bactéries plus élevée dans le sang (Buffy coat). Un afflux massive de bactéries dans le sang associé à une inflammation aigue semble être un facteur de risque très important pour la fibrose.

Exploration des mécanismes de la fibrose hépatique Nous venons d'identifier des communautés de gènes fortement associées à la fibrose, à la résistance à l'insuline et à la translocation bactérienne. L'interprétation fonctionnelle de ces communautés met en évidence la cohérence de nos résultats avec les hypothèses biologiques qui ont déjà été validées pour expliquer certains mécanismes de la fibrose et de la résistance à l'insuline chez les personnes souffrant d'obésité. Ce qui ressort des analyses précédentes :

1. Les gènes de la matrice extracellulaire (communauté 5) sont sur-activés chez les patients atteints de fibrose.
2. Les gènes du CMH 1 (communauté 14) s'activent en présence de bactéries dans le sang (chez les Espagnols). Quand la charge bactérienne est élevée et que ces gènes sont surexprimés, le risque de fibrose est grand.

3. Un gène potentiellement antidiabétique (IGFBP2, dans la communauté 24) est sous-exprimé chez les patients très résistants à l'insuline.
4. Le gène de la cytokératine 18 (communauté 9) est sur-exprimé en cas de résistance à l'insuline sévère et il est impliqué dans les processus d'apoptose.
5. La communauté 36 semble caractériser l'inflammation chronique en lien avec le métabolisme des lipides. Les gènes sont sur-exprimés chez quelques patients qui ont une fibrose.

Nous observons sur le PAG (Figure3.6) mettant en relation les différentes communautés via leur représentant (hub), que la communauté 36 fait le lien entre celles que nous venons de citer (hormis la 24). La seule « cause potentielle » de la communauté 36 est la communauté 14, c'est à dire celle de la réponse aux bactéries. La fiabilité estimée pour ce lien orienté est de 0.2. Les communautés 9, 19 et 5 ont une « cause commune » avec la communauté 19 (fiabilité 0.19, 0.28 et 0.6 respectivement). La relation la plus fiable est celle entre la communauté 36 et les gènes « de la fibrose » (communauté 5). Cependant, cette relation suggère une cause commune, et la seule communauté qui peut éventuellement « être une casue » de la communauté 36 est la communauté de « la charge bactérienne ». En gardant à l'esprit que l'estimation de la causalité est une problématique très difficile et que les résultats doivent être appréhendés avec beaucoup de réserves, nous avons cherché à analyser un peu plus finement la communauté 36 qui constitue un point de rencontre entre les différents facteurs de risques de la fibrose. Nous avons, d'une part, analysé les relations pouvant exister entre les 13 gènes de cette communauté, et d'autre part, étudié la fonction biologique des différents gènes. L'estimation des relations de causalité entre l'expression des gènes a été effectuée à l'aide de l'algorithme FCI pour un niveau de significativité des dépendances $\alpha = 0.05$. Le PAG obtenu est représenté sur la Figure 3.11. Nous avons également estimé (1000 échantillons bootstrap) la fiabilité de la structure non orientée (dépendance) du graphe obtenu, ainsi que celle de la structure orientée (causalité). La fiabilité des liens non orientés est globalement très bonne (au minimum 0.33 et en moyenne 0.81) et celle des liens orientés assez satisfaisante (0.34 en moyenne).

Les bactéries à l'origine de l'inflammation chronique ? Nous rappelons que tous les gènes de la communauté 36 semblent, quand ils sont sur-exprimés, être associés à un risque de fibrose élevé. Le gène EMILIN2 est placé au centre du graphe 3.11. Il s'exprime dans la matrice extra-cellulaire et il a été montré [42] que ce gène est impliqué dans le déclenchement de la mort des cellules (apoptose). Deux groupes de gènes sont placés de part et d'autre du gène EMILIN2 sur le graphe. D'un côté, on observe un groupe de gènes impliqués dans la réaction inflammatoire aigüe. Le gène LGALS3 ou Galectin-3 est une « cause potentielle » de l'activation de EMILIN2 (et la seule). Ce gène s'exprime dans la matrice extra-cellulaire et il intervient dans la réponse aux bactéries. Cela suggère, que l'afflux de bactéries provoque l'activation de LGALS3 qui va à son tour potentiellement activer EMILIN2 pour provoquer l'apoptose des cellules infectées (présence de bactéries pathogènes). LGALS3 est relié à d'autre gènes (TREM2, UBD) qui sont eux-mêmes impliqué dans la réponse immunitaire, et susceptibles de provoquer la production de cytokines pro-inflammatoires. Dans un contexte d'inflammation aigue, des macrophages sont recrutés sur le site de l'inflammation (par LGALS3 en particulier). On observe un autre groupe de gènes (FABP5 et LPL) qui s'activent en présence de lipides (impliqués dans le métabolisme des lipides). Nous émettons alors

l'hypothèse, que les macrophages arrivés sur le site de l'inflammation en présence de bactéries pathogènes, se chargent de lipides par l'action de FABP5 et LPL, et libèrent en continu des cytokines pro-inflammatoires qui vont attaquer la matrice extra-cellulaire et maintenir le tissu dans un état d'inflammation chronique, provoquant ainsi la fibrose.

