Analyses de données omiques : clustering et inférence de réseaux

Audrey Hulot

Analyse de données multi-omiques : clustering et inférence de réseaux

Thèse de doctorat de l'université Paris-Saclay

École doctorale n° 577, Structure et dynamique des systèmes vivants (SDSV)
Spécialité de doctorat : Sciences de la vie et de la santé
Unité de recherche : Université Paris-Saclay, INRAE, AgroParisTech, GABI, 78350, Jouy-en-Josas, France
Référent : Université de Versailles -Saint-Quentin-en-Yvelines

Thèse présentée et soutenue en visioconférence totale, le 26/11/2020, par

Audrey HULOT

Composition du jury :

Marie-Laure Martin-Magniette Directrice de recherches, INRAE, Institut des Sciences des Plantes Paris-Saclay	Présidente
Grégory Nuel Directeur de Recherches, CNRS - Sorbonne Université	Rapporteur
Nathalie Vialaneix Directrice de recherches, MIAT, INRAE Toulouse	Rapportrice
Guillaume Assié Professeur, AP-HP Hôpital Cochin	Examinateur
Guillemette Marot Maître de conférence, Université de Lille	Examinatrice
Henri-Jean Garchon Professeur, INSERM-UVSQ	Directeur
Florence Jaffrézic Directrice de recherches, INRAE Jouy-en-Josas	Co-Directrice
Julien Chiquet Directeur de recherches, Université Paris-Saclay, AgroParisTech INRAE	Encadrant

Thèse de doctorat
NNT : 2020UPASL034

Table des matières

Communications et papiers

Articles publiés dans une revue à comité de lecture

Hulot, A., Chiquet, J., Jaffrézic, F. et Rigaill, G. Fast tree aggregation for consensus hierarchical clustering. BMC Bioinformatics 21, 120 (2020). https://doi.org/10.1186/s12859-020-3453-6

Hulot A., Laloë D., Chiquet J., Jaffrézic F., A unified framework for the integration of multiple hierarchical clusterings or networks from multi-source data, *soumis à PLOS One (Juillet 2020), en cours de révision (majeures)*

Charlotte Dupont, Audrey Hulot, Florence Jaffrezic, Céline Faure, Sébastien Czernichow, et al.. Female ponderal index at birth and idiopathic infertility.. Journal of Developmental Origins of Health and Disease, Cambridge University Press, 2019, pp.1-5. (10.1017/S2040174419000394). (hal-02197023)

Communications orales

Audrey Hulot, Julien Chiquet, Florence Jaffrezic, Guillem Rigaill. Fused-ANOVA : une méthode de clustering en grande dimension. *50èmes Journées de Statistique*, 2018, Palaiseau, France. (hal-02483532)

Audrey Hulot, Julien Chiquet, Florence Jaffrezic, Guillem Rigaill. Fast tree aggregation for consensus hierarchical clustering : application to multi-omics data analysis. *Statistical Methods for Post-Genomic Data (SMPGD)*, 2019, Barcelona, Spain. (hal-02483674)

Audrey Hulot, Julien Chiquet, Florence Jaffrezic, Guillem Rigaill. Fast tree aggregation for consensus hierarchical clustering : application to multi-omics data analysis. *22ème Séminaire des Doctorants du Département de Génétique Animale*, 2019, Montauban, France

Audrey Hulot, Julien Chiquet, Florence Jaffrezic, Guillem Rigaill. Fast tree aggregation for consensus hierarchical clustering : application to multi-omics data analysis. *Netbio*, 2019, Palaiseau, France

Poster

Audrey Hulot, Henri-Jean Garchon, Florence Jaffrézic et Julien Chiquet, Omics Data Analysis : Clustering and Network Inference, *21ème Séminaire des Doctorants du Département de Génétique Animale*, 2018, Les Mureaux, France

Chapitre 1

Introduction

Table des matières

1.1 Contexte général : l'analyse de données -omiques

Les -omiques. On désigne par -omiques l'ensemble des données qui proviennent d'un des domaines de recherche en rapport avec la génomique. On peut notamment citer :

— la génomique, étude sur tout le génome, recherche de variants génétiques. Les données produites peuvent par exemple prendre la forme de modalités, ou de comptages avec le calcul de la fréquence de l'allèle mineur.

— la transcriptomique, étude de la transcription, expression des transcrits et des gènes, données obtenues par puces à ADN (données continues) ou séquençage haut-débit (données de comptages) par exemple ;

— l'épigénomique, qui est l'étude des changements du génome non imputables à des mutations. La méthylation est le mécanisme le plus commun de l'épigénétique. Les données de méthylation sont exprimées sous forme de pourcentages ;

— la protéomique étudie l'expression des protéines (jeux de données en général plus réduits). Les données sont des données continues obtenues par puce ou spectrométrie de masse ;

— la métabolomique, expression des métabolites, obtenues sous forme de données continues souvent par spectrométrie de masse ;

— la métagénomique, étude de tout le matériel génétique identifié dans un même milieu (sol, intestins, ...). Données sous formes d'abondances, elles peuvent être de grande dimension

lorsqu'on séquence l'ensemble des gènes, auquel cas on parle de données *whole genome*. Les données dites 16S permettent d'obtenir des jeux de données plus réduits.

Le phénotype se définit quant à lui comme l'ensemble des caractères observables d'un organisme.

L'analyse d'une couche -omique. L'existence de l'ADN a été pour la première fois identifiée en 1869, et les gènes ont été définis depuis 1909. L'analyse des données -omiques remonte donc au siècle dernier. Cependant, ce domaine de recherche s'est beaucoup développé ces dernières années, notamment grâce à l'arrivée des technologies de séquençage haut-débit qui permettent l'obtention d'un volume conséquent de données : des milliers voire millions de variables sont disponibles, et la baisse des prix associée à ces technologie offre la possibilité de séquencer plus d'échantillons.

Lier des données -omiques à un phénotype s'est imposé comme crucial ces dernières années, notamment dans la recherche médicale, pour pouvoir identifier des biomarqueurs associés à une maladie, arriver à repérer des individus plus susceptibles de développer une certaine maladie, et pouvoir poser des diagnostics précoces. La recherche médicale n'est pas le seul domaine où les données -omiques sont importantes : la sélection génomique est très employée par exemple chez les animaux destinés à la consommation, ou encore chez les plantes pour identifier les caractères de résistances, etc.

L'étape d'identification de biomarqueurs se fait souvent par analyse différentielle : deux conditions, ou plus, sont comparées dans le but de trouver quelles variables diffèrent et permettent de discriminer les groupes. La sélection de variables en vue de l'explication des conditions est aussi développée.

Les technologies NGS (Next Generation Sequencing) ont permis de raffiner cette recherche, en mettant à disposition plus de biomarqueurs possibles, et en permettant une connaissance plus fine des caractères étudiés. Ceci n'est pas venu sans son lot de problèmes, puisque les méthodes jusque-là disponibles et utilisées pour de petits jeux de données se sont vues dépassées. Les analystes des jeux de données -omiques sont souvent confrontés à des données de grande dimension, où le nombre de variables est largement supérieur au nombre d'individus. De plus, les données -omiques sont connues pour être très bruitées, et très corrélées, ce qui peut perturber certaines méthodes, comme par exemple la méthode Lasso, (Tibshirani, 1996), connue pour être peu robuste lorsque les variables sont corrélées (Zhao and Yu, 2006).

Un défi apporté par le développement de nouvelles technologies est l'adaptation des méthodes aux nouveaux types de données produites. Dans le cas de l'analyse transcriptomique, les données produites par des puces à ADN sont des données continues et gaussiennes, alors que les données produites par RNA-Seq sont des données de comptages. Les modèles adaptés pour les données gaussiennes étant non-utilisables sur les données de comptages. De nouveaux modèles et méthodes associées ont dû être mis en place pour parfaire l'étude des données de RNA-seq.

Intégration de données - Analyses multi-omiques. L'intégration de données n'est pas une problématique propre à l'analyse de données -omiques : tous les domaines de recherche confrontés à des données provenant de divers sources communicantes cherchent à analyser conjointement ces données. On qualifie ces données d'"hétérogènes", les données étant de natures différentes, émises par des lois différentes, de différentes dimensions et paramètres, voire sous forme de graphes ou arbres. L'intégration des données désigne tout le processus d'analyse d'un ensemble de données, parfois hétérogènes, étudiant les connexions entre les jeux de données aussi bien que les particularités de chacun de ces jeux de données.

Dans le cadre des données -omiques, les différentes composantes -omiques et les entités étudiées à l'intérieur de chacun de ces domaines sont connues pour interagir ensemble, autant à l'intérieur d'une

couche (régulation génique à l'intérieur de la couche de transcriptomique) qu'entre les différentes couches (transcriptomique → protéomique).

L'analyse d'une seule table de données peut permettre, typiquement par analyse différentielle, d'identifier des entités impliquées dans le processus biologique étudié. Un processus biologique mobilisera souvent plusieurs couches d'-omiques avec parfois des répercussions sur le phénotype (par exemple avec développement d'une maladie). C'est toute une suite d'interactions entre les diverses couches, voire avec l'environnement, qui provoque son apparition.

Le dérèglement n'est parfois observable qu'après certaines interventions. L'analyse d'une simple table de données est une étape intéressante mais ses résultats sont limités. On peut envisager d'analyser chacune des tables séparément et de réunir ensuite les différents résultats, mais analyser *a posteriori* les différentes entités n'a pas la même portée que de directement sélectionner ces entités conditionnellement à l'action des autres

L'analyse conjointe des données permet non seulement d'identifier les diverses entités impliquées à l'intérieur d'une couche, mais aussi de mettre en relation ces entités, et de comprendre la propagation d'un dérèglement sur les différentes couches.

Les problèmes rencontrés lors de l'analyse d'une seule table, à savoir les données bruitées, les variables très corrélées et la grande dimension, sont encore plus présents et importants à prendre en compte lorsque l'on cherche à mettre en relation plusieurs tables de données.

L'intégration de données ne vise pas seulement à sélectionner des variables, mais aussi à chercher à comprendre les mécanismes sous-jacents. Pour cela, les méthodes de clustering (pour co-expression, ou classement des individus, en première intention, par exemple) et les méthodes d'inférence de réseaux (réseaux de régulation, corrélations) sont appréciées. Ce sont des méthodes qui permettent d'obtenir des représentations visuelles très interprétables, et souvent employées de manière exploratoires, le clustering permettant de trouver des structures de groupes dans les données, et un réseau permettant d'observer les divers liens qu'il existe entre les entités. L'objectif de cette thèse est de développer des méthodes de clustering et d'inférence de réseaux, en prenant en compte les aspects multi-tables, données hétérogènes et de grandes dimensions rencontrés lors de l'analyse des données multi-omiques.

1.2 Contexte statistique

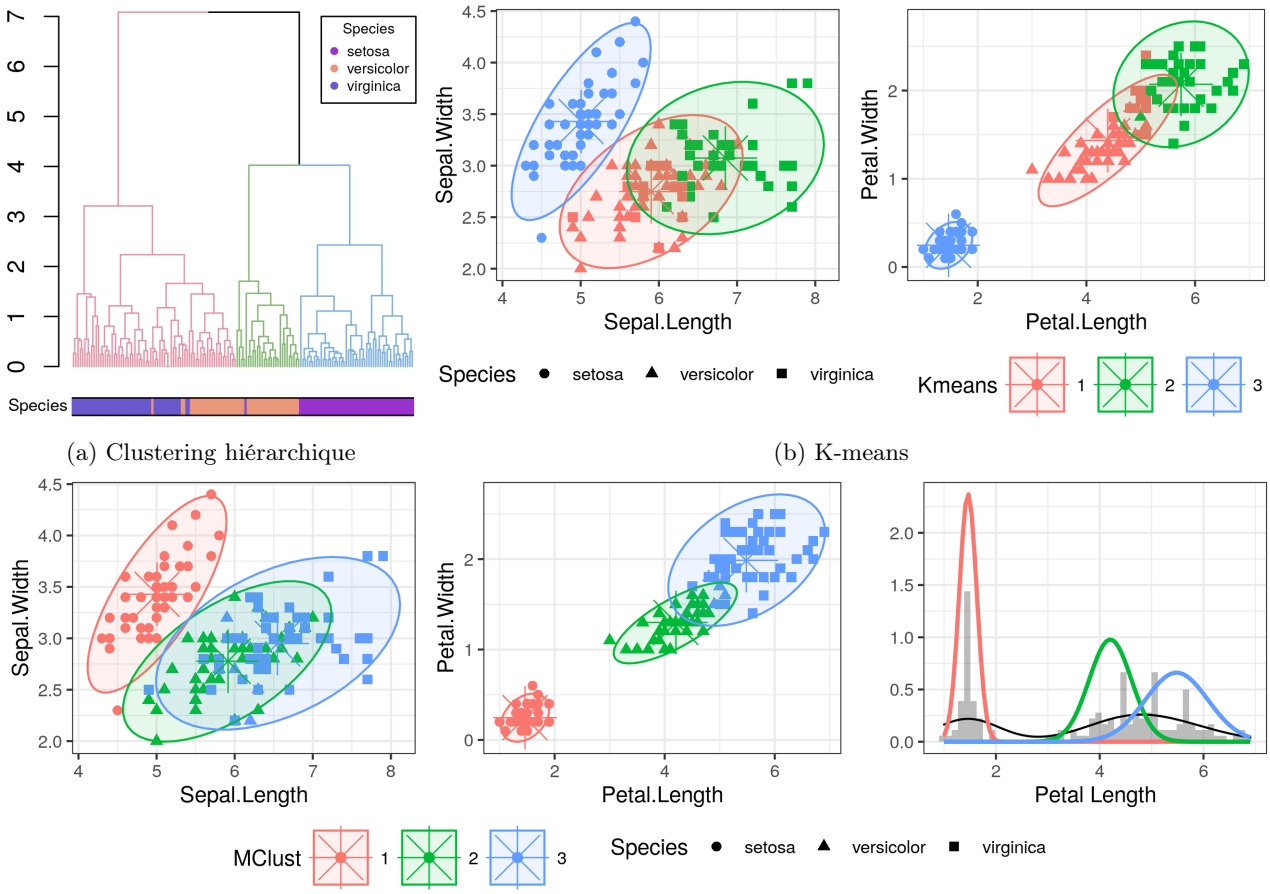
Dans cette partie sont abordées de façon très générale les méthodes de clustering et d'inférence de réseaux.

1.2.1 Le Clustering

Le clustering, ou classification non supervisée, désigne le regroupement des individus d'une population en groupes homogènes, selon un critère, l'un des plus utilisés étant la maximisation des distances inter-clusters. Cette partition se fait sur la base de données récoltées sur tout ou partie des individus considérés.

La classification permet d'identifier des groupes d'individus présentant des caractéristiques communes, ou proches, au vu des données utilisées pour le réaliser, et permet d'étudier la particularité des groupes formés, voire d'étudier la différence entre les groupes.

On distingue plusieurs types de classifications. La première distinction se fait entre classification non supervisée (clustering) et classification supervisée. Dans la deuxième famille de méthodes, une ou plusieurs variables sont utilisées comme *a priori* pour former les groupes. On s'intéresse ici plus



(a) Clustering hiérarchique

(b) K-means

(c) Modèle de mélanges

FIGURE 1.1 – Illustration des méthodes de clustering sur jeu de données des iris (Anderson, 1935) **(a)** Clustering hiérarchique, distance euclidienne, critère d'agrégation de Ward. Coupure de l'arbre à 3 groupes. **(b)** Partition des données obtenue par K-means. Nombre de départs : 100, iter max = 1000, centers = 3. Les ellipses ont été tracées avec un niveau à 95% pour chacun des groupes trouvés par les K-means. **(c)** Partition des données obtenue par modèle de mélange, en utilisant `mclust` et en cherchant 3 groupes et illustration du mélange de lois obtenu pour la variable Petal Length. Les ellipses ont été construires de la même manière que celles des K-means.

particulièrement à la classification non supervisée, dont les méthodes sont souvent employées dans des analyses exploratoires pour observer la répartition des individus.

Le clustering est utilisé dans tous les domaines où des données doivent être analysées. Dans le cadre des données -omiques, la classification est à la fois utilisée pour classer des individus, ou des variables (gènes, protéines, etc.). L'hypothèse est faite que les variables classées ensembles partagent des propriétés similaires. Cela permet notamment d'identifier des groupes fonctionnels, et des voies métaboliques.

1.2.1.1 Les différentes approches de clustering

On considère dans cette partie une table de données Y $(n \times p)$, avec $Y_i = (y_{i1}, ..., y_{ip})$ le vecteur d'observations de l'individu i et K le nombre de groupes supposés dans les données. On suppose dans cette section que l'on veut produire une classification des individus (lignes de la table).

On aborde les méthodes de clustering classiquement utilisées sur des tables de données, que l'on répartit en trois groupes. Les méthodes utilisées pour réaliser un clustering sur un réseau seront mentionnées plus tard.

Modèles de mélanges - *Model-based Clustering.* Ces méthodes font l'hypothèse que les données sont structurées en groupes, et que les lois de génération des données sont différentes d'un groupe à l'autre. Les données peuvent donc être écrites sous la forme d'un modèle de mélange de lois. On suppose que les données son divisées en un nombre fini de clusters $1, ..., K$ et qu'il existe, pour chaque individu i, une variable latente Z_i donnant son appartenance à un cluster. La distribution des données observée Y_i dépend de la valeur de cette variable latente (du cluster auquel appartient i).

Dans ces modèles, on suppose que les variables Z_i sont iid. La probabilité *a priori* pour l'individu i d'appartenir au cluster k est donnée par $\mathbb{P}(Z_i = k) = \pi_k$, proportions des individus dans les groupes. On suppose aussi que les observations Y_i sont indépendantes conditionnellement à Z_i. La loi des observations est conditionnelle au groupe d'appartenance : $Y_i|(Z_i = k) \sim f_k(\gamma_k)$. On a alors :

$$f(y_i; K, \theta_K) = \sum_k \pi_k f_k(y_i; \gamma_k) \tag{1.1}$$

où $\theta_K = (\pi_1, ..., \pi_K, \gamma_1, ..., \gamma_K)$ est le vecteur des paramètres des lois. On cherche à estimer les éléments de θ_K.

L'estimation des paramètres du modèle de mélange se fait à l'aide d'un algorithme *Expectation - Maximization* (EM) (Dempster et al., 1977). L'avantage d'utiliser ce type de méthodes réside dans l'existence de critères de sélection, pour sélectionner un nombre de groupes, et les paramètres associés, via notamment le *Bayesian Information Criterion* (BIC) (Schwarz et al., 1978; Fraley and Raftery, 1998) ou l'*Integrated Completed Likelihood* (ICL) (Biernacki et al., 2000). Ils permettent aussi de prendre en compte plusieurs types de lois : gaussienne, Poisson, binomiale, voire même des mélanges de ces lois.

Le désavantage majeur de ces méthodes est leur lenteur : l'étape d'estimation des paramètres peut prendre du temps. Les algorithmes EM peuvent tomber dans des minimums ou maximums locaux, ce qui demande de les lancer plusieurs fois avec des initialisations différentes.

Bien que les modèles de mélanges soient usuellement utilisés pour produire une partition des données, il existe des algorithmes qui permettent l'obtention d'un arbre, ou de plusieurs classifications imbriquées (Fraley, 1998).

Partition des données. Une deuxième catégorie contient les méthodes de partitionnement du type k-means (MacQueen et al., 1967), k-medians ou k-médoids (PAM) (Kaufman and Rousseeuw, 1987), qui reposent sur l'utilisation d'un algorithme itératif pour créer une partition des données en K groupes demandés par l'utilisateur. Les k-means sont une des méthodes de partitions les plus utilisées.

L'initialisation se fait aléatoirement, pour créer K groupes de données, et les centres (moyennes des observations dans cette classe) G_k, $k = 1, \ldots, K$, associés aux clusters $C_1, \ldots, C_K$. A chaque étape de l'algorithme, et jusqu'à convergence, les individus sont assignés à la classe dont le centre de gravité est le plus proche. Les centres G_k sont ensuite recalculés. La méthode k-means++ (Arthur and Vassilvitskii, 2006) propose une modification des k-means permettant de fixer les centres des $k-1$ classes de manière non aléatoire après que C_1 ait été fixé aléatoirement.

L'objectif des k-means est de trouver la classification qui minimise l'inertie intra-classe, soit qui cherche à résoudre le problème d'optimisation suivant :

$$\underset{C}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i \in C_k} \|y_i - G_k\|^2 \tag{1.2}$$

La méthode des k-means est une méthode rapide et appréciée pour le fait qu'elle puisse s'appliquer sur des grands jeux de données, la complexité étant de $\mathcal{O}(nKpj)$ pour l'algorithme de Lloyd (Lloyd, 1982), où j est le nombre d'itérations demandé. Cependant, la classification des données obtenue peut grandement dépendre du choix de l'initialisation, ce qui contraint à lancer plusieurs fois la méthode avec des initialisations différentes pour pouvoir obtenir une classification stable. Elle est de plus sensible aux valeurs atypiques.

Le fait que l'utilisateur doive connaître à l'avance le nombre de groupes est aussi un désavantage. Plusieurs partitions avec des nombres de groupes différents peuvent être demandées, et la partition retenue sera choisie selon un critère (par exemple en utilisant la méthode de la silhouette (Rousseeuw, 1987) ou la statistique de gap (Tibshirani et al., 2001)). Là encore, cette démarche demande de lancer plusieurs fois l'algorithme.

Des méthodes voisines sont utilisées pour réduire la sensibilité aux outliers : l'algorithme k-medians prend les médianes des différents clusters comme centres de gravité, et les k-medoids utilisent des points réels, et ne sont pas réduits à l'utilisation d'une distance euclidienne. Ces deux variantes ont une complexité supérieure à celle des k-means, qui ne permet pas leur utilisation sur des grands jeux de données.

Clustering hiérarchique Le clustering hiérarchique désigne l'ensemble des méthodes de clustering qui renvoient un arbre. L'arbre retrace les regroupements (ou les divisions) des individus à chaque étape. Le bas de l'arbre désigne la situation où chaque individu constitue son propre groupe ($K = n$), le haut de l'arbre correspond à la situation où tous les individus sont dans le même groupe ($K = 1$).

La construction peut se faire de façon divisive (de haut en bas de l'arbre) ou agglomérative (de bas en haut). Dans le cadre du clustering hiérarchique dit classique, la construction de cet arbre se fait à partir d'une matrice de distance et d'un critère d'agrégation, qui permet de calculer une distance entre les clusters, et de manière agglomérative : à chaque étape, les deux clusters les plus proches sont regroupés, jusqu'à ce que tous les individus soient dans le même groupe.

Un des grands avantages de cette méthode est le fait qu'un arbre peut être obtenu quel que soit le type de données, tant que la distance utilisée est appropriée (euclidienne pour des données continues, bray-curtis pour des abondances, etc.). Du fait du calcul de toutes les distances pour toutes les paires d'individus, l'arbre obtenu permet en général de bien différencier les outliers du reste de la population.

Le clustering hiérarchique est de plus grandement apprécié pour sa représentation visuelle, dans laquelle on peut observer le schéma de regroupement ou division des individus. En coupant l'arbre à différents niveaux, on obtient des classifications de tailles différentes, mais qui peuvent être mises en relation les unes aux autres.

La structure de l'arbre obtenu dépend grandement du critère d'agrégation utilisé, même si la distance reste la même. Les critères de *single linkage* et *complete linkage* sont souvent utilisés, ainsi que le critère de Ward lorsque la distance est euclidienne. On note d la distance utilisée, C_1 et C_2 deux clusters dans la partition considérée et G_1 et G_2 leurs centres de gravité respectifs.

$$d(C_1, C_2) = \min_{y_1 \in C_1, y_2 \in C_2} d(y_1, y_2) \tag{Single-linkage}$$

$$d(C_1, C_2) = \max_{y_1 \in C_1, y_2 \in C_2} d(y_1, y_2) \tag{Complete-linkage}$$

$$d^2(C_1, C_2) = \frac{|C_1||C_2|}{|C_1| + |C_2|} d^2(G_1, G_2) \tag{Critère de Ward}$$

Le désavantage principal de cette méthode est la mémoire requise pour effectuer tous ces calculs, les algorithmes utilisés pour construire l'arbre ayant au moins une complexité quadratique $\mathcal{O}(n^2)$. Le deuxième désavantage est le choix du nombre de groupes pour fixer une classification. Les méthodes de la silhouette ou l'utilisation de la statistique de gap, déjà évoquées pour les k-means, peuvent être ici aussi utilisées. Enfin, les clusterings hiérarchiques sont peu robustes face à des perturbations des données.

1.2.1.2 Le clustering dans les données -omiques

Les méthodes de clustering utilisées pour analyser une unique table de données -omiques ne diffèrent pas de celles décrites dans la partie précédente, tant que la dimension des données le permet. L'analyse de données multi-omiques requiert par contre de nouvelles méthodes, adaptées aux données hétérogènes de grande dimension. La concaténation de toutes les tables de données n'est parfois pas appropriée, puisqu'on perd la particularité de chacune des tables. Cette partie détaille les limites rencontrées dans l'utilisation du clustering pour les données -omiques, et les méthodes développées pour analyser conjointement plusieurs tables de données.

Les limites rencontrées. Les limites rencontrées dans le clustering des données -omiques sont les mêmes que pour tous jeux de données de grande dimension hétérogènes, où le nombre de variables (colonnes) est très supérieur au nombre de lignes et les types de données peuvent être différents entre les tables. Dans le cas des données -omiques, cela se rencontre par exemple en transcriptomique (nombre de gènes très supérieur au nombre d'individus, données continues ou de comptage) ou en métagénomique (données de comptages ou abondances).

Sur une table seule, le nombre de variables dépasse souvent la dizaine de milliers. Lorsque l'on cherche à réaliser un clustering sur les variables (gènes), toutes les approches de clusterings basées sur l'obtention d'une matrice de distance sont désavantagées, voire même impossible à appliquer dans certaines situations. Les méthodes de *model-based clustering* sont elles aussi rapidement dépassées. Les k-means sont toujours applicables, mais le fait qu'ils doivent être relancés plusieurs fois, avec plusieurs initialisations et plusieurs nombres de groupes est un inconvénient dans ce type de situations. De plus, les données -omiques sont connues pour être très bruitées : les k-means ne sont pas une solution assez stable pour ce genre de données. Compte tenu de la quantité de bruit présente dans les données -omiques, une étape de sélection, via une analyse différentielle par exemple, est la première étape effectuée.

En ce qui concerne le clustering des individus, il est souvent possible de le réaliser si on considère une seule table de données, pouvant être le produit de la concaténation des tables à disposition. Le problème se complexifie lorsqu'on s'intéresse à plusieurs tables de données en même temps, en cherchant à prendre en compte les caractéristiques des individus et des variables provenant de toutes les tables. Les modèles doivent prendre en compte la potentielle hétérogénéité des données (comptages, continues, etc.) et leurs liens.

Les méthodes de clustering multi-omiques. Il existe un certain nombre de méthodes qui ont été développées ces dernières années dans le cadre des données -omiques, bien qu'elles soient pour la plupart adaptées à tous jeux de données hétérogènes. La plupart des méthodes sont développées dans le cadre du clustering d'individus. Peu de méthodes ont été développées dans le sens de partitionner les variables de tables différentes ensemble (Hidalgo and Ma, 2018). On s'appuie dans ce paragraphe sur les reviews suivantes : Rappoport and Shamir (2018), Tini et al. (2019) et Pierre-Jean et al. (2019). Une classification possible des méthodes d'analyse de données hétérogènes, valable pour les

méthodes de clustering mais aussi pour toute méthode d'analyse de données multi-tables, est la suivante :

— Early integration : méthodes qui consistent à travailler sur la concaténation des matrices (icluster+ (Mo et al., 2013), SNF (Wang et al., 2014), ...) ;
— Intermediate integration : transformation des tables effectuée avant concaténation et analyse ;
— Late integration : regroupe les méthodes qui consistent à réaliser une analyse sur chacune des tables avant d'en agréger les résultats (COCA (Hoadley et al., 2014), Bayesian consensus clustering (Lock and Dunson, 2013), ...) ;

Certaines méthodes sont adaptées spécifiquement aux données -omiques, plutôt que d'être applicables à tous jeux de données hétérogènes. Ces méthodes prennent notamment en compte les annotations fonctionnelles et les sens de régulations entre les différentes couches d'-omiques (PARADIGM (Vaske et al., 2010)).

Ces méthodes souffrent parfois d'un temps de calcul très long (Pierre-Jean et al., 2019), qui ne permet pas de les appliquer aux données -omiques sans effectuer un filtre préalable sur les variables que l'on prend en compte. Les méthodes de réduction de dimensions (PCA, SVD) sont aussi utilisées pour permettre de réduire la dimension des tables et pouvoir appliquer des méthodes plus classiques de clustering.

Bien qu'un certain nombre de méthodes existent pour réaliser du clustering sur des données multi-omiques, peu permettent d'obtenir un arbre (LRACluster (Wu et al., 2015)). Ce problème est mentionné dans le Chapitre 2.

1.2.2 L'inférence de réseaux

Un réseau est un objet mathématique utilisé pour représenter des relations entre des entités. L'inférence de réseaux désigne l'ensemble des méthodes permettant de créer ce réseau, à partir d'un ensemble de données, généralement sous la forme d'une table de données X, $n \times p$. Un réseau $\mathcal{G}$ est défini comme un ensemble de sommets $\mathcal{V} = \{1, ..., p\}$ (*vertices* en anglais), ou nœuds, et d'arêtes, aussi appelées arcs, $\mathcal{E} \in \mathcal{V} \times \mathcal{V}$ (*edges* en anglais) qui relient ces sommets.

Les réseaux sont un outil de visualisation apprécié et utilisé dans de nombreux domaines (écologie, biologie, sociologie, ...) pour étudier les liens et interactions entre différentes entités. La quantité d'information contenue dans un réseau peut être très importante : la présence ou absence d'arc, la largeur des arcs, la dimension des nœuds, la couleur des nœuds, la forme des nœuds... chaque paramètre du réseau peut être adapté pour représenter une information différente, superposée sur une même image.

En plus de servir pour visualiser des données et des processus complexes, ils peuvent aussi être utilisés pour des analyses supplémentaires, notamment pour du clustering, en regroupant les entités les plus connectées entre elles.

En biologie, les nœuds du réseau représentent souvent les -omiques que l'on étudie. Les réseaux construits à partir de données -omiques sont répandus, couramment utilisés, et exploités. On peut citer notamment les réseaux d'interaction protéines-protéines, les réseaux de régulation génique ou les réseaux métaboliques.

Un graphe peut être représenté sous forme de matrice,

— par sa matrice d'adjacence, A, représentation binaire de $\mathcal{G}$: $A_{ij} = 0$ s'il n'existe pas d'arête entre les deux sommets i et j, et $A_{ij} = 1$ sinon ;
— par W, matrice de poids, $W_{ij} = 0$ s'il n'existe pas d'arête entre les deux sommets i et j, et $W_{ij} \neq 0$ sinon. Lorsque les poids peuvent être de signes différents, on parle de réseaux signés.

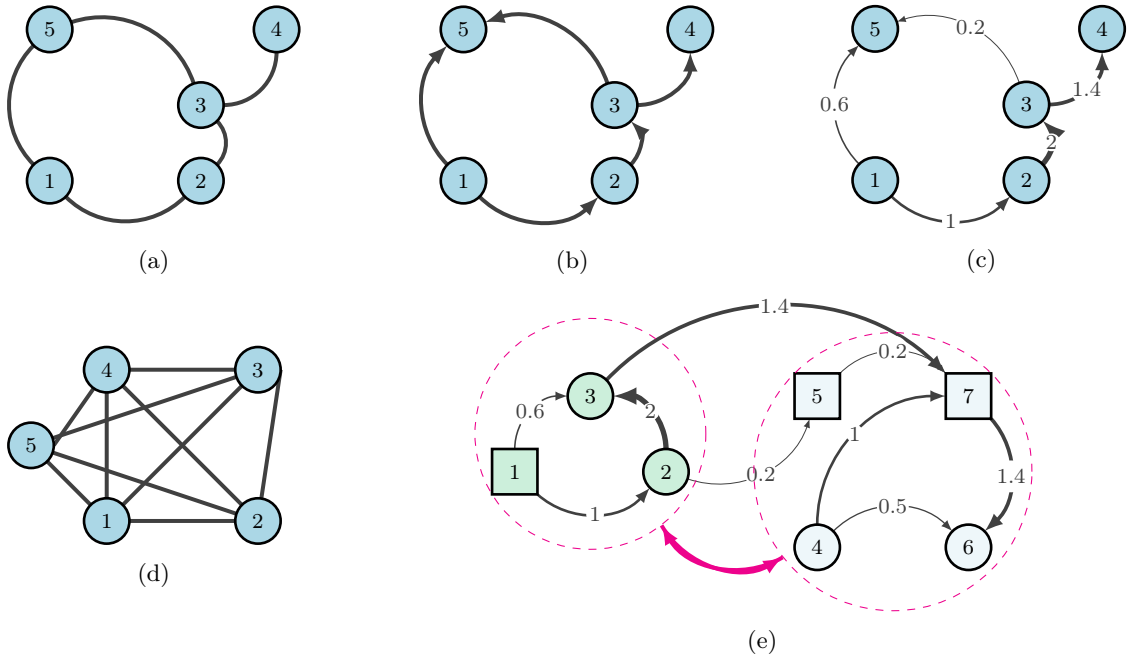


FIGURE 1.2 – Exemples de réseaux. **(a)** Réseau binaire, cyclique, non dirigé. **(b)** Réseau binaire, acyclique, dirigé. Ordres topologiques : 12345 ou 12354. **(c)** Réseau pondéré, acyclique, dirigé. **(d)** Réseau binaire, complet, non-dirigé. **(e)** Un réseau plus complexe.

On considère dans les chapitres de la thèse abordant les réseaux que $A_{ii} = W_{ii} = 0$ pour tout i, donc qui ne comportent aucune boucle.

Les réseaux peuvent être de plusieurs types, qui se traduisent par des propriétés différentes sur les matrices A et W. La Figure 1.2 page 9 donne quelques exemples. La plus grande distinction peut se faire entre les réseaux non dirigés, dans lesquels les matrices de poids et d'adjacence sont symétriques, et les réseaux dirigés, dans lesquels ces matrices ne sont plus obligatoirement symétriques, et chaque arc est caractérisé par un sens, avec un nœud de départ et un nœud d'arrivée.

La section suivante présente quelques méthodes classiquement utilisées pour inférer des réseaux, non dirigés puis dirigés. Puis, on abordera les méthodes employées pour l'analyse de données -omiques, ainsi que leurs limites.

1.2.2.1 Méthodes d'inférence de réseaux

Les arcs représentent une relation entre deux nœuds. Cette relation possède différentes interprétations selon la manière dont elle a été obtenue et le modèle statistique qu'il y a derrière. Les méthodes utilisées diffèrent selon que l'on veuille obtenir un réseau dirigé, un réseau pondéré, etc.

Réseaux dirigés. Les réseaux dirigés apportent une quantité d'information plus importante que les réseaux non dirigés, les arcs ayant une caractéristique supplémentaire : un sens d'action.

Les réseaux bayésiens sont particulièrement étudiés et développés, pour l'accès à des modèles mathématiques simplifiés, et du fait que les arcs estimés représentent l'action directe d'un nœud sur ses descendants. Ces réseaux sont supposés des DAG (*Directed Acyclic Graph*), impliquant que les nœuds sont ordonnés. Cet ordre est appelé ordre topologique. Lorsque la matrice d'adjacence du réseau est ordonnée selon un ordre topologique, elle est triangulaire supérieure.

Les réseaux bayésiens satisfont la propriété de Markov : chacun des nœuds est conditionnellement indépendant à tous les nœuds auquel il n'est pas directement relié, sachant ses parents. En utilisant

l'ordre des nœuds, cette propriété s'exprime par :

$$\mathbb{P}(X_i|X_1, ..., X_{i-1}) = \mathbb{P}(X_i|pa(X_i)) \tag{1.3}$$

en notant $pa(X_i)$ l'ensemble des parents du nœud X_i. La loi jointe du graphe, donnée par la formule :

$$\mathbb{P}(X) = \prod_i \mathbb{P}(X_i|X_1, ..., X_{i-1}). \tag{1.4}$$

peut s'exprimer sous la forme factorisée :

$$\mathbb{P}(X) = \prod_i \mathbb{P}(X_i|pa(X_i)). \tag{1.5}$$

Pour estimer entièrement ces réseaux, les données demandées sont plus importantes, et différentes, que celles qui sont demandées pour obtenir un réseau non dirigé. En effet, pour pouvoir obtenir tous les arcs et leurs directions, il est nécessaire de pouvoir quantifier l'action de cette entité sur les autres. Des données temporelles ou interventionnelles (knock-down ou knock-out de gènes, par exemple) sont nécessaires. Les réseaux bayésiens peuvent notamment servir, sous la forme de réseaux bayésiens dynamiques, à représenter les interactions des entités au cours du temps.

Dans cette thèse, nous nous sommes concentrés sur l'inférence de réseaux non dirigés.

Réseaux non dirigés. Les réseaux non dirigés sont plus facilement accessibles que les réseaux dirigés. Une simple matrice de dissimilarité entre les entités peut être vue comme un réseau. Cette méthode peut notamment être appliquée sans transformation à des données non continues, en utilisant par exemple les corrélations de Spearman.

En biologie, les réseaux de corrélation sont les réseaux les plus accessibles. La matrice de corrélation P obtenue sur une table de données est une matrice symétrique et non dirigée, qui peut être facilement calculée. Les corrélations les plus faibles sont éliminées via l'usage d'un seuil ou d'un test de significativité de la corrélation.

La corrélation apporte un certain nombre d'informations, et permet d'obtenir un réseau présentant des associations entre les nœuds. Ces réseaux présentent aussi l'avantage d'être signés et pondérés. Pour obtenir la relation propre entre deux entités, sans effets des autres variables du réseau, les corrélations partielles, soit la corrélation entre deux entités sans l'interaction des autres entités du réseau, peuvent être utilisées.

Si les données sont gaussiennes et P, matrice des corrélations entre les nœuds, est inversible, la corrélation partielle peut être accessible. Les coefficients de P^{-1} sont en effet reliés aux coefficients de corrélation partielle. En notant ρ_{ij} la corrélation entre deux entités, et ω_{ij} le coefficient i, j de la matrice inverse $\Omega = \mathrm{P}^{-1}$ entre ces deux mêmes entités, on a d'après (Lauritzen, 1996) :

$$\rho_{ij|\mathcal{V}\setminus\{i,j\}} = -\frac{\omega_{ij}}{\sqrt{\omega_{ii}}\sqrt{\omega_{jj}}} \tag{1.6}$$

On peut donc utiliser la matrice inverse des corrélations pour obtenir un réseau plus informatif qu'un réseau de corrélation. Cette matrice reste symétrique. Il convient là-encore d'utiliser un seuil pour enlever les plus faibles coefficients.

Cette relation est utilisée dans les modèles graphiques gaussiens, où les données gaussiennes sont générées par une loi $\mathcal{N}(\mu, \Sigma)$. L'objectif est d'estimer $\Omega = \Sigma^{-1}$, matrice de précision, et matrice du graphe. La méthode la plus utilisée pour estimer cette matrice est le Graphical Lasso (Glasso) (Friedman et al., 2008a; Banerjee et al., 2008a). La matrice renvoyée par la méthode

est l'estimation parcimonieuse de la matrice Ω, et permet d'obtenir directement les corrélations partielles significatives dans le réseau. Cette partie sera plus détaillée dans le Chapitre 3.

1.2.2.2 L'inférence de réseaux dans les données -omiques

Les réseaux construits en biologie peuvent l'être sur la base des liens établis précédemment dans la littérature. On s'intéresse ici uniquement à la construction d'un réseau obtenu à partir d'un ou plusieurs jeux de données concernant l'expression de différentes entités, et aux méthodes permettant de construire un réseau d'interactions entre ces entités. Les méthodes incluant des réseaux destinés aux patients (comme SNF (Wang et al., 2014)) ne seront pas mentionnées.

Dans le cas de l'analyse d'une seule table de données, comme pour le clustering, les techniques d'inférence de réseaux appliquées aux données -omiques ne diffèrent pas des méthodes appliquées à d'autres jeux de données.

On rencontre pour l'inférence de réseaux le même problème qu'avec le clustering multi-omique : doit-on considérer chaque table séparément et trouver les liens entre les différentes entités après avoir construit les réseaux intra-tables ? Ou directement inférer l'ensemble des liens ? La deuxième solution est la plus attractive, mais demande le développement de modèles plus complexes, et surtout de modèles adaptés pour les données hétérogènes (Lee et al., 2020).

On s'appuie dans les paragraphes suivants sur les articles (Hawe et al., 2019) et (Altenbuchinger et al., 2020), qui sont des reviews des méthodes utilisées en multi-omiques pour produire des réseaux non dirigés. Les réseaux basés sur l'information mutuelle (*Mutual Information*, MI) sont parfois employés, mais ne sont pas les plus répandus.

Les réseaux de corrélation, faciles à obtenir, sont très utilisés, via notamment le package WGCNA (*Weighted Correlation Network Analysis*) sous `R` (Langfelder and Horvath, 2008). Ce dernier peut aussi servir à la détection de communautés dans les graphes.

Pour l'estimation d'un réseau sparse de corrélations partielles, la méthode Glasso, pour les données gaussiennes, est très appliquée. Cette méthode sera détaillée dans le chapitre 3. Des modèles adaptés à d'autres types de données que les données gaussiennes, voire mélangeant des types de données, ont été développées : les *Mixed Graphical Models* (MGM (Lauritzen et al., 1989; Lee and Hastie, 2015)), peu utilisés cependant en pratique.

En ce qui concerne les approches bayésiennes, un certain nombre d'algorithmes ont été développés pour estimer une structure à partir des données (Ellis and Wong, 2008; Young et al., 2014; Cho et al., 2016), mais demandent des données parfois peu accessibles en biologie humaine, notamment un grand nombre de points pour estimer au mieux les réseaux (Banf and Rhee, 2017), ou des knock-out de gènes.

Malgré plusieurs méthodes développées, l'intégration de données multi-omiques pour réaliser un réseau reste une question ouverte, du fait de plusieurs limitations.

Limites de l'inférence de réseaux. Le premier aspect limitant dans l'inférence de réseaux est le fait de s'appuyer sur des modèles mathématiques, souvent sans aucun *a priori* biologique. Les réseaux sont estimés seulement sur la base des données, la méthode utilisée peut trouver des arêtes là où il est impossible d'en avoir. Une étape de correction *a posteriori* des arêtes détectées peut être obligatoire, en se basant sur la littérature. Les arêtes inférées n'ont peut-être aucune valeur biologique. Ces limites sont prises en compte par certaines méthodes, qui s'appuient sur des bases de données recensant les voies métaboliques pour avantager la création d'arcs déjà existants dans la littérature (pLasso (Wang et al., 2013)).

Le deuxième aspect est d'ordre computationnel : compte tenu du passage par une matrice d'adjacence ou de poids, la construction d'un réseau est souvent limitée à un nombre de nœuds réduit. On rejoint ici les limites rencontrées pour les méthodes de clustering : les données sont de grande dimension, et souvent hétérogènes. Si certains types de données peuvent être transformés pour se rapprocher d'une loi normale, il serait plus précis que les modèles nouvellement développés prennent en compte les différentes lois. Pour les réseaux dirigés, on a parfois besoin de données d'intervention (knock-out) pour pouvoir estimer correctement l'impact d'une entité sur les autres.

Le problème de la grande dimension se rencontre aussi dans l'estimation : plus le nombre de nœuds considéré est grand, plus le nombre d'individus requis pour l'estimation des arcs est grand. Dans les applications, le nombre de nœuds des réseaux est extrêmement réduit, et le réseau final ne représente qu'une petite fraction de la réalité.

1.3 Contributions de la thèse

Cette thèse se décompose en deux partie : la première partie décrit les contributions méthodologiques apportées sur l'analyse de données -omiques. La deuxième partie concerne un projet appliqué d'intégration de données.

1.3.1 Développement méthodologique

1.3.1.1 Méthode rapide de clustering hiérarchique en grande dimension

Dans ce chapitre, on introduit une méthode permettant de réaliser un clustering hiérarchique sur des millions de données avec une complexité sous-quadratique. Cette méthode s'appuie sur un algorithme d'agrégation d'arbre, pour réaliser un arbre consensus à partir d'un ensemble d'arbres ayant les mêmes feuilles, ainsi que sur une approche utilisant le clustering convexe. L'algorithme d'agrégation d'arbres a fait l'objet d'une publication, qui constitue la première partie du chapitre.

> Hulot, A., Chiquet, J., Jaffrézic, F. et Rigaill, G. Fast tree aggregation for consensus hierarchical clustering. BMC Bioinformatics 21, 120 (2020).
> https://doi.org/10.1186/s12859-020-3453-6

Le reste de la méthode de *convex clustering* est détaillée, et a été appliqué à un jeu de données simulé pour démontrer ses performances. Un axe d'amélioration par une approche spectrale est aussi développée.

La méthode démontre un intérêt certain au vu de sa rapidité, et l'amélioration spectrale permet d'envisager son application dans une situation réelle.

1.3.1.2 Estimation d'un réseau multipartite avec modèle stochastique par blocs

Ce chapitre présente la base d'un article de recherche, effectué en collaboration avec Sophie Donnet (AgroParisTech MIA).

Dans ce chapitre, nous proposons une méthode d'estimation de réseau, en se basant dans le cadre d'une partition des nœuds en groupes fonctionnels (GF) connue *a priori*. On se place dans le cadre des modèles graphiques gaussiens. Cette partition *a priori* permet un découpage du réseau en blocs de relations intra-GF (symétriques) et inter-GF (non-symétriques).

Basé sur un processus itératif, la méthode procède en deux étapes :

1. Estimation de la matrice de précision des données grâce à un Graphical Lasso (Glasso), avec matrice de pénalité Λ adaptée ;
2. Estimation des groupes des nœuds, grâce à une utilisation des méthodes de Latent Block Model (LBM) et de Stochastic Block Model (SBM), qui respectent la classification des GF. Grâce aux probabilités de connexions estimées, la matrice Λ est adaptée arête par arête pour pénaliser plus fortement les relations inter-blocs.

Les deux étapes sont répétées jusqu'à convergence. L'introduction d'une partition *a priori* des données nous permet de respecter une classification déjà documentée dans la littérature, cas notamment rencontré dans les données -omiques, et d'en trouver une partition plus fine.

La méthode est appliquée à un jeu de données simulées, ainsi qu'à un jeu de données réelles multi-omiques. On compare ses performances avec celles obtenues par un Glasso simple, ainsi qu'une combinaison similaire Glasso - SBM qui n'introduit pas de partition *a priori*.

1.3.1.3 Intégration de données via l'analyse factorielle multiple.

Ce chapitre est composé du texte d'un article soumis.