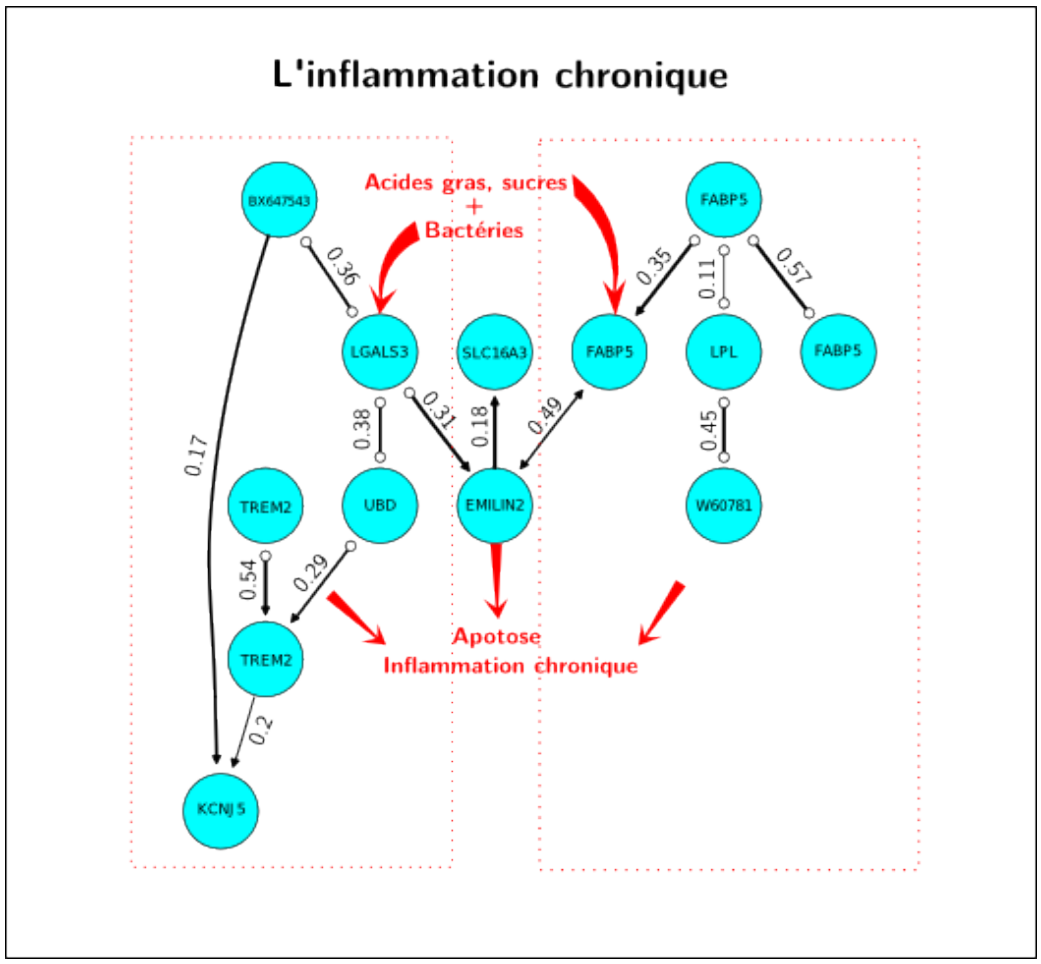


FIGURE 3.11 – Représentation du PAG estimé sur les variables d'expression des gènes de la communauté 36. L'épaisseur des liens est proportionnelle à la fiabilité (bootstrap, 1000 échantillons) du lien sans son orientation, et la valeur précisée sur chaque lien est la fiabilité du lien avec son orientation. Un gène (sommet) peut apparaître plusieurs fois sur le graphe car différents transcris placés dans cette communauté peuvent coder pour le même gène.

Conclusions Nous avons mis en évidence une petite communauté de gènes qui fait le lien entre les différents facteurs de risque de la fibrose : l'afflux de bactéries d'une part, la résistance à l'insuline et l'obésité d'autre part. Le gène LGALS3 s'active pour répondre à l'afflux des bactéries dans le sang. Son expression corrèle avec celles des gènes de la communauté 14, identifiée pour sa forte relation avec la qPCR16S chez les Espagnols. Le gène FABP5 (et LPL) s'active en présence de lipides/sucres et corrèle fortement avec les gènes de la communauté 9 associée à l'expression de la cytokeratine 18 qui nous a permis de caractériser un phénomène d'aptoptose (inflammation) accru

en cas de diabète sévère. Et enfin, le gène EMILIN2 impliqué lui aussi dans les phénomènes d'apoptose : il est sur-exprimé chez les patients avec une fibrose et son expression corrèle fortement avec les gènes de la communauté 5 qui participent au remodelage de la matrice extracellulaire. Ces différents mécanismes sont résumés sur la Figure 3.12.

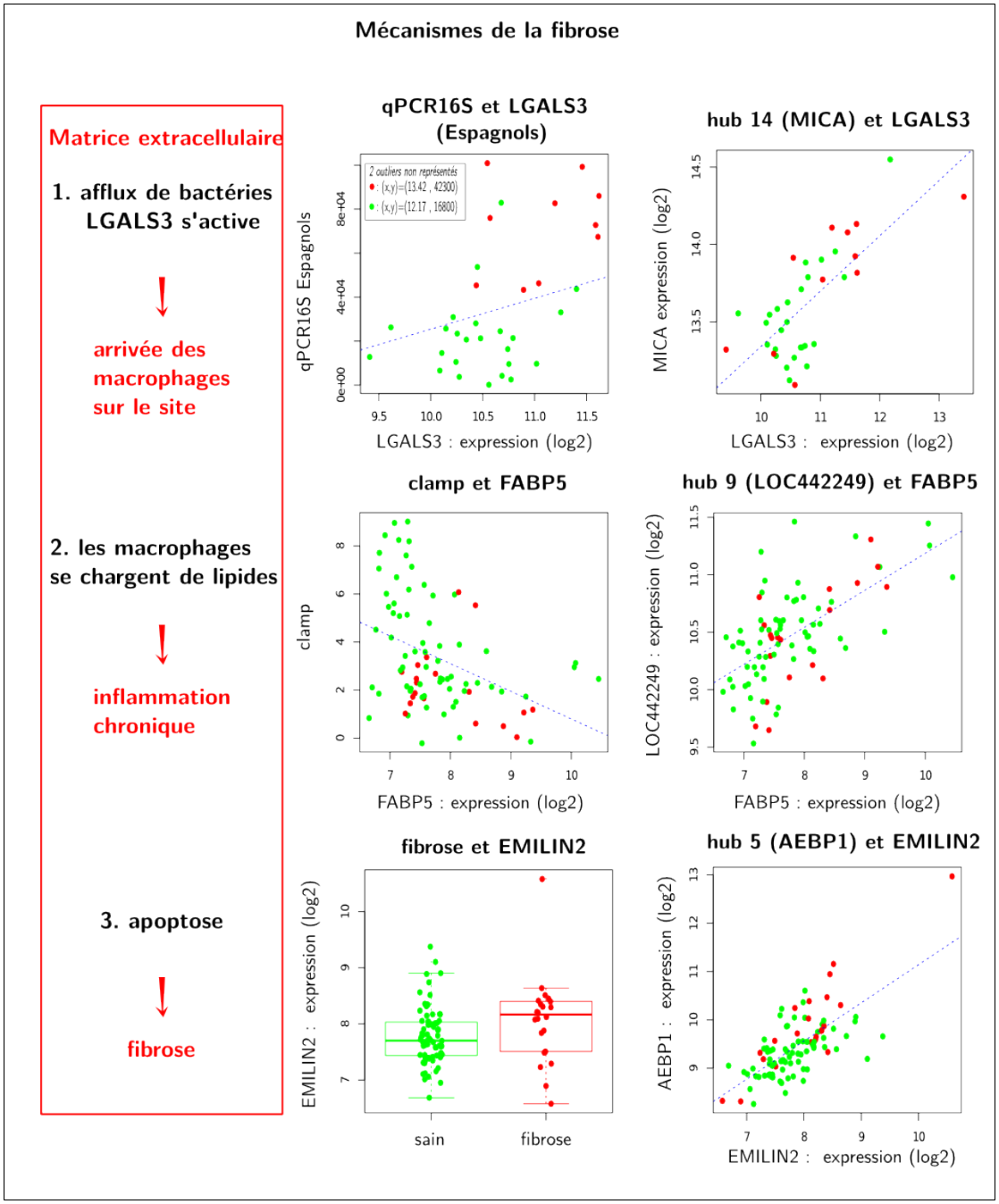


FIGURE 3.12 – Figures récapitulatives des mécanismes identifiés pour expliquer la fibrose. Les points rouges sur les biplots correspondent à des patients avec fibrose et les points verts à des patients sans fibrose.

Pour aller plus loin : une réponse au stress oxydant affaiblie, une cause de la multiplication des bactéries pathogènes sur le foie ? Nous venons de mettre en évidence l'implication des bactéries dans les mécanismes pouvant conduire à une fibrose hépatique. Pour aller plus loin, nous avons recherché la cause possible à l'afflux des bactéries dans le sang. En se repenchant sur les résultats des sélections de communautés par les tests de permutation (Figure 3.5), effectués en vue d'identifier des marqueurs fonctionnels de la translocation bactérienne, la communauté 37 a retenu notre attention. Les gènes de cette communauté sont globalement moins corrélés à la qPCR16S chez les Espagnols que ceux de la communauté 14 (CMH 1), mais les corrélations restent dans l'ensemble significatives : la pvaleur minimum est à 0.0069 et la pvaleur médiane à 0.05. Cette communauté est d'autant plus intéressante qu'elle est retrouvée chez les Italiens parmi celles qui sont les plus corrélées à la qPCR16S (avec la communauté 36), même si ces corrélations ne sont pas significatives. Nous avons alors poursuivi notre investigation en analysant les annotations GO et KEGG sur-représentées dans cette communauté :

index	type	annotation	pvaleur (FDR)	description
37	GO	GO :0019370	9.376e-22	leukotriene biosynthetic process
	GO	GO :0006750	8.833e-12	glutathione biosynthetic process
	KEGG	00430	4.171e-07	Taurine and hypotaurine metabolism
	KEGG	00460	7.282e-07	Cyanoamino acid metabolism