> Hulot A., Laloë D., Chiquet J., Jaffrézic F., A unified framework for the integration of multiple hierarchical clusterings or networks from multi-source data
> *soumis à PLOS One (Juillet 2020), en cours de révision (majeures)*

On propose ici d'intégrer tous types de données pouvant se résumer sous la forme d'une matrice de dissimilarité, par une procédure en deux étapes :

1. **Projection.**
   (a) Représenter toutes les données sous la forme de matrices de dissimilarités
   (b) Projeter ces distances dans un espace commun de coordonnées
2. **Intégration.**
   (a) Appliquer une méthode d'analyse multi-tables
   (b) Utiliser une représentation factorielle pour comparer les projections et créer un consensus (si l'objectif est de créer un consensus)

Des données continues ou qualitatives sous forme de tables peuvent être intégrées à l'analyse par l'étape 2. On utilise le *Multidimensional Scaling* (MDS) pour la première étape, et l'*Analyse Factorielle Multiple* (AFM) comme analyse multi-tables pour la deuxième étape. L'accent est mis sur l'intégration de données provenant d'arbres, via la distance cophénétique, et de réseaux, via la distance shortest path.

La combinaison des deux étapes a été appliquée à des simulations, ainsi qu'à deux jeux de données réelles.

1.3.2 Analyse de données -omiques pour l'étude du développement de la Spondyloarthrite Ankylosante (Multi-Spa)

Cette partie concerne un projet d'analyse et d'intégration de données -omiques en vue de mieux comprendre le développement de la Spondyloarthrite Ankylosante (SpA), maladie inflammatoire. L'apparition de cette maladie est en très grande partie expliquée par la présence de l'allèle B27 chez les patients, cependant, certaines personnes porteuses de cet allèle ne développeront jamais la maladie, et certains patients sont B27 négatifs. On cherche ici à comprendre ce qui peut expliquer

l'apparition ou non apparition de cette maladie en dehors de, ou en interaction avec, ce facteur déjà connu. Nous présentons deux analyses :

1. Une analyse différentielle des données de transcriptome, menant à la découverte d'un certain nombre de gènes d'intérêts ;
2. Une analyse conjointe de données de métagénome et de transcriptome, utilisant l'AFM, et les Random Forests.

Première partie

Développement méthododologique

Chapitre 2

Méthode rapide de clustering hiérarchique en grande dimension

Table des matières

Une partie des résultats de ce chapitre a fait l'objet d'une publication, qui constitue la première section. L'objectif de la méthode présentée ici est de répondre à la question de la création d'un arbre dans le cas d'un grand nombre de feuilles, individus ou variables, et notamment dans le cas des données -omiques.

La méthode présentée dans l'article se base sur l'obtention d'un arbre par dimension et l'agrégation des arbres obtenus en un arbre consensus. Il est démontré que le processus jusqu'à l'arbre final a une complexité sous-quadratique en n, nombre de feuilles des arbres, permettant son application dans la grande dimension. Cette démonstration est présentée à la suite de l'article. Les sections suivant l'article détaillent le contexte original dans lequel a été développé cet algorithme, présentent les bases du *convex clustering*, ainsi que la méthode fused-ANOVA (Chiquet et al., 2017), et les résultats de simulations. Une piste de généralisation de la méthode par utilisation d'un noyau spectral pour améliorer les résultats est aussi introduite.

2.1 Fast tree aggregation for consensus hierarchical clustering

Hulot, A., Chiquet, J., Jaffrézic, F. et Rigaill, G. Fast tree aggregation for consensus hierarchical clustering. BMC Bioinformatics 21, 120 (2020).
https://doi.org/10.1186/s12859-020-3453-6

2.1.1 Résumé

Contexte. Dans de nombreux domaines de recherches, intégrer des données de différentes sources et différents types reste une question ouverte, en particulier en ce qui concerne le clustering et l'apprentissage non supervisé. Dans le domaine de l'analyse de données -omiques, des dizaines de méthodes de clustering ont été développées dans les dernières années. En effet, les méthodes classiquement utilisées jusque-là se sont retrouvées dépassées par données de grande dimension, souvent très bruitées et souvent de types différents (continues, binaires, etc.) qu'il faut arriver à traiter de façon simultanée. Quand une seule source de données est considérée, le clustering hiérarchique (HC) est une méthode populaire : un arbre est une structure hautement interprétable et plus informative qu'une simple partition des données. Cependant, effectuer un clustering hiérarchique sur de multiples sources de données demande de se pencher sur l'interprétation de l'arbre en résultant, et sur les problèmes computationnels posés.

Résultats. Nous proposons ici *mergeTrees*, une méthode qui aggrège un ensemble d'arbres ayant les mêmes feuilles, pour créer un arbre consensus. On considère ici que les arbres ont été crées avant et sont disponibles, sans regard sur les méthodes utilisées pour les construire. Dans le consensus, un cluster présent à la hauteur h contient tous les individus étant dans le même cluster dans tous les arbres à cette même hauteur h. La méthode est déterministe et sa complexité est $\mathcal{O}(nq\log(n))$, n étant le nombre d'individus et q le nombre d'arbres à agréger. L'implémentation que nous proposons est extrêmement efficace dans les simulations, permettant d'agréger un très grand nombre d'arbres en un temps réduit. La méthode a été appliquée sur deux jeux de données issus d'études cliniques, et une alternative spectrale, robuste et efficace, a été introduite.

Conclusions. Notre méthode d'agrégation peut être utilisée en complément du clustering hiérarchique pour obtenir un arbre sur des données hétérogènes. Les résultats montrent la robustesse de la méthode à l'absence d'information dans certaines tables de données, ainsi qu'à la variabilité intra-clusters. La méthode est implémentée en `R`/`C++` et disponible dans le package `R` `mergeTrees`, ce qui la rend facilement employable dans plusieurs domaines de recherche.

2.1.2 Abstract

Background. In unsupervised learning and clustering, data integration from different sources and types is a difficult question discussed in several research areas. For instance in omics analysis, dozen of clustering methods have been developed in the past decade. Indeed, the methods usually employed until then have encounter many limitations, as the data are often high-dimensional, noisy and of different types (continuous, binary, etc.) and need to be analyzed simultaneously. When a single source of data is at play, hierarchical clustering (HC) is extremely popular, as a tree structure is highly interpretable and arguably more informative than just a partition of the data. However, applying blindly HC to multiple sources of data raises computational and interpretation issues.

Results. We propose *mergeTrees*, a method that aggregates a set of trees with the same leaves to create a consensus tree. We consider here a situation where the trees are already available, and do not take into account the way they were created. In our consensus tree, a cluster at height h contains the individuals that are in the same cluster for all the trees at height h. The method is exact and proven to be $\mathcal{O}(nq\log(n))$, n being the individuals and q being the number of trees to aggregate. Our implementation is extremely effective on simulations, allowing us to process many large trees at a time. We also rely on *mergeTrees* to perform the cluster analysis of two real -omics data sets, introducing a spectral variant as an efficient and robust by-product.

Conclusions. Our tree aggregation method can be used in conjunction with hierarchical clustering to perform efficient cluster analysis. This approach was found to be robust to the absence of clustering information in some of the data sets as well as an increased variability within true clusters. The method is implemented in `R`/`C++` and available as an `R` package named `mergeTrees`, which makes it easy to integrate in existing or new pipelines in several research areas.

2.1.3 Background

Data integration has become a major challenge in the past decade as an increasing amount of data is being generated from diverse sources, leading to heterogeneous and possibly high-dimensional data. It is thus essential to develop new methods to analyze multiple data sets at the same time, by taking into account the relationships between the sources and the different underlying mechanisms originating the data. This paper is part of this scope by introducing unsupervised tools to explore multiple hierarchies, built from heterogeneous and multi-source data, typically found in the omics field.

With omics, many studies were successful for linking a particular phenotypic trait to one kind of omic features (Guasch-Ferré et al., 2016; Quesnel-Vallières et al., 2018). However, multi-omics data is the new standard, since integrating several sources (genotyping, transcriptomics, proteomics, and more) is needed to have a finer understanding of the biological processes underlying the phenotypes. Typically, having a better omics-characterization of a disease could help to adjust the prediction of the outcome and the treatment of the patients. Therefore, multi-omics data analyses have recently received much interest in medical research (Hasin et al., 2017; Proctor et al., 2019).

Unsupervised methods – and in particular clustering – are routinely used in omics in order to discern grouping patterns between the observations and link the groups to an outcome such as death or disease. Hierarchical clustering (HC) builds an attractive tree structure with a simple interpretation and is therefore a method of choice in exploratory analyses. Indeed, HC allows to efficiently visualize group structures in the data for various numbers of groups. However, it is not directly adapted to the analysis of multiple, heterogeneous data sources.

In this paper, we propose a novel method and compare it to two existing ones for recovering a single hierarchy – or tree structure – between individuals for which multiple sources of data are available. Although the most natural way to reach this goal is to merge the data sets or the dissimilarities before applying HC, we propose a method that aggregates the result of several HC into a single hierarchy. To this end we introduce a fast tree aggregation algorithm that can deal with many hierarchies to merge. The overall complexity of our tree aggregation method is $\mathcal{O}(nq\log n)$, with q being the number of sources and n the number of individuals.

The rest of the paper is organized as follows: first, we give an overview of the methods that address a similar problem in the literature, in different yet related communities (machine learning, phylogenetics, bioinformatics). This leads us to introduce the rationale for developing our own

method for recovering a single hierarchy from multiple data sets, that we describe in the next section. In particular, we detail the algorithm that aggregates multiple tree structures with a low computational burden. Numerical and statistical performances of the aggregation methods are then studied on simulations. Finally, we illustrate our method on two multi-omics data sets, in breast cancer and cell differentiation.

2.1.3.1 Related work

Retrieving a consensus classification out of several possible classifications is a recurring topic in many fields, such as machine learning, multi-omics and phylogenetics. In this section, we present some of the existing methods that yield a tree in these research areas and discuss the novelty of the proposed algorithm.

Machine Learning In machine learning, the problem of aggregating multiple hierarchies is encountered when using convex clustering with the ℓ_1-norm.

Convex clustering (Pelckmans et al., 2005; Hocking et al., 2011) is a reformulation of hierarchical clustering into a convex optimization problem. It ensures that a unique solution is found at a given regularization parameter. The form of the regularization path depends on the choice of the norm and the weights. While algorithms exist for all weights and norms (Weylandt et al., 2019), they are generally computationally expensive. Moreover, if the weights are not chosen appropriately, individuals can fuse at one point and split later (Chiquet et al., 2017).

Using the ℓ_1 norm in the optimization problem leads to an improvement of the computation time and resources. In this case the method results, however, in a set of trees, one per feature, and needs a posterior treatment to obtain a consensus clustering, typically a tree aggregation method like the one we introduce hereafter.

Multi-omics Many clustering methods have been specifically developed to analyse multi-omics data. Several authors provide full reviews and benchmarks (Wang and Gu, 2016; Huang et al., 2017; Rappoport and Shamir, 2018). In particular, Wang and Gu (Wang and Gu, 2016) suggest the following typology: *i*) direct integrative clustering, consisting in a preprocessing of the original data set before concatenation into a single data set ready for some standard clustering analysis (Mo et al., 2013; Zhang et al., 2012); *ii*) regulatory integrative clustering, which are based on pathways (Vaske et al., 2010); *iii*) clustering of clusters, *i.e.,* methods that take clustering made on different data sets and find a consensus (Lock and Dunson, 2013; Kirk et al., 2012).

The methods that we introduce to recover a consensus tree are related to the clustering of clusters. However, the latter dœs not yield a hierarchical structure as a result. To our knowledge, no consensus tree method has been developed or applied to multi-omics data analysis. Our paper seems to be the first effort in this direction.

Phylogenetics In phylogenetics it is common to bootstrap sequence alignments to compute trees to assess the robustness of a tree (Felsenstein, 1985). It is also quite common to build multiple trees from different data sets (e.g. one tree per gene). Those forests of trees are often reduced to a consensus tree.

Methods that build consensus trees in phylogenetics consider the tree as a set of bipartitions (one per edge) and keep or delete bipartitions based on their occurrence frequency in the forest and/or their compatibility with previously selected bipartitions.

Adams (Adams, 1972; Rohlf, 1982) was the first to address the problem, and proposed to build a consensus tree by keeping bipartitions present in all trees of the forest. Margush and McMorris (Margush and McMorris, 1981) relaxed the constraint by including all bipartitions present in at least half of the trees, leading to the majority rule consensus. Both of these methods suffer from conservatism and lead to polytomies in the tree. Finally Barthélémy and McMorris (Barthélemy and McMorris, 1986) introduced the median tree, which has an algorithmic complexity of $O(n^3)$ and may not be unique.

All these methods consider only the tree topology, not the branching times. In HC fusion heights are an indication of the distance between clusters and are therefore important for the statistical interpretation of the tree.

In the rest of the paper, we stick to methods yielding a single consensus tree, with at most a quadratic complexity, and relying on mathematical distances for the branching pattern.

2.1.4 Methods

In this section we present our method for aggregating trees, and give the details of two other natural methods. We also investigate the complexity of these methods and different ways of applying them to get a consensus hierarchy.

2.1.4.1 Notation

Let $X_1, ..., X_q$ be q data sets, each sharing the same set of n individuals. For conciseness we consider that all the data sets share the same number of features p. Let d be the function used to build the dissimilarity matrix $d(X)$ computed between all individuals of X. Also denote by $\mathcal{T} = \{T_1, ..., T_q\}$ the set of q trees obtained from these data with any HC method, and by $\mathcal{C}(\mathcal{T})$ the consensus tree based on $\mathcal{T}$. The HC method used to obtain the initial set $\mathcal{T}$ dœs not matter. Note, however, that the tree heights should be comparable before the merge: if all the divisions in one tree T_a happen before the divisions of any of the other trees, then the consensus tree will be T_a.

This raises the question of the scaling of the tables associated to each data source. Scaling is a challenge common to all methods in data integration since each source may come from different technologies or correspond to different types of signal. Therefore, they have different ranges of values and distributions (like proteomics and transcriptomics). Typically, applying HC on unscaled features can lead to a tree dominated by the table with the largest variance or range of values. In this section, we assume that the data have already been transformed so that scaling is no longer an issue. We address this question in the Results section when dealing with real-world data.

2.1.4.2 Fast tree aggregation algorithm

In this section we introduce a fast algorithm called mergeTrees to build a consensus from a collection of q trees $\mathcal{T} = \{T_1, ..., T_q\}$ having the same n leaves. It can be summarized as follows:

> *For any observations i and j in $\{1, ..., n\}$, $i \neq j$, if i and j are not in the same cluster in at least one of the trees of $\mathcal{T}$ at height h, then they are not in the same cluster in $\mathcal{C}(\mathcal{T})$ at height h.*

or, equivalently:

> *For any observations i and j in $\{1, ..., n\}$, $i \neq j$, if i and j are in the same cluster in all of the trees of $\mathcal{T}$ at height h, then they are in the same cluster in $\mathcal{C}(\mathcal{T})$ at height h.*

Algorithm 1 mergeTrees

Input: A list of trees $\mathcal{T} = \{T_1, ..., T_q\}$
Output: A consensus tree $\mathcal{C}(\mathcal{T})$

```
n_group ← 1, currentnumberofgroups
Convert each tree to a list of splits
Order all possible splits by decreasing height
current_split ← 1
while n_group < n do
  n_new_group ← 0
  for each current group K do
    n_out ← number of individuals that split from K
    If n_out ≠ 0 and n_out ≠ #K,
        n_new_group++
  end for
  if n_new_group > 0 then
    current_split is active
    Move the individuals that split to their new groups
  else
    current_split is inactive.
  end if
  current_split++
end while
Build C(T) with the selected splits
```

Properties The consensus tree $\mathcal{C}(\mathcal{T})$ reconstructed by mergeTrees satisfies the following properties mentioned by (Steel et al., 2000) and (Bryant et al., 2016), in the phylogenetic context:

- **P1 (Anonymity).** Changing the order of the trees in $\mathcal{T}$ dœs not change $\mathcal{C}(\mathcal{T})$
- **P2 (Neutrality).** Changing the labels of the leaves of the trees in $\mathcal{T}$ simply relabels the leaves of $\mathcal{C}(\mathcal{T})$ in the same way.
- **P3 (Unanimity).** If the trees in $\mathcal{T}$ are all the same tree T, then $\mathcal{C}(\mathcal{T}) = T$

These properties ensure that we can use the method on any set of trees, as long as the trees have the same leaves and labels.

Also note that if multiple divisions occur at the same height in several binary trees, it is possible that the result is not a binary tree.

Algorithmic details Our tree aggregation method proceeds in a divisive manner, by starting with all individuals in the same group and then identifying all splits of the consensus tree from the highest to the lowest. Full details of the proposed algorithm are provided in Algorithm 1 and in the following paragraph in a more intuitive manner.

In our implementation, a tree is represented by a succession of $(n-1)$ splits characterized by (i) the height of the split and (ii) the two clusters coming from this split. These two new clusters are stored as a range of indices rather than a list of indices. This is done by re-labeling in $\mathcal{O}(n)$ the leaves in such a way that the tree is ordered or plane. The algorithm takes as input q trees and thus $(n-1) \times q$ splits. The algorithm initializes a unique cluster with all n leaves. It then processes all splits from the highest to the lowest and checks whether they create a new cluster or not. A split that creates a new cluster is labelled as *active* and the group structure is updated. A split that will not impact the current group structure is labelled as *inactive*. The key idea of the algorithm is to detect active splits using only the smallest cluster of each split.

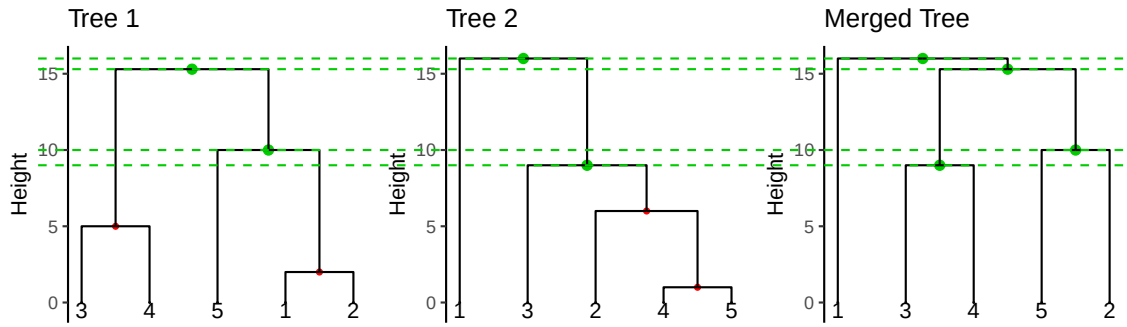


FIGURE 2.1 – **Illustration of the aggregation tree method.** Aggregation of 2 trees, Tree1 (left) and Tree2 (middle) in a Merged Tree (right). The two first trees are built with 4 splits each, the consensus tree is constituted of 2 splits from the first tree and 2 splits from the second tree. Green nodes represent active splits and red nodes inactive splits.

TABLE 2.1 – Description of the trees in Fig 2.1. Splits are ordered by overall height.

	Tree	Split	Height	Cluster 1	Cluster 2	Active
1	2	1	16	1	2, 3, 4, 5	active
2	1	1	15.3	3, 4	1, 2, 5	active
3	1	2	10	5	1, 2	active
4	2	2	9	3	2, 4, 5	active
5	2	3	6	2	4, 5	inactive
6	1	3	5	3	5	inactive
7	1	4	2	1	2	inactive
8	2	4	1	1	4	inactive

This is done with four loops over all leaves of the smallest cluster. The first loop increments the leaf group counter by one. The second loop checks whether the leaf group is active by checking whether the group counter is strictly smaller than the group size. The third loop allocates the leaf to its new group if necessary. The fourth resets the leaf group counter to zero.

Figure 2.1 and Table 2.1 provide a toy example to illustrate how the method works. The third hierarchical clustering is the result of the merging of the first two. Green horizontal dashed lines indicate the active splits.

Space complexity The structures of the trees are stored using matrices of size $n \times 3$. All operations are made through vectors of length n. The space complexity of our algorithm is thus $\mathcal{O}(nq)$.

Time complexity The complexity of Algorithm 1 to merge q trees with n leaves each can be shown to be in $\mathcal{O}(qn\log(n))$. The proof is given in Additional File 1 (section 2.1.7 page 31). Intuitively the $n\log(n)$ appears because the algorithm only uses the smallest cluster of each split. This complexity allows the merging of a large number of trees with a high number of individuals/leaves.

2.1.4.3 Methods for data integration

In the previous section the set of trees $\mathcal{T}$ is assumed to be known. Here, we include the cost of the construction of $\mathcal{T}$ from the data sets $X_1, \ldots, X_q$ into the build of the final consensus tree $\mathcal{C}(\mathcal{T})$. Recall that we assume that all data sets have the same number of features p for clarity.

In the following, we will refer as MC (short for *mergeTrees Clustering*) for the combination of a method that yields a set of trees and the aggregation of these trees with the mergeTrees algorithm. We will focus, for now, on the use of the classical hierarchical clustering method to build the trees.

Apart from using our mergeTrees algorithm on several trees, two other natural methods come to mind. The first idea (*Direct Clustering*, in short DC) is to directly merge the data into a single table: the aggregation criterion is applied on $d(X^c)$ where $X^c = [X_1, ..., X_q]$ is the aggregated table. The second idea (*Averaged Distance*, or AD) is to make the consensus on the dissimilarity matrices before applying HC, by averaging these matrices. Here, the aggregation criterion is applied on $\frac{1}{q}\sum_{j=1}^{q} d(X_j)$.

Time complexity including clustering There are two major operations to build the consensus tree in AD and DC: computation of the dissimilarity matrices and computation of the hierarchical clusterings. The computation of q distance matrices of size $n \times n$, using p features has a complexity of $\mathcal{O}(n^2qp)$. This is the same complexity to create one unique $n \times n$ distance matrix out of a $n \times (pq)$ matrix. The complexity of the agglomerative step of hierarchical clustering is at least $\mathcal{O}(n^2)$ (Murtagh and Contreras, 2012).

To sum-up,

— DC is in $\mathcal{O}(n^2pq)$ (complexity for computing a $n \times n$ dissimilarity matrix out of a $n \times qp$ matrix and building the final HC).

— AD is in $\mathcal{O}(n^2pq)$ (complexity of making q dissimilarity matrices of dimension $n \times n$ using p features, averaging the matrices and building the final HC).

— MC is in $\mathcal{O}(n^2pq)$ (complexity of making q distance matrices of dimension $n \times n$ using p features, building all HC and aggregating them).

In MC, note that the complexity of mergeTrees is dominated by the computational cost of the q dissimilarity matrices. Hence, all methods have the same time complexity when using a classical way of building the hierarchical clusterings. In case of a large number of leaves, this quadratic computation is a liability and the log-linear computation time of the tree aggregation method does not lead to any advantage.

We propose in the next paragraph an approach using the mergeTrees algorithm combined with dimension reduction to reach an overall log-linear complexity.

Dimension reduction and improvement of time complexity In the previous paragraph, we detailed the complexity of the mergeTrees algorithm when combined with a classical hierarchical clustering. The algorithm can be applied on any set of trees, regardless of the method used to build them. This allows to use faster approaches than HC.

In this paragraph, we introduce a way of reducing the overall time complexity of MC by considering a dimension reduction before building the trees.

For both statistical and algorithmic reasons, we suggest to perform a spectral decomposition on the concatenated data sets (i.e. one table of dimensions $n \times pq$), taking only a small amount of the new features and to create the consensus clustering on them. Using truncated SVD (tSVD) to retrieve $k \ll pq$ axes leads to a complexity of $\mathcal{O}(npqk)$ (Halko et al., 2011). In certain cases, using randomized SVD (rSVD) to retrieve k can be a better choice as the complexity of this procedure is $\mathcal{O}(npq \log k)$.

Although it makes sense to simply apply an HC algorithm on the results of the SVD, we propose a different approach. As the new features obtained by the SVD are orthogonal, each of them carries different but complementary information extracted from all the data sets. We therefore feel it makes sense to form a consensus tree out of the set of trees given by the vectors.

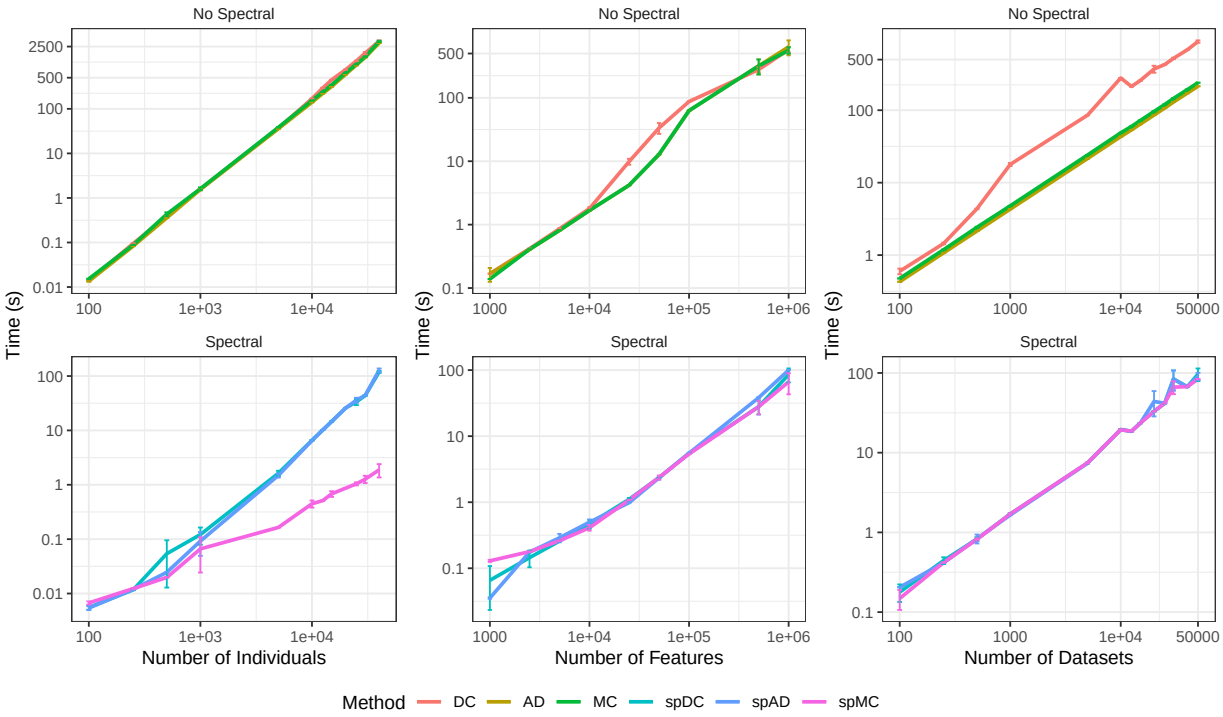


FIGURE 2.2 – **Timing simulations for the different methods.** Number of tables, individuals and features are set to 3, 100 and 100, respectively, when they are not the variable of interest. For the spectral methods, 3 axes were computed by random SVD.

Combining a tSVD, a hierarchical clustering and an aggregation method leads to an overall complexity of $\mathcal{O}(kn^2 + npqk)$. When considering the data in the form of vectors, a hierarchical clustering using Ward's aggregation criterion and Euclidean distance can be obtained directly without computing a distance matrix. Building a tree with this method has a complexity of $\mathcal{O}(n\log(n))$ per feature, so using such an approach to build the collection of q trees before applying mergeTrees leads to a complexity of $\mathcal{O}(qn\log(n))$ for the MC method. Combining this method with the tSVD dimension reduction technique, the overall complexity is $\mathcal{O}(kn\log(n) + npqk)$ for MC.

Note that $kn^2 + npqk$ is larger than $kn\log(n) + npqk$, which means that MC using a spectral decomposition is faster for large n and k small enough.

This direct way of obtaining a clustering in $\mathcal{O}(n\log(n))$ is not possible for DC and AD methods. Indeed, AD relies on the computation of the distance matrices, and DC concatenates all features available into a unique matrix. DC on the spectral vectors is actually the result of a hierarchical clustering performed on the tSVD decomposition of the concatenated data sets.

We will call spAD, spDC and spMC the spectral variants of the methods, i.e. the methods applied on the vectors of an SVD decomposition.

Timing simulations. Results for timing simulations are shown in Figure 2.2.

Timing simulations were performed for all methods and their spectral alternatives. They were repeated three times and averaged. The influence of the number of individuals per data set, the number of features and the number of data sets was studied. In the first simulation design, the number of features per table was set to 100 with 3 tables, and the number of individuals was increased up to a very large number. The opposite design was used for the second simulation scheme, with the number of features set to 100 with 3 tables, and the number of individuals increasing. For the last simulation, the number of individuals and features were set to 100 and the number of tables

available increased. For all the spectral applications, $k = 3$ axes were computed with randomized SVD. The time needed for concatenating all data sets into one before applying the rSVD procedure is included in the time dislayed in the spectral panels.

The three methods in the non spectral case have the same complexity, which is verified in the graphs for the individuals and features per table, as the three curves have the same trend. DC was found to be the most time consuming when dealing with a lot of data sets. The step of computing the distance out of the concatenation result causes an increase in the total time.

When increasing the number of individuals, spMC clearly outperforms its competitors by several orders of magnitude.

The spectral decomposition allowed to considerably reduce the computing time required for all the methods, especially in the case of large numbers of individuals.

Implementation We implemented the `mergeTrees` algorithm in an `R/C++` package called **mergeTrees** available on CRAN (R Core Team, 2019). In our analyses, we always rely on Ward's hierarchical clustering and Euclidean distances. With multivariate data, we use the implementation available in the `R`-base function *hclust* (Murtagh and Legendre, 2014). With vector data, we use the $\mathcal{O}(n \log(n))$ implementation available in the `ward_1d` function of the package **univarclust** (Chiquet, 2019).

2.1.5 Results

2.1.5.1 Simulation study

To compare the performance of AD, DC, MC and their spectral variants (spAD, spDC and spMC), we generated 5 tables of $n = 125$ individuals and $p = 150$ features. Tables were generated vector by vector, $\{\mathbf{y}_j, j = 1, \ldots p\}$ so that $\mathbf{y}_j = (y_{1j}, \ldots, y_{nj})^\intercal \in \mathbb{R}^n$ are realizations of Gaussian variables defined by

$$Y_{ij} = \begin{cases} \mu_{i(k)} + \varepsilon_{ij}, & \text{for} j = 1, \ldots, 50 \\ \varepsilon_{ij}, & \text{for} j = 1, \ldots, 100 \end{cases}$$

where $\varepsilon_{ij} \sim \mathcal{N}(0, 1)$. Hence, only the first 50 features of each table carry some information about a group structure defined by the means $\mu_{i(k)}$ as follows: the n individuals are divided into $K = 5$ balanced groups so that $\mu_{i(k)} = s \times k$ with $i(k)$ the group of individual i, and s a separability factor. This separability factor is introduced to control the difficulty of retrieving the underlying classifications of the individuals: a larger separability factor means more distant groups, while the within-group variance remains the same. Two scenarios are considered: one where all informative features describe the 5 groups, and one where the group information is split among the tables (only 2 groups are represented in each table). For the spectral variants, the feature vectors are bound into one data set on which the SVD is performed. Two axes are retained to form the new set of feature vectors on which AD, DC or MC are applied.

To compare the accuracy of the different methods, we rely on the *Normalized Information Distance* (NID) (Vinh et al., 2010), a distance between partitions based on mutual information. A value of 1 means two partitions with nothing in common, while a distance of 0 means identical partitions. The NID is computed for 5 repetitions of the experiment and averaged, at each level of the reconstructed trees.

Figure 2.3 shows the results of the simulations. The same pattern is observed in both scenarios: when the separability factor is low, all methods struggle to find the correct classification. As the

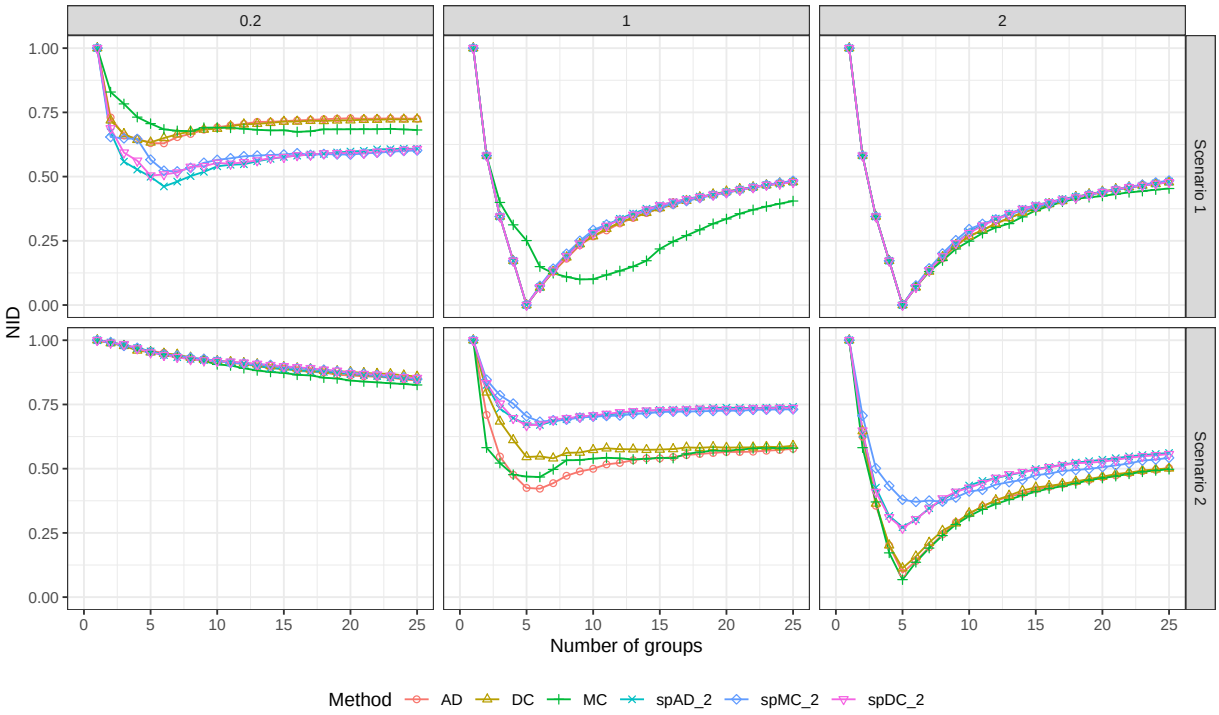


FIGURE 2.3 – **Simulation results for different separability factors and two scenarios.** Scenario 1 (top row): each informative vector contains information on the 5 groups. Scenario 2 (bottom row): each informative vector contains information about 2 of the 5 groups. Columns represent the separability factor between the groups, from the most difficult situation to the easiest one.

separability increases, the NID is minimized for the true number of groups ($k = 5$) for most of the methods. The spectral alternative improves the results for MC when considering the first scenario.

In the second scenario, where the group information is spread among the informative features, the non-spectral alternatives perform better. Having two groups per table allows a better differentiation of the groups, hence, each data set provides a more precise classification, which is reflected on the consensus trees. However, even when the separability is high, the spectral alternatives have trouble finding the classification.

2.1.5.2 Multi-omics data

To illustrate our approach in the context of unsupervised analysis of real-world data with multiple tables, we consider two multi-omics data sets from breast cancer and cell differentiation.

In order to avoid differences in the distances and heights of the trees that would hamper the tree aggregation process, each table are centered and standardized by means of its maximum singular value. The spectral decomposition was performed on the modified data sets, and the new features were neither scaled or centered. Hierarchical clusterings were first built on the separate tables to show the NID values obtained when considering only one type of data. Then the three methods presented above: Direct clustering (DC), Averaged distance (AD) and the proposed mergeTrees Clustering algorithm (MC) were applied, as well as their spectral versions. For AD and DC, the distance matrices and trees are calculated on each table separately, then aggregated.

For each of the obtained trees, we retrieve the classification they provide at each level of division, and compare them to one or more clinical outcomes, using the NID. The results we present in this section are the minimum NID values found and the associated number of groups.

TABLE 2.2 – NID values and number of groups, results for the cell-type data set, taking 3 axes for the spectral decomposition.

	Nb Groups	NID
Data sets		
Gene expression	**4**	**0.14**
Methylation	4	0.47
Spectral axes		
Gene expression-sp	**4**	**0.16**
Methylation-sp	6	0.53
Multivariate Methods		
AD	4	0.27
DC	4	0.27
MC	**6**	**0.22**
Spectral Methods		
SpAD	4	0.27
SpDC	**4**	**0.26**
SpMC	3	0.29

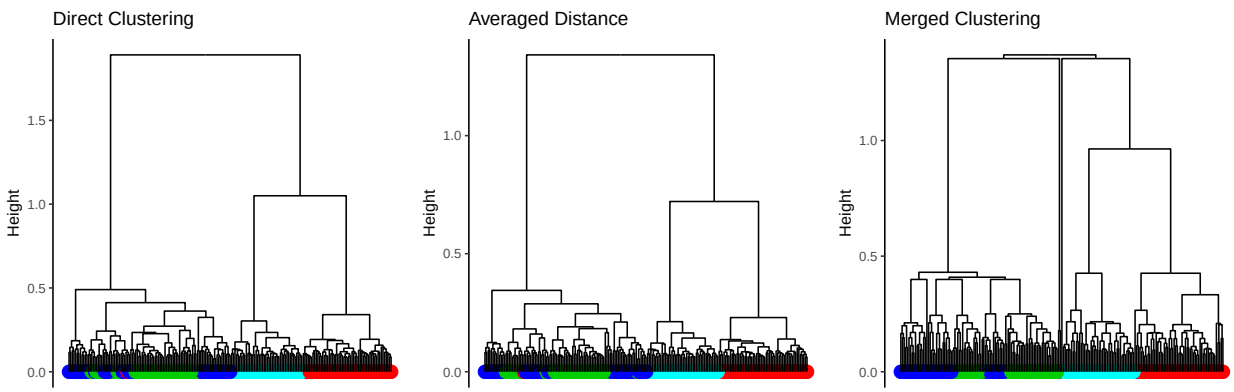


FIGURE 2.4 – **Celltype data sets.** Tree results for the three multivariate non spectral methods for the cell-type data set. Colors at the bottom correspond to the leaves cell-type. Red: CD14, Green: CD4, Blue: CD8, Cyan: whole blood.

Cell-type differentiation The first data set concerns the inflammatory bowel disease and is presented by Ventham *et al.* (Ventham et al., 2016). Methylation (485577 features) and gene expression (46835 features) data were available for 199 samples. Different cell-types were considered: CD14 (57 samples), CD4 (51 samples), CD8 (47 samples) and whole blood (44 samples) were sequenced, originating from 61 individuals. All methylation and gene expression data are freely available at NCBI GEO database (Edgar et al., 2002) (accession GSE87650). Individual clusterings based on the methylation and gene expression data show that the observations tend to cluster based on the cell-type of the sample. We therefore compared the results of the three clustering methods to the cell-type repartition of the samples.

Results are presented in Table 2.2 and in Figure 2.4. Gene expression data obviously contains signal largely related to the cell-type information, since HC leads to a NID value of 0.14. On the contrary, methylation data only reaches a NID of 0.47. The spectral decomposition of the separate tables, retaining 3 axes for each, do not improve the classification.

When analyzing the two data sets together with AD, DC and MC, all methods perform in a similar way.

Regarding the NID value, MC seems to be less impacted by the lack of information concerning the cell-type classification in the methylation data. It, however, selects more groups than expected.

The three spectral variants of the methods perform in a similar way as well. It is worth mentioning that the spectral approach helps MC to select a number of groups closer to the ground truth (from 6 to 3 groups), although the NID is higher. Overall, the three methods seem to be quite robust to this difficult case.

Figure 2.4 shows the trees obtained from the three non-spectral methods. The color bar at the bottom of each dendrogram represents the cell-type of the leave. MC leads to a non binary tree in this case. All the methods seem to have trouble finding the difference between CD4 (green leaves) and CD8 (blue leaves) samples.

It has to be pointed out that the consensus methods provide better NID results than the methylation data but are less efficient than the gene expression data alone. This example shows very well the behaviour of the methods when integrating data sets that are carrying different information. However, this raises the question of the choice of the data sets to be jointly analyzed to be biologically relevant.

TCGA multi-omics breast cancer data The data used in this section was downloaded from the TCGA website. It consists in four types of omics to be integrated for 104 patients: methylation (21123 features), miRNA expression (725 features), protein expression (156 features), gene expression (RNA-seq, 19738 features). The RNA-seq table was log-transformed.

Clinical features such as the age at diagnosis, cancer status, cancer subtype, œstrogen and progesterone receptor status (designated by ER and PR status respectively, in the following paragraphs) are available for all patients with no missing value. The individuals ($n = 104$) are patients with breast cancer distributed into four existing subtypes: Luminal A ($n = 44$), Luminal B ($n = 20$), HER2-enriched ($n = 18$) and Basal-like ($n = 22$). These subtypes are related to the ER and PR status, as the luminal subtypes are associated with positive ER and PR, and the two others are related to negative ER and PR. Clustering was first performed for each dataset separately. These individual clusterings were not found to be related to age or stage of the cancer. The protein and RNA-seq analyses reflected the ER/PR status the best. We therefore compared the results of the consensus methods to these clinical variables in order to quantify their medical relevance. The subtype was also included, as it is related to the ER/PR status and is often of interest in such studies. Results are shown in Table 2.3.

Regarding the NID values based on the individual clusterings at the top of Table 2.3, the protein expression dataset is the most informative in the task of retrieving the ER/PR status, as well as the cancer subtype. The RNA-seq data perform nearly as well, whereas the methylation and miRNA data provide very little information with regard to these clinical variables. When considering the spectral variants, there is an increase in the performance of the RNA-seq data for the subtype classification while it decreases for the ER and PR status. On the other hand, miRNA performance is slightly improved for the ER status. Other data sets do not seem to have improved performance for any of the three clinical variables after a spectral decomposition.

When combining all these data within a multi-omics clustering approach (second part of Table 2.3), all the methods perform better than the methylation or miRNA data alone. They, however, often perform worse than the most informative individual table, i.e. protein. They are closer to the RNA-seq results. The proposed method (MC) for finding a consensus tree performs well to retrieve the ER/PR status, and has better performances for that purpose than the two others. AD performs better for finding a consensus for the subtype classification. MC has a close result for the NID on the subtype, but identifies 8 groups instead of 4.

Table 2.3 – NID values and number of groups, results for the TCGA breast cancer dataset, taking 5 axes for the spectral decomposition.

	ER status		PR status		Subtype	
	NID	N	NID	N	NID	N
Data sets						
methyl	0.77	3	0.78	4	0.69	9
mirna	0.72	2	0.71	2	0.67	4
protein	**0.32**	**2**	**0.45**	**2**	**0.53**	**5**
rna	0.40	2	0.55	2	0.59	4
Spectral DataSets						
methyl-sp	0.78	3	0.84	3	0.70	6
mirna-sp	0.66	2	0.70	2	0.64	5
protein-sp	**0.46**	**2**	**0.48**	**2**	0.58	4
rna-sp	0.71	2	0.73	2	**0.44**	**4**
Non spectral consensus						
AD	0.61	2	0.66	2	**0.54**	**4**
DC	0.68	2	0.70	2	0.57	4
MC	**0.40**	**2**	**0.51**	**3**	0.56	8
Spectral consensus						
SpAD	0.60	2	0.61	2	0.49	4
SpDC	0.46	2	**0.54**	**2**	**0.43**	**4**
SpMC	**0.40**	**2**	0.55	2	0.56	5

The spectral analyses show a similar pattern in the results. The NID values for the MC approach remain nearly the same, but the number of groups found for the subtype with the spectral version is now equal to 5. The DC performances are improved in the spectral setting, as well as the AD approach concerning the subtype.

The stability of the methods was assessed by generating 100 subsamples with a 0.8 proportion in each subsample. Results are shown in Figure 2.5. For each of them, the three methods and their spectral variants are applied and the minimum NID values were computed. The first panel shows the minimum value for each method, the second panel shows the difference of these values between the methods. All violin plots illustrate the high variablity of the results, i.e. the classification is highly dependent on the individuals chosen in the subsamples.

For the standard version of the methods, the violin plots show that DC and AD lead to similar results for the three classifications. MC leads to lower NID values for ER and PR but higher for subtype, when compared to the two others.

When considering the spectral approaches, there is an improvement for MC for the ER classification. However, MC and DC do not seem to benefit as much as AD from the spectral decomposition. Comparison of the methods shows that SpDC and SpAD perform in a similar way for the ER and PR status. SpAD is better at retrieving the subtype. SpMC seems to yield higher NID values for the subtype than the two others but lower for ER and PR status.

2.1.6 Discussion and conclusion

The joint analysis of multi-omics data is a challenging research question. We presented in this paper an algorithm for aggregating multiple hierarchical trees to obtain a consensus clustering. Several advantages of the proposed method have to be pointed out. First of all, it requires no a priori knowledge concerning the optimal number of groups.

Secondly, it is highly computationally efficient on large data sets, with a complexity of $\mathcal{O}(nq\log(n))$,

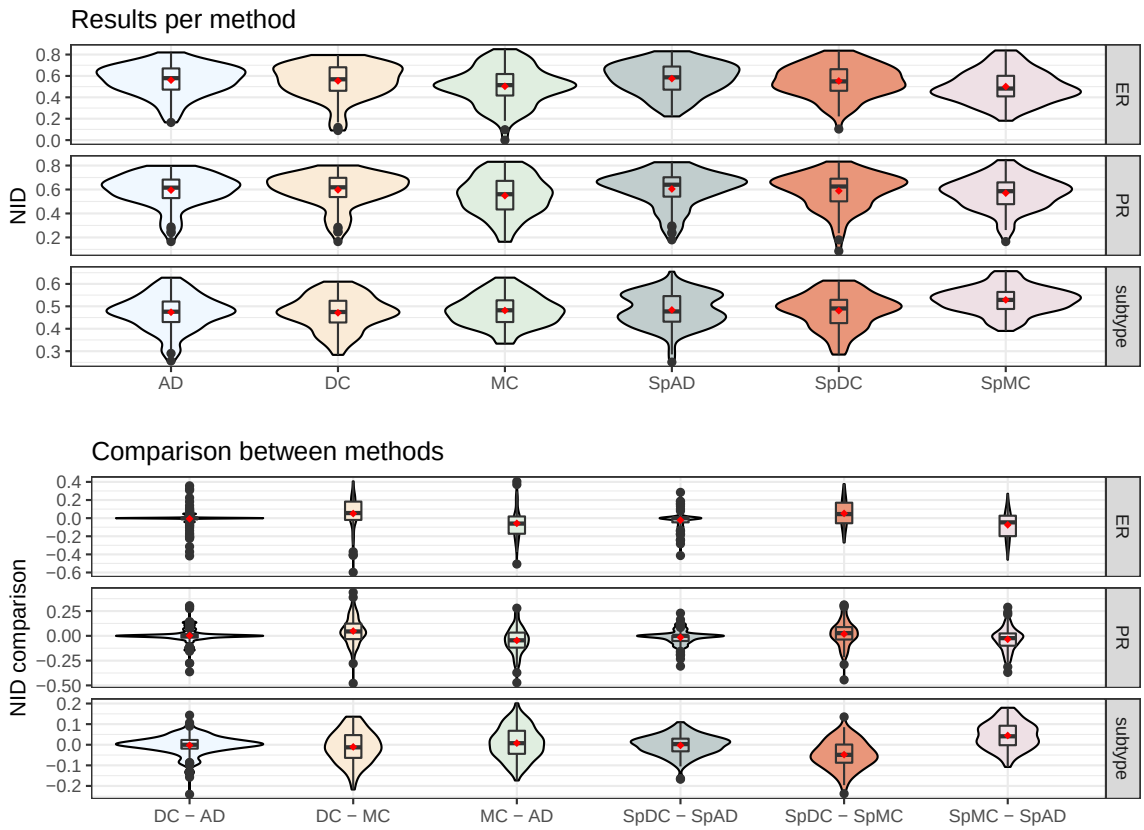


FIGURE 2.5 – **Breast Cancer data sets.** Violin plots of bootstrap results (100 iterations, 80% individuals). Minimum NID obtained for each bootstrap run, without taking into account the number of groups, and differences between these minimum NID values. Spectral method results were obtained with truncated svd taking the first 5 axes of the decomposition.

n being the individuals/leaves and q the number of trees to aggregate. We also introduced a way of combining dimension reduction with building and aggregating the trees in a sub-quadratic overall complexity, allowing to deal with high-dimensional data. This spectral variant can help to retrieve the predominant clustering pattern of the data in a non-linear way. Finally, our approach requires very little data pre-processing, as only centering and standardization by the first singular value is necessary to ensure similar heights in the trees and proper integration. Note that the method can also be of interest when only a set of trees is known.

Several scenarios were investigated in the simulation study. We considered the case where all the features share the same classification information, and then divided the information among the features. The proposed method was compared to two other approaches that either merge all the data sets or vectors into one table, or average the distance matrices obtained on each dimension separately. As expected, the more noise was introduced in the groups, the less the methods were able to retrieve the underlying simulated classification. Our spectral alternative was able to improve the MC performances in the case where all the data sets carry the same information. Two real data sets were also analyzed. The same pattern was observed for both applications. The information contained in one data set was diluted when merged with another data set that did not have the same underlying classification. For the TCGA breast cancer data, the MC approach retrieved well the ER/PR status and performed close to the most informative individual -omics data set for these two clinical variables. In the cell-type case, the three methods performed in a similar way being impacted by the methylation data set.

To conclude, these analyses show that it is important that the data tables integrated in multi-source data provide coherent information to deliver a meaningful global analysis. In the case of contradictory information, it is difficult to automatically merge these data without hampering the interpretation. Nevertheless, our data integration approach is robust to the presence of data tables that do not carry any information.

An interesting research direction is to use the consensus tree approach to compare a set of hierarchical clusterings sharing the same leaves, for instance in a boostrap framework. Indeed, using a distance measure between classifications such as NID or *the Adjusted Rand Index* (Vinh et al., 2010; Hubert and Arabie, 1985) at each level of divisions between the individual trees and the consensus provides a quantification of the distance between the trees and their average.

Acknowledgements. We thank Mahendra Mariadassou for his help and remarks on the phylogenetics related work part.

The results shown here are in part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

2.1.7 *Additional File 1*: Démonstration de la complexité sous-quadratique de la méthode

La démonstration est incluse dans l'article en tant que *Additional File 1 — Proof of time complexity of the* mergeTrees *procedure*. Les notations sont celles de l'article. La démonstration présentée ici diffère de celle de l'article, cette dernière présentant une erreur.

Proof. We aim to prove a recurrence relation on n, the number of individuals in the data. Let $T(n)$ be the worst computation time for a $n-$tree, with the convention $T(1) = 0$. We make the assumption that we only consider, at each step, the individuals for the smallest cluster created by the division (i.e. $n/2$ individuals at most).

Then,

— $T(2) = 1$ (only one split and one element on the smallest cluster) ;
— $T(3) = 2$ (two successive splits, two individuals to consider) ;
— $T(4) = \max(1 + T(3), 2 + 2T(2)) = 4$

Building a tree with four individuals can lead to two different tree configurations: one that is balanced, and one where all the children are on the same side. This leads us to the following recurrence relation for the worst computation time:

$$T(n) = \max_{i=1}^{n/2}\{i + T(i) + T(n-i)\}$$

The first split will consider i individuals in the smallest new cluster, and the remaining complexity is the one of the two trees generated by this split.

Let us now prove that the above expression reduces to

$$T(n) = \frac{n}{2}\log_2(n), \quad \text{for all } n.$$

The initialization for this relation is already proven, as we showed that $T(4) = 4 = 2\log_2(4)$. We

now assume the relation holds true at rank n, and consider a $(n+1)$-individuals tree. Then,

$$\begin{aligned} T(n+1) &= \max_{i=1}^{(n+1)/2} \{i + T(i) + T(n+1-i)\} \\ &\leqslant \max_{i=1}^{(n+1)/2} h(i), \end{aligned}$$

where, by the recurrence relation,

$$h : x \longmapsto x + \frac{x}{2}\log_2(x) + \frac{n+1-x}{2}\log_2(n+1-x).$$

We find the maximum of h by a basic study on $x \in \{1, \ldots, (n+1)/2\}$. Straightforward differentiation leads to

$$h'(x) = 1 + \frac{1}{2\log(2)} \log\left(\frac{x}{n+1-x}\right),$$

which is positive when $x \geq (n+1)/5$. The maximum is then reached either in $x = 1$ or $x = \frac{n+1}{2}$. The two candidates for the maximum are thus

$$\begin{aligned} h(1) &= 1 + \frac{n}{2}\log_2(n) \\ h\left(\frac{n+1}{2}\right) &= \frac{n+1}{2}\log_2(n+1) \end{aligned}$$

Since the function $g : n \longmapsto h(1) - h(\frac{n+1}{2})$ is negative for all n, h reaches its maximum for $x = \frac{n+1}{2}$ and thus

$$T(n+1) = \frac{n+1}{2}\log_2(n+1).$$

This gives us the complexity of the algorithm when we only consider the smallest created cluster. Having q trees to aggregate, we conclude that the final complexity of the mergeTrees Procedure is thus $\mathcal{O}(qn\log(n))$.

□

2.2 Méthode d'agrégation d'arbres dans le cadre du *convex clustering*

La méthode d'agrégation d'arbres, présentée dans l'article précédent, a été originalement développée dans le cadre de l'analyse de données métagénomiques *whole genome*. Le séquençage *whole genome* de l'ADN contenu dans le microbiote intestinal, ou dans le sol, amène à obtenir des comptages pour des millions de gènes, qui doivent ensuite être regroupés en clusters pour déterminer les génomes des bactéries présentes.

De manière plus générale, l'objectif de ce travail est de pouvoir effectuer un clustering hiérarchique pour réaliser un arbre comprenant un grand nombre de feuilles. Lorsque l'on traite des données qui ne relèvent pas de la grande dimension, l'une des méthodes les plus employées consiste à construire l'arbre sur la base d'une distance et d'un critère d'agrégation. Le calcul des distances entre tous les points n'est pas efficace en termes de mémoire et de temps utilisés dans le cadre des données de grande dimension. Lorsque le clustering hiérarchique n'est plus possible, les K-means sont une solution. Cette méthode ne renvoie cependant pas un arbre, et ne permet pas de répondre à une partie de nos objectifs.

La méthode Fused-ANOVA (Chiquet et al., 2017) offre une alternative rapide aux méthodes plus classiques qui permettent d'obtenir un arbre. Elle présente le défaut d'être univariée.

Cette section introduit brièvement les concepts de clustering convexe, Fused-ANOVA, et présente les résultats de la méthode Fused-ANOVA multivariée sur des données simulées.

2.2.1 Clustering convexe

On note X un jeu de données de dimensions $n \times p$.

Problèmes d'optimisation. Hocking et al. (2011) interprète le clustering hiérarchique comme la solution du problème d'optimisation suivant :

$$\underset{\beta \in \mathbb{R}^{np}}{\text{minimize}} \frac{1}{2} \sum_{i=1}^{n} \|X_i - \beta_i\|_2^2, \quad \text{s.t} \quad \sum_{i<i'} \mathbb{1}_{\beta_i \neq \beta_{i'}} \leqslant t, \quad t \in \{1, ..., n(n-1)/2\} \tag{2.1}$$

Lorsque $t > n(n-1)/2$, tous les individus sont dans leur propre groupe, et $\beta_i = X_i$, pour tout i. À l'inverse lorsque $t = 1$, tous les éléments sont dans le même groupe. La hiérarchie s'obtient en forçant les éléments à fusionner dans des groupes, en baissant la valeur de t par étape.

Le *convex clustering* est une relaxation convexe du problème d'optimisation du clustering hiérarchique (Pelckmans et al., 2005; Hocking et al., 2011), qui considère le problème d'optimisation suivant :

$$\underset{\beta \in \mathbb{R}^{nd}}{\text{minimize}} \frac{1}{2} \sum_{i=1}^{n} \|X_i - \beta_i\|_2^2 + \lambda \sum_{\substack{k,l:k \neq l \\ k,l=1}}^{n} \omega_{kl} \Omega(\beta_k - \beta_l), \tag{2.2}$$

où Ω est une norme convexe et β_k est le coefficient du groupe k, $k \in \{1, ...K\}$. λ est le paramètre de régularisation/pénalité. Les poids ω_{kl}, tels que $\omega_{kl} = \omega_{lk}$ et $\omega_{kl} > 0$, sont des éléments importants dans la résolution de ce problème, et contrôlent notamment la vitesse de fusion entre deux éléments et la forme du chemin de régularisation. Le choix de ces poids sera détaillé plus tard.

L'intérêt de l'utilisation de ce problème réside dans la norme convexe, qui apporte la garantie de trouver une solution unique au problème 2.2 à λ donné.

Définir un arbre via le problème d'optimisation. Le clustering hiérarchique est défini par le chemin de régularisation du problème d'optimisation 2.2. La Figure 2.6 montre un exemple de chemin de régularisation obtenu par du clustering convexe sur le jeu de données *aves*, issu du package `univarclust` (Chiquet, 2019), dans `R`[1]. Lorsqu'il n'y a aucune régularisation, qui correspond au cas $\lambda = 0$, les éléments sont tous dans des groupes différents, ce qui constitue le bas de l'arbre. Au fur et à mesure que λ augmente, les éléments se regroupent, jusqu'à être tous dans le même groupe. L'arbre est formé.

La structure de l'arbre obtenu dépend de la norme Ω choisie, mais aussi des poids ω_{kl}. Une condition nécessaire à l'obtention d'un arbre est que le problème d'optimisation n'autorise pas les divisions de groupes une fois que ceux-ci se sont formés. Certaines associations de poids et norme ne permettront pas de réaliser cette condition.

De plus, l'aspect de l'arbre est aussi dépendant des poids : il est nécessaire de choisir des poids qui permettent aux éléments proches de se regrouper rapidement pour garder une structure cohérente avec les données.

1. Le package `fusedanova` est une ancienne version du package `univarclust`.

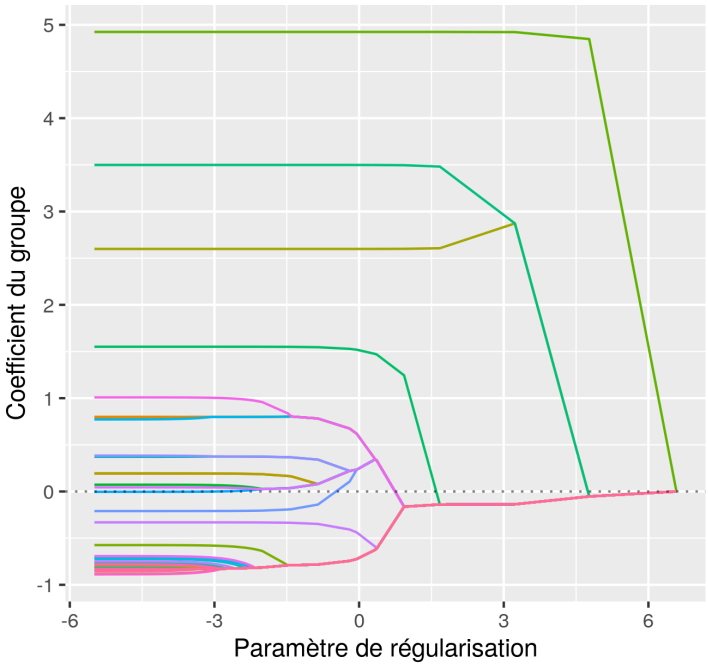


FIGURE 2.6 – Illustration d'un chemin de régularisation réalisé en utilisant Fused-ANOVA sur le jeu de donnée aves du package `fusedanova`.

Choix de la norme et des poids. Dans Hocking et al. (2011), les auteurs démontrent que l'utilisation de la norme ℓ_1 permet de réduire la complexité à $\mathcal{O}(pn\log(n))$, complexité sous-quadratique, mais le résultat donne alors un arbre par dimension/variable. Ils démontrent de plus que l'utilisation de poids unitaires associés à cette norme permet d'obtenir un chemin de régularisation sans séparation.

Développements précédents. Des algorithmes ont été développés pour résoudre ce problème d'optimisation, en utilisant notamment la norme ℓ_1 et ℓ_2, les packages `R` suivants peuvent être utilisés pour ce problème :

— ClusterPath (Hocking et al., 2011)
— cvxclustr (Chen et al., 2015; Chi and Lange, 2015).

Le principal problème rencontré dans l'utilisation de ces deux packages est la lenteur de la résolution du problème, ainsi qu'une grande complexité, lorsqu'on utilise la norme ℓ_2 ($\mathcal{O}(n^3)$, voir Hocking et al., 2011). De plus, aucune des méthodes de ces deux packages n'a été développée dans l'optique de produire un arbre en résultat, mais dans celle de trouver une partition des données.

Des développements plus récents ont été effectués, notamment (Weylandt et al., 2020), qui propose un algorithme permettant la reconstruction d'un arbre via le *convex clustering*, mais qui, au vu des temps de calcul, ne peut s'appliquer à des jeux de données impliquant un grand nombre de feuilles.

2.2.2 Fused-ANOVA

Dans (Chiquet et al., 2017), les auteurs introduisent la méthode Fused-ANOVA, une version contrainte de l'ANOVA, qui s'appuie sur les problèmes d'optimisation présentés dans la section précédente ainsi que sur celui de la MANOVA. Ils introduisent notamment des poids, dits *exponentially*

adaptative qui assurent l'obtention d'une structure d'arbre dans la solution. De plus, cette structure reflète au mieux celle des données, puisque deux éléments qui sont proches dans les données fusionnent rapidement.

Problèmes d'optimisation. Le problème d'optimisation de la MANOVA est le suivant :

$$\underset{\beta \in \mathbb{R}^{Kp}}{\text{minimiser}} \sum_{i=1}^{n} \left\| X_i - \boldsymbol{\beta}_{\kappa(i)} \right\|_2^2, \tag{2.3}$$

où $\kappa : i \mapsto \kappa(i)$ est une fonction d'a priori sur les groupes des éléments. Dans le cas où il n'y a pas d'*a priori* connu, $K = n$. Combiné à une pénalisation de type fused-Lasso (Tibshirani et al., 2005), on obtient le problème d'optimisation suivant :

$$\underset{\beta \in \mathbb{R}^{Kp}}{\text{minimize}} \frac{1}{2} \sum_{i=1}^{K} \left\| X_i - \beta_{\kappa(i)} \right\|_2^2 + \lambda \sum_{k,l:k \neq l} \omega_{kl} \Omega(\beta_k - \beta_l), \tag{2.4}$$

Ce problème d'optimisation, si l'on choisit une norme convexe, rejoint le clustering convexe, avec un *a priori* de la structure de groupes donné par la fonction κ. Les problèmes de lenteur de résolution en ℓ_2 sont toujours présents, ainsi que le problème du choix des poids. Ces deux problèmes sont détaillés dans l'article de Chiquet et al. (2017). Ils s'appuient sur les démonstrations de Hocking et al. (2011).

Garantir une structure d'arbre. Le théorème 2 de Chiquet et al. (2017) spécifie que le chemin de régularisation ne contient aucune division lorsque les poids ω_{kl} sont choisis sous la forme :

$$\omega_{kl} = n_k n_l f(|\bar{X}_k - \bar{X}_l|), \tag{2.5}$$

avec f une fonction décroissante positive. Ils utilisent les poids dits *exponentially adaptative* :

$$\omega_{kl} = n_k n_l \exp\{-\gamma \sqrt{n} |\bar{X}_k - \bar{X}_l|\}, \gamma > 0 \tag{2.6}$$

Ces poids permettent d'obtenir une structure d'arbre cohérente avec les données puisque deux éléments proches fusionnent rapidement. Le paramètre γ contrôle les vitesses de fusion dans ces poids. Le choix de ce paramètre est laissé à l'utilisateur.

Fused-ANOVA multivarié. L'utilisation de la norme ℓ_1 s'impose pour obtenir une méthode applicable pour pouvoir produire des clusterings sur un grand nombre d'individus. L'utilisation de la norme ℓ_1 et des poids *exponentially adaptative* conduit à obtenir un arbre par dimension/variable. Le résultat est donc donné sous forme d'un ensemble d'arbres. Il faut maintenant agréger ces arbres pour obtenir un arbre unique.

On utilise la méthode d'agrégation d'arbre `mergeTrees` exposée dans l'article de la section 2.1 pour résoudre ce problème et obtenir une méthode multivariée. Les sections suivantes détaillent les performances sur données simulées de la méthode.

La complexité de l'algorithme `mergeTrees` étant sous-quadratique, son utilisation conjointement avec Fused-ANOVA reste une procédure de complexité sous-quadratique et permet son application sur des données de grande dimension.

U/V	V_1	V_2	$\dots$	V_C	Sums
U_1	n_{11}	n_{12}	$\dots$	n_{1C}	$n_{\bullet 1}$
U_2	n_{21}	n_{22}	$\dots$	n_{2C}	$n_{\bullet 2}$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
U_R	n_{R1}	n_{R2}	$\dots$	n_{RC}	$n_{\bullet R}$
Sums	$n_{1\bullet}$	$n_{2\bullet}$	$\dots$	$n_{C\bullet}$	$\sum_{ij} n_{ij} = N$

Tableau 2.4 – Tableau de contingence entre deux classifications U et V sur les mêmes individus. Source : (Vinh et al., 2010)

2.2.3 Performances de Fused-ANOVA multivarié

Packages et version de R. Le package `univarclust` a été utilisé pour réaliser les tests de ce chapitre. Le fused-ANOVA est codé dans la fonction *clusterpath_l1*. On utilise fused-ANOVA sans *a priori* sur les groupes des individus/feuilles de l'arbre, le problème d'optimisation dans ce cas est le même que celui de clusterpath. Le choix des poids, de la norme, et de l'algorithme utilisé pour le résoudre sont ceux décrits dans (Chiquet et al., 2017).

Trois versions de `R` ont été utilisées pour produire les résultats de la section suivante : la version 3.6.1 via les serveurs Migale, et les versions 4.0.1 et 4.0.2.

Utilisation du NID *(Normalized Information Distance).* Pour comparer deux classifications, on utilise le NID (Vinh et al., 2010). Ce critère est une distance, bornée entre 0 et 1. Un NID de 0 signifie que les classifications sont identiques, tandis qu'un NID de 1 signifie que les classifications n'ont rien en commun. En prenant U et V deux classifications réalisées sur les mêmes n individus, on définit le NID de la façon suivante :

$$1 - \frac{I(U,V)}{\max\left(H(U), H(V)\right)} \tag{NID}$$

où $H(U)$ est l'entropie, et $I(U,V)$ l'information partagée entre U et V (*mutual information*). En se référant au tableau 2.4, l'entropie et la *mutual information* se définissent comme :

$$H(U) = -\sum_{i=1}^{R} \frac{n_{i\bullet}}{N} \log \frac{n_{i\bullet}}{N} \tag{Entropie}$$

$$I(U,V) = \sum_{i=1}^{R}\sum_{j=1}^{C} \frac{n_{ij}}{N} \log \frac{n_{ij}/N}{n_{i\bullet}n_{\bullet j}/N^2} \tag{Mutual Information}$$

Simulation de temps. Pour évaluer le temps de calcul, une table de données de $p = 100$ variables a été simulée pour n allant de 500 à 10^6 individus. On cherche à réaliser un clustering des individus.

La Figure 2.7 page 37 montre les temps de calcul obtenus pour *hclust* et la combinaison *mergeTrees* et *clusterpath_l1* (nommée FA-MT), en moyennant trois évaluations par *microbenchmark*, ainsi que la mémoire demandée pour un calcul. Les paramètres par défaut ont été utilisés pour les fonctions *dist* et *hclust*, le temps de calcul correspond donc à une distance euclidienne et un complete-linkage. Le temps et la mémoire montrés ne prennent pas en compte l'étape de création des tables, qui a été faite à part.

Lorsque le nombre d'individus/feuilles de l'arbre est bas (≤ 5000), le clustering hiérarchique est la solution la plus avantageuse. Pour des données de plus de 5000 individus/feuilles, le fused-ANOVA

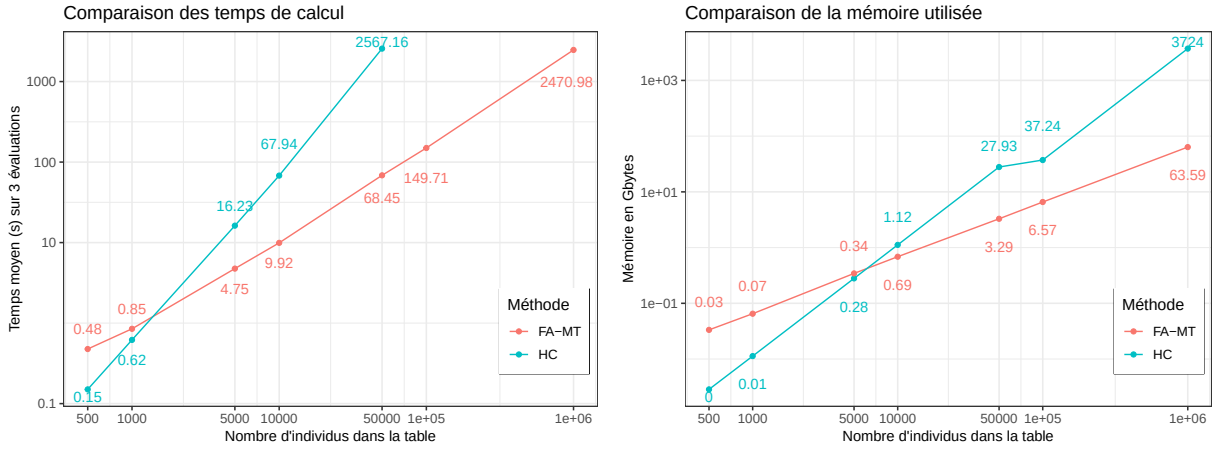


FIGURE 2.7 – Comparaison des temps de calcul entre hclust et Fused-ANOVA multivarié, sur 100 variables et de 500 à 10^6 individus. Points supérieurs à 50000 individus non calculés pour hclust du fait de la mémoire demandée. Échelle log-log.

multivarié a un intérêt majeur, autant sur le temps de calcul que sur la mémoire utilisée. Le fait que l'algorithme est sous-quadratique apparaît clairement sur la figure.

Performances. Pour évaluer les performances de la méthode Fused-ANOVA multivariée, nommée FA-MT dans la suite, par rapport au clustering hiérarchique classique, on se place dans un cas où le nombre de feuilles, ici le nombre d'individus, à considérer est grand. On simule une table de données de $p = 100$ variables, avec $n = 25\ 000$ et $n = 50\ 000$. Les individus sont répartis en 15 groupes de moyennes différentes. L'écart-type à l'intérieur d'un groupe est fixé à $\sigma = 0.1$, puis 0.5, pour étudier les cas de petites et grandes variances.

$$\begin{aligned} i = \{1, ..., n\}, i \in k = \{1, ...15\} \\ Y_i \sim \mathcal{N}(k, \sigma^2) \end{aligned} \tag{2.7}$$

On utilise deux distances (manhattan et euclidienne) et trois critères d'agrégation (*single-linkage*, *complete-linkage* et critère de Ward, tel qu'implémenté dans l'option *ward.D2* d'*hclust* (Murtagh and Legendre, 2014) pour le clustering hiérarchique classique.

On compare la combinaison FA-MT calculée avec différents γ pour les poids avec les résultats de ces arbres. Pour chaque arbre obtenu, on calcule le NID entre les classifications obtenues pour les niveaux de coupures pour $k = 1, ..., 1000$ sur l'arbre et la classification simulée.

La Figure 2.8 page 38 montre les performances que l'on obtient pour plusieurs γ, en comparaison avec les différentes distances et critères d'agrégation pour le clustering hiérarchique classique, pour des nombres d'individus différents (25 000 puis 50 000). La variance est laissée à 0.1^2 dans les deux cas. Le minimum du NID pour chaque méthode est présenté dans le Tableau 2.5 page 38. Dans les deux figures, la performance de la méthode FA-MT est très dépendante du paramètre γ utilisé. Dans le premier cas, $n = 25\ 000$, les NID trouvés par la méthode FA-MT sont très petits mais toujours différents de 0, la classification obtenue diffère sur quelques individus. En prenant en compte le temps de calcul et la mémoire utilisée, ainsi que les NID trouvés par HC, ce dernier est la méthode la plus avantageuse. FA-MT produit cependant des performances correctes. Dans le deuxième cas, $n = 50\ 000$, quel que soit le paramètre, le NID minimum obtenu est toujours très proche de 0, et inférieur à celui des méthodes de clustering hiérarchique, à l'exception de la combinaison euclidien-

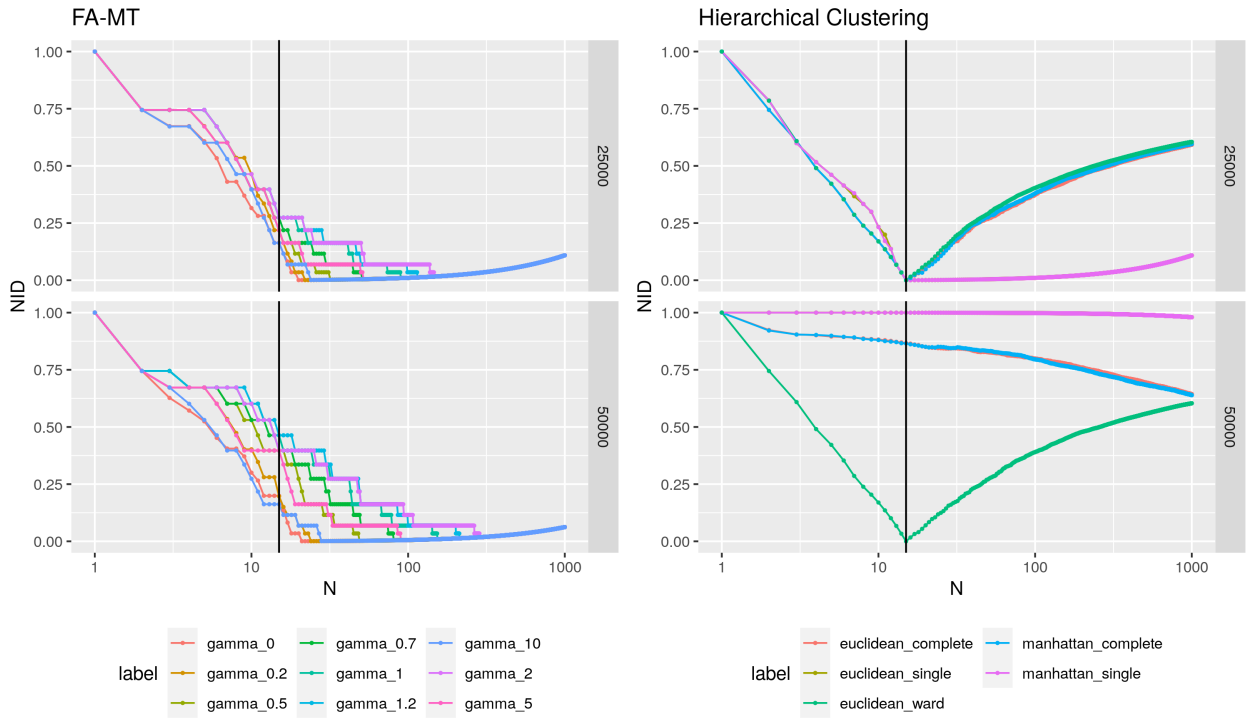


FIGURE 2.8 – Comparaison des performances entre hclust et Fused-ANOVA multivarié. Écart-type : 0.1. La droite verticale désigne le nombre de groupes simulés. Echelle log pour l'axe des abscisses.

FA-MT									
$n = 25\ 000$									
γ	0	0.2	0.5	0.7	1	1.2	2	5	10
min NID	0.0006	0.0009	0.0021	0.0045	0.0092	0.0123	0.0161	0.0046	0.0011
Nb Groupes	20	22	32	51	90	115	147	52	24
$n = 50\ 000$									
γ	0	0.2	0.5	0.7	1	1.2	2	5	10
min NID	0.0004	0.0006	0.0023	0.0045	0.0093	0.0132	0.0179	0.0050	0.0009
Nb Groupes	21	24	49	81	154	213	286	90	28

HC					
$n = 25\ 000$					
distance	euclidean	euclidean	manhattan	manhattan	euclidean
critère	single	complete	single	complete	ward
min NID	0	0	0	0	0
Nb Groupes	15	15	15	15	15
$n = 50\ 000$					
distance	euclidean	euclidean	manhattan	manhattan	euclidean
critère	single	complete	single	complete	ward
min NID	0.9800	0.6440	0.9800	0.6391	0.0000
Nb Groupes	1000	1000	1000	1000	15

TABLEAU 2.5 – Résultats des performances pour les méthodes FA-MT et HC pour $n = 25000$ et $n = 50000$ individus, avec variance de 0.1^2. Niveau de coupure maximum testé : 1000.

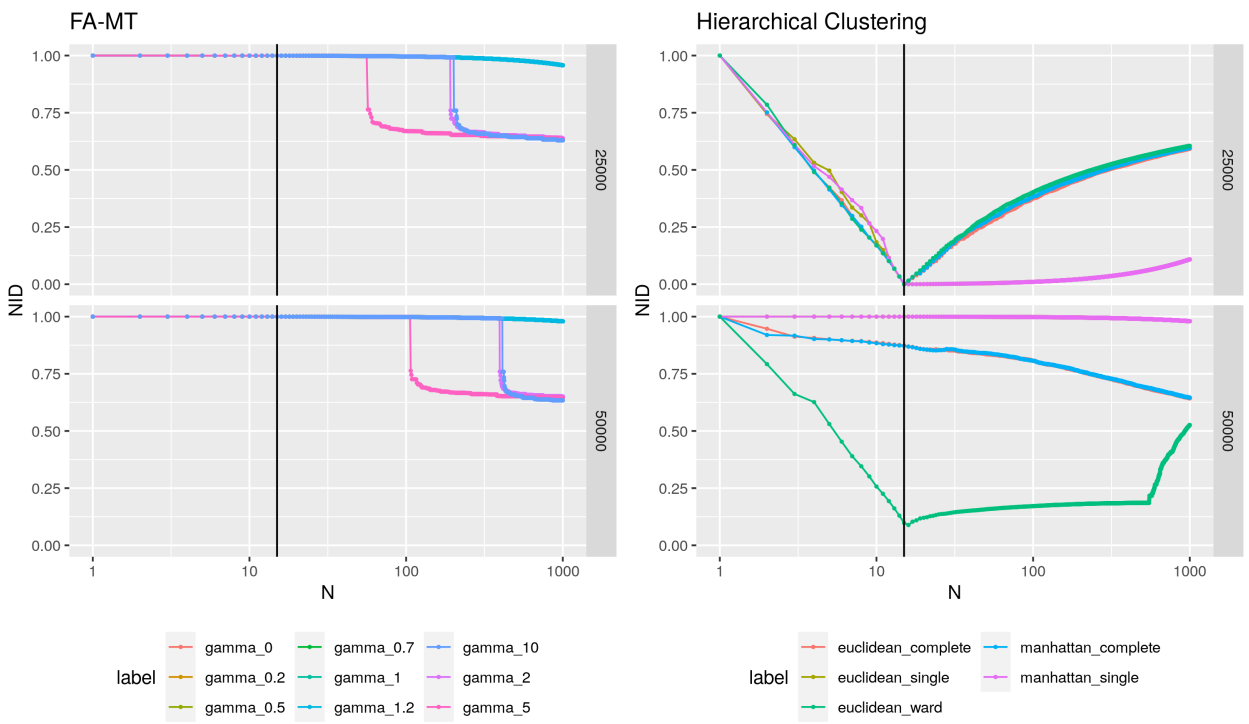


FIGURE 2.9 – Comparaison des performances entre hclust et Fused-ANOVA multivarié. Écart-type : 0.5. La droite verticale désigne le nombre de groupes simulés. Echelle log pour l'axe des abscisses.

ward.D2. Lorsque le nombre de variables est très inférieur au nombre d'individus, FA-MT est plus avantageux, dans le cas de petites variances.

Ces deux figures montrent des résultats obtenus sur des données très peu bruitées (variance faible). Dans la réalité, et en particulier en vue d'une application sur des données -omiques, on s'attend à une variance plus élevée. Les résultats présentés sur la Figure 2.9 montrent le comportement des méthodes lorsque la variance est fixée à 0.5^2. Les valeurs minimum de NID et le nombre de groupes associés sont disponibles dans le Tableau 2.6 page 40. Dans les deux cas étudiés, les performances de FA-MT sont très inférieures à ce que l'on obtient pour le clustering hiérarchique classique, le NID étant toujours très proche de 1 : la classification n'est pas du tout retrouvée. L'augmentation de la variance n'a que peu d'influence sur le comportement de HC en général. On peut remarquer ici que les courbes obtenues avec des paramètres γ plus élevés (2, 10 et 5) se dégagent des autres et semblent amener des NID plus petits, bien que leurs performances restent très en dessous de ce qui est attendu. A l'inverse, lorsque la variance était plus petite, la courbe associée à $\gamma = 0$ était la plus proche du nombre de groupes réels.

Ces résultats nous montrent que la méthode FA-MT ne peut pas être appliquée telle quelle à des données bruitées. Nous proposons d'effectuer une étape de réduction de dimension à l'aide de méthodes spectrales avant d'appliquer la méthode de clustering Fused-ANOVA multidimensionnelle.

2.2.4 Spectral fused-ANOVA

Cette section est en partie issue du résumé soumis aux Journées de Statistique de la Société Française de Statistique de 2018.

Clustering Spectral. L'idée du clustering spectral est d'utiliser le spectre d'une matrice de similarité comme nouvelles données pour effectuer le clustering (Ng et al., 2002). On propose ici

FA-MT									
$n = 25\ 000$									
γ	0	0.2	0.5	0.7	1	1.2	2	5	10
min NID	0.9577	0.9577	0.9575	0.9575	0.9574	0.9574	0.6380	0.6353	0.6292
Nb Groupes	1000	1000	1000	1000	1000	1000	997	994	995
$n = 50\ 000$									
γ	0	0.2	0.5	0.7	1	1.2	2	5	10
min NID	0.9795	0.9795	0.9795	0.9795	0.9795	0.9794	0.6513	0.6443	0.6337
Nb Groupes	1000	1000	1000	1000	1000	1000	998	1000	1000

HC					
$n = 25\ 000$					
distance	euclidean	euclidean	manhattan	manhattan	euclidean
critère	single	complete	single	complete	ward
min NID	0	0	0	0	0
Nb Groupes	15	15	15	15	15
$n = 50\ 000$					
distance	euclidean	euclidean	manhattan	manhattan	euclidean
critère	single	complete	single	complete	ward
min NID	0.9800	0.6427	0.9800	0.6456	0.0887
Nb Groupes 1000	999	1000	1000	16	

Tableau 2.6 – Résultats des performances pour les méthodes FA-MT et HC pour $n = 25000$ et $n = 50000$ individus, avec variance de 0.5^2. Niveau de coupure maximum testé : 1000.

d'appliquer un noyau gaussien pour le calcul de la matrice de similarités K $(n \times n)$, entre les différents vecteurs d'observations :

$$K_{ij} = \exp(-\|X_i - X_j\|_2^2 / 2\sigma^2), \tag{2.8}$$

avec σ un paramètre de dispersion. Le Laplacien normalisé de la matrice K est ensuite calculé comme $L = D^{-1/2} K D^{-1/2}$ où D est la matrice diagonale telle que $D_{ii} = \sum_{j=1}^{n} K_{ij}$. On applique alors une méthode de clustering quelconque (ici, Fused ANOVA) aux vecteurs propres associés aux r plus grandes valeurs propres, ces vecteurs propres étant considérés comme les nouvelles données d'entrée.

Approximation de Nyström. Dans un contexte de grande dimension, calculer et stocker toute la matrice K, de taille $n \times n$, est extrêmement coûteux en temps et en mémoire, avec une complexité en $\mathcal{O}(n^2)$. Le même problème est rencontré pour effectuer une SVD du Laplacien (lui-même de taille $n \times n$), avec une complexité en $\mathcal{O}(n^3)$ si l'on calcule tous les axes.

On utilise l'approximation de Nyström pour les matrices de Gram (Williams and Seeger, 2001; Drineas and Mahoney, 2005) qui permet d'approcher la matrice K, notée $K(n, n)$ dans la suite. En notant $s \subset n$ un sous-échantillon de n, l'approximation se fait ainsi :

$$K(n, n) \approx K(n, s) K(s, s)^{\dagger} K(s, n), \tag{2.9}$$

où $K(n, s)$ désigne la matrice de kernel limitée aux s colonnes du sous-échantillon choisi, et $K(s, s)^{\dagger}$ la matrice pseudo-inverse de $K(s, s)$.

On calcule ensuite non pas la SVD du Laplacien mais seulement sur $K(s, s) = U \Sigma V^{\intercal}$. On utilise de plus une SVD tronquée, pour limiter la complexité des opérations, celle-ci ayant une complexité en $\mathcal{O}(s^2 r)$ en notant r le nombre d'axes retenus. On obtient X^{new}, la matrice des nouvelles données,

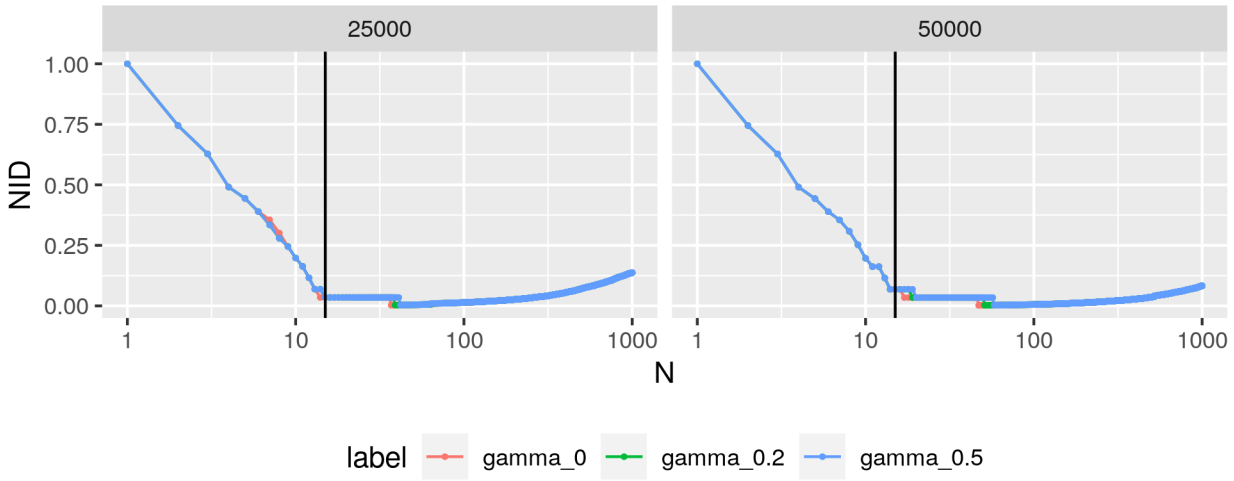


FIGURE 2.10 – Performances spectrales de la méthode fused-ANOVA multivariée. Noyaux calculés par approximation de Nyström en sélectionnant 10% des individus au hasard. 2 axes retenus pour réaliser le clustering. SD = 0.5 dans les deux cas.

de dimension $n \times r$, avec r les dimensions retenues, associées aux plus grandes valeurs propres :

$$X^{\text{new}} \approx K(n,s)V_{(r)}\Sigma_{(r)}^{-1}, \tag{2.10}$$

où $V_{(r)}$ $(s \times r)$ et $\Sigma_{(r)}$ $(r \times r)$ sont les matrices tronquées issues de la SVD. Le clustering s'effectue sur une version renormalisée de X^{new}.

La complexité de cette méthode est en $\mathcal{O}(s^3nr)$, $s \ll n$, $r \ll n$ du fait de la décomposition en valeurs propres. Le fait de prendre un sous-échantillon très petit par rapport à n permet d'avoir un temps de calcul moindre qu'une méthode à complexité quadratique. Appliquer la méthode de clustering sur les nouvelles données reconstituées donne une complexité en $\mathcal{O}(nr\log(n))$.

Nouvelles performances. On reprend les mêmes cas de simulations que pour les performances non spectrales. Pour calculer l'approximation du noyau, on prend 10% des individus de la table, sélectionnés au hasard. La méthode de sélection peut impacter les résultats (Kumar et al., 2012), mais n'a pas fait l'objet de simulations plus poussées ici. Le design des simulations implique que tous les vecteurs/individus portent la même information. Le fait que la sélection soit faite de manière uniforme parmi les individus de la table n'affecte pas négativement les résultats. Deux vecteurs propres sont retenus pour créer le clustering.

Les résultats des nouvelles simulations sont présentés sur la Figure 2.10 page 41. Le clustering hiérarchique classique n'a pas été étudié ici, l'approximation du noyau ne permettant pas de réduire le temps de calcul ou la mémoire utilisée puisque la méthode requiert une matrice complète de distances pour fonctionner. La variance a été fixée à 0.5^2. On observe que les résultats de FA-MT sont fortement améliorés, avec des NID très proches de 0. Le paramètre γ semble avoir beaucoup moins d'influence sur les performances dans les cas décrits.

La méthode FA-MT spectral apporte beaucoup d'améliorations aux performances et est prometteuse.

2.3 Conclusion

Dans ce chapitre, nous nous sommes intéressés à la problématique de l'agrégation d'arbres, problématique motivée par le clustering convexe et notamment la méthode Fused-ANOVA, univariée. Un algorithme rapide pour l'obtention d'un clustering consensus a été mis en place, et utilisé conjointement avec Fused-ANOVA pour une application multivariée. Les résultats des simulations démontrent la rapidité de la méthode, ainsi que des performances stables sur des données peu variables. Dans le cas de données plus variables, les performances peuvent être améliorée par une utilisation des méthodes spectrales.

Le nombre de paramètres à configurer est pour l'instant le frein principal du développement de la méthode. Le paramètre γ, qui contrôle la vitesse de fusion dans l'arbre, doit être fixé au mieux. Nous avons fait le choix ici de fixer le même paramètre γ pour toutes les dimensions de la table de données, mais il peut être plus judicieux pour des données réelles d'en fixer un par variable. Au vu de la rapidité de la méthode, une procédure par cross-validation peut être envisagée. On peut cependant observer que le paramètre semble avoir peu d'importance dans la variante spectrale de la méthode. Pour le cas spectral, le nombre d'individus choisi pour calculer l'approximation du noyau, ainsi que la méthode pour les choisir, doivent faire l'objet de plus de simulations. Toutes les étapes permettant de fixer ces paramètres doivent être développées de façon sous-quadratique pour éviter de perdre l'avantage de la vitesse de la méthode FA-MT. Enfin, la méthode FA-MT doit encore faire l'objet d'une application sur des données réelles pour pouvoir juger de ses performances.

Chapitre 3

Estimation d'un réseau multipartite avec modèle stochastique par blocs

Table des matières

Dans le cadre de l'inférence de réseaux, il n'est pas rare de connaître une partition *a priori* des nœuds du réseau, dits Groupes Fonctionnels (GF) dans la suite. Il est aussi courant de vouloir estimer des communautés dans un réseau : trouver des groupes de nœuds très connectés entre eux, pouvant présenter un intérêt. Il peut être intéressant dans l'analyse de vouloir identifier des groupes de nœuds tout en gardant la structure *a priori* et en prenant en compte les groupes et connexions estimées dans les autres GF. Dans le contexte des données multi-omiques, les GF sont souvent les types d'entités : protéines, métabolites, etc., qui sont connus à l'avance et proviennent de tables de données différentes. Créer des clusters d'entités à l'intérieur du réseau – tout en gardant la structure des -omiques – peut servir à mieux cerner des processus biologiques.

Nous proposons ici de combiner une méthode de recherche de groupes dans un cadre multi-réseaux, et une méthode d'inférence de réseaux. Plus particulièrement, nous employons une version multi-partite du SBM, appelée GREMLIN, et une version pondérée du Graphical Lasso. Les deux méthodes sont appliquées de façon itérative, pour parvenir à inférer un réseau respectant une structure de blocs connue *a priori*, et produisant une structure plus affinée par une recherche de blocs à l'intérieur des GF. L'itération des deux permet d'ajuster les pénalités du Glasso et d'obtenir une matrice d'adjacence respectant les contraintes. Le réseau inféré est un réseau non dirigé. Dans le cas où il n'y a pas d'*a priori* sur une partition en GF, le procédé revient à une itération d'un SBM classique et d'un Glasso.

Ce chapitre est constitué de trois parties : premièrement nous présentons les méthodes utilisées, soit le SBM, le LBM, GREMLIN et le Glasso. Ensuite nous introduisons l'algorithme itératif, appelé *janine*, mis en place pour inférer le réseau et estimer les blocs. Enfin, une application sur données simulées et une application sur un jeu de données réelles sont présentées.

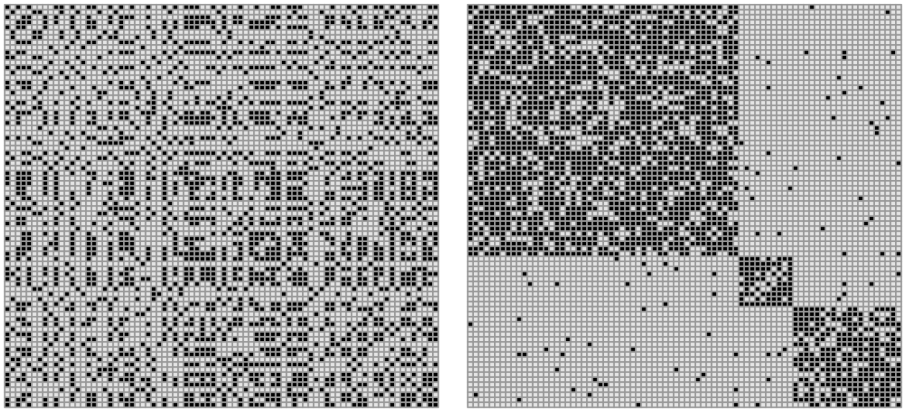


FIGURE 3.1 – À gauche : la matrice originale simulée (80×80) comportant 3 groupes de nœuds. À droite : la matrice re-ordonnée selon les groupes trouvés par un SBM.

3.1 Méthodes

3.1.1 *Stochastic Block Model* (SBM)

Un SBM (Snijders and Nowicki, 1997) est un modèle de loi de génération de graphe, notamment employé pour la détection de communautés (clusters) à l'intérieur de graphes non orientés. Dans ce modèle, on fait l'hypothèse que les nœuds appartiennent à des groupes différents. Les probabilités de connexion entre deux nœuds d'un même groupe ou de deux groupes séparés, sont différentes. Une illustration du SBM est proposée dans la Figure 3.1 page 44.

Les nœuds $1, ..., p$ sont séparés en K communautés. On note $\alpha_k = \mathbb{P}(i \in k)$ la probabilité que le nœud i appartienne au groupe k, avec $\sum_k \alpha_k = 1$.