TABLE 3.10 – Résultats des analyses d'enrichissement effectuées sur les gènes de la communauté 37

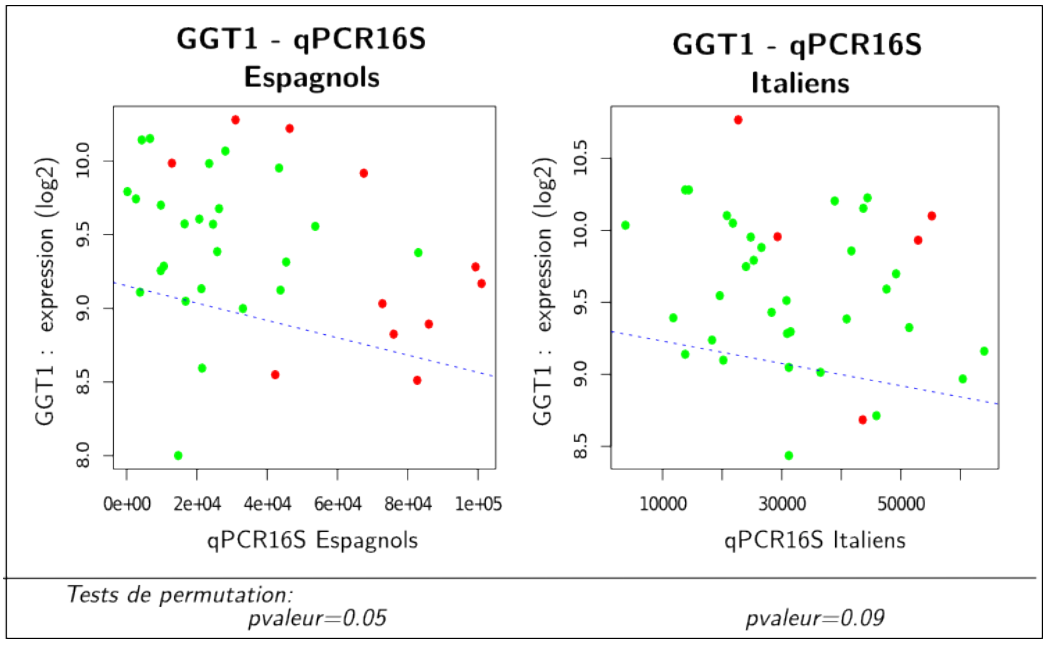


FIGURE 3.13 – Représentation des nuages de points, avec en abscisse l'expression de GGT1 (communauté 37) et en ordonnée la valeur de la qPCR16S, chez les Espagnols et chez les Italiens. La droite bleue en pointillés correspond à la droite de la régression linéaire. Les points rouges sont associés aux patients avec fibrose et les points verts aux patients sans fibrose

La réponse au stress oxidant : la communauté 37 (12 gènes) Cette communauté est de très petite taille, mais les annotations sur-représentées le sont de façon très significatives. La majeure partie des gènes sont impliqués dans le métabolisme du glutathion, un antioxydant très puissant qui intervient dans les réactions de détoxification et joue un rôle central dans le fonctionnement des lymphocytes (système immunitaire adaptatif). Les gènes GGT1, GGT2, GGTL4 et GGTLA4 appartiennent à cette communauté, et ils ont tendance à être sous-exprimés chez les patients chez lesquels la concentration bactérienne dans le sang est la plus importante (Figure 3.13). Nous émettons alors l'hypothèse, que la diminution du glutathion pourrait être à l'origine d'un affaiblissement du système immunitaire adaptatif qui conduirait à la multiplication des bactéries pathogènes dans le sang.

3.5 Comparaison de notre approche avec les méthodes d'apprentissage classiques

Afin de mettre en avant les performances de l'approche statistique que nous proposons pour l'analyse de données transcriptomiques, nous avons comparé nos résultats avec ceux obtenus par des méthodes d'apprentissage classiques telles que la régression pénalisée, la méthode algorithmique des forêts aléatoires et la méthode « filtre » des tests de permutation. Ces différentes méthodes ont été utilisées pour sélectionner les gènes qui expliquent au mieux la fibrose (variable binaire) et la résistance à l'insuline (clamp, variable continue) parmi les 5405 gènes préselectionnés (Figure 3.2). Nous avons ensuite comparé ces sélections avec celle des hubs identifiés par l'analyse du réseau de co-expression, et jugé leur pertinence sur la base de critères statistiques (ajustement, capacité prédictive, robustesse) et de critères d'interprétabilité (les fonctions biologiques des gènes).

Pertinence statistique des sélections Nous avons utilisé la fonction « glmnet » du paquet « glmnet » de R pour effectuer les régressions Lasso et Elastic-Net, et la fonction « randomForest » du paquet portant le même nom pour effectuer les sélections par la méthode des forêts aléatoires, en se basant sur la mesure d'importance définie à partir des échantillons OOB (nous avons fixé le nombre d'arbres à 10 000 et les valeurs par défaut pour les autres paramètres).

Pour chaque ensemble de gènes sélectionnés par l'une des méthodes, nous avons évalué par validation croisée, les capacités prédictives et la qualité de l'ajustement du modèle de régression classique qui explique la variable clamp (régression linéaire) ou la variable fibrose (régression logistique) à partir des gènes sélectionnés. La procédure de la validation croisée a été itérée 100 fois et les mesures de qualité ont été calculées en prenant les valeurs moyennes des mesures calculées à chaque itération. Pour la prédiction de la variable clamp, à chaque itération de la validation croisée, l'échantillon original a été subdivisé en 10 sous-échantillons par échantillonnage aléatoire simple. Pour la prédiction de la variable fibrose, nous avons subdivisé l'échantillon original en 5 sous-échantillons par un échantillonnage stratifié (patients avec et sans fibrose).

La sélection des hubs identifiés précédemment avec l'analyse du réseau de co-expression de gènes a été effectuée sur la base des p-valeurs obtenues pour les tests de permutation (Figure 3.5).