Il existe une variable latente Z_{ik}, telle que $Z_{ik} = 1$ si $i \in k$. Le vecteur $Z_i = (Z_{i1}, ..., Z_{iK})$ est donc distribué selon : $Z_i \sim \mathcal{M}(1, \alpha)$ avec $\alpha = (\alpha_1, ..., \alpha_K)$.

On se place ici dans le cas où l'on cherche à estimer une matrice d'adjacence, A, binaire. Des modèles plus généraux existent pour estimer des matrices pondérées, utilisant notamment la loi normale, ou Poisson, ainsi que Multinomiale. Dans notre contexte, les éléments de A sont tirés selon la loi suivante, conditionnelle aux groupes auxquels appartiennent les nœuds :

$$A_{ij} | (Z_{ik} Z_{jl} = 1) \sim \mathcal{B}(\pi_{kl}), \tag{3.1}$$

où π_{kl} représente la probabilité d'existence d'un arc entre deux nœuds des groupes k et l. On considère que la probabilité pour qu'il existe une arête entre deux nœuds d'une même communauté est plus forte que celle entre deux nœuds de communautés différentes. Ces hypothèses se traduisent par une structure en blocs dans la matrice d'adjacence, lorsque les nœuds sont rangés par communautés.

Estimation des paramètres. On rencontre deux problèmes majeurs dans l'estimation des paramètres : le premier est un problème d'identifiabilité, le deuxième une loi que l'on ne peut écrire sous forme explicite. On note θ l'ensemble des paramètres à estimer, soit $\{\pi_{kl}, \forall (k,l) \in \{1, ..., K\}^2; \alpha\}$. De manière générale, on estime les paramètres Z et α en calculant l'estimateur du maximum de vraisemblance, soit la solution de :

$$\underset{\theta}{\operatorname{argmax}}\, \mathbb{P}(A; \theta). \tag{3.2}$$

Or dans notre contexte, cela revient à devoir calculer $\mathbb{P}(A;\theta) = \sum_Z \mathbb{P}(A, Z;\theta)$. D'après Allman et al. (2009, 2011), les paramètres du modèles sont identifiables dans un SBM binaire à un nombre de groupes donné, pour un nombre de nœuds minimum, mais sont connus à une permutation près. Calculer la somme sur l'ensemble des classifications possibles devient de plus rapidement infaisable, le nombre de possibilités étant trop important, surtout si l'on ne connaît pas le nombre de groupes.

Dans ce cas, on utilise un algorithme EM (Dempster et al., 1977), basé sur une alternance des deux étapes *Expectation* (E) et *Maximization* (M), pour calculer le maximum de vraisemblance, jusqu'à convergence :

— Etape E : Calculer $\mathbb{P}(Z|A;\hat{\theta})$ en utilisant l'estimation $\hat{\theta}$ de l'étape M ;
— Etape M : Calculer $\hat{\theta} = \text{argmax}_\theta \, \mathbb{E}[\log \mathbb{P}(A, Z;\theta|A)]$. On note $\mathbb{Q}(Z) = \mathbb{E}[\log \mathbb{P}(A, Z;\theta|A)]$.

Dans notre cas, la loi de $\mathbb{P}(Z|A)$ est inconnue, les variables Z_{ik} et Z_{jl} ne sont pas indépendantes conditionnellement aux observations. On utilise alors une variante de l'EM : l'EM variationnel (Jordan et al., 1999; Jaakkola, 2001), dans laquelle on considère une borne inférieure $\mathcal{J}$ pour $\mathbb{P}(A;\theta)$. Pour toute distribution $\mathcal{R}(Z)$, approximation de $\mathbb{P}(Z|A;\theta)$, on a :

$$\begin{aligned}\mathcal{J}(A, \mathcal{R}(Z)) &= \log \mathbb{P}(A;\theta) - D_{KL}\{\mathcal{R}(Z)||\mathbb{P}(Z|A;\theta)\} \\ &= \mathcal{H}(\mathcal{R}(Z)) + \mathbb{E}_{\mathcal{R}(Z);\theta}[\log \mathbb{P}(A, Z)] \leq \log \mathbb{P}(A;\theta)\end{aligned} \tag{3.3}$$

où D_{KL} désigne la divergence de Kullback-Leibler, distance entre deux distributions (Kullback and Leibler, 1951) :

$$D_{KL}\{\mathcal{R}(Z)||\mathbb{P}(Z)\} = -\mathcal{H}(\mathcal{R}(Z)) - \sum_{Z\in\mathcal{Z}} \mathcal{R}(Z) \log \mathbb{P}(Z|A;\theta). \tag{3.4}$$

Dans l'algorithme EM, on utilise alors des étapes modifiées :

— Etape E : $\hat{\mathcal{R}}(Z) = \text{argmin}_{\mathcal{R}} \, D_{KL}\{\mathcal{R}(Z)||\mathbb{P}(Z|A;\theta)\}$, trouver la distribution $\mathcal{R}(Z)$ la moins éloignée de ce que l'on voudrait pouvoir calculer ;
— Etape M : Calculer $\hat{\theta} = \text{argmax}_\theta \, \mathbb{E}_{\hat{\mathcal{R}}(Z)}[\log \mathbb{P}(A, Z;\theta|A)]$

Le fait de chercher l'approximation $\mathcal{R}(Z)$ sur l'ensemble des lois disponibles ne simplifie en rien le problème original, il faut réduire l'espace de recherche. On se restreint pour la suite à la famille de lois qui prennent la forme factorisée suivante (Mariadassou et al., 2010) :

$$\mathcal{R}(Z) = \prod_i \mathcal{R}(Z_i) = \prod_i \prod_k \tau_{ik}^{Z_{ik}}, \tag{3.5}$$

on a donc : $\tau_{ik} \approx \mathbb{P}(Z_{ik} = 1|A_i)$, avec la contrainte $\sum_k \tau_{ik} = 1$. τ_{ik} est la probabilité que le nœud i appartienne au groupe k, sachant les données, le groupe d'appartenance du nœud sera assigné en regardant le maximum a posteriori des τ_{ik}.

On peut calculer le terme d'entropie de $\mathcal{J}(A, \mathcal{R}(Z))$:

$$\mathcal{H}(\mathcal{R}(Z)) = -\mathbb{E}_{\mathcal{R}(Z)}[\log \mathcal{R}(Z)] = -\sum_{k=1}^{K}\sum_{i=1}^{p} \mathbb{P}(Z_{ik} = 1) \log \mathbb{P}(Z_{ik} = 1) = -\sum_{k=1}^{K}\sum_{i=1}^{p} \tau_{ik} \log \tau_{ik} \tag{3.6}$$

Vraisemblance et estimation des paramètres. La vraisemblance complète et log-vraisemblance, pour une partition K des données, peuvent s'exprimer par les équations suivantes :

$$\log \mathbb{P}(A, Z) = \log \mathbb{P}(A|Z) + \log \mathbb{P}(Z) \tag{3.7}$$

On connaît la loi de Z :

$$\log \mathbb{P}(Z) = \log \prod_{i=1}^{p} \mathbb{P}(Z_i) = \log \prod_{i=1}^{p} \prod_{k=1}^{K} \alpha_k^{Z_{ik}} = \sum_{i=1}^{p} \sum_{k=1}^{K} Z_{ik} \log \alpha_k \tag{3.8}$$

On a aussi :

$$\log \mathbb{P}(A|Z) = \sum_{(i,j)} \sum_{(k,l)} \mathbb{P}(A_{ij}|Z_{ik}Z_{jl} = 1) = \sum_{(i,j)} \sum_{(k,l)} \log \left(\pi_{kl}^{A_{ij}} (1 - \pi_{kl})^{(1-A_{ij})} \right)^{Z_{ik}Z_{jl}} \tag{3.9}$$

La borne $\mathcal{J}(A, \mathcal{R}(Z))$ s'écrit alors de la manière suivante :

$$\mathcal{J}_\tau(A, \mathcal{R}(Z)) = \sum_{i=1}^{p} \sum_{k=1}^{K} \tau_{ik} \log \alpha_k + \sum_{(i,j)} \sum_{(k,l)} \tau_{ik}\tau_{jl} \left[A_{ij} \log \pi_{kl} + (1 - A_{ij}) \log(1 - \pi_{kl}) \right] - \sum_{k=1}^{K} \sum_{i=1}^{p} \tau_{ik} \log \tau_{ik} \tag{3.10}$$

On doit maintenant estimer les paramètres. En utilisant les équations précédentes et la forme factorisée de $\mathcal{R}(Z)$, on a par le théorème de Bayes :

$$\tau_{ik} = \mathbb{P}(Z_{ik} = 1|A_i) = \frac{\alpha_k \mathbb{P}(A_i|Z_{ik} = 1)}{\sum_q \alpha_q \mathbb{P}(A_i|Z_{iq} = 1)} \propto \alpha_k \mathbb{P}(A_i|Z_{ik} = 1) \tag{3.11}$$

avec $\mathbb{P}(A_i|Z_{ik} = 1) = \prod_{l=1}^{K} \prod_{j=1}^{p} (\pi_{kl}^{A_{ij}} (1 - \pi_{kl})^{1-A_{ij}})^{Z_{ik}Z_{jl}}$. Cette étape correspond à l'étape E de l'algorithme EM. En reprenant l'expression de $\mathcal{J}(A, \mathcal{R}(Z))$ en supposant les paramètres τ_{ik} connus, on peut estimer les autres paramètres, pour l'étape M de l'algorithme EM. En dérivant la borne $\mathcal{J}_\tau(A, \mathcal{R}(Z))$ par rapport à α_k, en introduisant la contrainte $\sum_k \alpha_k = 1$, on obtient :

$$\left. \frac{\partial \mathcal{J}_\tau(A, \mathcal{R}(Z)) - \eta(\sum_k \alpha_k - 1)}{\partial \alpha_k} \right|_{\tau_{ik}} = \sum_{i=1}^{p} \tau_{ik} \frac{1}{\alpha_k} - \eta = 0 \Leftrightarrow \alpha_k = \frac{1}{\eta} \sum_{i=1}^{p} \tau_{ik} \tag{3.12}$$

En utilisant de nouveau la contrainte $\sum_k \alpha_k = 1$, on en déduit le paramètre $\eta = p$, on obtient l'estimation suivante :

$$\hat{\alpha}_k = \frac{1}{p} \sum_{i=1}^{p} \tau_{ik} \tag{3.13}$$

Le dernier paramètre s'obtient de manière similaire, en dérivant la borne par rapport à π_{kl} :

$$\left. \frac{\partial \mathcal{J}_\tau(A, \mathcal{R}(Z))}{\partial \pi_{kl}} \right|_{\tau_{ik}} = - \sum_{(i,j)=1}^{p} \tau_{ik}\tau_{jl} \frac{A_{ij}}{\pi_{kl}} + \sum_{(i,j)=1}^{p} \tau_{ik}\tau_{jl} \frac{1 - A_{ij}}{1 - \pi_{kl}} = 0 \tag{3.14}$$

La résolution donne l'estimation pour $\hat{\pi}_{kl}$:

$$\hat{\pi}_{kl} = \frac{\sum_{(i,j)} \tau_{ik}\tau_{jl} A_{ij}}{\sum_{(i,j)} \tau_{ik}\tau_{jl}} \tag{3.15}$$

3.1.2 *Latent Block Model* (LBM).

Un LBM est un modèle de co-clustering (Govaert and Nadif, 2003). Une illustration du LBM est proposée dans la Figure 3.2 page 47.

On rejoint ici le cadre des modèles de mélanges évoqués dans l'introduction, en supposant une structure de groupes dans les données $X_{n \times p}$, telle que la loi de génération des données est une loi

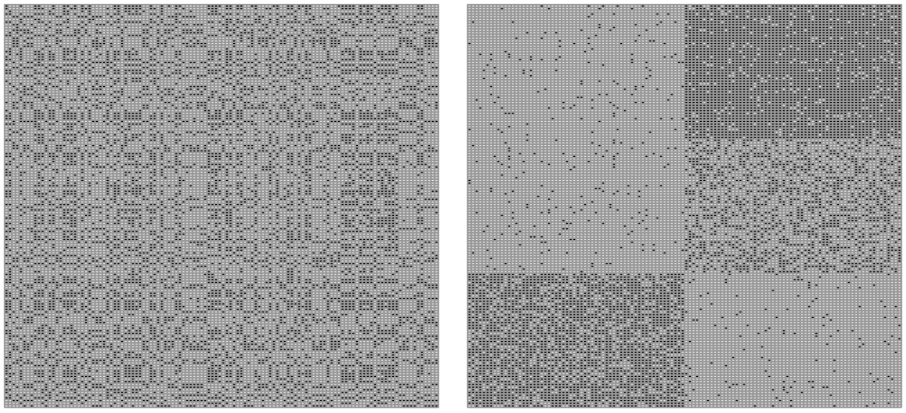


FIGURE 3.2 – À gauche : la matrice originale simulée (150×120) avec 3 groupes sur les lignes et 2 groupes sur les colonnes. À droite : la matrice re-ordonnée selon les groupes trouvés par un LBM.

de mélange avec structure de groupes à la fois sur les colonnes et sur les lignes. On a donc une probabilité d'appartenance à un groupe sur les lignes, α_k, et une probabilité d'appartenance à un groupe sur les colonnes β_q, associées aux variables latentes respectives Z et W. Le SBM peut être vu comme un cas particulier du LBM, où la matrice est symétrique, et les groupes des lignes et des colonnes sont les mêmes. Les équations données dans cette partie sont très proches de celles de la partie précédente. Les problèmes d'identifiabilité et de résolution de l'algorithme EM sont aussi rencontrés pour ce cas, on utilise à nouveau un algorithme EM variationnel. Les résolutions des équations et estimations sont moins détaillées.

Les variables Z_i et W_j sont iid respectivement et indépendantes entre elles : les groupes définis sur les lignes sont indépendants des groupes définis sur les colonnes.

$$\begin{cases} i \in \{1, \dots, n\},\ k \in \{1, \dots, K\},\ \mathbb{P}(Z_i = k) = \alpha_k \\ j \in \{1, \dots, p\},\ q \in \{1, \dots, Q\},\ \mathbb{P}(W_j = q) = \beta_q \end{cases} \tag{3.16}$$

Les variables Z_i et W_j sont distribuées selon des lois multinomiales : $Z_i = (Z_{i1}, ..., Z_{iK}) \sim \mathcal{M}(1, \alpha)$, $W_j = (W_{j1}, ..., W_{jQ}) \sim \mathcal{M}(1, \beta)$. Les deux paramètres α et β respectent la contrainte : $\sum_k \alpha_k = 1$ et $\sum_q \beta_q = 1$.

On se place ici dans un cas particulier, puisqu'on étudie une matrice X binaire :

$$X_{ij} | (Z_{ik} W_{jl} = 1) \sim \mathcal{B}(\pi_{kl}), \tag{3.17}$$

Les paramètres π_{kq}, pour $i \in k$, $j \in q$, sont aussi appelés probabilités de connexion. On note θ l'ensemble de tous les paramètres : $\theta = (\pi_{11}, ..., \pi_{KQ}; \alpha_1, ..., \alpha_K; \beta_1, ..., \beta_Q)$. On peut écrire la vraisemblance complète du modèle :

$$\mathbb{P}(X, Z, W; \theta) = \mathbb{P}(X|Z, W; \theta)\mathbb{P}(Z, W; \theta) = \mathbb{P}(X|Z, W; \theta)\mathbb{P}(Z; \theta)\mathbb{P}(W; \theta) \tag{3.18}$$

avec :

$$\mathbb{P}(X|Z, W; \theta) = \prod_{i,j,k,q} \mathbb{P}(X_{ij}; \pi_{kq})^{Z_{ik} W_{jq}}, \qquad \mathbb{P}(Z; \theta) = \prod_{i=1}^{n} \prod_{k=1}^{K} \alpha_k^{Z_{ik}}, \qquad \mathbb{P}(W; \theta) = \prod_{j=1}^{p} \prod_{q=1}^{Q} \beta_q^{W_{jq}} \tag{3.19}$$

On reprend les mêmes mécanismes explicités dans la partie SBM : on utilise un VEM et une loi

$\mathcal{R}$ pour définir une borne inférieure pour la vraisemblance. La formule utilisée pour la borne voit apparaître un deuxième terme d'entropie :

$$\mathcal{J}(A, \mathcal{R}(Z)) = \mathcal{H}(\mathcal{R}(Z)) + \mathcal{H}(\mathcal{R}(W)) + \mathbb{E}_{\mathcal{R}(Z)\mathcal{R}(W)}[\log \mathbb{P}(A, Z, W)] \leq \log \mathbb{P}(A) \tag{3.20}$$

On reprend la même famille de loi que précédemment pour $\mathcal{R}$ et on définit $\mathcal{R}(Z)$ et $\mathcal{R}(W)$ de la façon suivante :

$$\mathcal{R}(Z) = \prod_{i=1}^{n} \prod_{k=1}^{K} \tau_{ik}^{Z_{ik}} \qquad \mathcal{R}(W) = \prod_{j=1}^{p} \prod_{q=1}^{Q} \gamma_{jq}^{W_{jq}} \tag{3.21}$$

On a alors les estimations suivantes :

$$\hat{\tau}_{ik} = \mathbb{P}(Z_i = 1|X_i) \propto \alpha_k \mathbb{P}(X_i|Z_{ik} = 1), \qquad \hat{\gamma}_{jq} = \mathbb{P}(W_j = 1|X_j) \propto \beta_q \mathbb{P}(X_j|Z_{jq} = 1) \tag{3.22}$$

avec :

$$\begin{aligned} \mathbb{P}(X_i|Z_{ik} = 1) &= \prod_{q=1}^{Q} \prod_{j=1}^{p} (\pi_{kq}^{X_{ij}} (1 - \pi_{kq})^{1-X_{ij}})^{Z_{iq} W_{jq}}, \\ \mathbb{P}(X_j|W_{jq} = 1) &= \prod_{k=1}^{K} \prod_{i=1}^{n} (\pi_{kq}^{X_{ij}} (1 - \pi_{kq})^{1-X_{ij}})^{Z_{iq} W_{jq}} \end{aligned} \tag{3.23}$$

On peut en déduire :

$$\hat{\alpha}_k = \frac{1}{n} \sum_{i=1}^{n} \tau_{ik}, \qquad \hat{\beta}_q = \frac{1}{p} \sum_{j=1}^{p} \gamma_{jq}, \qquad \hat{\pi}_{kq} = \frac{\sum_{(i,j)} \tau_{ik} \gamma_{jq} X_{ij}}{\sum_{(i,j)} \tau_{ik} \gamma_{jq}} \tag{3.24}$$

3.1.3 *Integrated Completed Likelihood* (ICL).

Les paramètres sont estimés à une partition de K groupes donnée, l'étape suivante est le choix de la partition la plus adaptée. On utilise pour cela le critère d'ICL, développé par (Biernacki et al., 2000) dans le cadre des modèles de mélanges gaussiens. Dans le cas d'un LBM, en reprenant les notations utilisées précédemment, à K et Q donnés, respectivement nombre de groupes sur les lignes et sur les colonnes, l'ICL est le suivant :

$$\begin{aligned} \mathrm{ICL}(K, Q) &= \log \mathbb{P}(X, Z, W|(K, Q); \theta) \\ &= \log \mathbb{P}(X|Z, W, K, Q; \theta) + \log \mathbb{P}(Z|K, Q; \theta) + \log \mathbb{P}(W|K, Q; \theta) \end{aligned} \tag{3.25}$$

On sélectionne le couple (K, Q) maximisant cette quantité.

Ce critère a été adapté pour les SBM (Daudin et al., 2008), par une approximation de la vraisemblance complète. Si l'on note X les données, on estime K, le nombre de groupes, comme $\mathrm{argmin}_K \mathrm{ICL}(K; \theta)$ où le critère ICL prend la forme :

$$\mathrm{ICL}(K; \theta) = -\log(\mathbb{P}(X, K; \theta)) + \mathcal{H}(K; \theta) \tag{ICL}$$

L'entropie s'écrit sous la forme :

$$\mathcal{H}(K; \theta) = -\sum_{i=1}^{n} \sum_{k=1^K} \tau_{ik} \log \tau_{ik} \tag{3.26}$$

3.1.4 GREMLIN : association d'un SBM multipartite et d'un LBM

On se base ici sur l'article Bar-Hen et al. (2018), qui présente la méthode GREMLIN, une adaptation du modèle SBM au cas multipartite, à l'aide d'un *Latent Block Model* (LBM). Un réseau est dit multipartite lorsque les nœuds peuvent être répartis en groupes. Dans le sens général, les nœuds d'un groupe dans un réseau multipartite n'ont pas d'interaction entre elles, seuls deux nœuds de deux groupes différents interagissent.

Dans notre contexte, on considère que les nœuds sont à l'avance assignés à des groupes, mais aussi que les nœuds d'un même groupe ont des interactions. Dans ce cas, un réseau multipartite peut être vu comme un ensemble de réseaux : les réseaux intérieurs aux groupes, et les réseaux d'interactions extérieures aux groupes. La combinaison d'un SBM et d'un LBM, appliqués à un réseau multipartite, sera appelée *Multipartite Block Model* (MBM).

On suppose que l'on connaît une partition *a priori* des nœuds d'un graphe, en G sous-graphes, de dimensions $p_g, g \in \{1, ..., G\}$. L'indice g désigne ici le groupe des variables, les groupes fonctionnels. Les réseaux associés à ces tables sont désignés par leurs matrices d'adjacence et d'incidence :

— A^{gg}, matrice d'adjacence de la table g ;
— $A^{gg'}$, $g \neq g'$, matrices d'incidence (non carrées), représente les réseaux d'interactions entre les tables.

L'ensemble des réseaux que l'on a à disposition est l'ensemble indicé par $\{1, ..., G\}^2$.

GREMLIN est défini comme une combinaison de SBM et de LBM, dans le sens où la méthode estime les communautés dans les matrices d'adjacence A^{gg} sur la base des arêtes présentes dans ces réseaux (partie SBM), mais aussi sur la base des matrices d'incidence $A^{gg'}$ (partie LBM). Dans le cas où on ne connaît pas de structure *a priori* des nœuds, on se ramène à un SBM classique.

GREMLIN estime les groupes sur la base de l'ensemble des matrices d'adjacence et d'incidence, mais ne comporte pas d'étape d'estimation de la structure du réseau. Cette étape est essentielle. On propose d'utiliser un Glasso, en combinaison de GREMLIN, pour estimer simultanément les groupes, et le réseau.

3.1.5 *Graphical Lasso* (Glasso) : estimation d'un graphe

Modèles graphiques gaussiens. On considère un jeu de données X de taille $n \times p$. On suppose ici que les données sont centrées. Dans le cadre des modèles graphiques gaussiens (GGM), on fait l'hypothèse que les données sont obtenues à partir d'une loi normale : $X \sim \mathcal{N}(0, \Sigma)$, où Σ est la matrice de variance-covariance de dimensions $p \times p$ et définie positive.

On note $\Omega = \Sigma^{-1}$ la matrice de précision, aussi appelée matrice de concentration. Dans ce cadre particulier, les coefficients Ω_{ij} de la matrice Ω peuvent être reliés aux corrélations partielles, par la formule suivante (Lauritzen, 1996) :

$$\rho_{ij|\mathcal{V}\setminus\{i,j\}} = -\frac{\Omega_{ij}}{\sqrt{\Omega_{ii}}\sqrt{\Omega_{jj}}} \tag{3.27}$$

Dans cette configuration, si $\Omega_{ij} = 0$, alors les deux variables X_i et X_j sont indépendantes, conditionnellement au reste des données.

Estimer la matrice Ω est une étape qui peut s'obtenir, sous réserve que la dimension le permette, par inversion simple de Σ. Dans ce cas, une étape de seuillage de la matrice est requise pour identifier les dépendances les plus significatives. Sans cette étape, le graphe obtenu est complet.

Graphical lasso. La méthode de *Graphical Lasso* (Glasso), présentée dans (Banerjee et al., 2008a; Friedman et al., 2008a), est une méthode pénalisée d'estimation de Ω. Sa pénalisation permet d'estimer directement une matrice parcimonieuse (sparse), donc de sélectionner les corrélations partielles les plus significatives, qui correspondront aux arêtes du réseau. L'estimation de la matrice Ω passe par la maximisation de la log-vraisemblance des données, problème donné par :

$$\max_{\Omega} \ \log(\det \Omega) - \mathrm{Tr}(S\Omega) - \lambda \left\|\Omega\right\|_1 \tag{3.28}$$

où λ est le paramètre de régularisation et S est la matrice de variance-covariance empirique. Le problème est convexe et admet une unique solution, pour λ donné. Le choix de ce paramètres de régularisation est important : la régularisation gère la présence ou absence d'arêtes dans le graphe, plus la régularisation est forte, plus le graphe est sparse. A l'inverse, une pénalisation de 0 renverra un graphe complet.

Dans Friedman et al. (2008a), les auteurs proposent d'estimer la structure du graphe en résolvant une succession de problèmes de type lasso, jusqu'à convergence.

Choix de la pénalité. Le choix du paramètre λ est crucial pour déterminer le réseau estimé. On s'appuie sur les critères *Bayesian Information Criterion* (BIC) (Schwarz, 1978) et *Extended Bayesian Information Criterion* (EBIC) (Foygel and Drton, 2010), dans la suite. On note $\mathcal{E}$ l'ensemble des arcs du graphe considéré, ici les arcs estimés par Glasso, et présents dans $\hat{\Omega}$.

$$\mathrm{BIC}_\lambda = |\mathcal{E}| \log(n) - 2 \log L, \tag{3.29}$$

où $|\mathcal{E}|$, le degré de liberté, est le nombre d'arcs (éléments différents de 0) dans le graphe, n la taille des données et L la vraisemblance obtenue avec le niveau de pénalisation λ. Dans Chen and Chen (2008), les auteurs avancent que le BIC, dans le contexte des réseaux, où le nombre de variables (les arcs) à sélectionner est très important, a tendance à privilégier des modèles avec un nombre d'arêtes trop élevés par rapport à la réalité. Dans Foygel and Drton (2010), les auteurs proposent une adaptation du BIC, plus stricte :

$$\mathrm{EBIC}_{\lambda,\gamma} = \mathrm{BIC}_\lambda + 4\gamma|\mathcal{E}| \log(p), \tag{3.30}$$

où p est le nombre de nœuds (variables). Le paramètre γ de l'EBIC gère la parcimonie du modèle sélectionné : de plus grandes valeurs de γ conduiront à sélectionner un modèle plus parcimonieux. Un paramètre γ à 0 ramène au BIC classique.

3.2 Combinaison des méthodes GREMLIN et Glasso

L'algorithme simplifié utilisé est présenté en Algorithme 2 page 51. La méthode est implémentée dans le package `R janine` (*Just Another Network INference mEthod*) (Chiquet, 2020). Cet algorithme associe l'estimation globale du graphe pondéré (Glasso) avec l'estimation des groupes intra-tables (GREMLIN). La combinaison de ces deux méthodes revient à associer un niveau de pénalité globale, désigné par λ, traditionnellement utilisé dans le Glasso, à une matrice de pénalité provenant des probabilités de connexion estimées par GREMLIN, qui aura donc une structure par blocs, à la fois déterminée par la partition *a priori* des variables, et par la sous-partition trouvée par GREMLIN. On cherche à ajuster la pénalité de façon à moins pénaliser les connexions intra-blocs qu'inter-blocs. Lorsque l'on connaît une partition des données, pénaliser toutes les arêtes de la même manière n'a en effet pas forcément de sens. Le terme de pénalisation du problème d'optimisation 3.28 est légèrement

Algorithme 2 janine, à pénalité λ donnée et partition en groupes fonctionnels donnée

Input : Données $X_{n\times p}$ gaussiennes,
G, vecteur des groupes fonctionnels sur les colonnes de X
λ, pénalité globale pour le Glasso,
ϵ, seuil de convergence,
n_{iter}, nombre maximum d'itérations
Output : $\hat{\Omega}_{p\times p}$ une estimation du réseau des variables de X, avec une structure par bloc
G_{intra} partition affinée de G en sous-groupes
$\Pi_{p\times p}$ Probabilités de connexion entre les nœuds

1: S matrice var-cov empirique de X
2: $i \leftarrow 1$
3: Définir matrice de poids $W_{p\times p}$ telle que $W_{ij} = 1$ pour tout i et j dans $\{1, ..., p\}$
4: $objectif[1] \leftarrow \infty$ {Calcul du problème 3.31}
5: $Cond \leftarrow FALSE$
6: **while** !$Cond$ et $i \leq n_{iter}$ **do**
7: $\hat{\Omega} \leftarrow GLASSO(X, S, \lambda * W)$ {Estimation du réseau global par Glasso}
8: Définir A telle que $A_{ij} = 0$ si $\hat{\Omega}_{ij} = 0$, 1 sinon, $A_{ii} = 0$. {Matrice d'adjacence binaire associée à $\hat{\Omega}$}
9: $sparsity \leftarrow 1 - \frac{\|A\|_1}{p^2}$
10: $(\Pi_{p\times p}, G_{intra}) \leftarrow GREMLIN(A, G)$ {π_{ij} probabilité d'une connexion entre les nœuds i et j}
11: $W \leftarrow (1 - \Pi)/sparsity$
12: $objectif[i+1] = \log(\det \hat{\Omega}) - \mathrm{Tr}(S\hat{\Omega}) - \left\|\lambda * \hat{\Omega} * W\right\|_1$
13: **if** $(objectif[i+1] - objectif[i]) \leq \epsilon$ **then**
14: $Cond = TRUE$
15: **end if**
16: iter = iter ++
17: **end while**

transformé en :

$$\max_{\Omega} \ \log(\det \Omega) - \mathrm{Tr}(S\Omega) - \|\lambda * \Omega * W\|_1 \,, \tag{3.31}$$

l'opérateur $*$ dénotant le produit terme à terme entre deux matrices. On inclut dans la résolution un niveau de pénalisation λ général à tout le graphe, et une pénalisation W adaptée à chaque arête, dépendante de l'appartenance des nœuds aux divers groupes estimés par GREMLIN.

Pour obtenir une estimation stable du graphe, on introduit une itération jusqu'à convergence de la quantité définie à l'itération i comme $\log(\det \hat{\Omega}^{(i)}) - \mathrm{Tr}(S\hat{\Omega}^{(i)}) - \left\|\lambda * \hat{\Omega}^{(i)} * W^{(i)}\right\|_1$, soit le problème définit à l'équation (3.31).

L'algorithme présenté en Algorithme 2 est à λ donné pour le Glasso. En pratique, une grille de pénalités doit être étudiée, pour choisir la pénalité adéquate. A chaque itération, le nombre de groupes est sélectionné par ICL dans la partie GREMLIN de l'algorithme. La meilleure estimation $\hat{\Omega}$ de la matrice de précision est sélectionnée parmi les estimations des différentes pénalités sur la base de l'EBIC ou du BIC.

Le fait d'utiliser une combinaison de pénalisation ℓ_1 et de SBM, pour inférer un réseau a été proposé précédemment (Ambroise et al., 2009; Marlin and Murphy, 2009), de même que l'utilisation d'un Glasso et d'une adaptation de la pénalité en fonction des blocs, par exemple dans Hosseini and Lee (2016), où les auteurs utilisent une adaptation du Glasso et du SBM pour estimer une structure

comprenant des blocs se chevauchant.

Lorsque le vecteur des groupes fonctionnels n'est pas connu, ou qu'il n'y a pas de partition *a priori* des données, on se ramène à une combinaison d'un Glasso avec un SBM classique, itéré plusieurs fois pour avoir une certaine stabilité. Ce processus est aussi présent dans la littérature (Pircalabelu and Claeskens, 2020).

Notre méthode introduit le fait que l'on puisse utiliser une partition *a priori* des données, permettant de fixer des groupes d'intérêt, que l'on veut conserver, pour observer la structure du réseau formée selon ces blocs.

3.3 Performances de la méthode

On se compare dans cette section aux résultats obtenus en utilisant un Glasso avec pénalité commune à tous les nœuds et à une alternance entre le Glasso et un SBM classique, nommée GlassoBM, sans *a priori* sur la partition des nœuds. La combinaison Glasso et GREMLIN sera notée GlassoGREMLIN.

Le package `huge` a été utilisé pour le Glasso. Pour les combinaisons GlassoBM et GlassoGREMLIN, le package `janine` (Chiquet, 2020) a été utilisé, l'un sans entrer de partition *a priori*, l'autre avec une partition. Ce package utilise les packages `GREMLIN` (Bar-Hen et al., 2018) et `blockModels` (Leger, 2016) pour les différentes estimations. Pour chacune des trois méthodes, on choisit les modèles retenus par EBIC, avec $\gamma = 0.5$.

Le Glasso estime à chaque itération i une matrice de précision, $\Omega^{(i)}$. L'estimation du Glasso comporte l'estimation de la diagonale de cette matrice. On définit la matrice de support $A^{(i)}$ par :

$$\begin{cases} A_{ij}^{(i)} = 1 & \text{si } \Omega_{ij}^{(i)} \neq 0 \text{ et } i \neq j \\ A_{ij}^{(i)} = 0 & \text{sinon.} \end{cases} \tag{3.32}$$

On considère dans toutes les matrices de support tracées que la diagonale est à 0 (il n'y a aucune boucle dans le réseau). L'estimation du Glasso se fait cependant avec la diagonale incluse.

3.3.1 Données simulées

On simule un jeu de données à partir d'un *Multipartite Block Model* (MBM) : la matrice de précision des données Ω est simulée avec 60 nœuds, répartis en 3 grands groupes de 20 nœuds chacun, les groupes fonctionnels (GF), appelés Gènes, Métabolites (Met) et Protéines (Prot) pour l'exemple. Une sous-partition en 3, 2 et 2 sous-groupes respectivement a été simulée. Les groupes sont de tailles différentes, avec respectivement 7, 8, 5, 7, 13, 10 et 10 nœuds. Les données sont donc divisées en 7 groupes, dont 3 majeurs (GF) qui seront indiqués à GREMLIN.

Les probabilités de connexion ont été fixées à 0.81 pour les connexions intra-blocs, et à 0.01 pour les inter-blocs, à l'exception de quelques groupes qui ont une probabilité de connexion de 0.2, comme montré sur la Figure 3.3 page 53. La matrice d'adjacence simulée est présentée sur la même figure.

Au total, 50 valeurs de pénalités globales λ ont été testées. GlassoBM cherche, pour chaque pénalité, la meilleure combinaison de groupes entre 1 et 15 groupes. La partie GREMLIN de GlassoGREMLIN cherche, pour chaque GF, entre 1 et 10 groupes. L'EBIC nous indique pour GlassoBM et GlassoGREMLIN le même niveau de pénalité globale de $\lambda = 0.1401$. Le Glasso classique indique une pénalité de $\lambda = 0.020$. On compare les matrices d'adjacence, ainsi que les matrices de précision estimées aux matrices simulées.

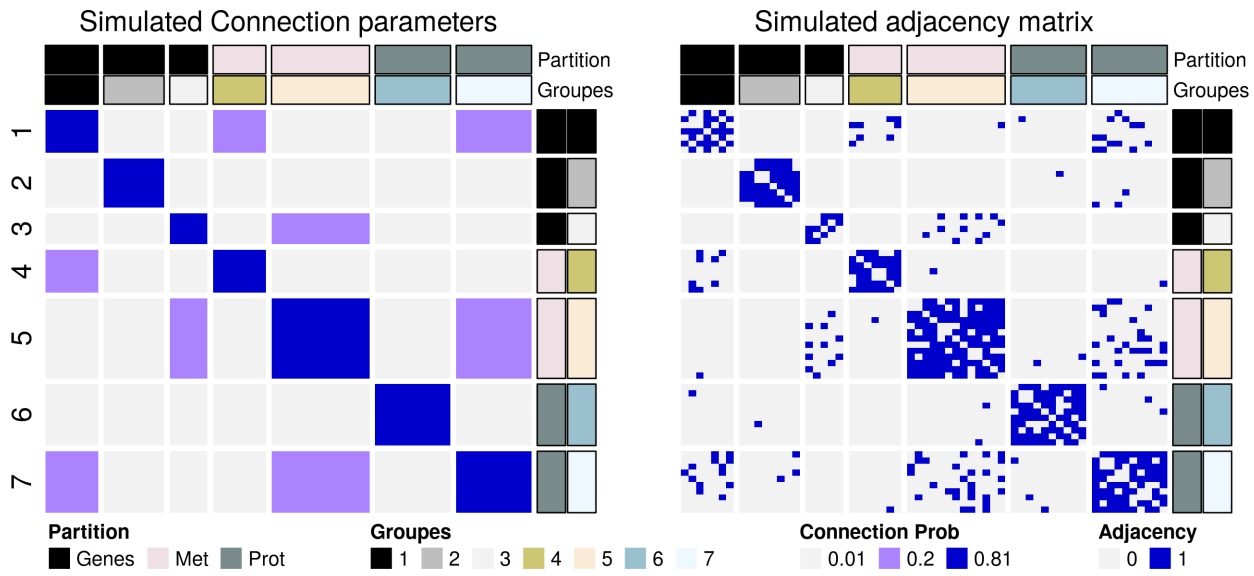


FIGURE 3.3 – Probabilités de connexion entre les blocs des données simulées et matrice d'adjacence associée.

Pour comparer les matrices de précision, on utilise la formule de distance entre matrices suivante : $\sum_{i,j,i\neq j} |\hat{\Omega}_{ij} - \Omega_{ij}|$. Les matrices d'adjacence sont comparées à l'aide du nombre d'arcs Vrais Positifs (*True Positives*, tp), Faux Positifs, (*False Positives*, fp), et de leurs taux associés, ainsi que du *True Discovery Rate* (tdr). En notant $A^{(t)}$ et $A^{(e)}$ les matrices d'adjacence réelle et celle estimée, on a $n_{arcs}^{(t)} = \frac{1}{2}\sum_{i,j} A_{ij}^{(t)}$, le nombre d'arcs de la réalité, $n_{arcs}^{(e)} = \frac{1}{2}\sum_{i,j} A_{ij}^{(e)}$, le nombre d'arcs dans le réseau estimé, et $n_0 = \frac{1}{2}\sum_{i,j}(\mathbb{1}_{A_{ij}^{(t)}=0})$, le nombre d'absence d'arcs dans la réalité, on définit tp, fp, tpr, fpr et tdr par les formules suivantes :

$$fp = \frac{1}{2}\sum_{i,j} \mathbb{1}_{(A_{ij}^{(e)} - A_{ij}^{(t)})=1}, \qquad tp = n_{arcs}^{(e)} - fp,$$
$$fpr = \frac{fp}{n_0}, \qquad tpr = \frac{tp}{n_{arcs}^{(t)}}, \qquad tdr = \frac{tp}{n_{arcs}^{(e)}} \tag{3.33}$$

La *sparsity* (taux de parcimonie du graphe) d'un réseau est quant à elle définie comme : $1 - 2n_{arcs}/(p(p-1))$ en notant p le nombre de nœuds.

Le Tableau 3.1 page 55 récapitule les caractéristiques des différentes matrices estimées, en comparaison avec les matrices simulées. Le graphe simulé comporte 268 arcs, et a une *sparsity* de 0.85. Le Glasso estime un nombre d'arcs bien trop important, mais trouve tous les arcs simulés. À l'inverse, les deux méthodes GlassoBM et GlassoGREMLIN trouvent un nombre d'arcs s'approchant plus de la réalité, mais ne trouvent pas tous les arcs simulés. Le nombre d'arcs faux positifs est faible devant le taux de vrais positifs, et GlassoGREMLIN trouve plus d'arcs correspondant à la réalité que GlassoBM, ce qui se retrouve dans le score de *tpr*, avec un taux de 0.75 pour GlassoBM et 0.84 pour GlassoGREMLIN. Concernant les performances de la classification, le fait d'introduire une classification *a priori* permet d'obtenir une estimation des groupes beaucoup plus en accord avec la simulation, avec un ARI de 0.92 pour GlassoGREMLIN et de 0.68 pour GlassoBM.

La Figure 3.4 page 54 montre les matrices d'adjacence estimées par les différentes méthodes, ainsi que les partitions estimées par GlassoBM et GlassoGREMLIN. Les arcs présents sur les blocs diagonaux sont globalement bien retrouvés, et les différents groupes simulés sont pour la plupart retrouvés. GlassoBM regroupe les deux premiers groupes des Genes et Métabolites, et associe quelque

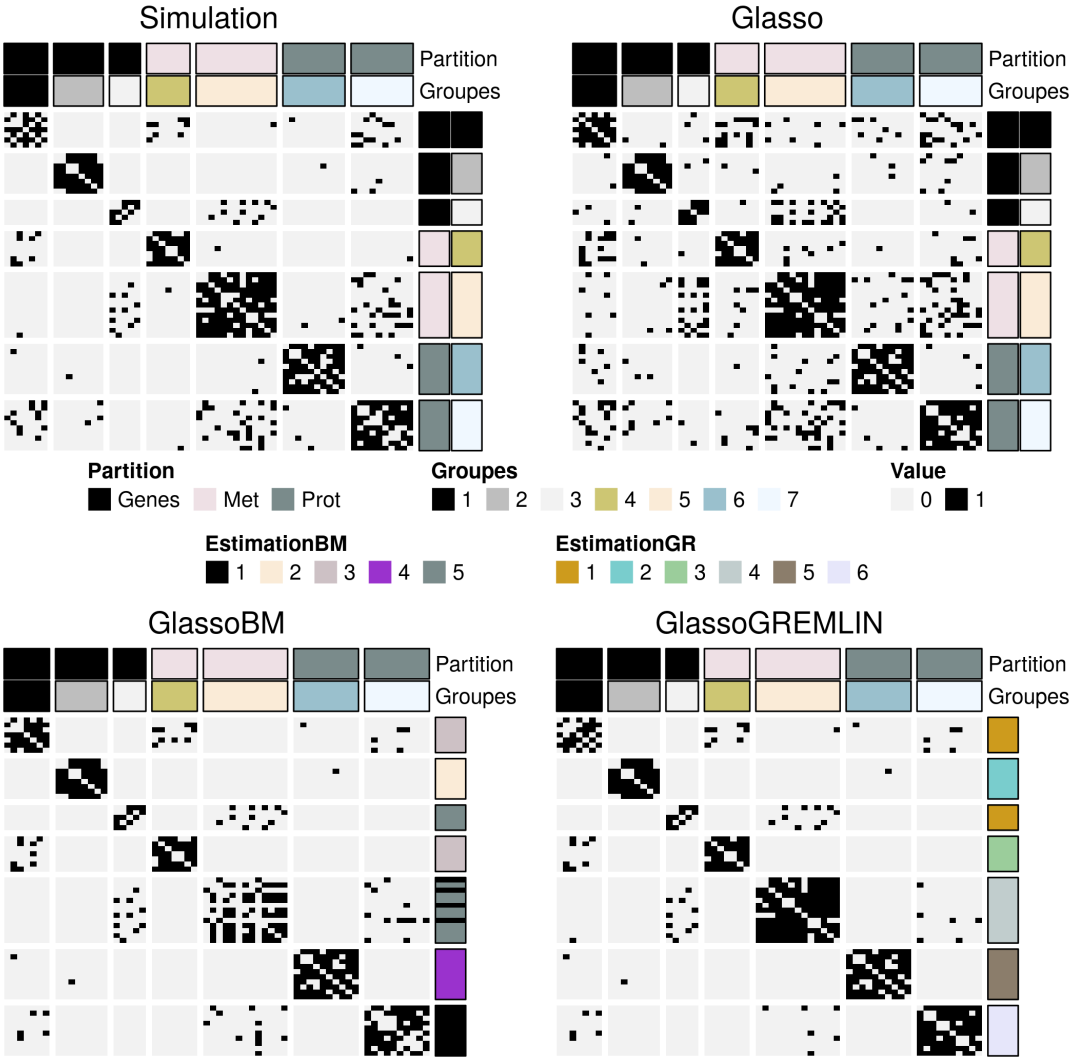


FIGURE 3.4 – Matrices d'adjacence estimées par les différentes méthodes. Pour GlassoBM et GlassoGREMLIN, les nœuds ordonnés selon leurs groupes simulés. La diagonale des matrices estimées a été mise à 0 (aucune boucle).

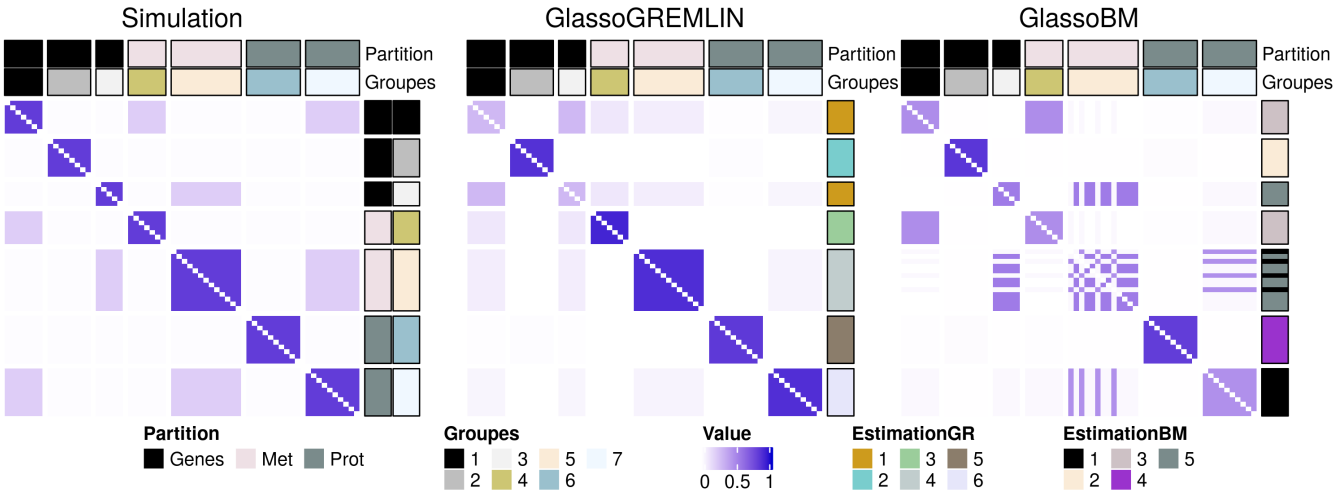


FIGURE 3.5 – Probabilités de connexion estimées par GlassoBM et GlassoGREMLIN. Les nœuds sont ordonnés selon leurs groupes simulés.

	arcs	tp	fp	tpr	fpr	tdr	distMatrix	*sparsity*	nbGroupes	ARI
Glasso	378	268	110	1	0.07	0.71	4.56	0.79	-	-
GlassoBM	208	200	8	0.75	0.03	0.96	15.66	0.88	5	0.68
GlassoGREMLIN	238	225	13	0.84	0.03	0.95	12.40	0.87	6	0.92

Tableau 3.1 – Comparaison des matrices estimées et simulées. La *sparsity* du réseau simulé est de 0.85, avec 268 arcs.

métabolites à un groupe de protéines. Le fait que GlassoGREMLIN prenne une partition supérieure des données permet de trouver tous les groupes simulés, à l'exception de deux des groupes des Gènes, qui sont confondus.