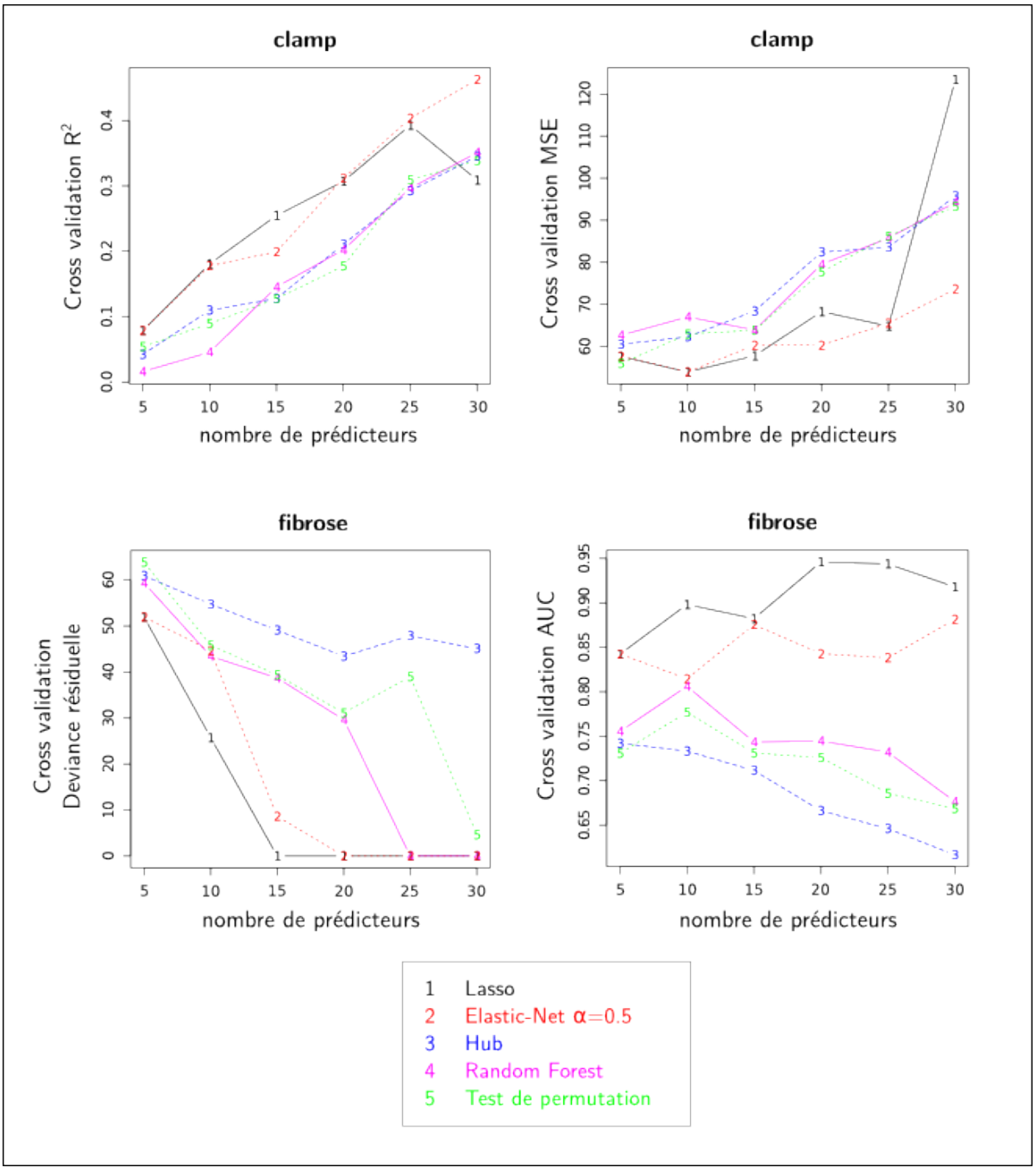


FIGURE 3.14 – Mesures de la qualité de l'ajustement et des prédictions calculées par validation croisée

La Figure 3.14 représente les mesures de la qualité de l'ajustement et des prédictions des modèles de régression classiques qui expliquent le clamp ou la fibrose à partir d'un ensemble de 5 à 30 gènes sélectionnés par l'une des différentes méthodes (régressions Lasso et Elastic-Net, forêts aléatoires, tests de permutation et hubs).

Pour les modèles de régression linéaires visant à expliquer la variable clamp, la qualité de

l'ajustement a été évaluée à l'aide du R^2 et celle des prédictions avec l'erreur quadratique moyenne (MSE). Globalement, les modèles les plus performants pour expliquer le clamp sont ceux qui intègrent les gènes sélectionnés par le Lasso ou l'Elastic-Net. Avec le Lasso cependant, les performances du modèle sont dégradées si l'on sélectionne un trop grand nombre de gènes (30 gènes). Si la pénalité est trop faible, les phénomènes de multicolinéarité deviennent problématiques et ne sont pas correctement gérés par le Lasso. La pénalité Elastic-Net permet de sélectionner un plus grand nombre de variables pouvant être fortement corrélées. Les modèles qui intègrent les hubs ont des performances très comparables à ceux qui intègrent les gènes sélectionnés par les forêts aléatoires ou les tests de permutation.

La qualité de l'ajustement des modèles de regression logistiques visant à expliquer la fibrose a été évaluée avec la déviance résiduelle (comparaison du modèle avec le modèle saturé), et leurs performances prédictives avec l'aire sous la courbe ROC (AUC). Ici encore, les modèles obtenus en sélectionnant des gènes par les régressions Lasso et Elastic-Net présentent globalement de meilleures performances. Ces modèles séparent parfaitement les patients dans les deux groupes (avec et sans fibrose) dès que le nombre de gènes sélectionnés est supérieur à 15. Néanmoins, ce phénomène est plus vraisemblablement associé à un problème de sur-ajustement qu'à la réelle pertinence des modèles car il est d'autant plus probable de trouver un modèle parfait quand le nombre de prédicteurs candidats à la sélection est très élevé. On ne rencontre pas ce problème du sur-ajustement pour les modèles qui expliquent la fibrose à partir des hubs, et pour un nombre de prédicteurs assez faible (5 gènes). ils ont des performances équivalentes à celles des modèles obtenus en sélectionnant les gènes par la méthode des forêts aléatoires ou par les tests de permutation.

Sur le plan statistique, c'est en sélectionnant les gènes par les méthodes de régression pénalisées que l'on obtient les meilleurs modèles pour expliquer/prédire le clamp et la fibrose à partir des gènes. La pertinence statistique n'est cependant pas nécessairement corrélée à la pertinence biologique des modèles et c'est cette dernière qui prime pour la validation d'un modèle.

Pertinence biologique des sélections Les analyses menées avec notre approche (identification de noyaux, de communautés, sélection de communautés, analyse des relations entre les hubs des communautés) nous ont permis de proposer des hypothèses biologiques qui font sens pour expliquer les mécanismes de la fibrose en lien avec les problèmes de résistance à l'insuline et de translocation bactérienne. Nous allons tenter d'évaluer la pertinence biologique des gènes sélectionnés (dans au moins l'un des modèles qui intégrent de 5 à 30 gènes) par les différentes méthodes pour expliquer le clamp ou la fibrose, et comparer ces résultats avec ceux que nous avons obtenus avec notre approche.

Dans un premier temps, Nous allons nous intéresser aux gènes sélectionnés pour expliquer la variable clamp. Les deux méthodes de régression pénalisées, Lasso et Elastic-Net, conduisent à sélectionner beaucoup de gènes communs. Seulement trois gènes (ITGBL1, FBXO2, PHLDA3) appartiennent à l'une des communautés identifiées par l'analyse du réseau de co-expression (la numéro 5, 21 et 22 respectivement) mais pas à celles que nous avions observées comme étant les plus fortement liées aux problèmes de résistance à l'insuline (les communautés 9, 24 et 36). La liste des gènes sélectionnés est la suivante (précision entre parenthèses si le gène est sélectionné uniquement avec le Lasso ou Elastic-Net) :

A_24_P490011, ALKBH2, ARHGEF19 (Lasso), ARSE (Lasso), BC070363, BX647987, C9orf125, CHRM2 (Lasso), CHRM3, CR617033, DOK7, ENST00000366569 (EN), FBXO2, FLJ37644, FMO5, FUT3, GPLD1 (Lasso), IL3RA, ITGBL1, KRT18P42 (Lasso), KRTCAP3, LGALS4, LOC644728, LRG1, LRP4, NEK10, PEX6 (Lasso), PHLDA3, PLGLB1 (EN), SLC23A2, SLC6A16, SSPO, STMN2 (EN), THC2657593, THC2660361, THC2755690 (Lasso), USP34 (Lasso), VIL1 (EN), WTAP (Lasso)

Les analyses d'enrichissement menées sur cette liste de gènes pour les annotations GO et KEGG n'aboutissent pas à des résultats significatifs (p-valeur< 0.05).