La Figure 3.5 page 54 montre les probabilités de connexion entre les différents blocs, estimées par GlassoBM ou GlassoGREMLIN. Les deux méthodes attribuent des probabilités de connexion plus fortes là où il n'y avait pratiquement pas de connexion dans la simulation, ce qui correspond bien au fait qu'il y ait des arcs trouvés en dehors des arcs simulés. Les probabilités de connexion estimées sur les blocs diagonaux sont, à l'inverse, plus faibles que ce qui a été simulé.

Les deux méthodes ont des performances similaires, mais cette simulation démontre l'intérêt que l'on peut avoir à fixer une partition *a priori* des nœuds pour estimer au mieux des groupes intérieurs.

On peut remarquer que l'estimation par Glasso a une distance plus réduite avec la matrice simulée que les deux autres estimations : la matrice de précision estimée comporte plus d'arêtes, mais celles-ci sont de faibles valeurs. Elles ont tout de même été sélectionnées par l'algorithme. La Figure 3.6 page 56 montre la matrice de précision simulée et les trois estimations qui en ont été faites. La matrice Glasso est visuellement plus proche, mais comporte beaucoup d'arêtes de petites intensités qui n'ont pas été sélectionnées par les deux autres méthodes. Les matrices estimées par GlassoBM et GlassoGREMLIN sont par contre de moindre intensité sur la totalité des arêtes trouvées.

3.3.2 Données réelles

On applique les trois méthodes utilisées dans les simulations, Glasso, GlassoBM et GlassoGREMLIN, à un jeu de données réelles. On considère dans cette sous-partie le jeu de données utilisé dans l'article présenté au chapitre 2, provenant du site de *The Cancer Genome Atlas* (TCGA). Pour cette application, trois tables de données sont retenues, les miRNA ($p = 725$), RNA-seq ($p = 19\ 738$) et protéines ($p = 156$). Les données concernent 104 patientes atteintes de cancer du sein, qui peuvent être réparties en quatre sous-types : Basal-like ($n = 22$), HER2-enriched ($n = 18$), Luminal A ($n = 44$) et Luminal B ($n = 20$).

Les deux tables de données de RNA-seq et miRNA comportant un nombre important de variables, on fait une première sélection des 500 variables par ordre décroissant de variance. Ensuite, pour chacune des tables, une analyse différentielle par `limma` (**?**) est faite, cherchant les variables différentiellement epxrimées pour tous les contrastes entre sous-types de cancers. Les 15 variables les plus globalement différentiellement exprimées sont sélectionnées.

On a 45 nœuds, répartis en trois groupes fonctionnels, avec des observations sur 104 individus pour estimer le réseau.

La Figure 3.7 page 57 présente les matrices d'adjacence estimées par les trois méthodes Glasso, GlassoBM et GlassoGREMLIN, ordonnées selon leur table d'appartenance (miRNA, Protein et RNA-seq) (Figure 3.7 (a)) et selon les groupes estimés par GlassoBM (Figure 3.7 (b)). Les réseaux sont assez denses, avec une *sparsity* de 0.67, 0.75 et 0.72, respectivement, et un total d'axes estimés de 338, 258 et 288. On retrouve un des résultats des simulations dans le fait que Glasso estime

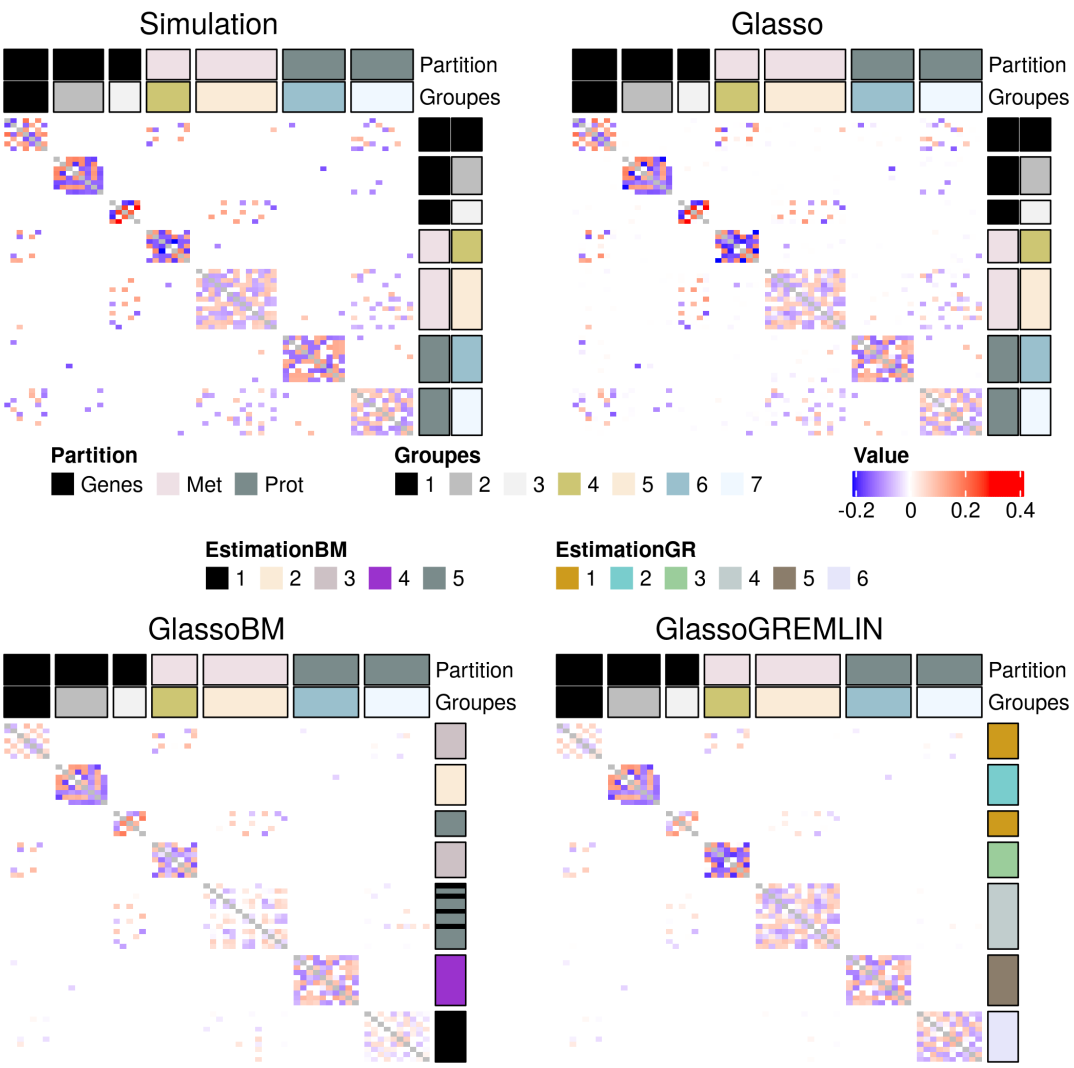


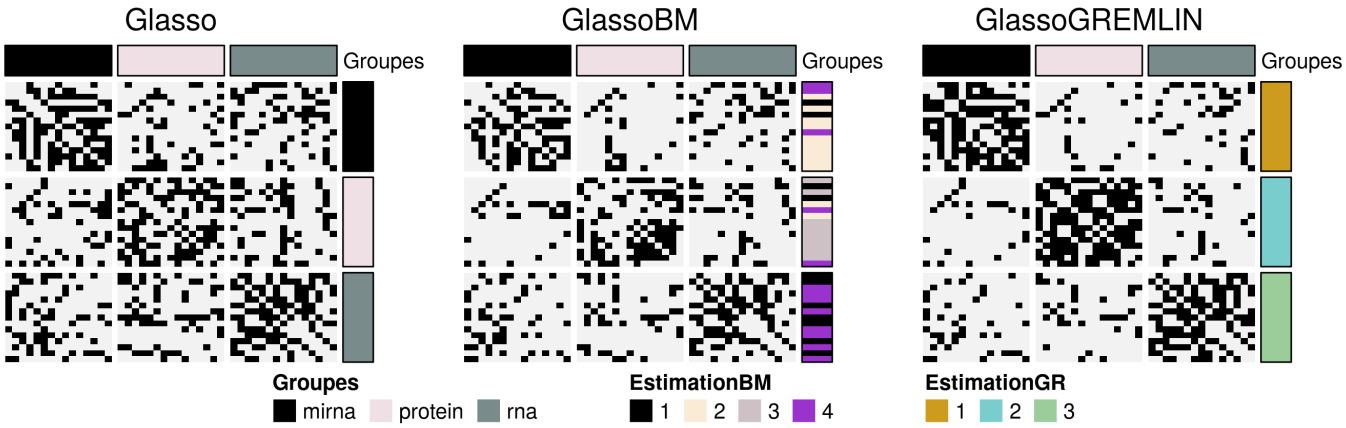
FIGURE 3.6 – Matrices de précision simulées et estimées pour les simulations. Diagonales mises à NA pour le tracé.

beaucoup plus d'arcs que les deux autres méthodes et GlassoBM renvoie l'estimation la plus sparse des trois. La Figure 3.8 page 57 montre un diagramme de venn des arêtes des différentes matrices estimées. La majorité des arêtes trouvées par les méthodes sont communes à toutes les méthodes.

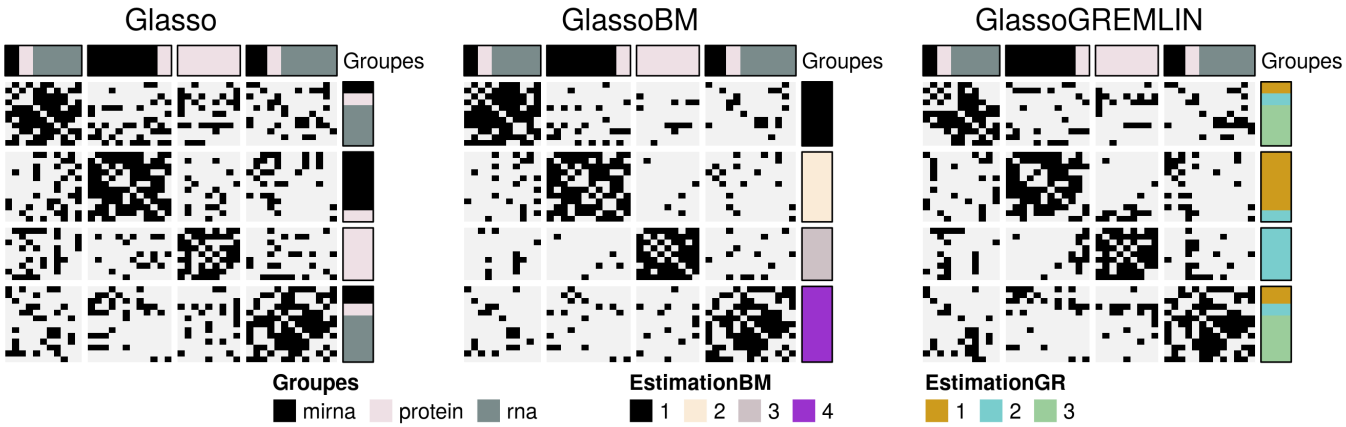
En ce qui concerne les groupes, la méthode GlassoGREMLIN ne trouve pas de sous-groupes dans aucune des tables et renvoie la classification d'entrée. Cette méthode estime des probabilités de connexion plus importantes intra-blocs que GlassoBM. A l'inverse, la méthode GlassoBM trouve 4 groupes distincts qui ne reflètent pas les tables originales, sauf pour un groupe constitué de protéines seulement. Si à première vue on pourrait penser que les matrices estimées diffèrent, l'agencement de la matrice d'adjacence GlassoGREMLIN selon les groupes de GlassoBM montre que leur estimation est en fait ressemblante : les connexions intra-groupes sont très importantes.

3.4 Conclusion

Dans ce chapitre, nous avons présenté une méthode d'estimation de réseaux via une combinaison de Glasso et de GREMLIN (GlassoGREMLIN). Cette méthode permet d'estimer le réseau à partir de données, tout en estimant des clusters de nœuds, en prenant en compte une appartenance *a*



(a) Ordre des nœuds selon leur table d'appartenance.



(b) Ordre des nœuds selon les groupes trouvés par GlassoBM.

FIGURE 3.7 – Matrices d'adjacence estimées dans l'application sur données réelles TCGA.

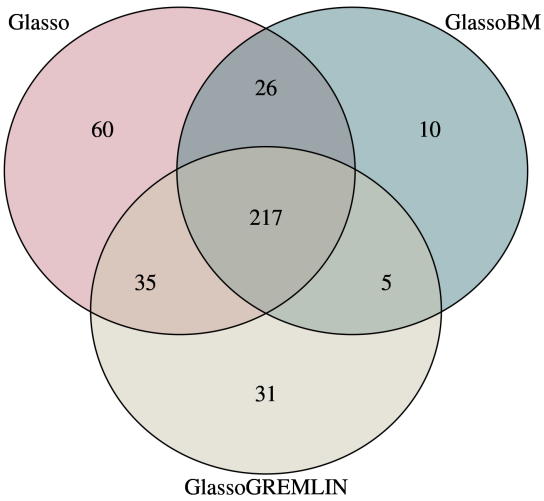


FIGURE 3.8 – Comparaison des arêtes des matrices d'adjacences estimées, pour l'application sur données réelles TCGA.

priori des nœuds. Nous avons appliqué cette combinaison à un jeu de données simulées, ainsi qu'à un jeu de données réelles concernant des patientes atteintes de cancer du sein.

Dans les deux applications, la méthode a été comparée à l'estimation trouvée par un Glasso simple, ainsi que par la combinaison d'un Glasso et d'un SBM (GlassoBM) classique itérés, sans *a priori* sur les groupes des nœuds. Les résultats montrent que GlassoGREMLIN permet d'obtenir une partition plus fine des données, sans perdre la qualité de l'estimation de la matrice d'adjacence du réseau. Les matrices d'adjacence obtenues par GlassoGREMLIN et GlassoBM comportent en effet moins d'arêtes fausses en comparaison de la matrice estimée par Glasso. En revanche, Glasso estime mieux la matrice de précision : les arêtes fausses positives sont en effet d'intensité proche de 0.

La méthode GlassoGREMLIN a montré un intérêt certain et des performances tout à fait correctes dans l'estimation de réseau et de blocs. Des pistes d'améliorations peuvent être envisagées, notamment au niveau d'un modèle d'estimation continue : le SBM utilisé ainsi que GREMLIN estiment tous deux des matrices binaires, le Glasso est pour l'instant la seule méthode permettant d'estimer la matrice de précision, ce qui peut être à l'origine de la baisse de performance dans l'estimation de la matrice de précision. Il peut aussi être intéressant, en particulier dans le contexte de données -omiques, de s'intéresser à une estimation de réseaux MBM à partir de données de comptages.

Chapitre 4

Intégration de données via l'analyse factorielle multiple

Table des matières

L'article présenté dans cette section a été soumis à *PLOS One*.

Résumé. L'intégration de données provenant de différentes sources est un problème récurrent en bio statistique et bio informatique. La plupart des recherches statistiques se sont tournées vers la résolution du problème de l'intégration de données du même types, souvent des tables de données continues. Cependant, les données à traiter sont souvent hétérogènes : il n'est pas rare d'avoir à disposition des données sous la forme d'arbres, de réseaux, ou de cartes factorielles, ces représentations étant fortement appréciées pour la visualisation des données et l'étude de groupes ou d'interactions entre les différentes entités. Dans ce papier, nous nous intéressons à l'intégration de ces différentes représentations.

Nous proposons une procédure simple qui permet de comparer des données de différents types, en particulier les arbres et les réseaux, en deux étapes : la première étape trouve un système de coordonnées communs dans lequel projeter les différentes représentations ; la seconde étape utilise une méthode d'intégration multi-table pour comparer les projections obtenues. Ces deux étapes utilisent des méthodes déjà connues et efficaces : la projection est obtenue en transformant les objets en matrices de dissimilarités ou distances, puis en appliquant du Multidimensional Scaling (MDS), qui fourni un nouveau jeu de coordonnées à partir de ces dissimilarités. L'étape d'intégration est obtenue quant à elle en effectuant une Analyse Factorielle Multiple (AFM) sur ces nouvelles coordonnées. Cette procédure permet de comparer et intégrer des jeux de données, notamment des

arbres et des réseaux. En comparaison des méthodes à noyaux, qui sont utilisées dans le même esprit d'intégration de données sous différentes formes, notre approche est plus facile à utiliser et permet l'utilisation de toutes les méthodes de visualisations et d'interprétations qui sont utilisées sur les résultats d'une AFM.

La procédure présentée est évaluée sur des données simulées ainsi que des données issues d'études : on compare d'abord des clusterings de gènes réalisés pour différents types de cellules, les données provenant d'une étude single-cell sur des embryons de souris ; dans un second temps on intègre des données -omiques provenant de patients atteints de cancer du sein, dans le but de comparer différents réseaux de protéines.

Abstract. Integrating data from different sources is a recurring question in computational biology. Much effort has been devoted to the integration of data sets of the same type, typically multiple numerical data tables. However, data types are generally heterogeneous: it is a common place to gather data in the form of trees, networks or factorial maps, as these representations all have an appealing visual interpretation that helps to study grouping patterns and interactions between entities. The question we aim to answer in this paper is that of the integration of such representations.

To this end, we provide a simple procedure to compare data with various types, in particular trees or networks, that relies essentially on two steps: the first step projects the representations into a common coordinate system; the second step then uses a multi-table integration approach to compare the projected data. We rely on efficient and well-known methodologies for each step: the projection step is achieved by retrieving a distance matrix for each representation form and then applying Multidimensional Scaling (MDS) to provide a new set of coordinates from all the pairwise distances. The integration step is then achieved by applying a Multiple Factor Analysis (MFA) to the multiple tables of the new coordinates. This procedure provides tools to integrate and compare data available, for instance, as tree or network structures. Compared to multiple kernel methods that could be used to answer the same question, our approach remains easier to use as it requires very little tuning, and provides the appealing toolkit of data analysis via MFA that eases the interpretation.

Our approach is evaluated on simulation and used to analyze two real-world data sets: first, we compare several clusterings for different cell-types obtained from a transcriptomics single-cell data set in mouse embryos; second, we use our procedure to aggregate a multi-omic data set from the TCGA breast cancer database, in order to compare several protein networks inferred for different breast cancer subtypes.

4.1 Introduction

When integrating data in computational biology, we are often confronted with the problem of comparing outcomes from different types of data, with various forms of representation (Gligorijević and Pržulj, 2015; Mariette and Villa-Vialaneix, 2017; Li et al., 2018). These representations may either result from a learning algorithm (e.g. dimension reduction, hierarchical clustering or network inference) or they may be extracted from a data base, reflecting our knowledge about a complex biological process.

As a simple example in genomics, several hierarchical clusterings of individuals can be obtained based on transcriptomics, proteomics or metagenomics experiments, giving birth to several tree-like representations which need to be compared and eventually aggregated. Such an analysis is essential to better understand the data and to obtain a consensus clustering from coherent trees. In fact, it is important to be able to compare these trees and to quantify how similar or different they are

before trying to aggregate them. Similarly in the context of regulatory network analysis, several networks associated with the same genes can be inferred from gene expression data gathered in different tissues. It is of crucial biological interest to be able to compare them to highlight their differences and similarities.

The question of comparing hierarchical clusterings and networks, as well as creating consensus, is recurrent in the literature: (Tantardini et al., 2019) provides a detailed review about methods existing for comparing networks. As for the clusterings, comparison of a set of trees often relies on distances between trees, for example using Robinson-Foulds metric (Robinson and Foulds, 1979, 1981) as in phylogenetics. However, the comparison of clusterings or networks has been treated as different questions: the procedure that we introduce in this paper answers both questions simultaneously, and can be applied to a broader variety of data representation than just tree or network structures. In this sense, our procedure is similar to unsupervised multiple kernel methods, which is a powerful, general way for performing integration of heterogeneous data (Schölkopf et al., 2004; Zhuang et al., 2011; Mariette and Villa-Vialaneix, 2017). Multiple kernel learning requires that the user chooses appropriate kernels to define the data. When heterogeneous data are considered, adjusting one kernel per type of data, their parameters and their weight for the integration, may be demanding in terms of computation time and memory. We define here a procedure where everything is automated for the integration process, as the user has very few, if none, parameter to define.

In a nutshell, the contribution of this paper is a unified and simple way of integrating data with various forms of representation (like trees, networks or factorial maps), that relies on a two-step strategy which philosophy is close to unsupervised multiple kernel: the first step consists in finding a way to project all these objects into a comparable coordinate system. This leads to new collection of data tables which are analyzed in a second step by means of any multi-table integration method. The specificity of our approach is to combine Multidimensional scaling (MDS) (Torgerson, 1958; Borg and Groenen, 2005) and Multiple Factor Analysis (MFA) (Escofier and Pages, 1994; Abdi et al., 2013; Rau et al., 2019) to perform these two steps: the MDS allows us to calculate coordinates from distances or dissimilarities, obtained from trees, networks or factorial maps.

Then, MFA provides a canonical framework to perform multi-table analysis, bringing powerful tools to study the relationships between tables of data, and to quantify the similarities and differences between them.

Our procedure is particularly useful in the case where we are given a set of trees or networks, or any object set we want to compare, without the original data. For example, networks of protein-protein interaction or ecological networks are available on databases without any indication of the data they have been built on (Szklarczyk et al., 2019; Fortuna et al., 2014; Poisot et al., 2016). This can also be useful to compare different ways to transform the data, e.g. using different distances or aggregation criteria to build the trees.

The rest of the paper is organized as follows: first, we give details about the proposed methodology. Then its performance is evaluated on simulated data, and two real-world data sets are analyzed: the first one is a single-cell data set that illustrates the comparison of clusterings for different cell-types in mouse embryos. The second one is a -omic data set from the TCGA breast cancer database, for which several PPI networks are compared and aggregated for different breast cancer subtypes.

4.2 Methods

In order to compare and aggregate trees or networks, in the context of multi-source data analysis, we adopt the general 2-step approach:

1. **Projection.**
   (a) Represent all data sources in the form of distance or dissimilarity matrices
   (b) Place these distances in the same coordinate system
2. **Integration.**
   (a) Apply multi-table analysis
   (b) Use factorial representation for comparing the projected data and creating a consensus

Step 1 is done by retrieving a distance matrix specific to either trees or networks (see details below) and then applying Multidimensional Scaling (MDS), which provides a new set of coordinates from all these pairwise distances. These new coordinates can be interpreted the same way as original multi-source data and all methods available for the analyses of such data sets can be used for Step 2 (integration). We chose Multiple Factor Analysis (MFA), which allows us to position the different objects on a factorial map.

Any object that can be summarised in the form of a dissimilarity matrix can be integrated using the two steps. We would like to point out that any data, categorical or quantitative (original data, factorial maps, clinical outcome...), as long as it is computed on the same individuals, can be integrated with the others in step two.

This provides the identification of clusterings (or networks), that have similar patterns across various conditions, or data types, and to aggregate them by performing a consensus on these identified sub-groups of data using the individual coordinates of the MFA.

4.3 A common representation for trees and networks

This section details the different ingredients used in the method explained above: we explain how the distance matrices can be retrieved when focusing on network or tree structures, although any object that can be represented by a distance or dissimilarity matrix can be used in our procedure. MDS and MFA are also presented in more details, as well as the different steps of the procedure.

4.3.1 Retrieving a distance matrix from a tree

Consider a hierarchical tree obtained with any hierarchical clustering (it can be a non-binary tree). Recall that the cophenetic distance between two leaves of a tree is the height where the two leaves or their cluster are merged. Hierarchical trees can then be summarized by a symmetrical matrix using the cophenetic distance (Sokal and Rohlf, 1962). In the context of MDS, detailed later, it is best to use Euclidean distances to avoid numerical issues while computing the coordinates. It is shown in (Pavoine et al., 2005) that the distances extracted from a ultrametric tree can always be considered as Euclidean distances. All hierarchical clusterings built on a distance and aggregation criterion are ultrametric trees, therefore applying MDS to a cophenetic matrix requires no further transformation of the matrix in this particular case.

4.3.2 Retrieving a distance matrix from a network

Consider an undirected binary graph: we suggest to build a distance matrix from this graph by means of the shortest path distance between all pairs of nodes, before applying MDS. The shortest path distance is defined as the minimum number of edges to cross to go from one node to another. The shortest path distance between two unconnected nodes is generally set to infinity. This method can also be applied to weighted graphs with positive weights, where the cost of a path is understood as the sum of weights along the edges of the path.

4.3.3 Multidimensional scaling

We will refer here to the classical Multidimensional scaling (MDS), introduced by (Torgerson, 1958). The goal of the method is to find coordinates X of data given a dissimilarity matrix Δ between individuals.

Consider a matrix of dissimilarities Δ, and Δ^2 the matrix of squared coefficients of Δ, the double-centered matrix is defined as $B = -\frac{1}{2}J\Delta^2 J$, where $J = I - \frac{1}{n}\mathbb{1}\mathbb{1}^T$ is the centering matrix. The classical scaling (GOWER, 1966; Borg and Groenen, 2005) minimizes the strain: $\left\|XX^T - B\right\|^2$ where X are the coordinates we search for. The solution can be shown to verify $X = Q_+\Lambda_+^{1/2}$ with Λ_+ being the diagonal matrix with the non negative and non-zero eigenvalues of B, and Q_+ the corresponding eigenvectors.

If Δ is an Euclidean distance matrix[1], the MDS coordinates X are actually the original coordinates up to a rotation and a translation if X is not column-centered.

Several variants of MDS exist to deal with matrices that are not positive semi-definite, such as the Cailliez' method (Cailliez, 1983), which consists in adding a positive constant to the element outside of the diagonal to make the matrix positive definite. (Lingoes, 1971) proposed a similar method by adding a constant to the squared dissimilarities and taking the square root as the modified distances. One of the simplest method is to take the square root of the distances (Legendre and Legendre, 2012).

When the dissimilarities are not produced by a distance function (metric), solutions for non-metric MDS are also available (Shepard, 1962; Kruskal, 1964). In all our applications, we chose to take only the positive eigenvalues of B when needed.

4.3.4 Multiple Factor Analysis

Multiple Factor Analysis (MFA) is a method that allows the joint analysis of several data sets with different types of data (Escofier and Pages, 1994; Abdi et al., 2013). Let $X_1, ..., X_Q$ be Q data tables, which can be quantitative or qualitative data, with $p_1, ..., p_Q$ features observed on the same n individuals. The principle of MFA is to divide each data table by its first singular value to ensure the contribution of the data sets in the first axis will be equal. Data tables are then concatenated into one and a PCA is performed on $X = [X_1, ..., X_Q]$. This step is called global PCA (gPCA) in (Abdi et al., 2013).

A great advantage of the use of MFA in integrating data is that it provides several scores to compare the different tables, as well as axis coordinates that allow the visualization of features, individuals and tables on a factorial map. In this analysis, we will use the group coordinates obtained from the MFA analysis.

Group coordinates $X_1, ..., X_Q$ can be positioned on each component using their contribution to the gPCA. Let $\tilde{X}$ be the concatenation of $X_1, ..., X_Q$ each divided by its first singular value. The gPCA step decomposes $\tilde{X}$ by singular value decomposition into $U\Lambda V^T$. V is the matrix of the loadings, its column corresponding to the variables in $\tilde{X}$. The loadings can be decomposed into subsets delimited by the number of variables in each table: $V = [V_{(1)}, ..., V_{(Q)}]$. Let λ_ℓ be the ℓth eigenvalue of the gPCA. The coordinate of table X_q in the ℓ axis is defined as:

$$\text{coord}_{q,\ell} = \lambda_\ell \times \sum_{j=1}^{p_q} V_{(q)\ \ell,j}^2 = \lambda_\ell * \text{ctrb}_{q,\ell}$$

1. According to (Gower, 1982; Dokmanic et al., 2015), a distance matrix D is an Euclidean distance matrix if $-\frac{1}{2}JD^2J$ is a positive semi-definite matrix.

with p_q being the number of variables of table X_q, and $\text{ctrb}_{q,\ell}$ the contribution of table q on dimension ℓ of the gPCA.

Using these group coordinates, we propose to create a clustering of the tables. In the following, we will use a hierarchical clustering, but any clustering method can be considered. The tables are then gathered according to their similarity and can be analyzed together within groups.

4.3.5 RV-coefficient

The RV-coefficient (Robert and Escoufier, 1976) is described by its authors as a measure of closeness of two data matrices X_q and $X_{q'}$ observed on the same individuals. It is defined by

$$RV_{q,q'} = \frac{tr((X_q X_{q'}^T) \times (X_{q'} X_q^T))}{\sqrt{tr((X_q X_q^T) \times (X_q X_q^T)) \times tr((X_{q'} X_{q'}^T) \times (X_{q'} X_{q'}^T))}} \tag{4.1}$$

The RV coefficient varies from 0 to 1, 0 meaning the data matrices share no common information. It is often used to compare and investigate the general relationship between data tables, as in STATIS (L'Hermier des Plantes, 1976; Abdi et al., 2012), a method close to MFA. The difference relies in the weights they use for the global PCA step: STATIS uses weights derived from the RV-coefficient whereas the MFA uses the singular value of each data table.

In (Josse et al., 2008), the authors argue that the RV depends on a number of factors (dimensions and covariance structure of the tables) and that a high coefficient dœs not "necessarily signify a significant relationship between the data sets". Based on this statement, we chose to build data set clustering on the MFA group coordinates, and to refer to the RV to investigate the general relationships between the data tables inside a cluster.

4.3.6 Creating a consensus from MFA results

To compute a consensus hierarchical clustering given the MFA restults, we will refer to the clusters made on the group coordinates. Let $\mathcal{T}_1, \ldots, \mathcal{T}_{k_1}$ be a group of trees defined as previously described. The same process of cophenetic distances, MDS and MFA is applied on these trees. A consensus clustering is then obtained by performing a hierarchical clustering (or any other clustering method) on the individual coordinates obtained by the MFA.

To compute a consensus graph out of the MFA results, we will use majority-rule consensus, i.e. an edge is kept in the consensus if it is present in more than half of the networks in the groups we identified. Any method to create a consensus network could be used.

4.4 Results

In this section we describe the results obtained on simulated data, in order to evaluate the performances of the proposed method, as well as on two real data sets. Analyses were performed with R 4.0.1 (R Core Team, 2020a). All code and data are available at `https://github.com/AudreH/intTreeNet`.

Hierarchical clustering was performed using Euclidean distance and Ward's aggregation criterion as implemented in the "ward.D2" option of the hclust R function (Murtagh and Legendre, 2014). All trees were transformed using the *cophenetic* base function. Using *cmdscale*, the new data coordinates from the MDS approach were retrieved. MFA was performed using the MFA function from the FactoMineR package (Lê et al., 2008).

To assess the differences between clusterings, we used the *Adjusted Rand Index* (ARI) (Vinh et al., 2010; Hubert and Arabie, 1985), which measures the agreement between two classifications.

To determine the groups in a hierarchical clustering, we used the DynamicTreeCut method as implemented in the R-package (Langfelder et al., 2007). This method identifies groups based on the structure of the tree and the distance matrix used to build the tree.

In the graph application, the shortest path distance is computed using the *distances* function of the *igraph* R-package (Csardi and Nepusz, 2006) and default parameters.

4.4.1 Simulation study in the case of clusterings

In this first set of simulations, $Q = 9$ tables with $p = 1000$ variables and $n = 100$ individuals were generated according to three different patterns of classification with $K = 4, 3$ and 5 groups for each pattern, respectively. The chosen patterns of classifications are very different, with an ARI close to 0 between them.

Observation j for individual i of table q when i is in group k follows the Gaussian distribution

$$i \in \{1, \dots, n\}, \quad j \in \{1, \dots, p\}, \quad k \in \{1, \dots, K_q\}, \quad q \in \{1, \dots, Q\}, \\ i \in k, \quad Y_{i,j}^q = \mathcal{N}(\mu_k, q^2) \tag{4.2}$$

Each observation is generated according to Eq. 4.2, with the mean depending on the group of the individual and the variance depending on the table number.

To analyze the generated data, the MFA was performed on 99×9 axes from the MDS.

Fig. 4.1 presents the hierarchical clustering obtained on the coordinates of the tables, the RV between the tables and the factorial maps of the data set. Tables with the same classification are grouped together in the hierarchical clustering, as well as on the first two axes of the MFA. The first axis differentiates the tables from the first classification from the others, the second axis differentiates the tables from classification 3 from the rest. These observations made on the group coordinates are visible in the hierarchical clustering as the level of division between elements reflects the axis on which the separation is found (*e.g.* third division in the tree separates Table 6 from its group, and is found on axis 3 of the MFA).

We now have access to new axes for the individuals, with the principal components of the MFA. The hierarchical clustering performed on the group coordinates and the classification of the tables made by DynamicTreeCut can help identify the tables that are close in terms of underlying information. The three groups of tables that we identify using DynamicTreeCut are the three groups of tables we simulated.

This approach allows to visualize and compare the different clusterings before calculating a consensus tree. In this example, it would not make sense to try to aggregate all the trees, as they have very different structures. It would be more adequate to calculate three different consensus trees, based on the three different groups of classifications.

The consensus trees can be obtained by performing a hierarchical clustering on the individual coordinates of the MFA axes. Results of the three consensus trees based on the identified sub-groups of data are presented in Fig 4.2. As expected, inside a group of tables we retrieve the original classification. We do not find any information on the other classifications. On the other hand, in the consensus tree obtained with all the tables, none of the simulated classification patterns was recovered, with a maximum ARI of 0.51 obtained for classification 1.

4.4.2 Simulation study on network data

A similar simulation setup was used for the network data: $Q = 9$ adjacency matrices with $n = 100$ were simulated according to three different classification patterns, with an ARI close to 0 between them, of $K = 4$, 3 and 5 groups respectively. The presence or absence of an edge between two nodes is generated according to Eq. (4.3), with connection probabilities depending on the group the nodes are in. We chose $\pi_{kl} = 0.05$ for $k \neq l$ and $\pi_{kk} = 0.8$.

$$\begin{aligned} i, j \in \{1, \dots, n\}, \quad k, l \in \{1, \dots, K_q\}, \quad q \in \{1, \dots, Q\}, \\ i \in k, \ j \in l \quad A^q_{i,j} = \mathcal{B}(\pi_{kl}) \end{aligned} \tag{4.3}$$

The shortest path was then computed, and transformed into new data using the MDS. Results of the MFA are shown in Fig 4.3, presenting the factorial maps as well as a clustering obtained from the MFA coordinates.

Fig 4.4 shows the majority-vote consensus obtained with the groups formed by the hierarchical clustering. The original classifications are recovered very well in the networks, as the nodes are grouped in the network according to their simulated classification. The connection probability inside a cluster is far superior to the one between groups, which is exactly what we simulated.

4.4.3 Application to single-cell data

The data we use in this section are presented by Pijuan-Sala et al. (2019). They come from 411 mouse embryos, collected at different time points, from day 6.5 to day 8.5. Transcriptome expression is available for 116,312 cells. The authors divided these cells into 37 groups that we will call cell-types.

For this application we only used the samples from the first stage (E6.5), deleted all genes that had a mean count of less than 10^{-3}, as well as genes on the Y chromosome and the xist gene. These two steps led to the analysis of 15,086 genes and 3,520 samples.

Following the procedure explained by Pijuan-Sala et al. (2019), we selected the most variable genes using the scran R package and the function modelGeneVar. In total, 318 genes were selected by taking a threshold of 0.1 for the adjusted p-value.

Samples were then divided according to their cell-type. Cell-types with only one sample were discarded. The cell-types and the number of samples for each one are presented in Table 4.1. We applied the method presented above on these data in order to obtain gene clusterings for the different cell-types, compare the trees and aggregate the most coherent ones using the MFA coordinates. First, the MDS was applied to the data set in which 317 axes were obtained for each cell-type and used for the MFA analysis.

The cell-types are then grouped in clusters using a hierarchical clustering on their coordinates. Fig 4.5 shows this hierarchical clustering, as well as the RV coefficients obtained between the groups. Using the DynamicTreeCut with minimal cluster size of 1, we define three groups of cell-types. In the heatmap, the cell-types are reordered according to the dendrogram and the groups are highlighted with a black grid.

In the supplementary data of (Pijuan-Sala et al., 2019), the authors presented a map of the cell-types for every timepoint. The map of E6.5 shows roughly three groups of cell-types: the first one consisting in Epiblast, Rostral neurectoderm, Primitive Streak, Surface ectoderm and Nascent mesoderm, the second one in ExE endoderm and Visceral endoderm and the third one of Parietal endoderm and ExE ectoderm. The samples from Rostral neurectoderm and Surface ectoderm were

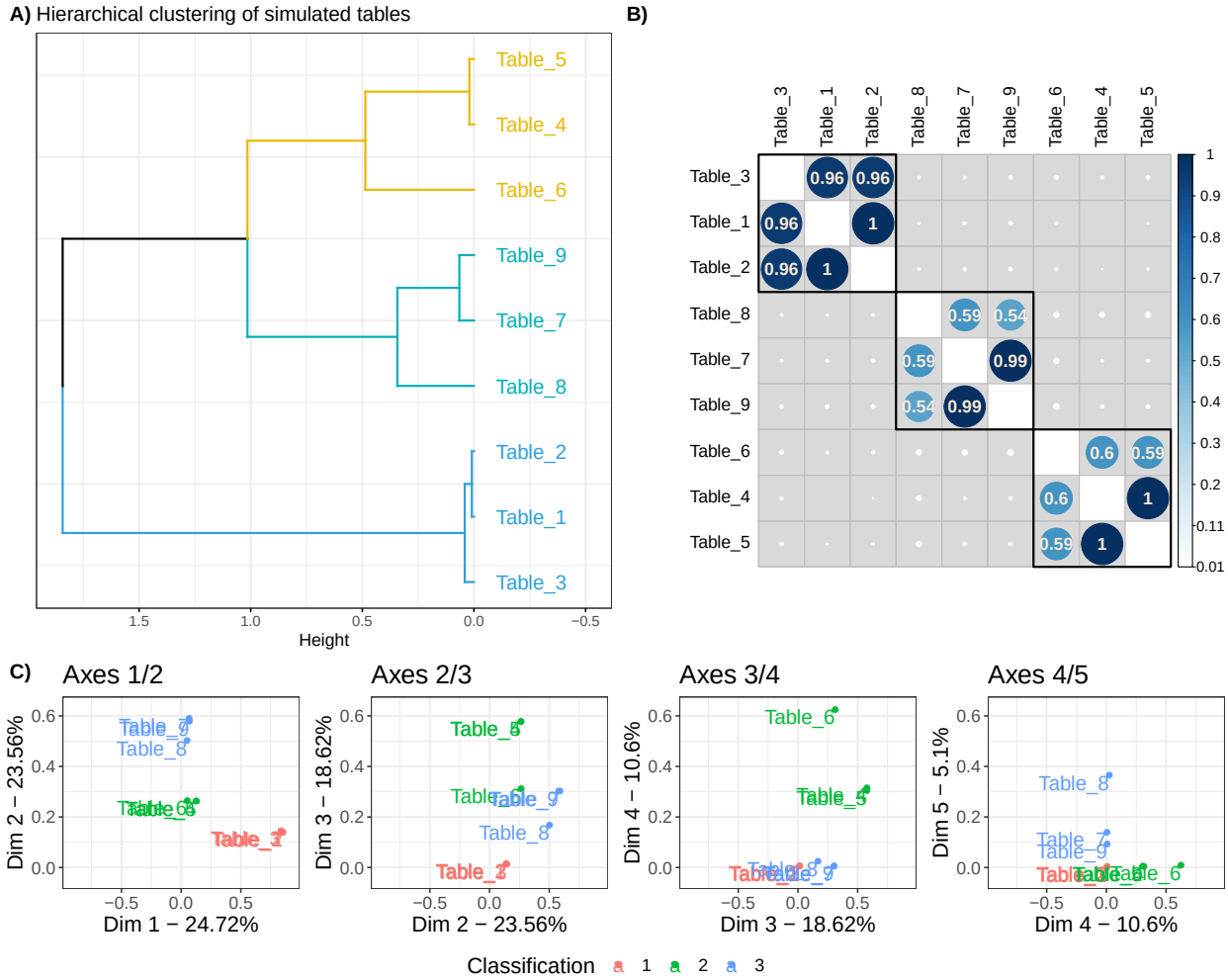


FIGURE 4.1 – **Results for the simulation study on hierarchical clustering data.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following the Gaussian in Eq. 4.2 Panel A) represents the hierarchical clustering obtained in the MFA group coordinates, performed with euclidean distance and ward.D2 aggregation criterion. Panel B) represents the RV-coefficient between the cophenetic distances tables, ordered according to the hierarchical clustering of panel A). Panel C) represents the factorial map for axis 1 to 5 of the MFA, these groups coordinates were used to compute the hierarchical clustering on panel A).

TABLE 4.1 – **Number of samples per cell-type for the single cell application.**

Group	Nb Samples
Epiblast	2276
ExE ectoderm	633
ExE endoderm	126
Nascent mesoderm	4
Parietal endoderm	10
Primitive Streak	381
Visceral endoderm	52

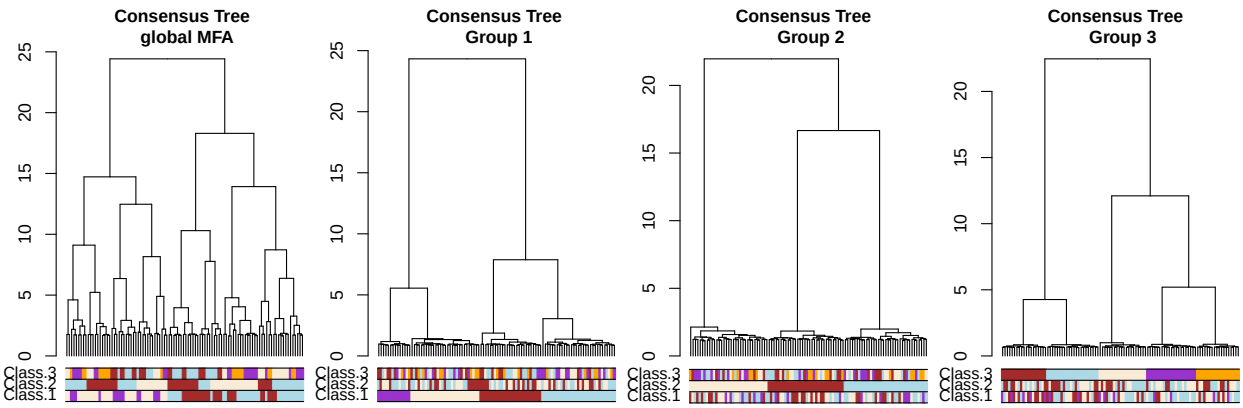


FIGURE 4.2 – **Results for the simulation study on hierarchical clustering data.** Consensus trees obtained on 4 configurations, with colored bars representing the simulated classifications.

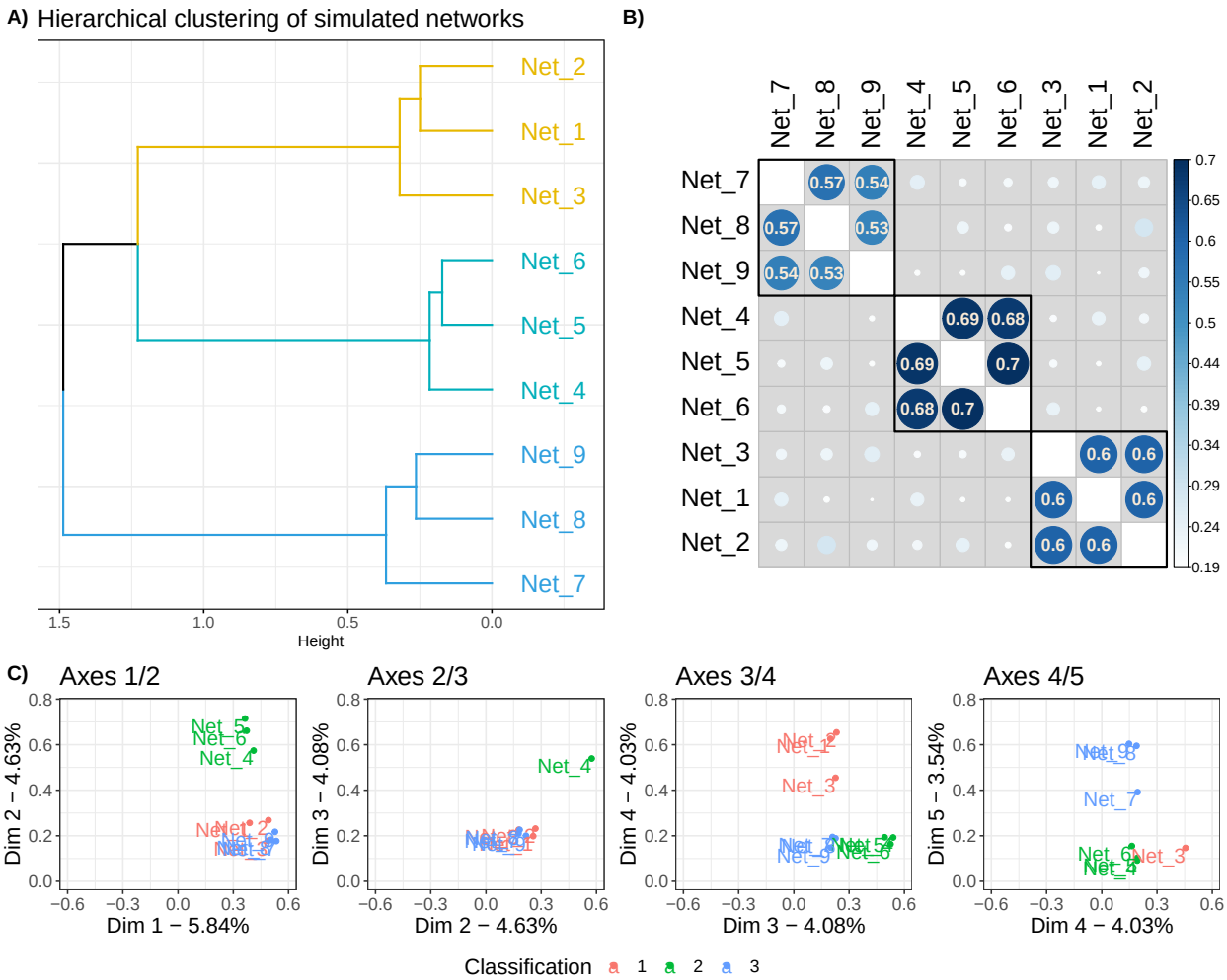


FIGURE 4.3 – **Results for the simulation study on network data.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following Eq 4.3. A) presents the hierarchical clustering of the networks based on the group coordinates of the MFA. B) presents the RV coefficients of the network data. C) shows the factorial maps for the 5 first axes of the MFA. These coordinates are the group coordinates on which A) was made.