Avec la méthode des forêts aléatoires, on retrouve certains gènes que nous avions identifiés comme étant potentiellement associés aux problèmes de résistance à l'insuline : trois gènes qui appartiennent à la communauté 36 (TREM2, W60781 et FABP5), deux gènes de la communauté 24 (NUDT13 et PLEKHA4), deux gènes de la communauté 22 (ZMAT3 et PHLDA3), et un gène dans les communautés 5, 6 et 19 (COL1A1, IL1RAP et AASS). La liste des gènes sélectionnés est la suivante (* gène sélectionné également avec le Lasso ou Elastic-Net) :

A_23_P13232, A_32_P67303, AASS, B3GNT5, C9orf125 *, CCL20, COL1A1, FABP5, HS3ST2, IL1RAP, IL3RA *, KRT19, LPA, LRP4 *, MRAS, NUDT13, P2RY2, PHLDA3 *, PLEKHA4, PLGLB1, PTGFR, RASSF4, SLC6A16 *, STMN2 *, THC2657593 *, TREM2, W60781, ZMAT3

Aucune annotation GO ou KEGG n'est significativement sur-représentée dans cette liste de gènes.

Les gènes sélectionnés par les tests de permutation sont tous différents de ceux sélectionnés avec les régressions pénalisées ou la méthode des forêts aléatoires. On retrouve un gène dans la communauté 1, 6, 7, 10, 23 et 26 (C1QA, CPEB4, NSDHL, RPS7 et GFAP respectivement). Ces communautés ne font pas partie de celles que nous avions sélectionnées comme étant associées au clamp. Aucune annotation GO ou KEGG est significativement sur-représentée dans la liste de gènes :

A_23_P64962, A_24_P716249, A_32_P18630, AK056855, BC073929, BF436529, BG618474, BX537816, C19orf33, C1QA, C20orf46, CPEB4, CTSB, CTTNBP2NL, FLJ21272, GFAP, hCG_2026038, HPGD, ID1, ID2, LOC646161, MX2, NM_001018022, NSDHL, OTUD6A, RPS7, SEC61A2, TBX3, TCL1A, TMEM56

Nous allons maintenant nous intéresser aux gènes sélectionnés pour expliquer la fibrose. Avec les régressions Lasso et Elastic-Net, deux gènes appartenant à des communautés sélectionnées pour expliquer la fibrose sont retrouvés : un gène dans la communauté 11 (PCK1) et un gène dans la communauté 28 (AF161344). Trois gènes (APOA5, PCK1 et CPT1A) sont annotés avec l'annotation KEGG « 03320 : PPAR signaling pathway ». Nous avions montré la potentielle implication du réseau PPAR (métabolisme des lipides) dans les mécanismes pouvant conduire à la fibrose (gènes du réseau PPAR dans la communauté 36). Les gènes sélectionnés sont les suivants :

AF161344, AK021804, AK095791, AL832717 (Lasso), APOA5 (EN), ARHGEF19 (Lasso), AVPR1A, BC010544 (Lasso), BM717049, C14orf68, C14orf80, CGREF1, CPT1A (Lasso), CPZ (EN), DNAJC15, DUSP8, EFCAB4B, ENST00000252229 (EN), ENST00000273340 (Lasso), ENST00000376155 (EN), FANCC, ITIH3, KIF5C, KRTCAP3, LHX6, LOC220594 (Lasso), LSS (Lasso), OR2A9P (Lasso), PCK1, PPP2R1B

(EN), PTPRT, SLC6A13, STMN2 (EN), TFF2, THC2651501 (EN), THC2684461, THC2739159, TNC, ZNF783 (EN)

Pour la sélection par les forêts aléatoires on retrouve quatre gènes de la communauté 1 (CD6, JAK3, RAC2 et FCGR3B), quatre gènes de la communauté 2 (AMPD1, RP5-821D11.2, FER1L4, AF267875) et un gène de la communauté 5 (COL1A2), donc principalement des gènes participant à l'immunité (communauté 1 et 2). Les gènes sélectionnés sont les suivants :

A_32_P170814, AF267875, AKR1B1, AMPD1, ATOH8, AVPR1A *, BQ897248, CD6, COL1A2, COP1, CXorf6, EMR1, ENST00000376155 *, FCGR3B, FCN2, FCRL5, FER1L4, HLA-DQB1, JAK3, NBLA00301, NPNT, PTPRT *, QSOX1, RAC2, RP5-821D11.2, SLC16A14, SLITRK6, SPON1, SULF2, THC2560357

En sélectionnant par les tests de permutation on retrouve un gène dans les communautés 11, 24 et 40 (A_32_P20040, PLEKHA4, GSTM1) et aucune annotation GO ou KEGG n'est sur-représentée :

A_24_P152793, A_24_P482124, A_32_P20040, AF090933, AGXT, C5AR1, C8orf46, CPNE1, CPS1, CREM, ENST00000356572, EYA2, FLJ40722, GOLGA, GSTM1, HPD, ICA1, MXRA7, NBPF14, NXF3, OLFML2A, PKLR, PLEKHA4, RET, THC2739760, TREM1, TRPM2, UNQ5783, USP43, ZNF300

Robustesse face au bruit Pour tester la robustesse des méthodes nous avons ajouté un bruit aléatoire gaussien d'espérance nulle et d'écart-type $\sigma = 0.5$ à chaque observation de l'expression d'un gène. Nous avons utilisé notre approche pour identifier les communautés sur le graphe construit à partir des données bruitées (même paramètre que pour l'analyse des données non bruitées). Avec la même approche que pour les données non bruitées nous avons sélectionné par les tests de permutation les communautés les plus associées à la fibrose et au clamp. Les résultats sont représentés sur la Figure 3.15.

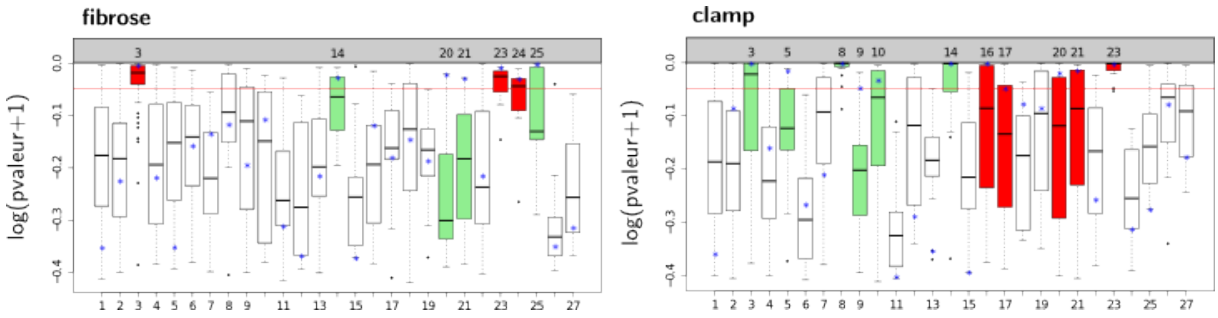


FIGURE 3.15 – Représentation « boxplot » des distributions des p-valeurs au sein des communautés pour les variables réponses fibrose et clamp.