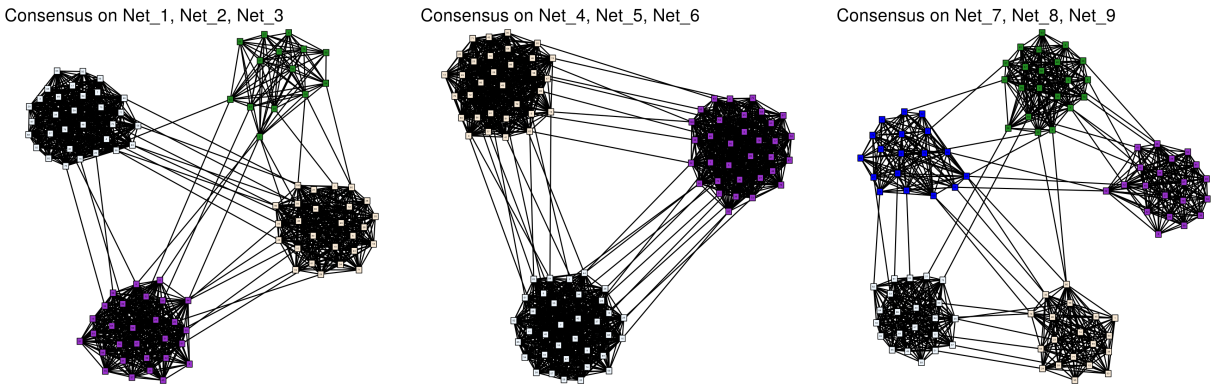


FIGURE 4.4 – **Simulations study on network data.** Consensus network obtained on the networks clusters found with MFA. Nodes are colored according to their group used for simulating the data.

discarded here as there was only one sample for each cell-type. In the clustering we obtain, the map is well refleted as the three main groups are retrieved, and the first and second groups are closer to each other than the third group.

4.4.4 Application to breast cancer data

The data used in this section are downloaded from the TCGA website. Data are protein expression from 777 patients with breast cancer, divided into 4 subtypes: Basal-like ($n = 151$), HER2-enriched ($n = 85$), Lumina A ($n = 283$), Luminal B ($n = 258$).

In this data set, $p = 173$ proteins were expressed in at least one sample of any subtype.

Using limma (Ritchie et al., 2015b) to perform a differential analysis, we selected the 5 first proteins by order of adjusted p-value, for each contrast between subtypes, which provided 15 unique proteins. Networks associated with each subtype were inferred using glasso (Friedman et al., 2008a; Banerjee et al., 2008a) on centered data, and the Bayesian Information Criterion (BIC) (Schwarz et al., 1978) was used to select the adequate level of penalty, as implemented in the *huge* R-package (Zhao et al., 2012). All non-zero coefficients were set to 1 in the adjacency matrices. New coordinates were obtained by MDS and an MFA was then performed. The hierarchical clustering based on the table coordinates provided two groups, consisting in the Luminal A and B subtypes in one group and the HER2-enriched and Basal-like subtypes in the other.

The consensus networks obtained by majority-rule, as well as the clustering of the subtypes, are shown in Figure 4.6.

4.5 Discussion

In this paper we proposed a procedure to compare multiple objects built on the same entities, with a focus on trees and networks, in order to define coherent groups of these kind of structures to be further integrated. The procedure relies on two well-known methodologies, namely Multidimensional scaling (MDS) and Multiple Factor Analysis (MFA), that offer a unified framework to analyze both tree or network structures. The proposed approach provides tools to compare the structures and to easily obtain consensus trees or networks.

Because its computation only relies on a singular value decomposition (SVD), and since we may

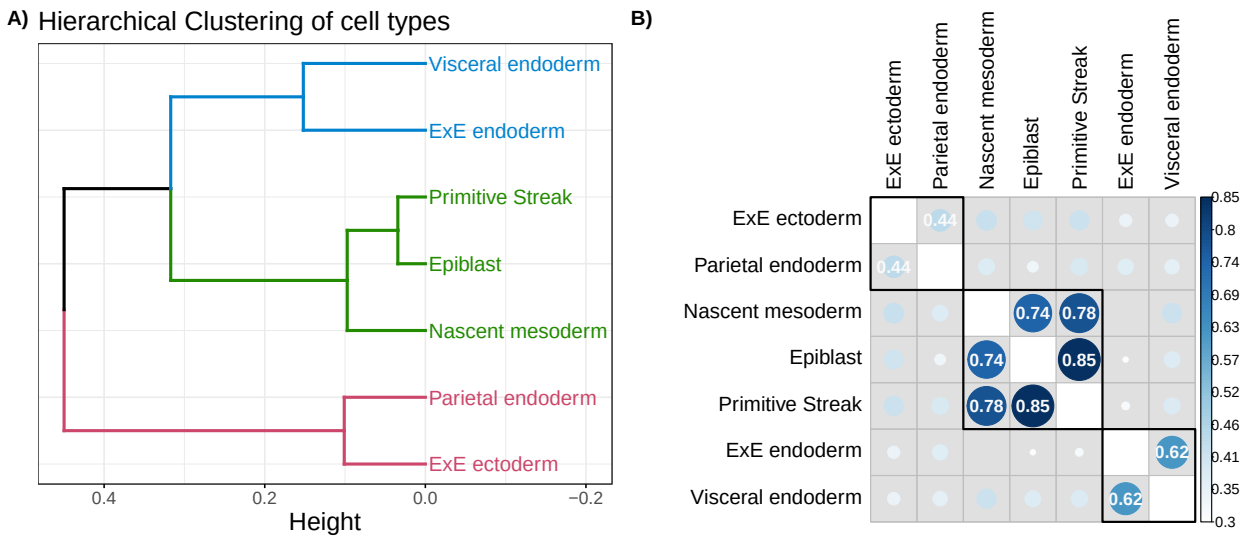


FIGURE 4.5 – **Visualization of groups given by MFA.** Dendrogram of the groups and associated RV coefficients between the cell-types. A) Dendrogram of the cell-types obtained on group coordinates of the MFA results using euclidean distance and Ward's aggregation criterion. Clusters were chosen using DynamicTreeCut and colored accordingly. B) Heatmap of RV coefficient between tables. The black grid shows the clusters as found in the dendrogram of panel A.

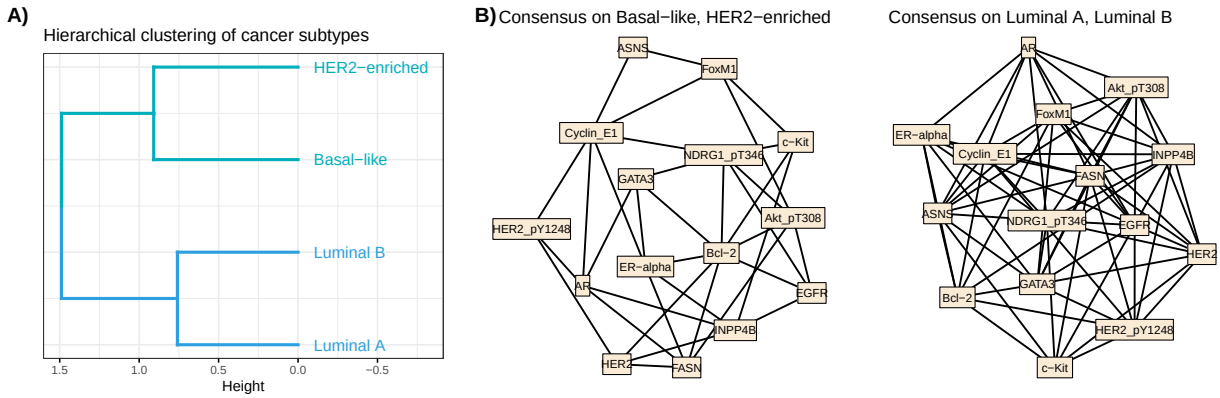


FIGURE 4.6 – **TCGA Breast cancer application.** Panel A) shows the hierarchical clustering of the breast cancer subtypes obtained with MFA group coordinates. Panel B) shows the two consensus networks, made with a majority rule from the adjacency matrices from the subtypes groups found in panel A).

have recourse to a truncated version of SVD, the procedure is very fast and appropriate to analyze a great number of objects. Our procedure was applied to simulated data, both in the context of trees and networks. In both cases, three very different grouping information were generated. The method was able to retrieve these three different structures. Consensus trees and networks were then obtained based on the MFA coordinates and were consistent with the simulated data for both the tree and network examples. We also analyzed two real data sets. A single-cell data set on mouse embryos was used to illustrate the performance of the methods on trees. Comparison with a clustering obtained in a previous study on these data (Pijuan-Sala et al., 2019) showed that the proposed methodology can integrate several trees while preserving the biological meaning of the

data. A TCGA breast cancer data set was also used to illustrate the process on network data. It emphasized two groups of breast cancer subtypes that are consistent with the literature. It also allowed to create two consensus networks that highlight differences in the protein interactions in these two groups. In both simulations and real data application, the procedure was shown to be an efficient and useful tool for the user to identify groups of data that are relevant to integrate.

We studied here the integration of data of the same types (trees or networks), but our procedure can integrate them together, along with other types of representations. An interesting point to be further investigated would be the integration of additional information such as clinical data. This would indeed be possible thanks to the use of MFA that can deal with data of various types (continuous and categorical).

In this paper, we used binary adjacency matrices with shortest path distance for the networks, and cophenetic distances for the trees. Any metric or transformation of the objects can be used as long as it yields a dissimilarity matrix usable in the MDS step.

Deuxième partie

Analyse de données -omiques pour l'étude du développement de la spondyloarthrite ankylosante (Multi-Spa)

Chapitre 5

Introduction

Table des matières

5.1 Spondylarthrite Ankylosante (SpA)

La Spondylarthrite Ankylosante est une maladie inflammatoire, de la famille des Spondyloarthrites, provoquant des inflammations aiguës et chroniques localisées sur la colonne vertébrale et le bassin, bien que d'autres articulations puissent être atteintes. La maladie peut aussi se manifester sous des formes extra-articulaires, comme avec du psoriasis, une uvéite et une Maladie Inflammatoire Chronique Intestinale (MICI, ou en anglais IBD pour *Inflammatory Bowel Disease)*. La maladie touche environ 1% de la population française, avec un ratio de 2 pour 1, pour les hommes (Dean et al., 2014). Elle atteint les individus de manière différente, avec une sévérité variable. Celle-ci est jugée notamment par l'échelle *Bath Ankylosing Spondylitis Disease Activity Index* (BASDAI).

L'héritabilité de cette maladie est importante, avec un facteur de prédisposition majeur identifié : l'allèle HLA-B27. Cet allèle se retrouve en effet chez la très grande majorité des patients : en France, 75% des patients sont HLA-B27 positifs (Breban et al., 2015; Costantino et al., 2015), contre 6.9% de la population générale. Ce pourcentage dépend de la population considérée, la fréquence de HLA-B27 positif étant estimée à 1% chez les Japonais, par exemple (Feltkamp et al., 2001), avec une apparition moins fréquente de la maladie. Cependant, cet allèle ne suffit pas, à lui seul, à expliquer l'apparition de la maladie : seules 3 à 6% des personnes HLA-B27 positives développeront la maladie et certaines personnes non porteuses de l'allèle développent la maladie. En fait, la part de la composante génétique de la maladie associée à cet allèle n'est que d'environ 20% (Breban et al., 2003). D'autres facteurs sont impliqués dans le développement de cette maladie. Comme pour d'autres maladies complexes, on soupçonne à la fois des causes génétiques et des causes environnementales à leur origine (Costantino et al., 2018).

Il est notamment considéré qu'une infection puisse être un déclencheur, en entraînant une réponse du système immunitaire et une inflammation. Le fait que la maladie se manifeste chez une part non négligeable des patients par une IBD permet de relier la maladie à une dysbiose intestinale, ce qui tend à être confirmé, sur modèle animal (Taurog et al., 1994; Rath et al., 1996) comme dans des études cliniques (Costello et al., 2015; Breban et al., 2017). Ces études tendent à montrer que

les patients atteints de SpA ont une composition du microbiote différant légèrement de celle des individus sains.

5.2 Projet Multi-Spa

On cherche dans ce projet à étudier les causes du développement de la maladie qui ne sont pas reliées à l'allèle HLA-B27. On dispose pour cela de données transcriptomiques et métagénomiques, incluant des patients et des contrôles, ces derniers répartis en deux groupes : les contrôles dits positifs, porteur de l'allèle HLA-B27, et les contrôles dits négatifs, qui ne possèdent pas l'allèle. Les patients sont tous HLA-B27 positifs.

Les données transcriptomiques ayant été obtenues en deux temps, une première étape du projet est d'analyser les deux jeux de données séparément, pour identifier des cibles potentielles. L'analyse des données de transcriptomique est présentée dans le chapitre suivant. Le plan d'expérience selon lequel les données ont été générées est détaillé dans la section suivante. L'analyse des données de métagénomique seules a été entièrement effectuée par Metagénopolis.

La deuxième étape concerne l'intégration de données : il est important de comprendre comment les espèces bactériennes du microbiote peuvent être corrélées, voire même interagir, avec l'expression des gènes, et quelle différence il peut y avoir dans les interactions observées entre les patients et les contrôles.

Les individus impliqués dans cette études proviennent de deux ensembles, aussi désignés par le mot set : le premier correspond à des inclusions effectuées entre 2010 et 2012 et le deuxième à des inclusions entre 2013 et 2015. Cette répartition en deux sets a son importance : les échantillons du premier set ont bénéficié d'une exposition plus longue à l'IL4 (Interleukine 4).

5.3 Données de transcriptomique

Les expressions de 114 individus ont été obtenues, 30 Contrôles Négatifs (NC, *Negative Control*), 44 Contrôles Positifs (PC, *Positive Control*) et 40 Patients Positifs (PP, *Positive Patient*). 5 individus marqués comme Contrôles ont été retirés de l'étude car ils présentaient un psoriasis.

Comme évoqué dans la présentation du projet, les individus ont été recrutés sur deux périodes différentes, et les échantillons ont été traités de manière différentes. On parlera de set 1 et de set 2 pour les désigner.

On détaille dans les sections suivantes le design utilisé pour l'étude, ainsi que l'impact qu'il a sur les données.

5.3.1 Plan d'expérience : introduction d'un facteur temps

L'étude vise en particulier à observer l'évolution de l'expression des gènes au cours du temps, après stimulation. Pour ce faire, les monocytes prélevés sur les individus à D0 sont mis en culture pendant 7 jours (D7), pendant lesquels ils se différencient en cellules dendritiques (voir Figure 5.1), cellules impliquées dans les mécanismes de réponse immunitaires. Au septième jour, les échantillons sont stimulés par du LPS (Lipopolysaccharide) bactérien pour produire une réaction inflammatoire, et des extractions d'ARN sont faites après 3 heures (H3), 6 heures (H6) et 24 heures (H24). Pour chaque individus, jusqu'à 5 échantillons, correspondant aux différents temps, sont disponibles. Comme cela a été déjà évoqué, les échantillons du set 1 ont été stimulés en présence d'une plus grande quantité d'IL4 que ceux du set 2.

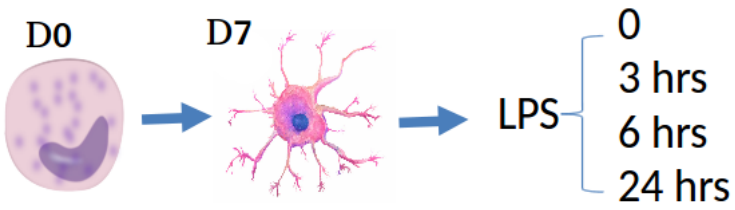


FIGURE 5.1 – Design du temps dans les données - différenciation et stimulation des cellules dendritiques.

	Contrôles		Patients	Total
	Négatifs	Positifs	Positifs	
Monocytes				
D0	21	24	32	77
Dendritiques				
H0 / D7	26	38	35	99
H3	27	33	34	94
H6	26	39	35	100
H24	26	38	32	96
Dend Total	105	148	136	389

TABLEAU 5.1 – Nombre d'échantillons par condition/temps utilisés pour les analyses. Dend désigne l'ensemble des échantillons des cellules dendritiques (tous temps sauf D0).

La différence d'expression des gènes au cours du temps conduit à une nette séparation des échantillons prélevés (D0, monocytes) des autres (dendritiques) sur l'ACP 5.2 page 76. Nous avons fait le choix d'analyser séparément les échantillons D0 et les échantillons provenant des cellules dendritiques dans la suite. D7 est appelé H0 dans la suite.

Les échantillons des 5 temps ne sont pas disponibles pour tous les individus. Le nombre d'échantillons à disposition par temps et condition est récapitulé dans la Table 5.1 page 75. Un temps Dend (dendritique) total a été défini, qui rassemble tous les échantillons H0 à H24.

5.3.2 Séquençage des données et traitement des gènes

Deux techniques d'alignement ont été utilisées pendant la thèse, STAR (*Spliced Transcripts Alignment to a Reference*) (Dobin et al., 2013) et SALMON (Patro et al., 2017). SALMON est la technique qui sera retenue, SALMON permettant d'avoir accès aux comptages des transcrits. Les corrélations par échantillons entre les comptages produits par les deux méthodes ont été calculées, et sont proches de l'unité. Un seul échantillon a été enlevé car sa corrélation avec lui-même est inférieure à 0.6 entre les deux techniques.

Les données renvoyées par SALMON sont des comptages qui ne sont pas entiers, du fait de la méthode utilisée qui n'opère pas un alignement exact entre les lectures et le génome, certaines lectures pouvant s'aligner sur plusieurs transcrits, et sont donc réparties entre ces transcrits. Les comptages sont arrondis.

Les alignements renvoient des expressions pour 60 179 gènes. Les gènes correspondants à des catégories non fonctionnelles (par exemple les pseudogènes) et les transcrits correspondant à des petits ARN, comme les ARN de transfert, qui ont des séquences largement homologues, ont été directement sortis de l'analyse, pour ne garder que les catégories présentant un intérêt dans ce que l'on recherche. Les biotypes suivants : *3-prime overlapping, ncRNA, antisense, bidirectional promoter, lncRNA, lincRNA, macro lncRNA, protein coding, sense intronic* et *sense overlapping*

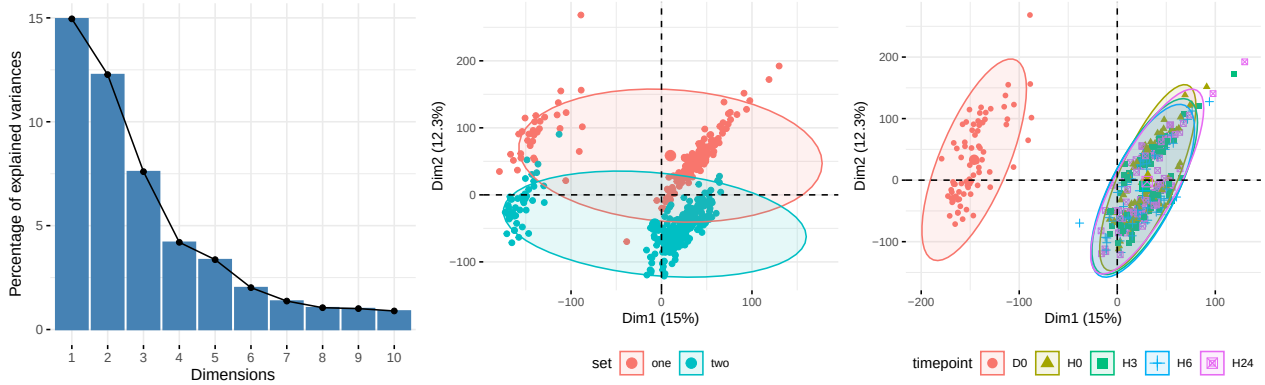


FIGURE 5.2 – ACP des données : séparation entre les deux sets et séparation entre les échantillons monocytes et dendritiques. ACP réalisée sur des données non filtrées, transformé par $\log_2$. Les échantillons ont été divisés par leur taille de librairie.

ont été conservés, pour un total de 33 404 gènes. Ces gènes seront ensuite filtrés selon le protocole décrit dans le chapitre 6.

Les sous-sections suivantes détaillent le design qui a été utilisé pour la production des données, et son impact sur le design utilisé pour l'analyse différentielle et l'intégration de données.

5.3.3 Deux sets

L'étude a été conduite en deux temps. L'ACP des données 5.2 page 76, montre une séparation des données de transcriptomique suivant les deux sets, sur le premier axe.

Les modalités de stimulation utilisées, ainsi que la profondeur de séquençage, n'ont pas été les mêmes dans les deux parties de l'étude. Cependant, ces deux justifications n'expliquent qu'en partie la différence observée entre les deux sets, puisque l'ACP, effectuée sur des données centrées et réduites, avec normalisation des échantillons par taille de librairie, indique une séparation nette dans les échantillons de D0, qui n'ont pas été mis en cluture. D'autres causes, matérielles peut-être, peuvent expliquer cette séparation.

Au vu de l'importance de cet effet set il a été décidé de l'inclure dans les designs utilisés pour les analyses différentielles, mais aussi de le prendre en compte dans les analyses d'intégration de données : lors de l'étape d'intégration des données de transcriptomique avec les données de métagénomique, le découpage de temps sera à nouveau utilisé.

5.4 Analyse des données de métagénomique

Les données de métagénomique (16S) ont été analysées séparément par l'équipe de Métagénopolis. Dans la partie du projet effectuée pendant ma thèse, les données de métagénomique n'ont été analysées qu'en interaction avec les données de transcriptomique.

Chapitre 6

Analyse des données transcriptomiques

Table des matières

6.1 Analyses différentielles

Contrastes étudiés. Trois contrastes ont été étudiés, pour prendre en compte les différentes conditions. Ils seront désignés par les acronymes suivants dans le reste du chapitre :

— PPNC : Positive Patients versus Negative Controls (effet maladie et B27 mélangés)
— PPPC : Positive Patients versus Positive Controls (effet maladie seulement)
— PCNC : Positive Controls versus Negative Controls (effet B27 seulement)

Design. Chaque ensemble d'échantillons à l'intérieur d'un temps a fait l'objet d'une analyse différentielle, pour les contrastes mentionnés précédemment. Les échantillons des cellules dendritiques (H0, H3, H6, H24) ont été utilisés ensemble pour une analyse supplémentaire en incluant le temps comme facteur dans la matrice de design, qui sera désignée par Dend dans les résultats.

Du fait des effets évoqués précédemment, le design utilisé dans les comparaisons temps par temps prend en compte le set, le sexe de l'individu et la condition combinant le statut morbide et la présence ou non de l'allèle B27. Le sexe de l'individu est inclus dans l'analyse, du fait de la légère prédominance des hommes chez les malades atteints de SpA.

6.1.1 Traitement des données

Filtres sur expressions et normalisation. Une fois les échantillons séparés, les filtres suivants ont été appliqués sur les gènes avant de lancer une analyse différentielle :

— Les gènes ne s'exprimant pas du tout sont supprimés ;

	H0	H24	H3	H6	D0	Dend
Avant filtre	30 576	30 576	30 576	30 576	28 366	30 576
Apres filtre (Analysés)	15 135	15 125	14 783	14 645	14 930	16 021
Gènes filtrés	15 441	15 451	15 793	15 931	13 406	14 555

Tableau 6.1 – Nombre de gènes avant et après application des filtres

— Les gènes ayant moins d'un comptage par million sur l'ensemble des échantillons servant à l'analyse sont supprimés. Ce filtre est tiré du *User guide* du package `edgeR`, version 3.24.3 [1]

Le Tableau 6.1 page 78 récapitule le nombre de gènes analysés pour chaque temps, ainsi que le nombre de gènes filtrés.

Les données sont ensuite normalisées par échantillon, en utilisant la normalisation TMM (Trimmed mean of M-values) (Robinson and Oshlack, 2010) implémentée dans le package `edgeR` de `R`.

Correction des données. L'effet set étant très important dans les données, lorsque l'utilisation de données continue est nécessaire, pour l'affichage des heatmaps et des violin-plots, par exemple, l'utilisation d'une simple transformation log des comptages n'est pas suffisante. Pour pouvoir afficher des données qui se rapprochent le plus des données corrigées, on utilise les fonctions *cpm* du package `edgeR` pour transformer les données en comptages par millions, puis la fonction *removeBatchEffect* du package `limma` en incluant le set comme effet.

6.1.2 Méthode utilisée pour les analyses différentielles

Présentation du modèle général utilisé. Notons X la matrice de design, dont les colonnes correspondent proviennent d'un codage disjonctif des variables incluses dans les analyses (Condition, set, sexe, temps si pertinent).

On utilise le package `edgeR` (Robinson et al., 2010; McCarthy et al., 2012) pour effectuer l'analyse différentielle. La méthode repose sur l'utilisation d'une loi négative binomiale. On note Y_{ij} le comptage du gène i pour l'échantillon j. On a :

$$\begin{aligned} Y_{ij} &\sim \mathcal{NB}(\mu_{ij}, \alpha_i), \\ \mu_{ij} &= s_j q_{ij}, \\ \log_2(q_{ij}) &= x_j \beta_i = \sum_r x_{jr}\beta_{ir}, \end{aligned} \tag{6.1}$$

avec α_i, paramètre de dispersion propre au gène i. x_j est la ligne j de la matrice de design X et β_i le vecteur des coefficients du gènes i associés à chaque modalités du desgin utilisé. La moyenne μ_{ij} se décompose en un terme de taille de librairie s_j et un deuxième paramètre q_{ij} qui est proportionnel au compte de reads pour l'échantillon j. s_j est estimé grâce à la méthode TMM. La variance de Y_{ij} est :

$$\mathbb{V}(Y_{ij}) = \mu_{ij}(1 + \mu_{ij}\alpha_i) \tag{6.2}$$

Lorsque $\alpha_i = 0$, ce modèle permet de retrouver la loi de Poisson. Un modèle linéaire généralisé avec un lien logarithmique utilisé, avec la décomposition suivante :

$$\log_2(\mu_{ij}) = \sum_r x_{jr}\beta_{ir} + \log_2 s_j \tag{6.3}$$

1. https://www.bioconductor.org/packages/3.8/bioc/vignettes/edgeR/inst/doc/edgeRUsersGuide.pdf

Le paramètre α_i est le coefficient de variation. La racine carrée de ce paramètre est appelé coefficient de variation biologique, qui représente la part de variation attribuable à la variabilité propre au système biologique étudié, se différenciant de la variabilité attribuable aux technologies utilisées pour la production des données. La variation totale se décompose en un terme dû à la variation biologique entre les échantillons, et un deuxième terme dû à la variation technique entre les échantillons.

Utilisation de la quasi-vraisemblance. On utilise la méthode décrite dans l'article Lun et al. (2016), utilisant la quasi-vraisemblance, dans une extension du modèle négatif binomial. Dans l'équation 6.2, on introduit le terme σ_i^2, paramètre de dispersion de la quasi-vraisemblance, propre au gène i :

$$\mathbb{V}(Y_{ij}) = \sigma_i^2 \mu_{ij}(1 + \mu_{ij}\alpha) \tag{6.4}$$

Dans ce modèle, le paramètre α est global à tous les gènes, et la variation propre au gène i est captée par le terme σ_i.

Tests et corrections. Pour conclure si un gène est différentiellement exprimé ou non, on utilise la méthode décrite dans Lund et al. (2012) sous le nom de "quasi-likelihood F-test", soit un test de Fischer, utilisant les estimations du modèle de quasi-vraisemblance, gène à gène, en considérant que les tests réalisés sont indépendants les uns des autres.

On cherche à comparer la moyenne d'expression du gène i sur les échantillons de la condition 1, comparée avec la moyenne d'expression du gènes i pour la condition 2, avec les hypothèses suivantes :

$$(H_0)\ \mu_{ij_1} = \mu_{ij_2} \ \text{ vs } \ (H_1)\ \mu_{ij_1} \neq \mu_{ij_2} \tag{6.5}$$

Le rejet de l'hypothèse nulle permet de conclure qu'un gène est différentiellement exprimé entre les conditions 1 et 2.

Pour contrôler le nombre de faux positifs trouvés, on corrige les p-valeurs des tests par la méthode de Benjamini-Hochberg (Benjamini and Hochberg, 1995). Dans la suite, on utilisera aussi l'appellation *False Discovery Rate* (FDR) pour la p-valeur corrigée.

6.1.3 Résultats des analyses

Les heatmaps présentées dans cette section ont été tracées avec le package `R ComplexHeatmap` (Gu et al., 2016). Les clusterings affichés sur ces heatmaps sont construits en utilisant la distance euclidienne et le *complete-linkage*.

Gènes différentiellement exprimés (DE). Le Tableau 6.2 page 80 montre le nombre de gènes DE trouvés pour chaque comparaison effectuée, à différents seuils de p-valeur ajustée, avec, entre parenthèses, le nombre de ces gènes ayant une valeur absolue de $\log_2$ *Fold-Change* supérieur à 0.5.

La plupart des comparaisons/temps donnent peu de gènes DE, à l'exception du temps H0 et Dend. L'utilisation de l'ensemble des échantillons Dend apporte un plus grand nombre de gènes différentiellement exprimés, même une fois le filtre sur le $\log_2$FC appliqué.

Gènes propres à l'effet maladie (PPPC − PCNC). Notre objectif est de repérer des gènes pouvant expliquer l'apparition de la maladie, qui ne soient pas reliés à l'effet B27+, ou qui soient complémentaires de cet effet. En effet, on peut supposer que B27 induit des modifications qui prédisposent à la maladie, mais on sait que son effet seul ne permet pas d'expliquer entièrement le développement de la SpA. On sélectionne les gènes ayant une p-valeur ajustée inférieure à 0.1.

	PPPC			PPNC			PCNC		
FDR	0.01	0.05	0.1	0.01	0.05	0.1	0.01	0.05	0.1
D0	2	3	10 (3)	0	3	5	0	13	14
H0	0	4	8	7	68 (29)	264 (89)	12	22	30 (27)
H3	0	1	2	0	0	1	1	1	2
H6	1	1	1	0	0	4	0	1	1
H24	2	2	2	1	2	2	3	6	7
Dend	184 (88)	714 (216)	1304 (302)	236 (105)	829 (245)	1414 (354)	209 (106)	606 (230)	1016 (316)

Tableau 6.2 – Résultats des analyses différentielles. Nombre de gènes différentiellement exprimés en fonction du seuil de p-valeur ajustée, pour chaque temps et chaque contraste étudié. Entre parenthèse : le nombre de gènes avec $|\log_2 \mathrm{FC}| > 0.5$

La Figure 6.1 page 81 montre les diagrammes de Venn de chacune des comparaisons effectuées. On recherche les gènes qui correspondent à : PPPC – PCNC, ou à PPNC – PCNC, ou à l'intersection de ces deux comparaisons. Il existe globalement peu de chevauchement entre les gènes DE pour chaque analyse. On peut remarquer que les gènes PPNC, qui sont réponses de la maladie mais aussi de B27, ne sont pas tous retrouvés dans la comparaison PCNC, les individus malades ont bien des expressions différentes pour certains gènes en comparaison des contrôles B27 positifs.

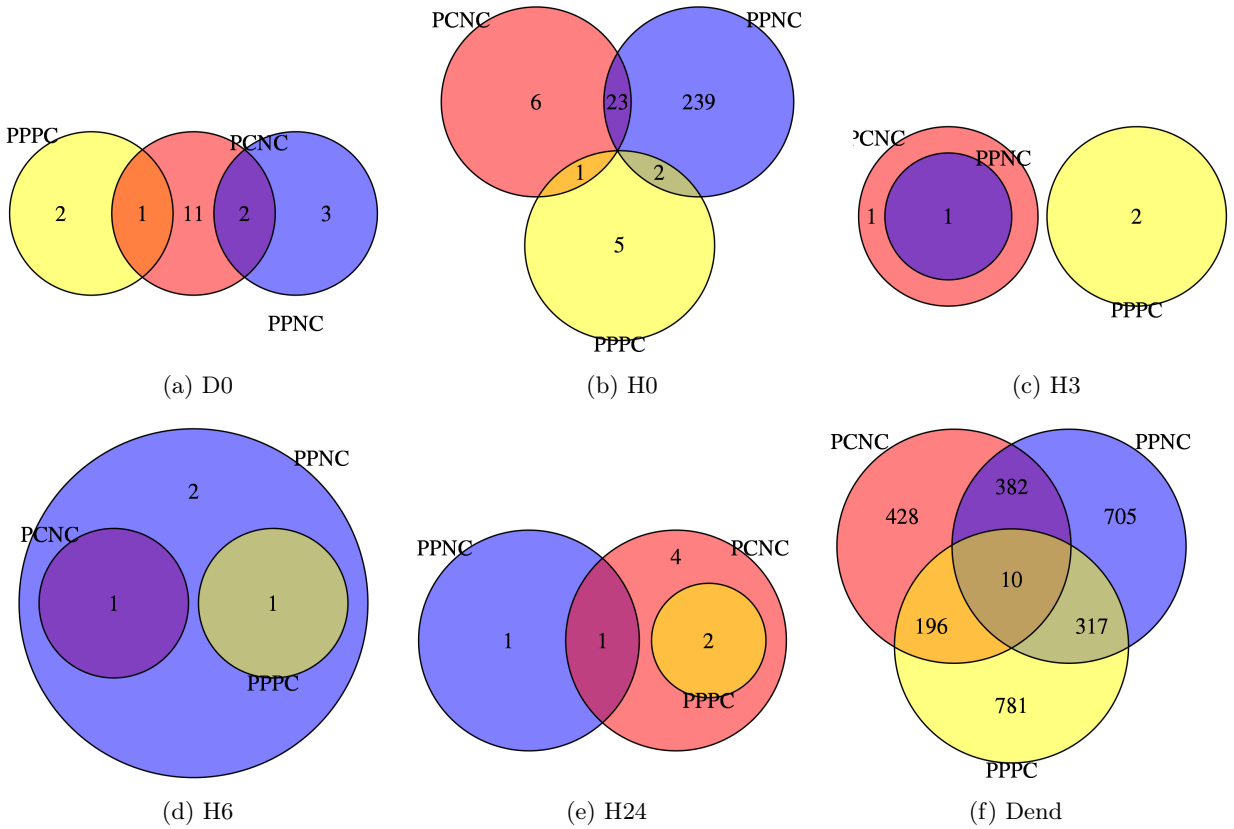
On s'intéresse particulièrement à la comparaison PPPC – PCNC. Dans les diagrammes de Venn, cela correspond à la partie jaune du diagramme (PPPC) combinée à son intersection avec la zone bleue (PPNC). Les gènes de la zone bleue, ainsi que son intersection avec la zone jaune, devraient être aussi des gènes impliqués dans l'effet maladie. Les gènes de l'intersection entre PPNC et PPPC, notamment les 317 de la comparaison Dend, sont très intéressants pour cette étude car ils sont en relation avec l'effet B27 et l'effet maladie, on peut donc considérer qu'ils sont impliqués de façon vraisemblable dans le développement de la maladie.

La plupart des temps mènent à un nombre de gènes réduit, sauf pour Dend, qui produit une liste de gènes DE conséquente. Les gènes contenus dans l'intersection des 3 comparaisons pour Dend (Figure 6.1 (f)) (ADGRG7, APBB1IP, LINC01500, MORC4, OSBP2, PPBP, FAT1, KCNQ5, ADTRP et CXCL5) sont intéressants puisqu'ils présentent la particularité d'être communs à toutes les analyses, ils sont à la fois trouvés dans l'effet maladie, et dans l'effet B27+. Ils ne seront cependant pas étudiés dans la suite, puisque l'on cherche dans cette partie des gènes spécifiques à PPPC ou PPNC.

Les figures 6.2 et 6.3 pages 82 et 82 représentent les heatmaps des expressions des gènes (après transformation en comptages par millions et *removeBatchEffect*), centrées par gène, des gènes DE qui correspondent aux gènes de réponse (PPPC – PCNC). Le temps H6 ayant donné un seul gène d'intérêt, son expression a été représentée sous la forme d'un violin plot. Les individus ont été ordonnés selon leur condition. Les gènes indiqués en bleus sont les gènes trouvés communs avec la comparaison (PPNC – PCNC).

On peut repérer dans la figure 6.2 (b) le gène PCOLCE2, dont l'expression chez les contrôles a tendance à être réduite comparée à ce qui est observé chez les patients, et qui sera discuté plus en détail dans un paragraphe suivant.

Dans la majeure partie des heatmaps, on observe que les populations, qu'il s'agisse des patients ou des contrôles, sont très hétérogènes en ce qui concerne l'expression des gènes. Certains patients sont très proches des contrôles en termes d'expression. Malgré le fait que ces gènes aient été sélectionnés par une analyse différentielle, et qu'on ait pris soin de supprimer les gènes qui pourraient être relié



(a) D0 (b) H0 (c) H3

(d) H6 (e) H24 (f) Dend

FIGURE 6.1 – Diarammes de Venn des gènes trouvés différentiellement exprimés pour chaque temps et contrastes étudiés. p-valeur ajustée : 0.1.

à l'effet B27+, la différence entre les patients et les contrôles reste modeste. Cette hétérogénéité explique la difficulté à identifier les acteurs du développement de la maladie.

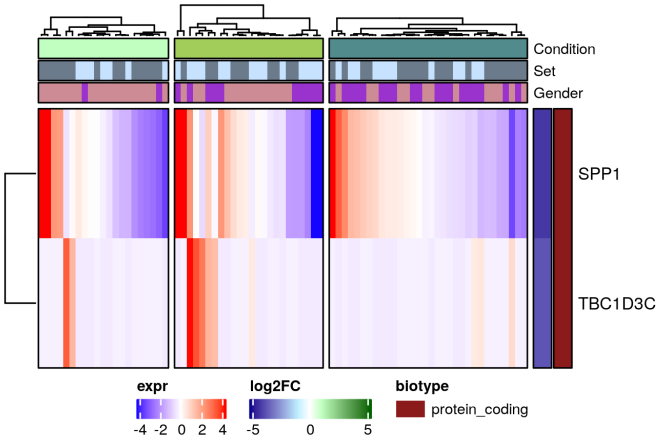
Dans le cas de l'analyse des échantillons dendritiques ensembles, la heatmap produite ne permet pas d'observer des effets propres à chaque gène, et aucun phénomène ne se dégage. Il est nécessaire de constituer des groupes de gènes pour les étudier ensemble, voire les actions de ces groupes en interaction.

Gènes d'intérêt : gènes de la comparaison PPNC - PCNC. Si l'on supprime les gènes qui sont en commun avec PCNC, on devrait trouver des gènes propres à l'effet maladie dans la liste des gènes différentiellement exprimés de PPNC. Dans cette comparaison, H6 n'a qu'un seul gène d'intérêt. H3 n'a pas de gène d'intérêt. À H0, le nombre de gènes est trop important pour pouvoir afficher le nom des gènes.

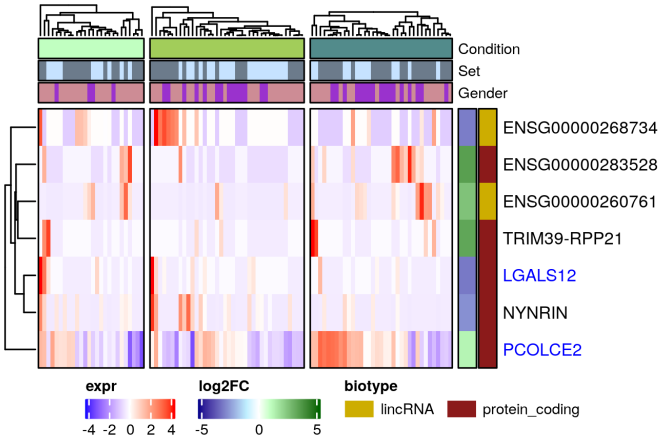
Les figures 6.4 et 6.5 page 83 et 83 présentent les heatmaps et violin plots des gènes d'intérêts de la comparaison PPNC. Les conclusions faites sur les gènes de PPPC sont encore ici valables : les populations de contrôles et de patients sont très hétérogènes, et définir des groupes de gènes pour les étudier en détail semble obligatoire.

PCOLCE2. La figure 6.6 page 84 montre l'expression normalisée (correction de l'effet set par *removeBatchEffect*) du gènes PCOLCE2 (*Procollagen C-endopeptidase enhancer 2*) pour les cellules dendritiques, pour chaque condition et chaque temps.

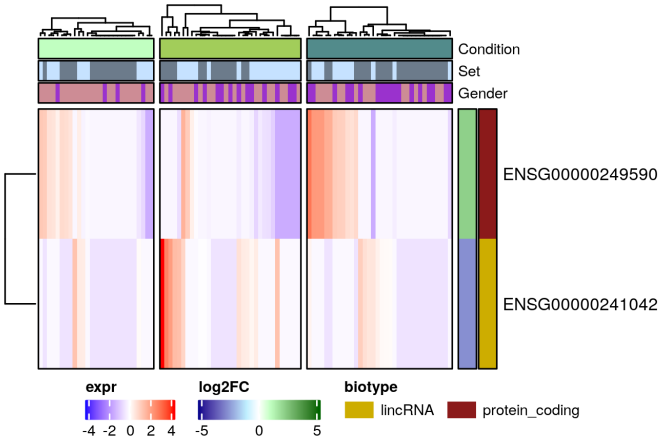
Ce gène a été trouvé DE pour les comparaisons H0 PPPC et H0 PPNC. Son expression apparait globalement plus élevée dans le groupe des patients que dans les deux groupes de contrôles. La



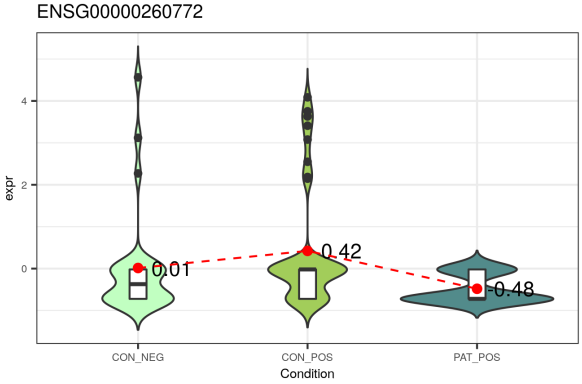
(a) D0



(b) H0



(c) H3



(d) H6

FIGURE 6.2 – **PPPC - PCNC.** Heatmaps des temps séparés. Gènes en bleu : gènes communs entre PPNC et PPPC

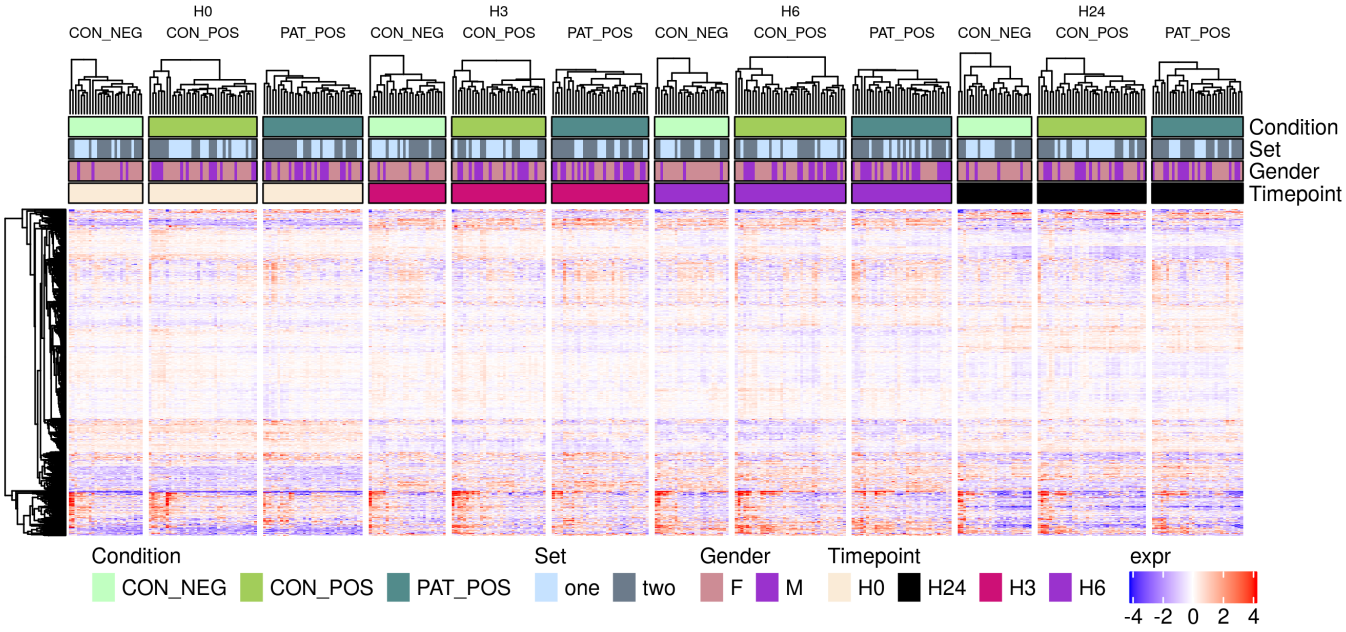
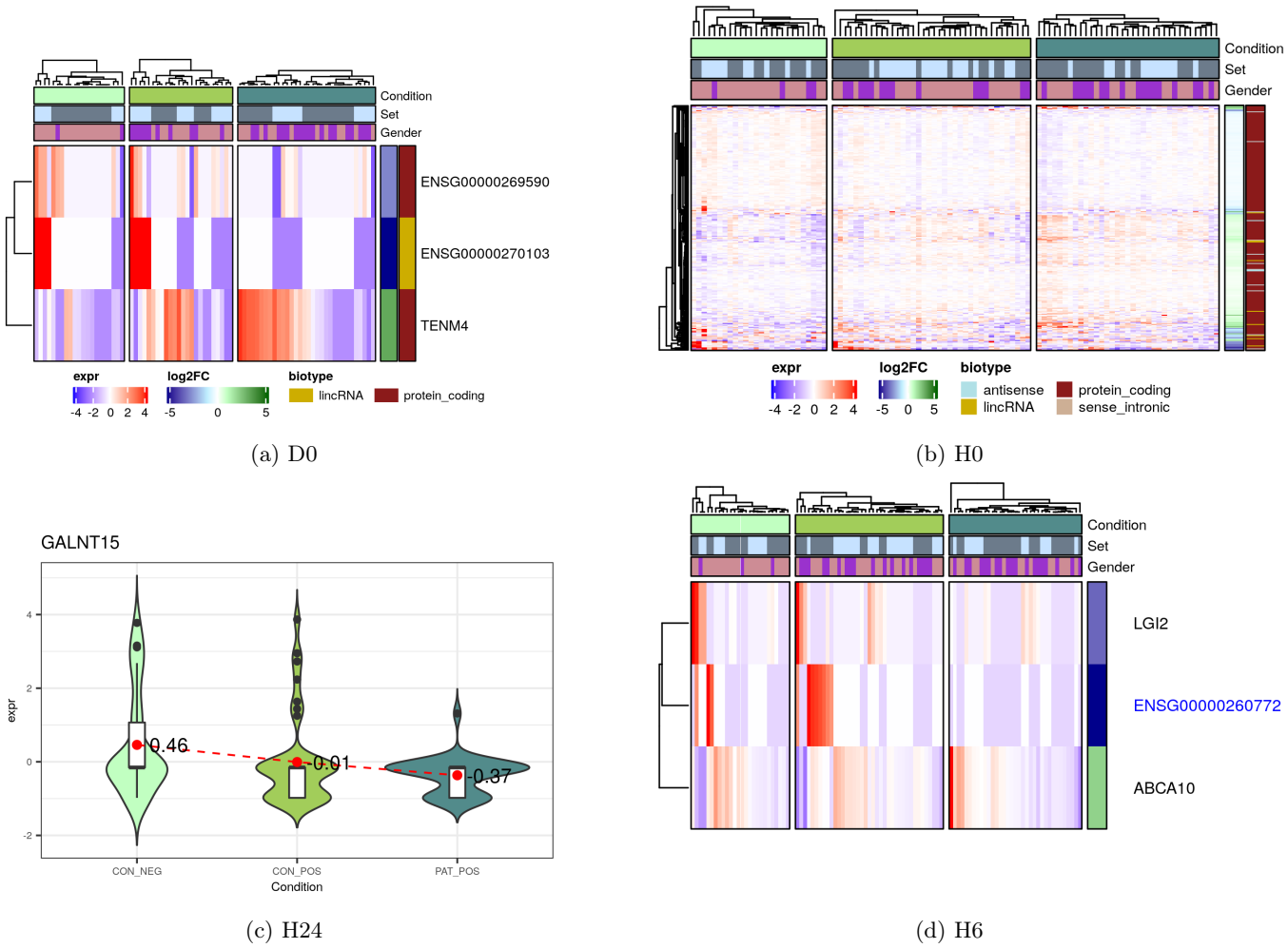


FIGURE 6.3 – **PPPC - PCNC.** Heatmaps des gènes DE pour la comparaison Dend.