C'est la communauté 3 qui est la plus significativement associée à la fibrose. Cette communauté contient 26 gènes (sur 39) qui sont dans la communauté 5 identifiée sur les données non bruitées, et également la plus liée à la fibrose (gènes de la matrice extracellulaire). Pour la variable clamp, on retrouve aussi des similitudes avec les communautés sélectionnées sur les données non bruitées :

numéro communauté		**nombre de gènes communs**	**description**
données bruitées	*données non bruitées*		
3 (39 gènes)	5	26	matrice extracellulaire
8 (20 gènes)	24	6	antidiabétique
14 (16 gènes)	36	7	métabolisme des lipides
	17	1	
23 (11 gènes)	34	11	chromosome Y

TABLE 3.11 – Intersection entre les communautés détectées sur données bruitées et non bruitées

Globalement, avec notre approche, les résultats semblent assez robustes. Avec les méthodes de régression pénalisées, des forêts aléatoires et des tests de permutation, nous avons sélectionné à partir des données bruitées (de la même façon que pour les données non bruitées), les gènes (une trentaine) les plus associés au clamp et à la fibrose. Parmi les gènes sélectionnés pour expliquer le clamp : avec le Lasso et Elastic-Net, on retrouve sept gènes (KRTCAP3, C9orf125, LOC644728, PEX6, VIL1, STMN2, LRG1) qui avaient été sélectionnés avec ces mêmes méthodes à partir des données non bruitées, trois avec les forêts aléatoires (C9orf125, STMN2, A_23_P13232) et un seul avec les tests de permutation (TCL1A). Parmi les gènes sélectionnés pour expliquer la fibrose : avec le Lasso et Elastic-Net, on retrouve six gènes (LHX6, STMN2, PCK1, KIF5C, ITIH3, AL832717), qui avaient été sélectionnés avec ces mêmes méthodes, aucun avec les forêts aléatoires et un seul avec les tests de permutation (AF090933).

Conclusions Les performances statistiques des modèles obtenus en sélectionnant les gènes par les méthodes de régression Lasso ou Elastic-Net semblent être meilleures mais peuvent souffrir de problèmes de sur-ajustement dès que l'on sélectionne un trop grand nombre de gènes. L'interprétation biologique des résultats obtenus par les différentes méthodes d'apprentissage classiques est plus laborieuse qu'avec notre approche. Nous n'avons pu identifier des annotations (GO ou KEGG) sur-représentées dans les différentes listes de gènes et il faudrait étudier un à un les gènes pour en valider la pertinence biologique. Les performances statistiques des modèles construits à partir des hubs ou des gènes sélectionnés par les forêts aléatoires ou les tests de permutation sont assez comparables. Les gènes sélectionnés par les tests de permutation sont cependant complètement différents de ceux sélectionnés avec les autres méthodes et donc peut être moins pertinents sur le plan biologique. On retrouve en sélectionnant par les forêts aléatoires certains gènes de la communauté 36 que nous avions proposée pour expliquer les mécanismes de la fibrose, certains gènes de l'immunité, et en sélectionnant par les régressions pénalisées, certains gènes impliqués dans le métabolisme des lipides (PPAR). Un travail plus approfondi mené par des biologistes sur ces listes de gènes pourrait éventuellement confirmer nos résultats. En ajoutant du bruit sur nos données, nous parvenons avec notre approche à retrouver les principaux résultats observés sur les données non bruitées alors qu'avec les méthodes d'apprentissage classiques, les intersections entre les ensembles de gènes sélectionnés à partir des données bruitées et ceux sélectionnés à partir des données non bruitées, sont faibles. Notre approche est plus robuste car elle intègre une information supplémentaire, celle de l'interaction entre gènes. On sélectionne uniquement des gènes qui appartiennent à une communauté de gènes en interaction, et l'identification de ces communautés favorise nettement les interprétations biologiques.

Conclusion

L'objectif de la thèse était de proposer des outils statistiques pour l'analyse des données transcriptomiques afin d'étudier les mécanismes génétiques d'une pathologie et de proposer des marqueurs génétiques de la pathologie.

Les analyses statistiques des données transcriptomiques sont délicates à cause du problème de la grande dimension, d'une part, et de l'existence de groupes de variables fortement corrélées, d'autre part. Nous avons mis en évidence les limites des méthodes d'apprentissage supervisé proposées pour traiter ce type de données, telles que les forêts aléatoires ou les régressions pénalisées. Bien que les capacités prédictives des modèles construits avec ces méthodes puissent paraître satisfaisantes, il est souvent difficile de les justifier sur le plan biologique et les sélections de variables au sein des groupes de variables corrélées sont effectuées uniquement sur des critères statistiques ce qui ne permet en rien d'en garantir leur pertinence biologique.

Nous proposons alors, d'identifier les groupes de variables fortement corrélées, c'est à dire les groupes ou communautés de gènes co-exprimés, avant de chercher à expliquer les mécanismes de la maladie à partir des variations d'expression observées au sein de ces communautés. Très souvent, les gènes co-exprimés ont des fonctions biologiques similaires dans la cellule et l'identification des communautés de gènes co-exprimés permet de retrouver (par des analyses d'enrichissement) ces fonctions biologiques et d'étudier les mécanismes impliquant ces fonctions et pouvant être associés à la maladie. L'analyse des réseaux de co-expression de gènes offre l'opportunité d'identifier des communautés de gènes en tenant compte non pas des interactions entre les gènes prise deux à deux, mais des interactions observées au sein d'un système complexe. Pour ce faire, nous proposons une approche efficace et originale qui permet d'identifier des communautés de gènes en intégrant de l'information portant sur le réseau de co-expression de gènes tout en s'affranchissant de certaines problématiques liées à ce type d'analyse. En effet, avec notre approche il n'est plus nécessaire de modéliser le réseau de co-expression par un unique graphe, un seul paramètre doit être choisi, il est possible de sélectionner les éléments les « plus aux centre »dans les communautés (coeur de classes), les communautés peuvent être disjointes ou chevauchantes...

Après avoir identifier des communautés de gènes, l'objectif est de faire le lien entre les variations d'expression observées au sein de ces communautés et la maladie. Dans ce contexte, nous proposons d'étudier la significativité des relations entre les profils d'expression observés dans chacune des communautés et la variable qui caractérise la maladie, en utilisant des tests de permutation. De plus, pour étudier plus finement les mécanismes de la maladie, nous proposons de choisir des représentants des communautés (sélection des hubs) et de construire des réseaux bayésiens sur ces représentants pour étudier les liens de causalités entre les différentes fonctions biologiques associées aux communautés. Nous avons montré sur un jeu de données réelles que notre approche nous permet d'obtenir des résultats pertinents sur le plan biologique et qui semblent suffisamment robustes.

Bibliographie

[1] Reka Albert. Scale-free networks in cell biology. *Journal of cell science*, 118(21) :4947–4957, 2005.

[2] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1) :47, 2002.

[3] Albert-László Barabási and Réka Albert. Emergence of scaling in random networks. *science*, 286(5439) :509–512, 1999.

[4] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology : understanding the cell's functional organization. *Nature reviews genetics*, 5(2) :101–113, 2004.

[5] Vladimir Batagelj and Matjaž Zaveršnik. Generalized cores. *arXiv preprint cs/0202039*, 2002.

[6] Peter J Bickel and Elizaveta Levina. Covariance regularization by thresholding. *The Annals of Statistics*, pages 2577–2604, 2008.

[7] D Montgomery Bissell. Sex and hepatic fibrosis. *Hepatology*, 29(3) :988–989, 1999.

[8] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics : Theory and Experiment*, 2008(10) :P10008, 2008.

[9] Leo Breiman. Random forests. *Machine learning*, 45(1) :5–32, 2001.

[10] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

[11] Aaron Clauset, Mark EJ Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical review E*, 70(6) :066111, 2004.