(a) D0

(b) H0

(c) H24

(d) H6

FIGURE 6.4 – **PPNC - PCNC**. Heatmaps des temps séparés. Gènes dont les noms sont en bleu : gènes commun entre PPNC et PPPC.

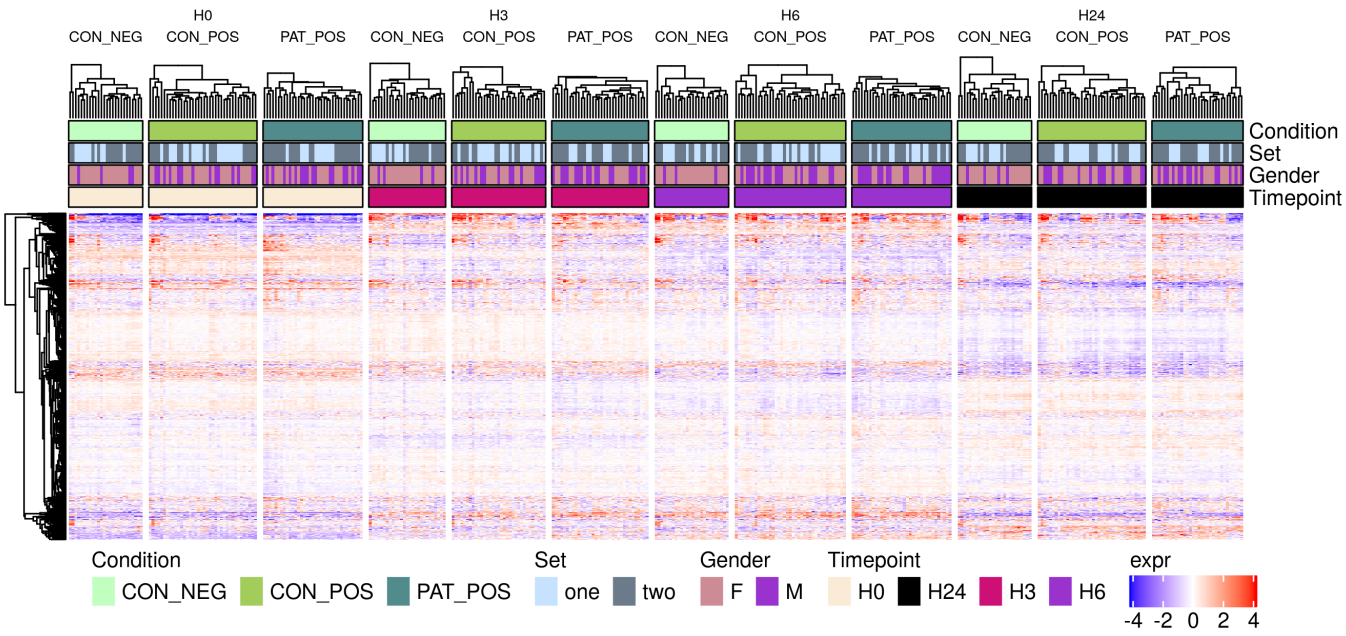


FIGURE 6.5 – **PPNC - PCNC**. Heatmaps des gènes DE pour la comparaison Dend

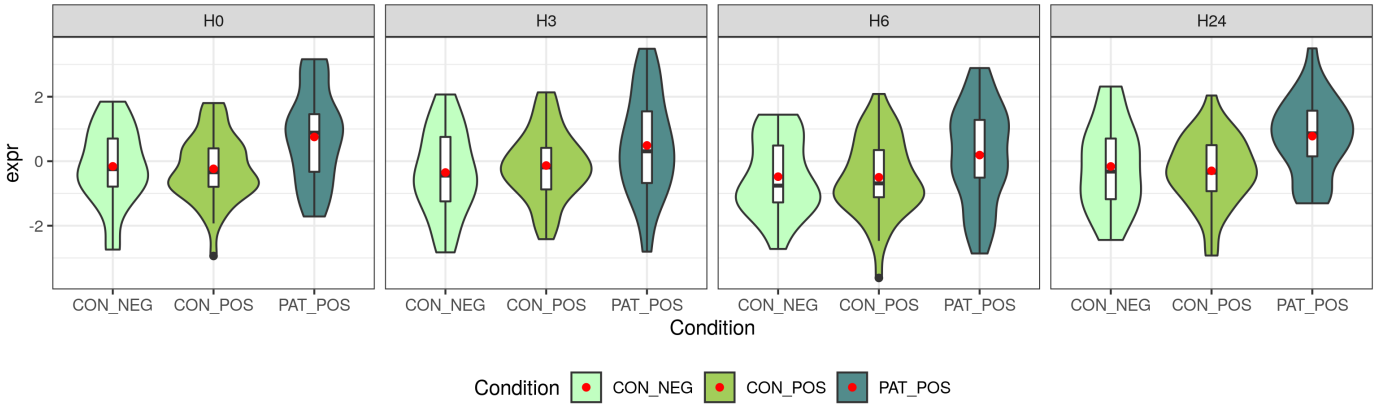


FIGURE 6.6 – Violin plot de l'expression corrigée pour l'effet set de PCOLCE2. Expression centrée sur tous les temps. Point rouge : moyenne de l'expression pour la condition et le temps donné.

	H3			H6			H24		
	PCNC	PPNC	PPPC	PCNC	PPNC	PPPC	PCNC	PPNC	PPPC
$\log_2$ FC	0.024	1.019	0.995	-0.164	0.764	0.928	-0.42	0.772	1.193
FDR	0.995	1.000	0.739	1.000	0.998	0.968	1.00	1.000	0.480

	H0			Dend		
	PCNC	**PPNC**	**PPPC**	PCNC	**PPNC**	**PPPC**
$\log_2$ FC	-0.165	1.105	1.270	-0.171	0.914	1.085
FDR	0.924	0.097	0.097	0.677	0.000	0.000

TABLEAU 6.3 – $\log_2$ FC et FDR données par `edgeR` pour le gène PCOLCE2 dans toutes les comparaisons effectuées. Le gène n'a pas été analysé à D0. Les comparaisons dans lesquelles le gène a été trouvé DE sont en gras. Valeurs arrondies à la troisième décimale.

heatmap donnait l'impression de deux groupes de patients pour ce gène, ce violin plot le confirme : si certains patients se détachent bien par leur expression élevée, d'autres sont au même niveau que les contrôles.

Ce gène est pertinent pour la pathogénèse de la spondyloarthrite par son implication dans la dégradation des procollagènes de type 1 et 2, et dans le clivage de ces collagènes par la protéine BMP1 (*Bone Morphogenetic Protein 1*), impliqué dans le métabolisme du tissu conjonctif et du tissu osseux.

Le Tableau 6.3 page 84 contient les $\log_2$ FC et les FDR du gène dans toutes les comparaisons effectuées. Il est à noter que le gène n'a pas passé le filtre pour D0 et n'a pas d'information pour cette comparaison. Le gène est trouvé DE dans les mêmes comparaisons PPPC et PPNC dans H0 et Dend. Dans aucun des autres temps, ce gène n'est trouvé DE.

Dendritiques : gènes communs entre PPNC et PPPC (-PCNC). Les deux comparaisons Dend PPNC et Dend PPPC renvoient trop de gènes pour distinguer des groupes de gènes, ou des gènes intéressants sur les heatmaps tracées, même en ayant enlevé les intersections avec PCNC. On s'intéresse ici aux gènes communs aux deux comparaisons PPNC et PPPC. Pour réduire le nombre de ces gènes (317), on prend seulement les gènes ayant un FDR < 0.01 et un $|\log_2 \text{FC}| > 0.5$. En retirant tous les gènes de PCNC trouvés DE à un seuil de 0.1 de FDR, on trouve 25 gènes d'intérêts, dont les expressions corrigées sont présentées sur la Figure 6.7 page 85.

Le gène PCOLCE2, déjà identifié comme un gène cible, fait partie de cette intersection. Ce gène

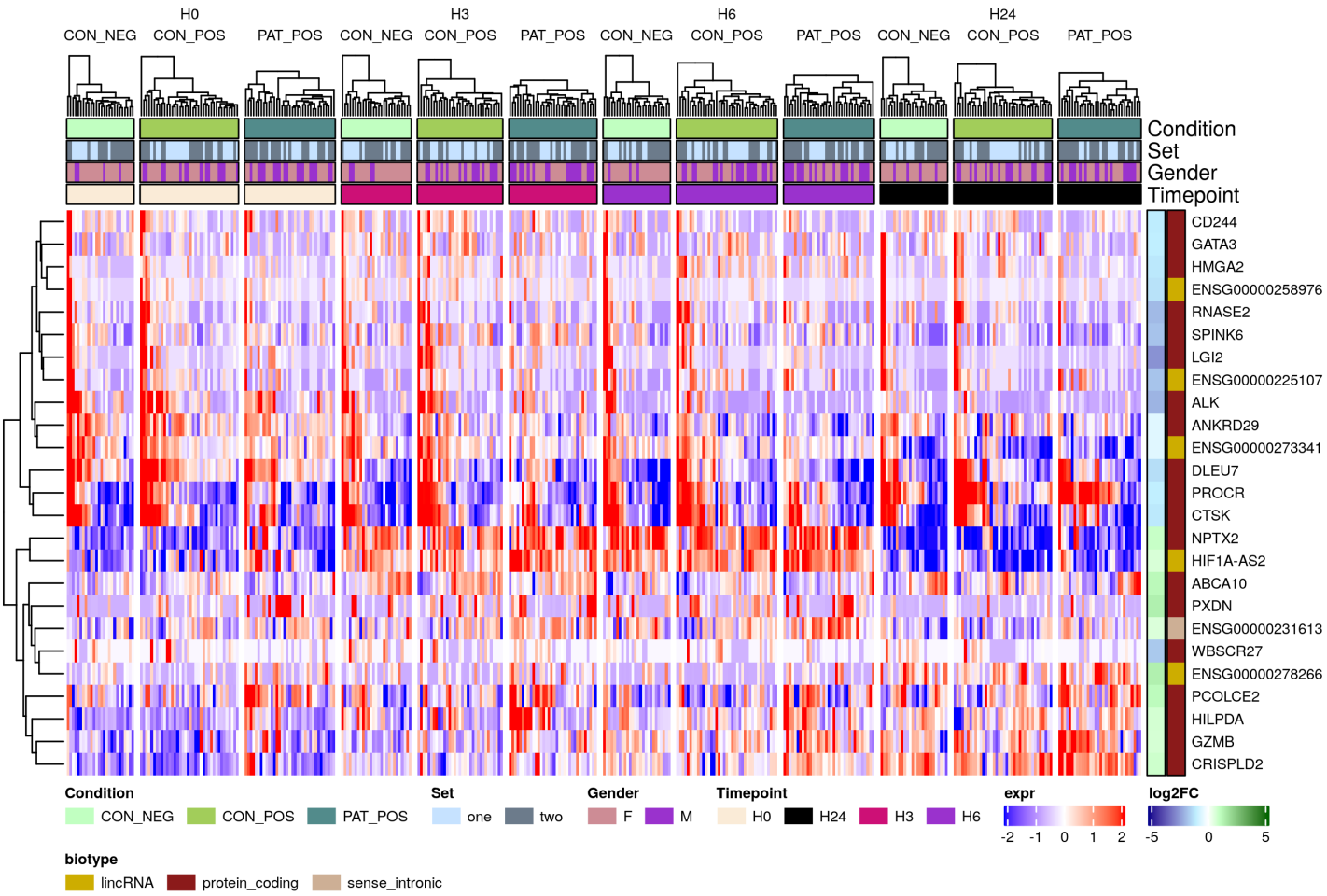


FIGURE 6.7 – Heatmap des expressions des gènes communs entre PPNC et PPPC dans la comparaison de tous les échantillons dendritiques, ayant une p-valeur ajustée à 0.01 et un $|\log_2 \mathrm{FC}| > 0.5$. Les gènes communs avec PCNC ont été retirés de l'analyse.

est regroupé avec trois autres gènes : HILPDA *(Hypoxia Inducible Lipid Droplet Associated)*, GZMB (*Granzyme B*) et CRISPLD2 (*Cysteine Rich Secretory Protein LCCL Domain Containing 2*), dont les expressions peuvent être intéressantes aussi. Leurs expressions sont présentées sur les violin plots de la Figure 6.8 page 86. Le gène PCOLCE2 avait une expression plus élevée chez les patients, même au temps H0, qui se retrouvait au temps H24. La variabilité d'expression de ces gènes est aussi très grande chez les patients, plus que chez les contrôles. L'expression du gène GZMB est légèrement supérieure chez les patients à celle des contrôles, quel que soit le temps, mais cette différence est plus marquée au temps H24. Comme pour PCOLCE2, on observe un effet prolongé jusqu'à 24h après la stimulation Alors que dans les deux groupes contrôles, l'expression augmente légèrement puis redescend au court du temps, on observe que l'expression chez les patients, ou chez une partie des patients, ne revient pas à son niveau d'origine au bout de 24 heures. Une partie des contrôles positifs a l'air de suivre la même évolution que les patients, avec une augmentation de la variabilité de l'expression dans le groupe entre H6 et H24.

Les deux autres gènes, CRISPLD2 et HILPDA montrent aussi une plus grande variabilité du groupe des patients que des deux groupes contrôles à certains temps. Globalement ces quatre gènes montrent des variations au court du temps, entre les groupes de contrôles et de patients, qui confirment leur intérêt.

Les fonctions auxquelles sont reliées ces gènes sont intéressantes et certaines peuvent être reliées

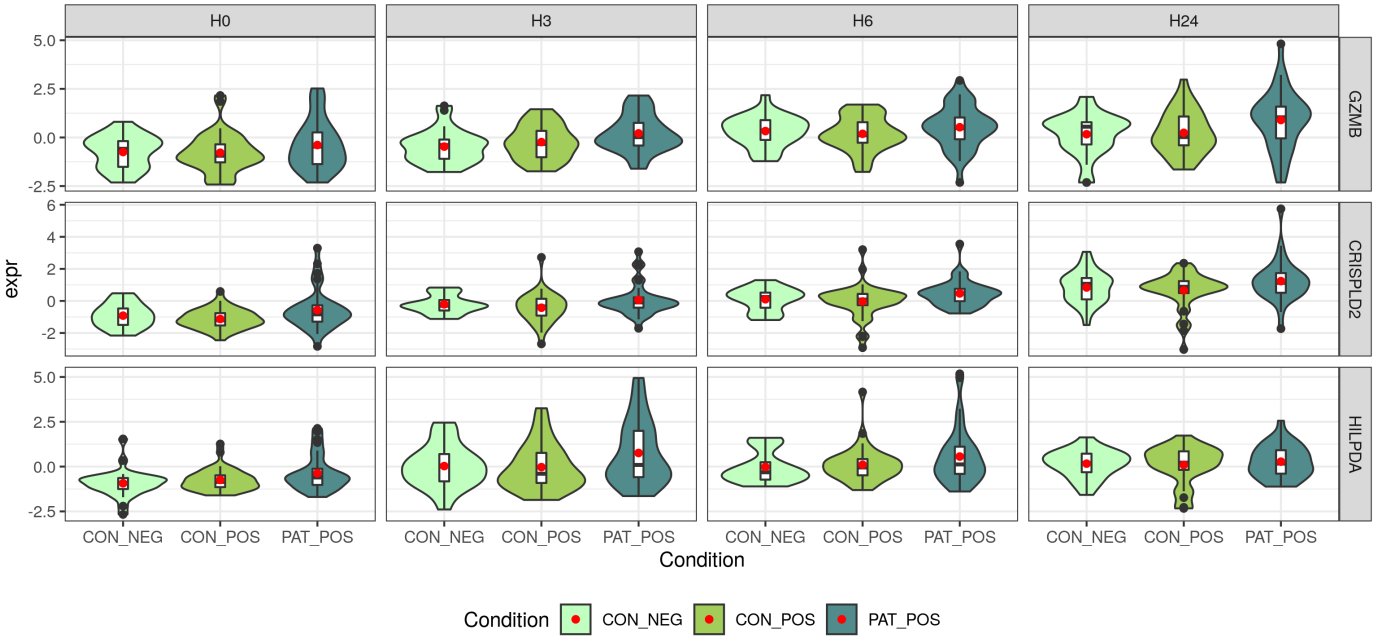


FIGURE 6.8 – Violin plot des gènes groupés avec PCOLCE2 dans la heatmap Figure 6.7. Point rouge : moyenne des expressions par condition. Expression centrée par gène et sur tous les temps affichés.

à des inflammations. Le gène HILPDA est relié aux gouttelettes lipidiques, des structures stockant les lipides intracellulaires, et à l'hypoxie (diminution du taux d'oxygène dans les tissus), un facteur important d'inflammation. Le gène CRISPLD2 est quant à lui relié à l'immunité innée, et à la formation des glycosaminoglycans (polymères de sucres), impliqués dans la structure du cartilage. Enfin, GZMB code pour pour une protéine connue pour être exprimée par les lymphocytes T cytotoxiques, les lymphocytes NK (*Natural Killer*) et les cellules dendritiques plasmacytoïdes, ayant un rôle dans les inflammations chroniques et le processus de guérison des blessures.

6.2 Recherche de termes associés aux gènes différentiellement exprimés

Pour étudier les termes de Gene Ontology (Botstein et al., 2000; Consortium, 2019) associés aux gènes que l'on a trouvés DE, et composer des groupes de gènes dans le cas des résultats des cellules dendritiques, on utilise le package `R` `gprofiler2` (Kolberg and Raudvere, 2020). Les termes KEGG (Kanehisa and Goto, 2000; Kanehisa et al., 2019; Kanehisa, 2019) et REACTOME (Jassal et al., 2020) (désignés parfois par REAC) ont aussi été analysés.

Pour chaque analyse, tous les gènes ayant une p-valeur ajustée inférieure à 0.1 sont sélectionnés et rentrés dans la fonction *gost* de `gprofiler` par ordre de p-valeur ajustée croissante. La recherche de termes enrichis se fait par rapport à la liste des gènes analysés (voir Tableau 6.1 page 78).

Le Tableau 6.4 page 87 présente le nombre de termes ayant une p-valeur ajustée significative à moins de 0.1, pour chaque comparaison, et pour chaque base de données interrogée. La dernière ligne (Gènes dans intersection) est le nombre de gènes uniques trouvés dans les intersections entre les gènes associés aux termes et les gènes entrés (DE). Ces gènes présents dans les intersections nous intéressent particulièrement : en plus d'être différentiellement exprimés, ils sont reliés à des fonctions biologiques similaires.

Base de données	D0 PPPC	H0 PPNC	H0 PCNC	H3 PCNC	H6 PPNC	Dend PPNC	Dend PPPC	Dend PCNC
REACTOME	**1**	0	**6**	0	0	**21**	**10**	**4**
KEGG	0	**2**	**28**	**2**	**1**	**5**	**2**	**19**
GO :BP	0	0	**11**	0	0	**1**	**7**	**12**
GO :MF	0	0	**5**	0	0	0	**1**	0
GO :CC	0	0	0	0	0	0	**9**	0
Gènes dans intersection	1	11	12	1	1	99	140	54

Tableau 6.4 – Nombre de termes trouvés par temps/comparaison. p-valeur ajustée demandée pour les termes : 0.1.

Beaucoup d'analyses n'ont pas donné de termes, quelle que soit la base de données interrogée. L'étude faite sur tous les échantillons dendritique est particulièrement intéressante, puisqu'elle est l'une des seules à donner des résultats pour la comparaison PPPC. Le nombre de gènes dans les intersections est aussi beaucoup plus important, ce qui nous offre un plus grand nombre de gènes d'intérêt.

Pour faciliter les représentations graphiques, les graphes représentant les termes trouvés n'affichent que les 50 premières lettres des noms des termes, ainsi que leur identifiant dans la base de données correspondante.

6.2.1 Termes des analyses des temps D0

On trouve un seul terme issu de la base de données REACTOME pour la comparaison D0-PPPC, appelé *RUNX3 Regulates Immune Response and Cell Migration, R-HSA-8949275*. L'intersection de ce terme avec la liste de gènes DE comporte un seul gène : SPP1 (*secreted phosphoprotein 1*), gène codant pour la protéine ostéopontine, servant notamment comme inhibiteur la minéralisation des os. Ce gène est aussi exprimé dans dans d'autres cellules, impliquées dans des processus de réaction immunitaire et inflammatoire, comme les cellules dendritiques et cellules T.

La Figure 6.9 page 88 présente les violin plots de l'expression de SPP1 au cours du temps, par condition. Les tableaux présentés Tableau 6.5 présentent quant à eux les différents $\log_2$ FC du gène lors des différentes comparaisons. Le gène est trouvé différentiellement exprimé seulement pour la comparaison D0-PPPC, ce qui le rend intéressant comme gène de réponse.

Cependant, les changements d'expressions constatés dans les violins plots sont minimes : la différence qui existe à D0 entre les groupes PP et PC semble être due à la présence d'outliers. On peut remarquer que la variabilité des expressions qui existe à D0 dans les trois groupes s'estompe après la période de culture de 7 jours des cellules. L'expression semble peu impactée par la stimulation au LPS, et ce dans les trois groupes.

6.2.2 Termes des analyses des temps H3

Les analyses sur les gènes DE de H3 PCNC donnent deux termes KEGG : *Mucin type O-glycan biosynthesis* (KEGG : 00512) et *Other types of O-glycan biosynthesis* (KEGG :00514), issus tous les deux du gène GALNT15 (*Polypeptide N-Acetylgalactosaminyltransferase 15)* (unique gène dans l'intersection). Malgré un $\log_2$ FC intéressant (-4.71) et deux termes KEGG lui étant associés, ce gène ne différencie les patients des contrôles (positifs ou négatifs) que par la présence d'un certain nombre d'individus non homogènes dans les populations contrôles (voir Figure 6.10 page 88 et Tableau 6.6 page 88). Le gène étant DE pour la comparaison PCNC, il est associé à l'effet B27.

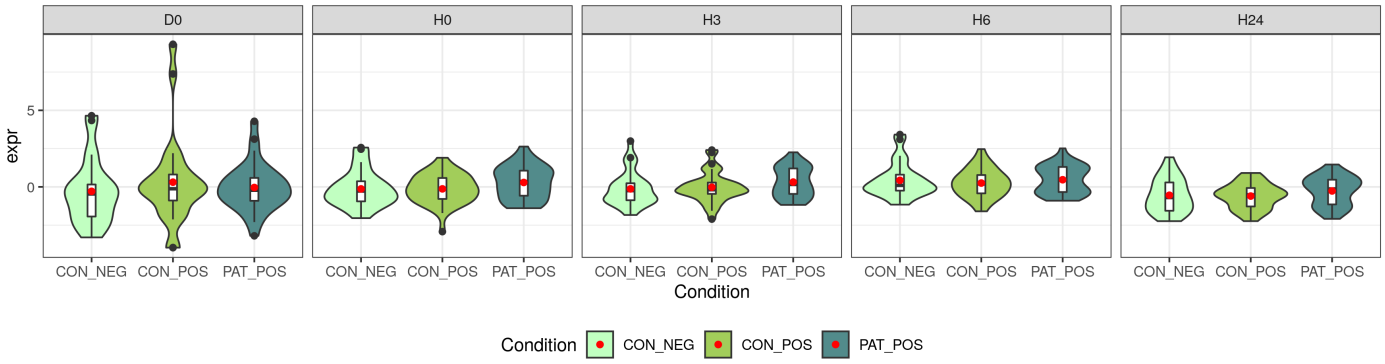


FIGURE 6.9 – Violin plot de l'expression du gène SPP1 au cours du temps. Expression centrée sur D0 et sur les temps Dend (H0, H3, H6, H24) séparément.

	H0			H3			H6			H24		
	PCNC	PPNC	PPPC	PCNC	PPNC	PPPC	PCNC	PPNC	PPPC	PCNC	PPNC	PPPC
$\log_2$ FC	-0.286	0.198	0.484	0.001	0.337	0.336	-0.466	-0.297	0.169	-0.317	0.084	0.4
FDR	0.824	0.784	1.000	1.000	1.000	0.899	1.000	0.998	0.983	1.000	1.000	1.0

	D0			Dend		
	PCNC	PPNC	**PPPC**	PCNC	PPNC	PPPC
$\log_2$ FC	2.771	-1.36	-4.130	-0.264	0.066	0.33
FDR	0.326	1.00	0.007	0.309	0.851	0.10

TABLEAU 6.5 – Valeurs de $\log_2$ FC et FDR pour le gène SPP1.

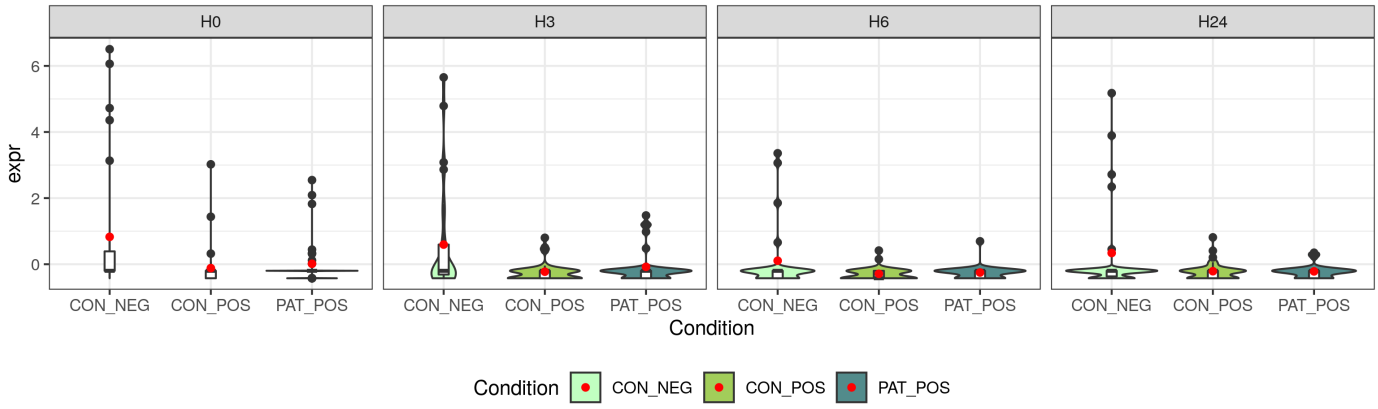


FIGURE 6.10 – Violin plot de l'expression du gène GALNT15 au cours du temps. Expression centrée sur les temps affichés.

	H0			H3			H24			Dend		
	PCNC	**PPNC**	PPPC	**PCNC**	PPNC	PPPC	**PCNC**	**PPNC**	PPPC	**PCNC**	**PPNC**	PPPC
$\log_2$ FC	-4.593	-3.667	0.926	-4.708	-3.578	1.130	-4.136	-4.488	-0.352	-4.338	-3.758	0.58
FDR	0.000	0.010	1.000	0.000	0.088	0.891	0.000	0.000	1.000	0.000	0.000	0.43

TABLEAU 6.6 – Valeurs de $\log_2$ FC et FDR pour le gène GALNT15.

6.2.3 Termes des analyses pour le temps H0

Les gènes DE pour le temps H0 donnent un certain nombre (50) de termes pour PCNC, et deux seulement pour PPNC. Dans les deux comparaisons, peu de gènes sont en fait à l'origine de la détection de ces termes. Le temps H0 correspond au J7, les cellules ayant été mises en culture pendant 7 jours. L'expression des gènes à ce temps représente une ligne de base d'expression pour les individus.

H0-PCNC. Les pathways trouvés pour la comparaison PCNC sont présentés Figure 6.11 page 90. On retrouve des termes associés à des maladies auto-immunes (Polyarthrite Rhumatoïde,...) et à l'inflammation (*inflammatory response*, terme GO :BP, Maladie Inflammatoire Chronique de l'Intestin (MICI)), alors qu'on ne compare que des contrôles entre eux (PCNC). Ces termes semblent être liés à l'effet B27 plus qu'à l'effet maladie, et confirment le rôle déjà connu de cet allèle dans le développement des maladies SpA.

H0-PPNC. Seuls deux termes KEGG ressortent des analyses pour PPNC : *ABC transporters* et *mTOR (mammalian Target of Rapamycin) signaling pathway*. Le premier pathway est largement relié aux gènes ayant le préfixe ABC (*ATP-binding cassette*), codant des protéines servant au transport d'éléments (nutriments, ions, etc.) entre cellules. Le deuxième est un pathway impliqué dans la régulation du métabolisme des cellules et de leur croissance. La rapamycine est une molécule notamment utilisée comme immunosuppresseur.

Les pathways, ainsi que la heatmap des gènes associés sont représentés sur la Figure 6.12 page 90. Globalement, ces gènes sont peu exprimés, avec des $\log_2$ FC peu élevés. Les gènes ABCB9, ULK1 et ABCC10 présentent des expressions plus fortes chez les patients et contrôles positifs que chez les contrôles négatifs. L'expression semble de plus supérieure chez les patients comparés aux contrôles positifs. Cependant, ces gènes n'ont pas été trouvés différentiellement exprimés à H0 pour PPPC.

6.2.4 Termes des analyses pour les cellules dendritiques regroupées

Les termes trouvés pour les comparaisons de tous les échantillons des cellules dendritiques sont présentés sur les Figures 6.13, 6.15, et 6.17, pages 91, 93, et 94 respectivement. Ces comparaisons sont celles qui donnent le plus de termes à étudier, avec des intersections qui comportent elles-mêmes un certain nombre de gènes d'intérêt.

Dend-PCNC. Cette comparaison donne un nombre de termes assez élevé, notamment 19 termes KEGG et 12 termes GO : BP. On retrouve de nouveau, dans les termes associés aux gènes DE pour PCNC (effet B27), les termes propres à une inflammation (Polyarthrite Rhumatoïde, MICI...), ce qui est cohérent. Plutôt que d'obtenir des termes GO :BP en relation avec une réponse inflammatoire, comme c'était le cas pour H0, on obtient ici des termes en relation avec les métaux (*copper ion, metal ion, zinc ion*, etc.). On trouve en effet un certain nombre de gènes DE dans cette comparaison DEND-PCNC dont le préfixe est MT1 (*Metallothionein*-1), dont l'expression est induite par les métaux divalents. La Figure 6.14 page 92 montre l'expression des gènes MT1 : ceux-ci sont effectivement tous sur-exprimés chez les individus B27 positifs, même à H0. Ces gènes MT1 ont aussi été trouvés différentiellement exprimés pour la comparaison PPNC, leur expression chez les patients étant aussi beaucoup plus forte comparée à celle des contrôles négatifs. Ils ont été enlevés de la liste de gènes différentiellement exprimés puisqu'ils sont communs avec PCNC. Ils n'ont pas été trouvés différentiellement exprimés dans la comparaison PPPC, ce qui indique que cette différence est rattachée à B27 plutôt qu'à la maladie. Cependant, ils ont un rapport avec l'inflammation. En effet,

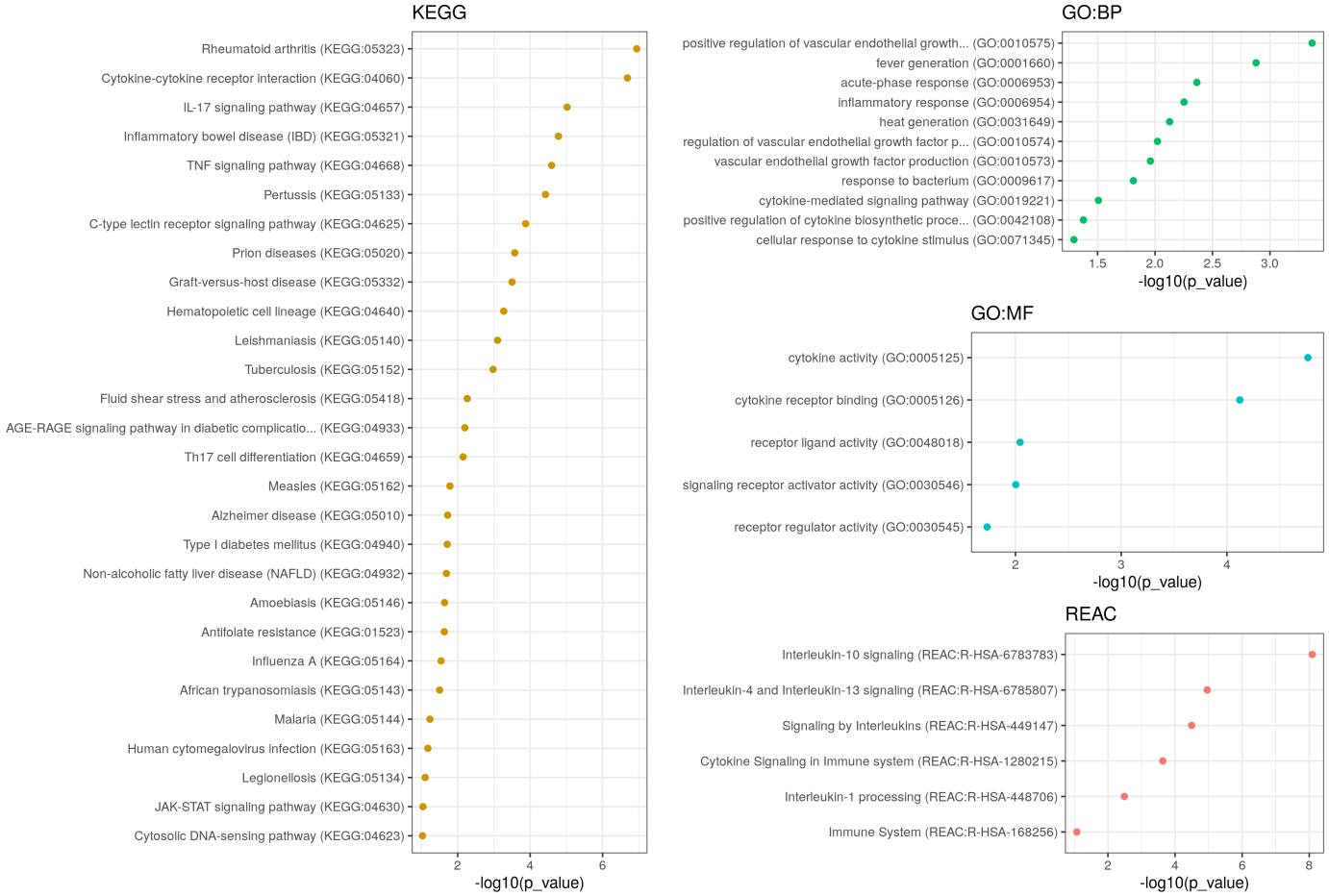


FIGURE 6.11 – **H0 PCNC.** Résultats de recherche de termes associés aux gènes DE trouvés pour ce temps et cette comparaison. Nombre de caractères des noms de termes limités à 50.

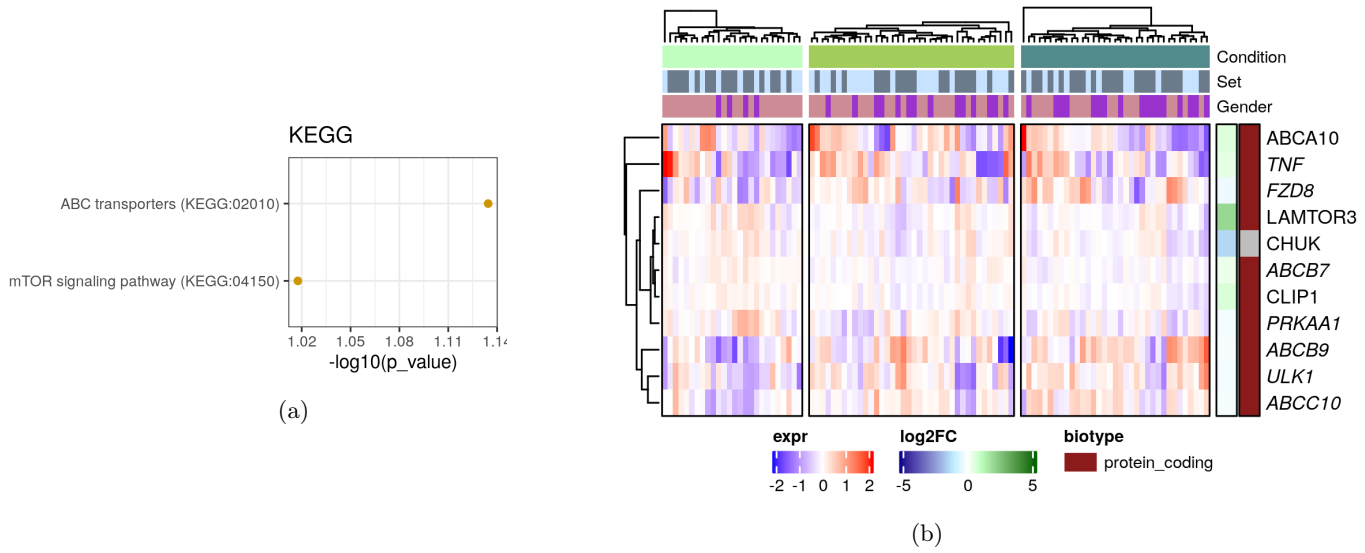


FIGURE 6.12 – **H0 PPNC. (a)** Termes KEGG associés aux gènes DE pour H0 PPNC - PCNC. Nombre de caractères des noms de termes limités à 50. **(b)** Heatmap des expressions des gènes trouvés dans les intersections des termes GO et des gènes DE pour PPNC H0. Les gènes en italiques ont un $|\log_2 \mathrm{FC}| < 0.5$.

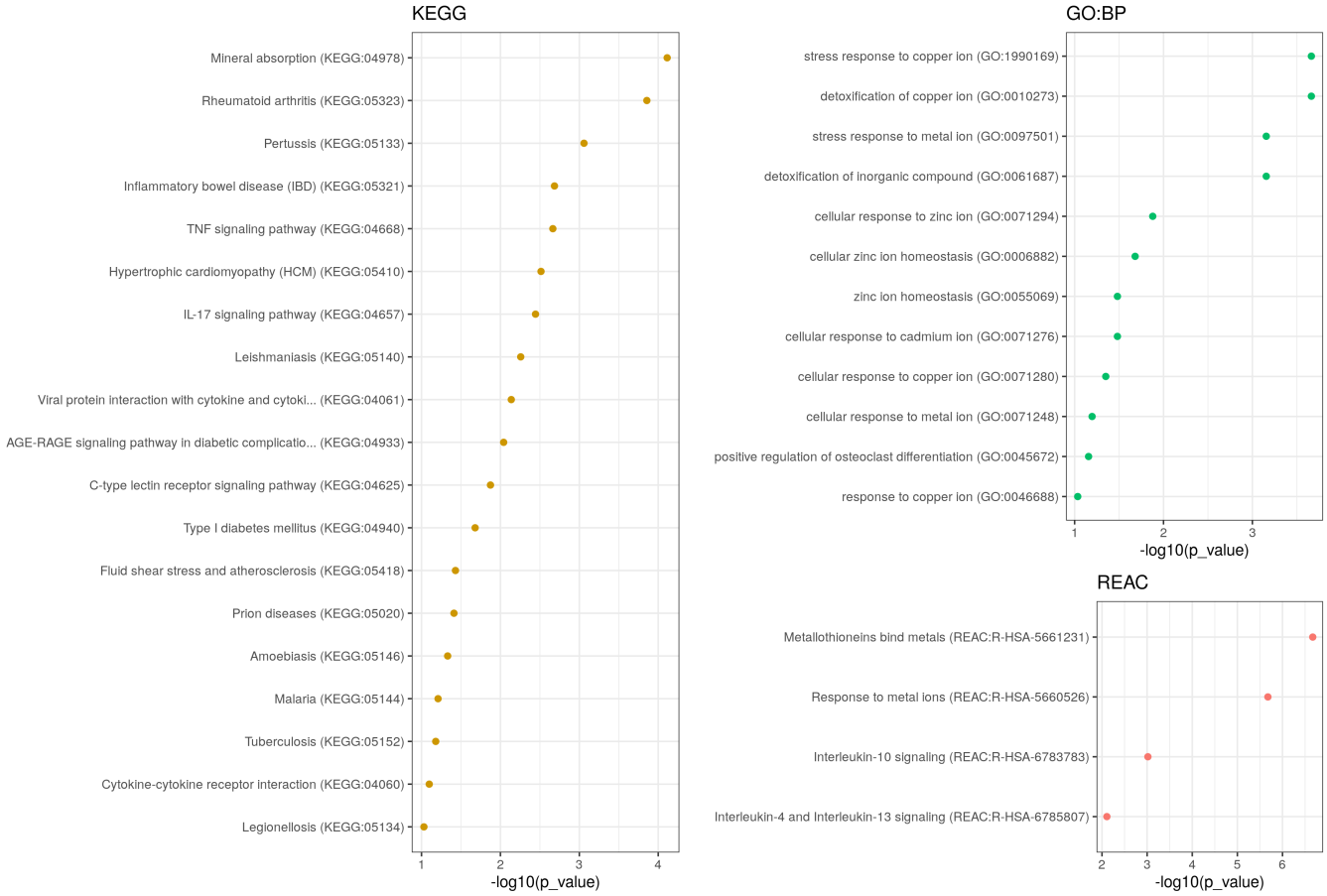


FIGURE 6.13 – **DEND PCNC** Termes associés aux gènes DE. Nombre de caractères des noms de termes limités à 50. Les termes sont ordonnés selon leur $-\log_{10} \mathrm{p}-$ valeur sur la p-valeur ajustée trouvée par le package `R` `gprofiler2`.

leur expression est modifiée par la stimulation, avec une sur-expression à H24 en comparaison de H0. Cette expression semble retardée en comparaison de la stimulation au LPS effectuée à H0, mais elle est commune à toutes les conditions des individus tout en étant plus marquée chez les patients. On remarque cependant que la différence d'expression (sur-expression chez les individus B27 positifs) se rencontre dès le temps H0, soit avant la stimulation, pour les gènes MT1G, MT1H et MT1X.

DEND-PPNC. Cette comparaison donne 21 termes REACTOME, 5 KEGG et un seul terme GO :BP. On trouve ici des termes en rapports avec la *transcription* et la *translation*. Dans les 99 gènes présents dans les intersections, une catégorie de gènes domine : les gènes ayant comme préfixe les lettres RPS (*Ribosomal Protein S*), un groupe de gènes impliqués dans la traduction des ARN messagers. On retrouve aussi quelques MRPS (*Mitochondrial Ribosomal Proteins*). Leurs expressions sont très faibles, à l'exception de celle de RPS16, ainsi que leur $\log_2$ FC, comme présenté sur la heatmap Figure 6.16 page 93. L'évolution de l'expression des RPS suit pour la plupart le chemin inverse de l'évolution des MT1 tracés plus tôt, l'expression à H24 étant la plus faible dans toutes les conditions : les expressions sont plus faibles à H24 qu'avant la stimulation, après une augmentation à H3 et H6. La différence entre les patients et les contrôles négatifs s'observe à H3, où une partie des patients a une expression plus faible, alors que tous les contrôles sont sur-exprimés en comparaison de leur expression à H3.

Pour les MRPS, l'expression à H24 est similaire à celle de H0 dans les groupes, à l'exception de MRPS16 et MRPS35, qui ont un niveau plus bas. Tous les MRPS observent une baisse d'expression

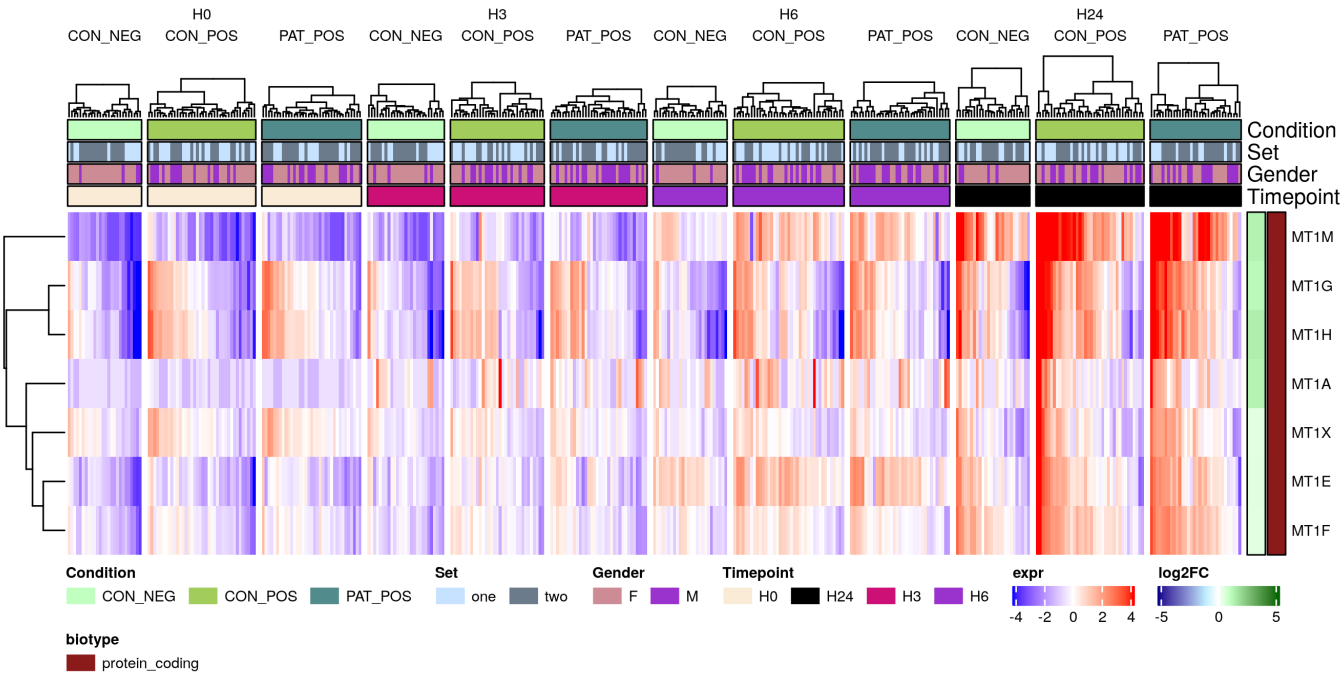


FIGURE 6.14 – **DEND PCNC** - Gènes avec préfixe MT1

à H6.

DEND-PPPC. Cette comparaison renvoie le plus de gènes dans les intersections avec les termes, avec 140 gènes associés aux termes. On trouve des termes dans toutes les bases de données consultées. Les termes associés à la mitochondrie prédominent, reliés aux gènes (M)RPL (*(Mitochondrial) Ribosomal Protein L*) et (M)RPS. Comme pour les deux autres comparaisons, on représente leur expression au cours du temps pour toutes les conditions Figure 6.18 page 94. À l'image des RPS identifiés dans la comparaison DEND-PPNC, on trouve des expressions peu élevées, associées à des $\log_2$ FC modestes. On remarque, à H3, que le groupe des patients a des expressions plus faibles que les groupes contrôles pour les gènes MRPL1, MRPL35, MRPS23, MRPL43 et MRPL57.