[12] Diego Colombo, Marloes H Maathuis, Markus Kalisch, Thomas S Richardson, et al. Learning high-dimensional directed acyclic graphs with latent and selection variables. *The Annals of Statistics*, 40(1) :294–321, 2012.

[13] Frederick E Dewey, Marco V Perez, Matthew T Wheeler, Clifton Watt, Joshua Spin, Peter Langfelder, Steve Horvath, Sridhar Hannenhalli, Thomas P Cappola, and Euan A Ashley. Gene coexpression network topology of cardiac development, hypertrophy, and failure. *Circulation : cardiovascular genetics*, 4(1) :26–35, 2011.

[14] Ramón Díaz-Uriarte and Sara Alvarez De Andres. Gene selection and classification of microarray data using random forest. *BMC bioinformatics*, 7(1) :3, 2006.

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

[16] Paul Erd6s and A Rényi. On the evolution of random graphs. *Publ. Math. Inst. Hungar. Acad. Sci*, 5 :17–61, 1960.

[17] Ariel E Feldstein, Anna Wieckowska, A Rocio Lopez, Yao-Chang Liu, Nizar N Zein, and Arthur J McCullough. Cytokeratin-18 fragment levels as noninvasive biomarkers for nonalcoholic steatohepatitis : A multicenter validation study. *Hepatology*, 50(4) :1072–1078, 2009.

[18] Sir Ronald Aylmer Fisher, Statistiker Genetiker, Ronald Aylmer Fisher, Statistician Genetician, Ronald Aylmer Fisher, and Statisticien Généticien. *The design of experiments*, volume 12. Oliver and Boyd Edinburgh, 1960.

[19] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3) :75–174, 2010.

[20] Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1) :1, 2010.

[21] Nir Friedman, Moises Goldszmidt, and Abraham Wyner. Data analysis with bayesian networks : A bootstrap approach. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pages 196–205. Morgan Kaufmann Publishers Inc., 1999.

[22] Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using bayesian networks to analyze expression data. *Journal of computational biology*, 7(3-4) :601–620, 2000.

[23] Chris Gaiteri, Ying Ding, Beverly French, George C Tseng, and Etienne Sibille. Beyond modules and hubs : the potential of gene coexpression networks for investigating molecular mechanisms of complex brain disorders. *Genes, Brain and Behavior*, 13(1) :13–24, 2014.

[24] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12) :7821–7826, 2002.

[25] Gene H Golub and Charles F Van Loan. *Matrix computations*, volume 3. JHU Press, 2012.

[26] Naftali Harris and Mathias Drton. Pc algorithm for nonparanormal graphical models. *The Journal of Machine Learning Research*, 14(1) :3365–3383, 2013.

[27] David Heckerman, Dan Geiger, and David M Chickering. Learning bayesian networks : The combination of knowledge and statistical data. *Machine learning*, 20(3) :197–243, 1995.

[28] Kristina Hedbacker, Kıvanç Birsoy, Robert W Wysocki, Esra Asilmaz, Rexford S Ahima, I Sadaf Farooqi, and Jeffrey M Friedman. Antidiabetic effects of igfbp2, a leptin-regulated gene. *Cell metabolism*, 11(1) :11–22, 2010.

[29] Arthur E Hoerl and Robert W Kennard. Ridge regression : Biased estimation for nonorthogonal problems. *Technometrics*, 12(1) :55–67, 1970.

[30] TC Hu. Letter to the editor-the maximum capacity route problem. *Operations Research*, 9(6) :898–900, 1961.

[31] Markus Kalisch and Peter Bühlmann. Estimating high-dimensional directed acyclic graphs with the pc-algorithm. *The Journal of Machine Learning Research*, 8 :613–636, 2007.

[32] Jin Kim and Judea Pearl. A computational model for causal and diagnostic reasoning in inference systems. 1983.

[33] Joseph B Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 7(1) :48–50, 1956.

[34] Peter Langfelder and Steve Horvath. Wgcna : an r package for weighted correlation network analysis. *BMC bioinformatics*, 9(1) :559, 2008.

[35] Peter Langfelder, Bin Zhang, and Steve Horvath. Dynamic tree cut : in-depth description, tests and applications. *November*, 22 :2007, 2007.

[36] Peter Langfelder, Bin Zhang, and Steve Horvath. Defining clusters from a hierarchical cluster tree : the dynamic tree cut package for r. *Bioinformatics*, 24(5) :719–720, 2008.

[37] Sophie Lebre, Jennifer Becq, Frederic Devaux, Michael PH Stumpf, and Gaelle Lelandais. Statistical inference of the time-varying structure of gene-regulation networks. *BMC systems biology*, 4(1) :130, 2010.

[38] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R news*, 2(3) :18–22, 2002.

[39] John Ludbrook and Hugh Dudley. Why permutation tests are superior to t and f tests in biomedical research. *The American Statistician*, 52(2) :127–132, 1998.

[40] Ulrike V Luxburg, Olivier Bousquet, and Mikhail Belkin. Limits of spectral clustering. In *Advances in neural information processing systems*, pages 857–864, 2004.

[41] Christopher Meek. Strong completeness and faithfulness in bayesian networks. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 411–418. Morgan Kaufmann Publishers Inc., 1995.

[42] Maurizio Mongiat, Giovanni Ligresti, Stefano Marastoni, Erica Lorenzon, Roberto Doliana, and Alfonso Colombatti. Regulation of the extrinsic apoptotic pathway by the extracellular matrix glycoprotein emilin2. *Molecular and cellular biology*, 27(20) :7176–7187, 2007.

[43] Mark EJ Newman. Fast algorithm for detecting community structure in networks. *Physical review E*, 69(6) :066133, 2004.

[44] Andrew Y Ng, Michael I Jordan, Yair Weiss, et al. On spectral clustering : Analysis and an algorithm. *Advances in neural information processing systems*, 2 :849–856, 2002.

[45] Regina Nuzzo. Statistical errors. *Nature*, 506(7487) :150–152, 2014.

[46] Judea Pearl. *Bayesian networks : A model of self-activated memory for evidential reasoning*. University of California (Los Angeles). Computer Science Department, 1985.

[47] Judea Pearl and Stuart Russell. *Bayesian networks*. Computer Science Department, University of California, 1998.

[48] Judea Pearl, Thomas Verma, et al. *A theory of inferred causation*. Morgan Kaufmann San Mateo, CA, 1991.

[49] Maurice Pollack. Letter to the editor-the maximum capacity through a network. *Operations Research*, 8(5) :733–736, 1960.

[50] Angela P Presson, Nam K Yoon, Lora Bagryanova, Vei Mah, Mohammad Alavi, Erin L Maresh, Ayyappan K Rajasekaran, Lee Goodglick, David Chia, and Steve Horvath. Protein expression based multimarker analysis of breast cancer samples. *BMC cancer*, 11(1) :230, 2011.

[51] Robert Clay Prim. Shortest connection networks and some generalizations. *Bell system technical journal*, 36(6) :1389–1401, 1957.

[52] Erzsébet Ravasz, Anna Lisa Somera, Dale A Mongru, Zoltán N Oltvai, and A-L Barabási. Hierarchical organization of modularity in metabolic networks. *science*, 297(5586) :1551–1555, 2002.

[53] Thomas Richardson and Peter Spirtes. Ancestral graph markov models. *Annals of Statistics*, pages 962–1030, 2002.

[54] Satu Elisa Schaeffer. Graph clustering. *Computer Science Review*, 1(1) :27–64, 2007.

[55] Stephen B Seidman. Network structure and minimum degree. *Social networks*, 5(3) :269–287, 1983.

[56] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8) :888–905, 2000.

[57] Peter Spirtes and Clark Glymour. An algorithm for fast recovery of sparse causal graphs. *Social science computer review*, 9(1) :62–72, 1991.

[58] Peter Spirtes, Clark Glymour, and Richard Scheines. Causation, prediction, and search. adaptive computation and machine learning, 2000.

[59] Peter Spirtes, Christopher Meek, and Thomas Richardson. Causal inference in the presence of latent variables and selection bias. In *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*, pages 499–506. Morgan Kaufmann Publishers Inc., 1995.

[60] René Thomas. Boolean formalization of genetic control circuits. *Journal of theoretical biology*, 42(3) :563–585, 1973.

[61] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[62] Deepak Verma and Marina Meila. A comparison of spectral clustering algorithms. *University of Washington Tech Rep UWCSE030501*, 1 :1–18, 2003.

[63] Irina Voineagu, Xinchen Wang, Patrick Johnston, Jennifer K Lowe, Yuan Tian, Steve Horvath, Jonathan Mill, Rita M Cantor, Benjamin J Blencowe, and Daniel H Geschwind. Transcriptomic analysis of autistic brain reveals convergent molecular pathology. *Nature*, 474(7351) :380–384, 2011.

[64] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4) :395–416, 2007.

[65] Ulrike Von Luxburg, Mikhail Belkin, and Olivier Bousquet. Consistency of spectral clustering. *The Annals of Statistics*, pages 555–586, 2008.

[66] Conrad H Waddington. The strategy of the genes allen and unwin, 1957.

[67] Jukka Westerbacka, Maria Kolak, Tuula Kiviluoto, Perttu Arkkila, Jukka Sirén, Anders Hamsten, Rachel M Fisher, and Hannele Yki-Järvinen. Genes involved in fatty acid partitioning and binding, lipolysis, monocyte/macrophage recruitment, and inflammation are overexpressed in the human fatty liver of insulin-resistant subjects. *Diabetes*, 56(11) :2759–2765, 2007.

[68] Zhigang Xue, Kevin Huang, Chaochao Cai, Lingbo Cai, Chun-yan Jiang, Yun Feng, Zhenshan Liu, Qiao Zeng, Liming Cheng, Yi E Sun, et al. Genetic programs in human and mouse early embryos revealed by single-cell rna [thinsp] sequencing. *Nature*, 500(7464) :593–597, 2013.

[69] Donghui Yan, Ling Huang, and Michael I Jordan. Fast approximate spectral clustering. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 907–916. ACM, 2009.

[70] Bin Zhang and Steve Horvath. A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology*, 4(1), 2005.

[71] Jiji Zhang. Causal reasoning with ancestral graphs. *The Journal of Machine Learning Research*, 9 :1437–1474, 2008.

[72] Jiji Zhang. On the completeness of orientation rules for causal discovery in the presence of latent confounders and selection bias. *Artificial Intelligence*, 172(16) :1873–1896, 2008.

[73] Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 67(2) :301–320, 2005.

RÉSUMÉ

Les nouvelles biotechnologies offrent aujourd'hui la possibilité de récolter une très grande variété et quantité de données biologiques (génomique, protéomique, métagénomique...), ouvrant ainsi de nouvelles perspectives de recherche pour la compréhension des processus biologiques. Dans cette thèse, nous nous sommes plus spécifiquement intéressés aux données transcriptomiques, celles-ci caractérisant l'activité ou le niveau d'expression de plusieurs dizaines de milliers de gènes dans une cellule donnée. L'objectif était alors de proposer des outils statistiques adaptés pour analyser ce type de données qui pose des problèmes de "grande dimension" (n«p), car collectées sur des échantillons de tailles très limitées au regard du très grand nombre de variables (ici l'expression des gènes).

La première partie de la thèse est consacrée à la présentation de méthodes d'apprentissage supervisé, telles que les forêts aléatoires de Breiman et les modèles de régressions pénalisées, utilisées dans le contexte de la grande dimension pour sélectionner les gènes (variables d'expression) qui sont les plus pertinents pour l'étude de la pathologie d'intérêt. Nous évoquons les limites de ces méthodes pour la sélection de gènes qui soient pertinents, non pas uniquement pour des considérations d'ordre statistique, mais qui le soient également sur le plan biologique, et notamment pour les sélections au sein des groupes de variables fortement corrélées, c'est à dire au sein des groupes de gènes co-exprimés. Les méthodes d'apprentissage classiques considèrent que chaque gène peut avoir une action isolée dans le modèle, ce qui est en pratique peu réaliste. Un caractère biologique observable est la résultante d'un ensemble de réactions au sein d'un système complexe faisant interagir les gènes les uns avec les autres, et les gènes impliqués dans une même fonction biologique ont tendance à être co-exprimés (expression corrélée).

Ainsi, dans une deuxième partie, nous nous intéressons aux réseaux de co-expression de gènes sur lesquels deux gènes sont reliés si ils sont co-exprimés. Plus précisément, nous cherchons à mettre en évidence des communautés de gènes sur ces réseaux, c'est à dire des groupes de gènes co-exprimés, puis à sélectionner les communautés les plus pertinentes pour l'étude de la pathologie, ainsi que les "gènes clés" de ces communautés. Cela favorise les interprétations biologiques, car il est souvent possible d'associer une fonction biologique à une communauté de gènes. Nous proposons une approche originale et efficace permettant de traiter simultanément la problématique de la modélisation du réseau de co-expression de gènes et celle de la détection des communautés de gènes sur le réseau. Nous mettons en avant les performances de notre approche en la comparant à des méthodes existantes et populaires pour l'analyse des réseaux de co-expression de gènes (WGCNA et méthodes spectrales).

Enfin, par l'analyse d'un jeu de données réelles, nous montrons dans la dernière partie de la thèse que l'approche que nous proposons permet d'obtenir des résultats convaincants sur le plan biologique, plus propices aux interprétations et plus robustes que ceux obtenus avec les méthodes d'apprentissage supervisé classiques.

ABSTRACT

Today's, new biotechnologies offer the opportunity to collect a large variety and volume of biological data (genomic, proteomic, metagenomic...), thus opening up new avenues for research into biological processes. In this thesis, what we are specifically interested is the transcriptomic data indicative of the activity or expression level of several thousands of genes in a given cell. The aim of this thesis was to propose proper statistical tools to analyse these high dimensional data (n«p) collected from small samples with regard to the very large number of variables (gene expression variables).

The first part of the thesis is devoted to a description of some supervised learning methods, such as random forest and penalized regression models. The following methods can be used for selecting the most relevant disease-related genes. However, the statistical relevance of the selections doesn't determine the biological relevance, and particularly when genes are selected within a group of highly correlated variables or co-expressed genes. Common supervised learning methods consider that every gene can have an isolated action in the model which is not so much realistic. An observable biological phenomenum is the result of a set of reactions inside a complex system which makes genes interact with each other, and genes that have a common biological function tend to be co-expressed (correlation between expression variables).

Then, in a second part, we are interested in gene co-expression networks, where genes are linked if they are co-expressed. More precisely, we aim to identify communities of co-expressed genes, and then to select the most relevant disease-related communities as well as the "key-genes" of these communities. It leads to a variety of biological interpretations, because a community of co-expressed genes is often associated with a specific biological function. We propose an original and efficient approach that permits to treat simultaneously the problem of modeling the gene co-expression network and the problem of detecting the communities in network. We put forward the performances of our approach by comparing it to the existing methods that are popular for analysing gene co-expression networks (WGCNA and spectral approaches).

The last part presents the results produced by applying our proposed approach on a real-world data set. We obtain convincing and robust results that help us make more diverse biological interpretations than with results produced by common supervised learning methods.