6.3 Conclusion de cette analyse

Dans ce chapitre, nous avons effectué une analyse différentielle des données de transcriptomiques, en vue d'identifier des gènes propres à l'effet maladie, dont l'impact viendrait s'ajouter celui des gènes liés à l'effet B27. Trois comparaisons ont été étudiées : PCNC, comparaison entre les groupes contrôles (effet B27), PPNC, comparaison entre le groupe des patients positifs et celui des contrôles négatifs (effet B27 et maladie conjoints) et PPPC, comparaison entre groupes d'individus B27 positifs (effet maladie seulement), à chacun des temps, ainsi que pour tous les temps des cellules dendritiques. Les gènes obtenus pour la comparaison PCNC ont été considérés comme étant reliés à l'effet B27 et ont été retirés des listes de gènes PPNC et PPPC.

Plus d'une centaine de gènes propres à l'effet maladie ont été identifiés, notamment grâce aux comparaisons utilisant tous les temps dendritiques. Ces gènes ont fait l'objet d'une recherche de termes associés, qui a mené à des groupes et pathways pouvant être affectés chez les malades : une dérégulation associée aux ribosomes et à la mitochondrie viendrait s'ajouter à un dérèglement de la réponse des gènes MT1, associée, elle, à l'effet B27, ce qui pourrait contribuer à l'apparition de la maladie.

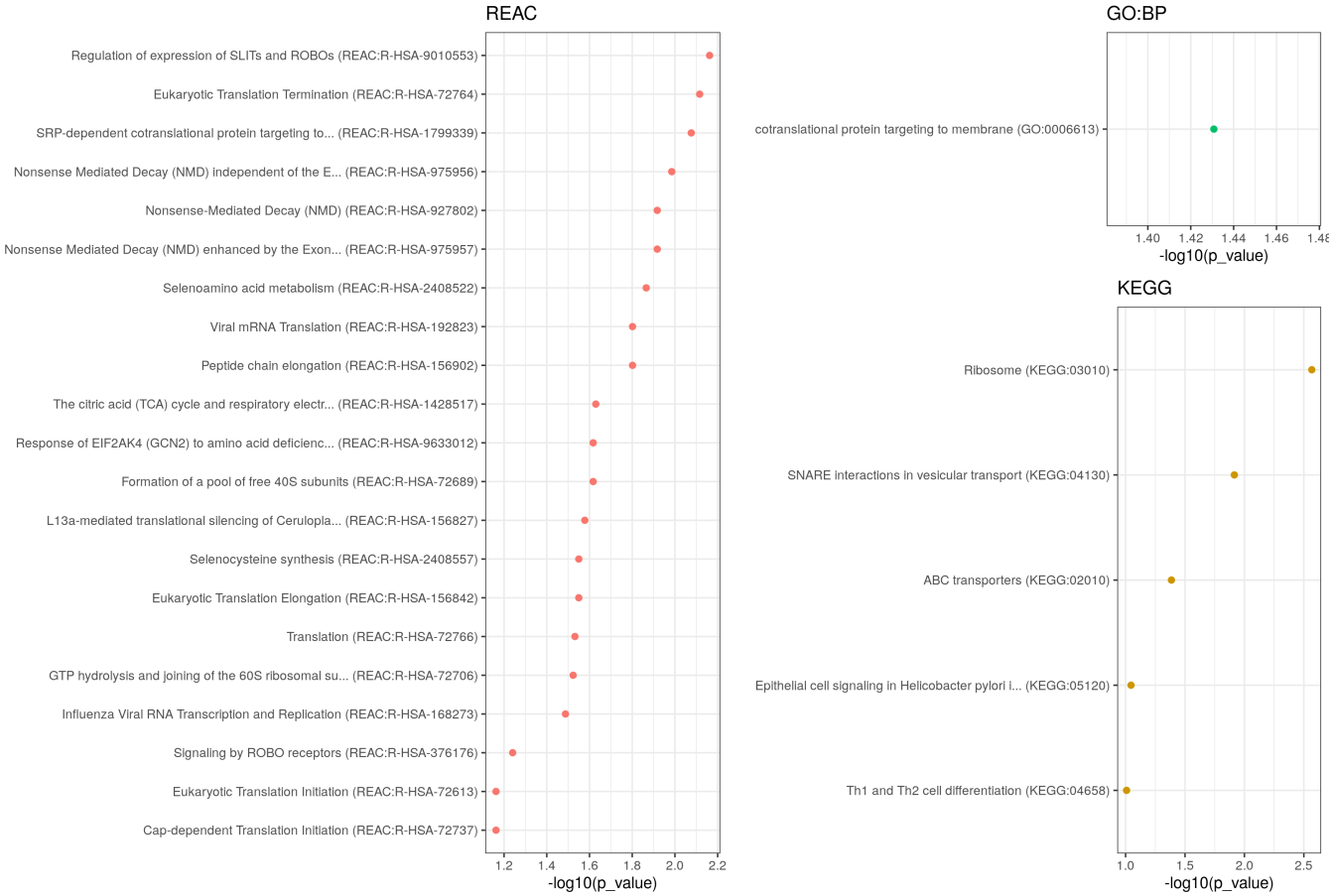


FIGURE 6.15 – **DEND PPNC** - Termes associés aux gènes DE. Nombre de caractères des noms de termes limités à 50. Les termes sont ordonnés selon leur $-\log_{10}$ p – valeur sur la p-valeur ajustée trouvée par le package `R` `gprofiler2`.

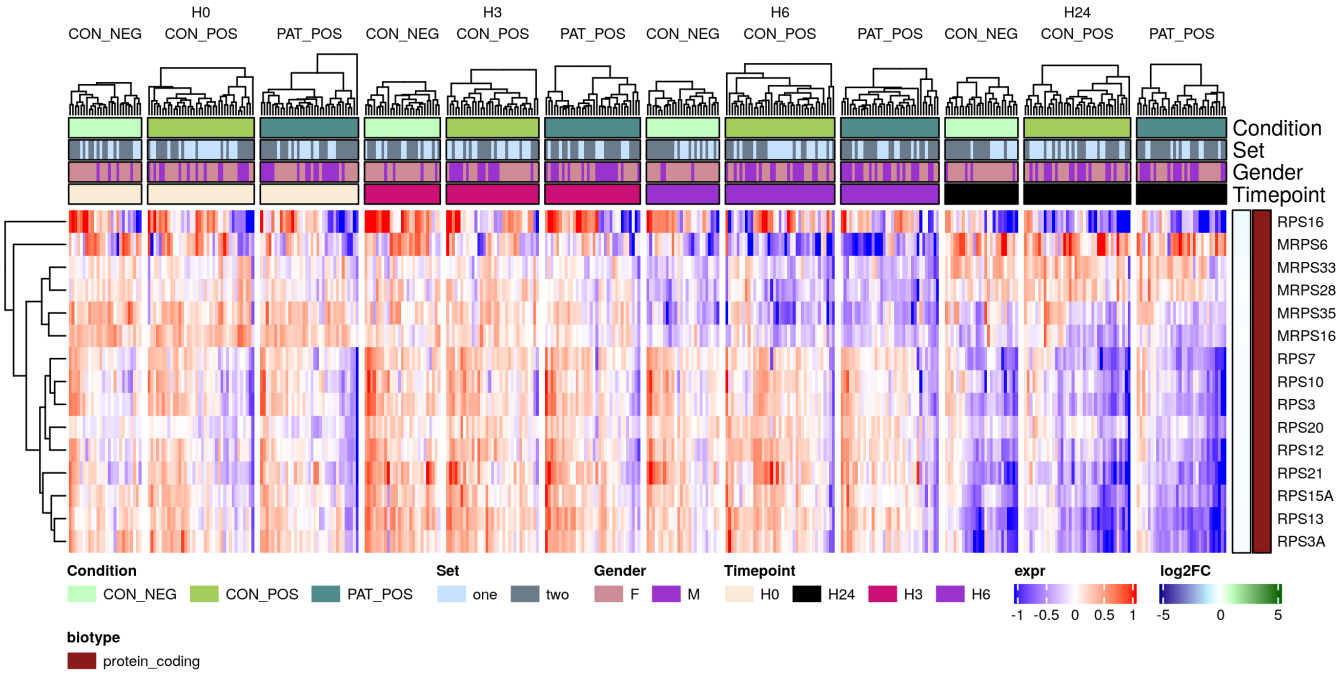


FIGURE 6.16 – DEND PPNC - Gènes RPS et MRPS.

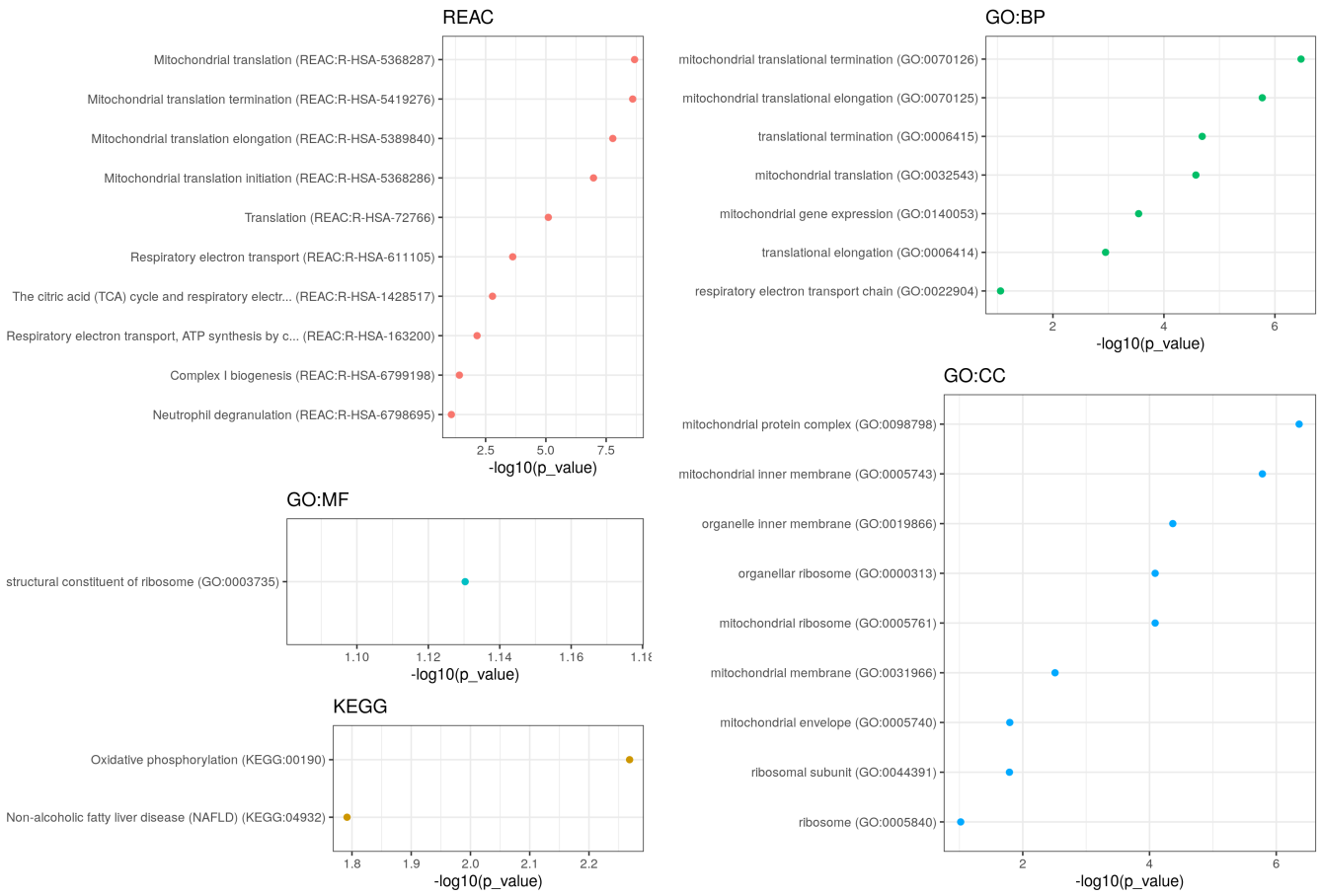


FIGURE 6.17 – **DEND PPPC** - Termes associés aux gènes DE. Nombre de caractères des noms de termes limités à 50. Les termes sont ordonnés selon leur $-\log_{10} p -$ valeur sur la p-valeur ajustée trouvée par le package R `gprofiler2`.

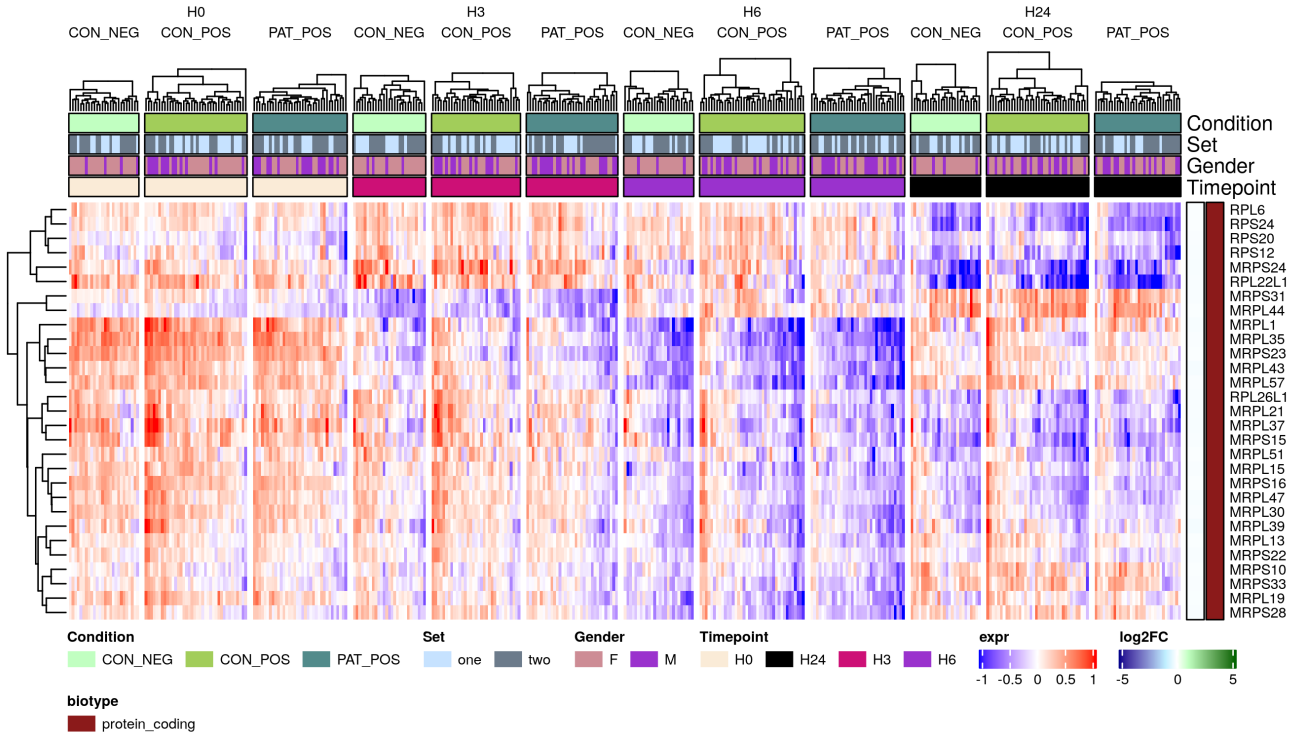


FIGURE 6.18 – DEND PPPC - Gènes RPS, MRPS et MRPL.

Ces groupes de gènes identifiés, ainsi que les dérèglement occasionnés, doivent faire l'objet d'analyses plus poussées et de validations expérimentales. Les données de cette étude permettent aussi d'envisager de créer des réseaux d'interaction dirigés, puisque nous avons à disposition des données temporelles.

Cette analyse a aussi révélé la grande hétérogénéité des données : des sous-groupes de patients et de contrôles sont en général visibles sur les heatmaps des expressions des gènes. Ces groupes ne correspondent cependant pas à une variable clinique connue, ce qui rend leur analyse complexe. On pourrait envisager d'étudier plus finement ces groupes de patients et de contrôles en cherchant à spécifier leurs caractéristiques au niveau de l'expression des gènes, voire tenter de relier les groupes de gènes avec ceux des patients, par l'utilisation du co-clustering.

Chapitre 7

Analyse conjointe des données de métagénomique et de transcriptomique

Table des matières

Dans cette partie, on cherche à mettre en lien les jeux de données de métagénomique et de transcritpomique. Plus particulièrement, on cherche à identifier des groupes d'espèces métagénomiques et de gènes qui interagissent entre eux, voire peuvent expliquer le développement de la maladie, une hypothèse étant que la flore intestinale participe à la stimulation du système immunitaire qui a été mimée in vitro par la stimulation par le LPS.

7.1 Traitement des données

Tous les individus n'ayant pas les 5 échantillons de transcriptomique, on prend pour chaque temps les individus ayant à la fois un échantillon de métagénomique et un échantillon de transcriptomique à ce temps. On obtient les effectifs présentés dans le Tableau 7.1 page 97.

Dans ce chapitre, on ne sélectionne que les individus B27+ pour l'apprentissage des randomForests, pour obtenir des scores de variables permettant d'identifier les variables impliquées dans le développement de la maladie.

Traitement des données de métagénomique. Le traitement des données de métagénomique s'effectue sur tous les individus disponibles. On a à disposition les abondances de 1 575 espèces métagénomiques (abrégé en MSP, pour *Metagenomic Species*), pour 68 individus. On supprime les MSP ayant une abondance de 0 dans plus de 85% des individus. On se ramène à l'étude de 748 MSP.

Temps	D0	H0	H3	H6	H24
Nb Individus	54	61	59	65	59
CON NEG	15	19	19	19	18
CON POS	11	13	11	16	14
PAT POS	28	29	29	30	27
Nb Gènes	946	1 043	1 034	1 036	1 044
Nb Gènes après traitement	780	907	915	934	847

Tableau 7.1 – Nombre d'individus en commun entre les tables de métagénomiques et de transcriptomiques pour chaque temps et nombre de gènes dans les tables de transcriptomique.

	D0		H0		H3		H6		H24	
	Meta.	Trans.	Meta.	Trans.	Meta.	Trans.	Meta.	Trans.	Meta.	Trans.
Meta.	1.00	0.42	1.00	0.38	1.00	0.38	1.00	0.35	1.00	0.37
Trans.	0.42	1.00	0.38	1.00	0.38	1.00	0.35	1.00	0.37	1.00

Tableau 7.2 – Coefficient RV issus des AFM des tables de transcriptomique et de métagénomique à chaque temps.

Traitement des données de transcriptomique. Dans ce chapitre, on utilise les données transcriptomiques corrigées pour l'effet set par *removeBatchEffect*, en log *count per millions*.

Pour réduire le nombre de gènes à étudier, mais garder un nombre important de gènes potentiels, on sélectionne les gènes trouvés dans les analyses différentielles, avec une p-valeur ajustée inférieure à 0.1 pour les comparaisons PPPC à tous les temps (y compris Dend), en enlevant les gènes DE pour les comparaisons PCNC, ce qui donne 1 105 gènes potentiels. Ces gènes n'ont pas forcément passé les filtres de pré-traitement des données qui ont été appliqués temps par temps, et parfois ne se retrouvent pas dans les tables analysées pour chaque temps séparément. Le nombre de gènes total pour chaque comparaison est indiqué dans le Tableau 7.1 page 97.

7.2 Analyse Factorielle Multiple des deux jeux de données

On réalise une AFM des deux jeux de données pour situer les deux tables l'une par rapport à l'autre, à tous les temps disponibles, en prenant pour les données de transcriptomique les gènes sélectionnés. Les données sont centrées et réduites pour l'AFM. La Figure 7.1 page 98 montre pour chacun des temps les individus selon les coordonnées trouvées par l'AFM. Les individus des trois conditions ont été pris en compte, bien que l'on travaille sur les gènes permettant de séparer les conditions PAT POS de CON POS. Les pourcentages de variance expliquée sont assez faibles (moins de 10% sur le premier axe). Il ne semble pas y avoir de séparation claire des groupes sur les axes. La variabilité des individus semble encore une fois poser problème ici. Les axes 3 et 4 présentent le même schéma et ne conduisent pas à une séparation claire en groupes de conditions.

Le Tableau 7.2 page 97 présente les coefficients RV (voir Section 4.3.5) calculés lors des AFM, pour chaque temps. Ces coefficients sont assez faibles, ne dépassant pas 0.45. On peut s'attendre à ce qu'il y ait un lien entre ces deux tables, mais aussi qu'il soit faible.

L'AFM peut déjà être vue, en soi, comme une intégration des données, cependant cette intégration ne montre qu'une petite séparation des groupes. Les RV nous permettent de conclure qu'il existe un lien entre les tables. Il est nécessaire d'aller plus loin dans la sélection de variables pour établir un véritable lien entre les gènes et MSP. On utilise pour cela des random forests.

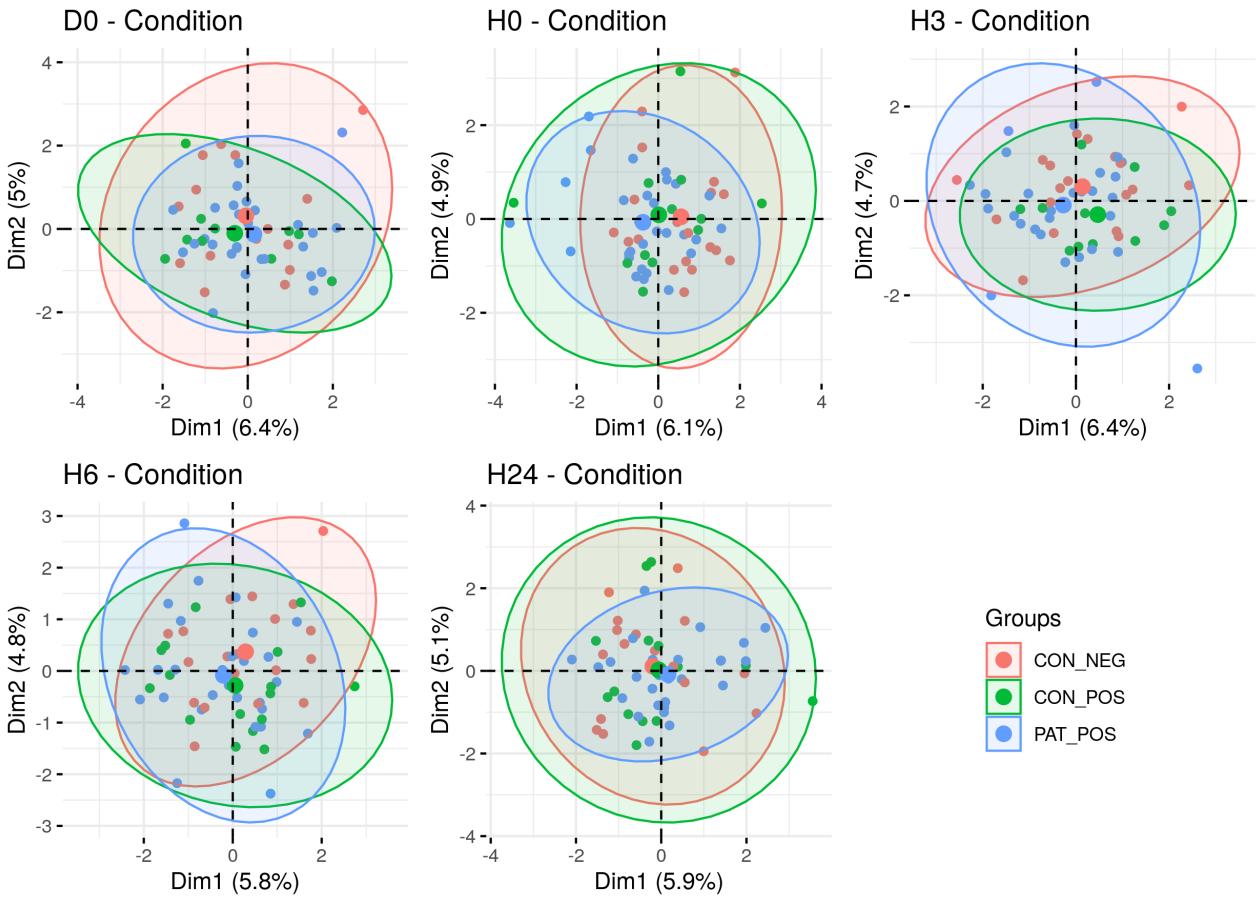


FIGURE 7.1 – Répartition des individus et des tables, selon les scores des AFM temps par temps, sur les données métagénomiques et transcriptomiques, sur les deux premiers axes. Les centroïdes des ellipses à 95% de chaque condition sont indiquées par des points de couleurs de taille supérieure.

7.3 Sélection de variables par random forest

7.3.1 Random Forests

Les *random forests*, ou forêts aléatoires, ont été introduites par Breiman (2001), sur la base d'un précédent travail de Ho (1995). Les random forests sont une combinaison d'arbres de décision. Chaque arbre est calculé sur un sous-échantillon des individus, mais aussi sur un sous-échantillon des variables. Les résultats de ces arbres sont moyennés, et permettent d'obtenir une mesure de l'importance d'une variable dans la classification.

Un arbre de décision est basé sur un algorithme de partitionnement des données, chaque nœud d'un arbre représentant une séparation (*split*) entre les données. Les nœuds n'étant pas des *splits* sont les nœuds finaux, donnant la classification des données obtenues sur l'échantillon des variables et des individus. Chaque split-nœud représente une classification utilisant un certain nombre des variables, déterminant une partition des individus.

Pour réaliser les random forests de ce chapitre, on utilise le package `R randomForest` (Liaw and Wiener, 2002). Les données sont centrées avant échantillonnage.

Les arbres créés sur des variables corrélées produiront la même classification, et n'apportent pas d'information. Les importances de deux variables très corrélées seront de plus similaires, ce qui conduira à les sélectionner. Pour pallier le problème de ces variables, on pré-traite les données de transcriptomique : pour chaque groupe de variables corrélées à plus de 0.8 en valeur absolue, on

ne laisse dans la table qu'un seul représentant de ce groupe. Les données de métagénomique sont beaucoup moins corrélées et n'ont pas subit de pré-traitement.

Les random forests utilisent le *bagging*, similaire au boostrap mais qui comporte une étape d'agrégation des modèles issus des bootstrap, avec une étape de sélection des individus, mais aussi des variables. Si l'on note B le nombre d'arbres crées, les random forests se décomposent de la manière suivante : Pour chaque itération $1, ..., B$:

1. Tirer, avec remise, un sous-échantillon des n_B individus
2. Entraîner un arbre de classification sur ce sous-échantillon. Pour chaque *split* de l'arbre, sélectionner un certain nombre de variables aléatoirement.
3. Calculer la précision du modèle sur les individus qui n'ont pas été sélectionnés (score de classification, ou *accuracy*)

Précision. On définit la précision d'un modèle par la part des individus prédits dans la bonne classe grâce au modèle.

Prédiction \ Référence	Classe 1	Classe 2
Classe 1	N_{11}	N_{12}
Classe 2	N_{21}	N_{22}

Tableau 7.3 – Matrice de confusion

On se réfère pour ces formules au Tableau 7.3. Les effectifs N_{11} et N_{22} représentent les individus bien classés. On définit la précision (*Accuracy*) et la précision équilibrée (*Balanced Accuracy*) par les formules suivantes :

$$Acc = \frac{N_{11} + N_{22}}{N_{11} + N_{12} + N_{21} + N_{22}} \qquad Acc_b = \frac{1}{2}\left(\frac{N_{11}}{N_{11} + N_{12}} + \frac{N_{22}}{N_{21} + N_{22}}\right) \tag{7.1}$$

Importance des variables. On définit l'erreur *out of bag* (OOBE) comme l'erreur de prédiction obtenue pour chaque individu lorsque la prédiction est calculée sur les arbres n'ayant pas pris cet individu en compte dans l'apprentissage. L'erreur OOB est calculée pour chacune des variables.

Les valeurs de cette variables sont ensuite permutées aléatoirement. On calcule l'erreur OOB obtenue à partir des prédictions obtenues sur ces données où les valeurs de la variables ont été permutées. Cette nouvelle erreur est notée OOBE_p.

Pour une forêt constituée de B arbres, on a pour une variable j :

$$Imp_j = \frac{\text{OOBE}_p - \text{OOBE}}{B} \tag{7.2}$$

L'importance est ensuite normalisée par l'écart-type des scores obtenus sur chaque échantillon.

7.3.2 Procédure utilisée

En prenant les individus patients et contrôles positifs, on a pour chaque temps une représentation plus importante de la classe patients. Ce déséquilibre peut poser des problèmes dans l'estimation d'un modèle : si tous les individus sont prédits comme appartenant à la classe patients, le modèle aura un score de précision correct, mais une performance peu intéressante, et les variables qui le composent ne devraient pas être retenues.

La procédure classique est de réaliser plusieurs échantillonnages des individus, en plusieurs jeux d'apprentissage et de validation. Les importances des variables sont ensuite moyennées pour pouvoir procéder à la sélection. Cela permet aussi d'éviter le sur-apprentissage : les modèles peuvent obtenir une bonne *accuracy* sur l'échantillon d'apprentissage, mais être médiocre lorsqu'on les teste sur un nouveau jeu de données. On ne cherche pas forcément des variables prédictives, mais on ne veut pas non plus sélectionner des variables qui soient adaptées seulement à un sous-échantillon d'individus, et dont on ne retrouve pas de caractère discriminant lorsqu'on l'étudie sur tous les individus.

On procède à un sous-échantillonnage, répété 50 fois, avec la procédure suivante :

1. Échantillonnage des individus, en respectant les proportions Patients/Contrôles, sélection de 70% des individus pour l'apprentissage (échantillon e_{train}, avec n_{train} individus), et 30% pour la validation (e_{test}, de cardinal n_{test}).
2. Procédure d'apprentissage,
   (a) Apprentissage : random forest : 500 arbres sont ajustés, avec une sélection de 50 variables par nœuds.
   (b) Test : test du random forest précédent sur s_{test}.
   (c) Modèle retenu si la *balanced accuracy* est supérieure à 0.5. Si ce n'est pas le cas, toutes les importances des variables sont mises à 0.
3. Moyenne des importances des variables des modèles sur les 50 itérations. Les variables non testées/retenues ont une importance à 0.
4. Sélection des 50 variables ayant les moyennes les plus grandes.

On obtient 50 importances pour chacune des variables, certaines ayant été mises à 0 si les modèles crées sont jugés peu pertinents.

7.4 Résultats Random Forests

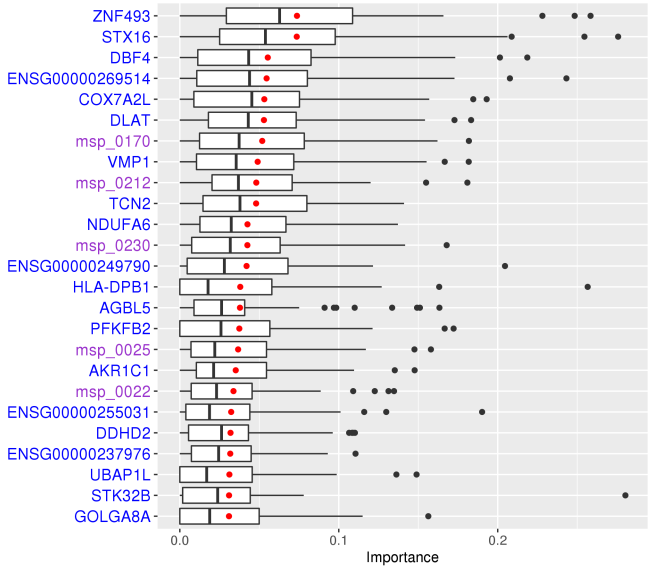
7.4.1 Sélection des variables

Pour chacun des temps, on a à disposition les moyennes des importances sur les 50 itérations effectuées. On classe les variables par ordre de moyenne décroissante. Les 25 premières variables pour chaque temps sont représentées sur la Figure 7.2 page 101, avec les boxplots des importances. Ces 25 variables sont peut-être corrélées avec des groupes plus importants qui ont été retirés de l'analyse, et qui auraient eu des résultats les amenant dans ce classement.

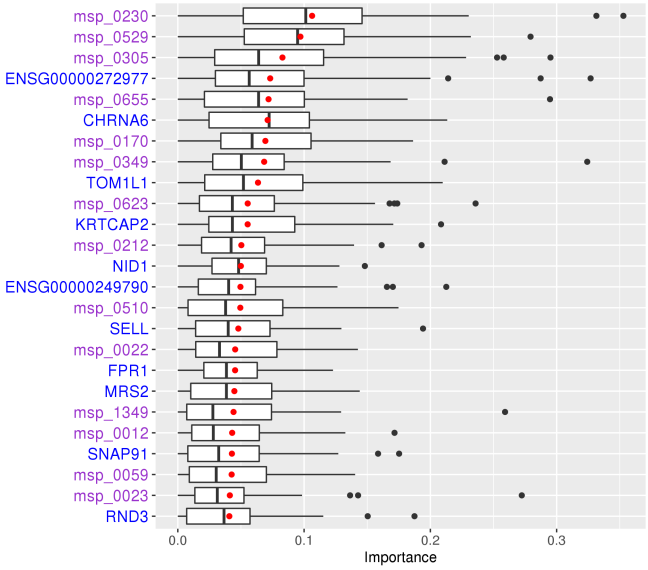
Beaucoup d'itérations n'ont pas donné de résultats de précision satisfaisants et ont été mises à 0, les médianes sont donc à 0. Pour les modèles ayant donné des résultats avec une précision supérieure à 0.5, les importances des variables sont faibles, pour la plupart ne dépassant pas les 0.05, avec une grande variabilité. On retrouve certains gènes identifiés précédemment dans l'analyse différentielle comme intéressants, notamment MRPS28 au temps H3.

On travaille dans la suite sur les 50 premières variables sélectionnées, en rajoutant les gènes corrélés à 0.8 qui avaient été enlevés dans pré-processing des données. Le nombre de gènes et de MSP trouvés est présenté dans le Tableau 7.2 (f). La colonne "Imp Mean" du tableau désigne l'importance "seuil" placée à chaque temps, soit l'importance moyenne trouvée pour la 50eme variable des random forests. La plupart des variables sélectionnées sont des gènes, mais la répartition varie beaucoup d'un temps à l'autre. Au temps H3, on sélectionne seulement 9 MSP, contre 42 gènes, par exemple, alors qu'on trouve 26 MSP dans la sélection pour H0.

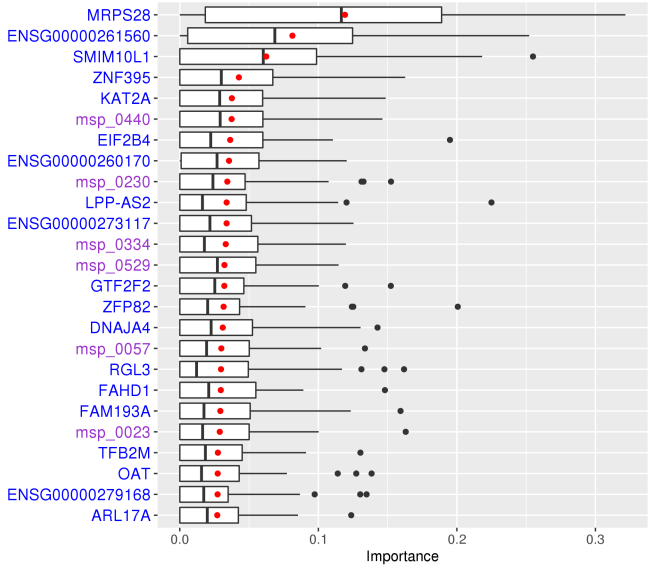
La Figure 7.3 page 102 montre les nombre de variables en commun entre les listes de gènes et MSP trouvées par les random forest pour chacun des temps. Cette figure s'interprète comme un



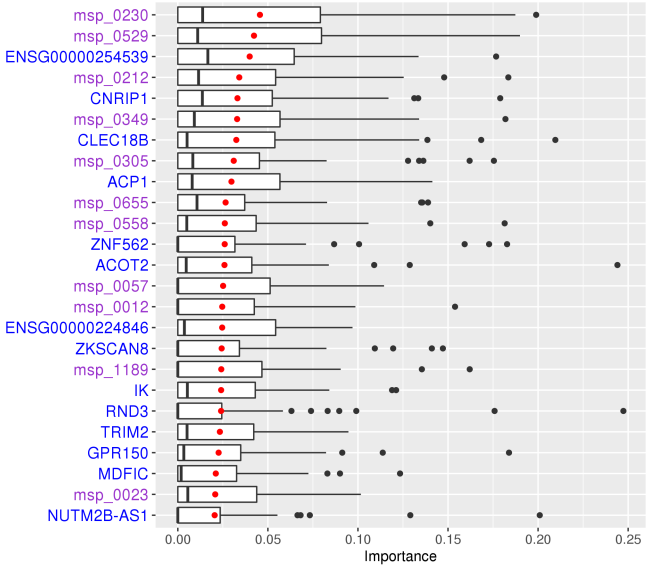
(a) D0



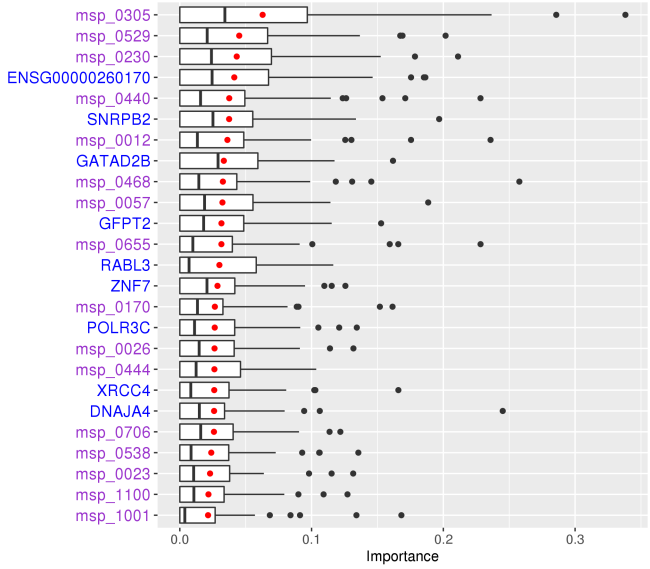
(b) H0



(c) H3



(d) H6



(e) H24

	Gènes	MSP	Total	Imp Mean
D0	42	10	52	0.0253
H0	25	26	51	0.0254
H3	42	9	51	0.0212
H6	31	19	50	0.0163
H24	23	28	51	0.0163

(f) Nombre de gènes et MSP sélectionnées par temps. La dernière colonne désigne le seuil d'importance moyenne utilisé pour sélectionner ces éléments.

FIGURE 7.2 – Résultats des Random Forests : boxplots des importances des 25 premières variables pour chaque temps, sur les 50 itérations. Les MSP sont indiquées en violet, les gènes en bleu. Le point rouge représente la moyenne, mesure retenue pour la sélection.

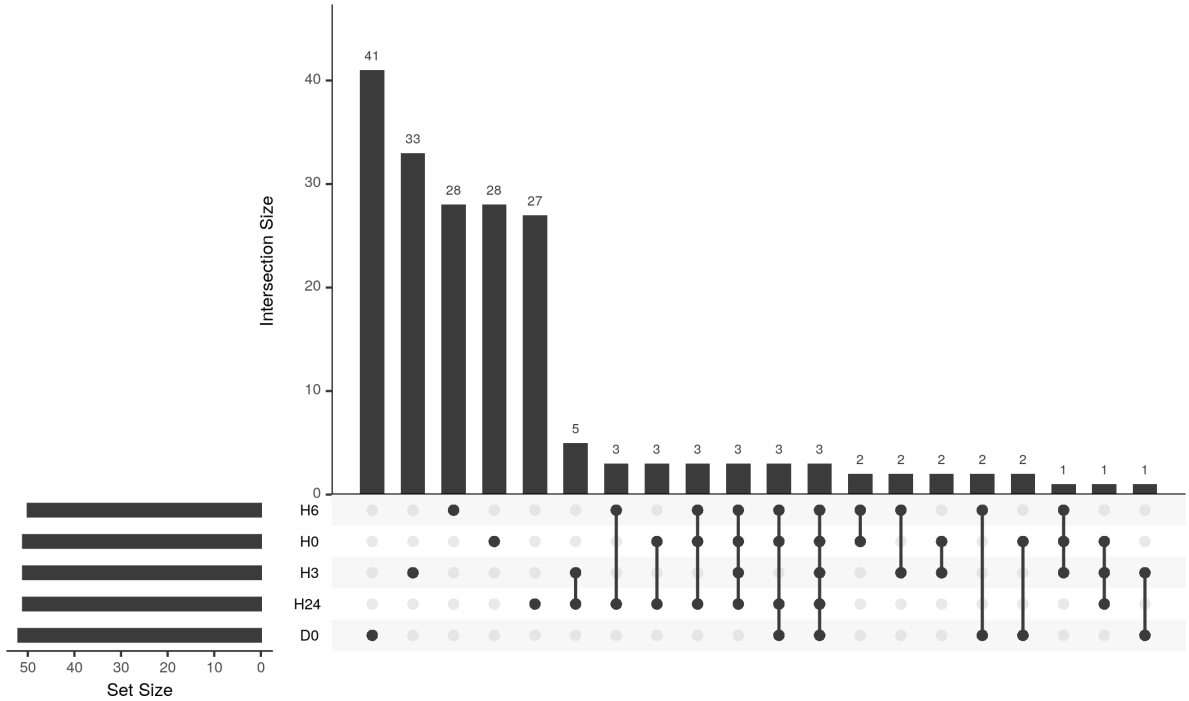


FIGURE 7.3 – Nombre de gènes et MSP dans les intersections (qui ne se retrouvent pas dans les autres intersections). Les temps sont rangés en fonction du nombre d'entités sélectionnées.

diagramme de Venn : les nombres indiqués désignent les gènes qui sont uniques à l'intersection considérée. On peut remarquer que la grande majorité des entités sélectionnées sont uniques à leur temps. Trois MSP sont communes à tous les temps : msp_0230 msp_0529 et msp_0212. Si l'on prend seulement les temps dendritiques, on trouve aussi les MSP msp_0023, msp_0057 et msp_0440 en commun. Aucun gène n'est commun à tous les temps, ou partagé entre les temps dendritiques.

Certaines de ces MSP ont déjà été identifiées précédemment comme étant caractéristiques de maladies inflammatoires. Au niveau du genus, Dialister (msp_0212) a déjà été trouvé en relation avec les maladies de type spondyloarthrites (Tito et al., 2017). Une baisse significative des Alistipes (msp_0230) et des Roseburia (msp_0057) a été observée chez des patients atteints de la maladie de Crohn (MICI), en comparaison à des sujet sains (Willing et al., 2010). Scher et al. (2015) retrouvent cette baisse des Alistipes également chez des patients atteints de psoriasis.

7.4.2 Réseaux multi-omiques

Pour visualiser les relations qui peuvent exister entre les gènes et les MSP sélectionnés à l'étape précédente, nous avons tracé les réseaux de corrélations partielles leur étant associés. Pour cela, on utilise le Glasso (voir Section 3.1.5), avec le package `R huge`. On sélectionne la pénalité et le réseau final par le critère stars (Liu et al., 2010), pour 20 sous-échantillonnage et un seuil de 0.1 sur l'instabilité. Les réseaux obtenus sont présentés Figure 7.4 et Figure 7.5 pages 103 et 104. Les statistiques des réseaux sont affichées dans le Tableau 7.5 page 103. Les nœuds qui ne sont pas reliés (ayant un degré de 0) ont été retirés des réseaux et ne comptent pas dans les statistiques.

Sur le nombre de variables sélectionnées, peu sont retirées : la plupart des variables a donc au moins une relation avec une autre variable. Les réseaux retenus sont très sparses, la majorité ayant un degré de parcimonie de plus de 0.9 : il y a peu de connexions entre les tables. De plus, les

	genus	family	order	class	phylum	superkingdom
msp_0212	Dialister	Veillonellaceae	Veillonellales	Negativicutes	Firmicutes	Bacteria
msp_0230	Alistipes	Rikenellaceae	Bacteroidales	Bacteroidia	Bacteroidetes	Bacteria
msp_0529	Eubacterium	Eubacteriaceae	Clostridiales	Clostridia	Firmicutes	Bacteria
msp_0023	Bacteroides	Bacteroidaceae	Bacteroidales	Bacteroidia	Bacteroidetes	Bacteria
msp_0057	Roseburia	Lachnospiraceae	Clostridiales	Clostridia	Firmicutes	Bacteria
msp_0440	Eubacterium	Eubacteriaceae	Clostridiales	Clostridia	Firmicutes	Bacteria

	species
msp_0212	Dialister invisus
msp_0230	Alistipes inops == Tidjanibacter massiliensis
msp_0529	Eubacterium sp. CAG :581
msp_0023	Bacteroides caccae
msp_0057	Roseburia sp. CAG :45 & sp. 2789STDY5608886
msp_0440	unclassified Eubacterium (genus, Eubacterium ramulus)

Tableau 7.4 – Correspondance des identifiants de MSP pour les espèces métagénomiques trouvées communes entre tous les temps, et entre les temps dend.

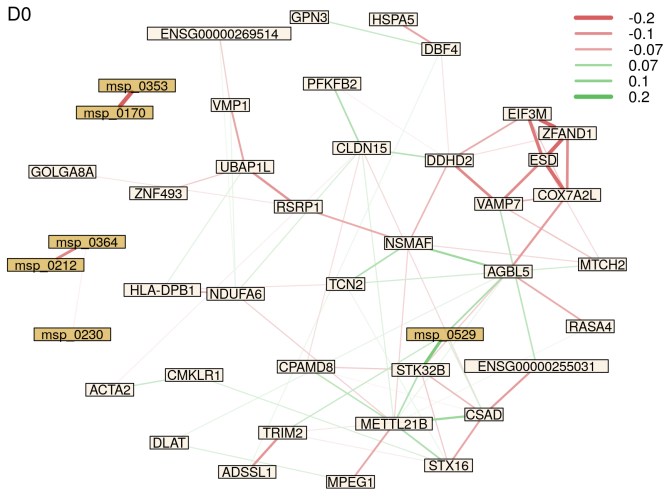


Figure 7.4 – **D0** - Réseau de corrélations partielles obtenu par Glasso et critère STARS, sur les entités sélectionnées par random forests. Les nœuds ayant un degré de 0 ont été supprimés. Les gènes sont représentés en beige, les MSP en jaune.

	Nœuds	Arcs	Arcs Positifs	Arcs Négatifs	*Sparsité*	Pénalité Glasso
H0	41	76	17	59	0.91	0.34
H3	43	132	40	92	0.86	0.40
H6	43	93	29	64	0.90	0.31
H24	43	72	32	40	0.92	0.33
D0	42	76	28	48	0.91	0.42

Tableau 7.5 – Statistiques des réseaux trouvés entre les entités sélectionnées par random forests, à chaque temps.

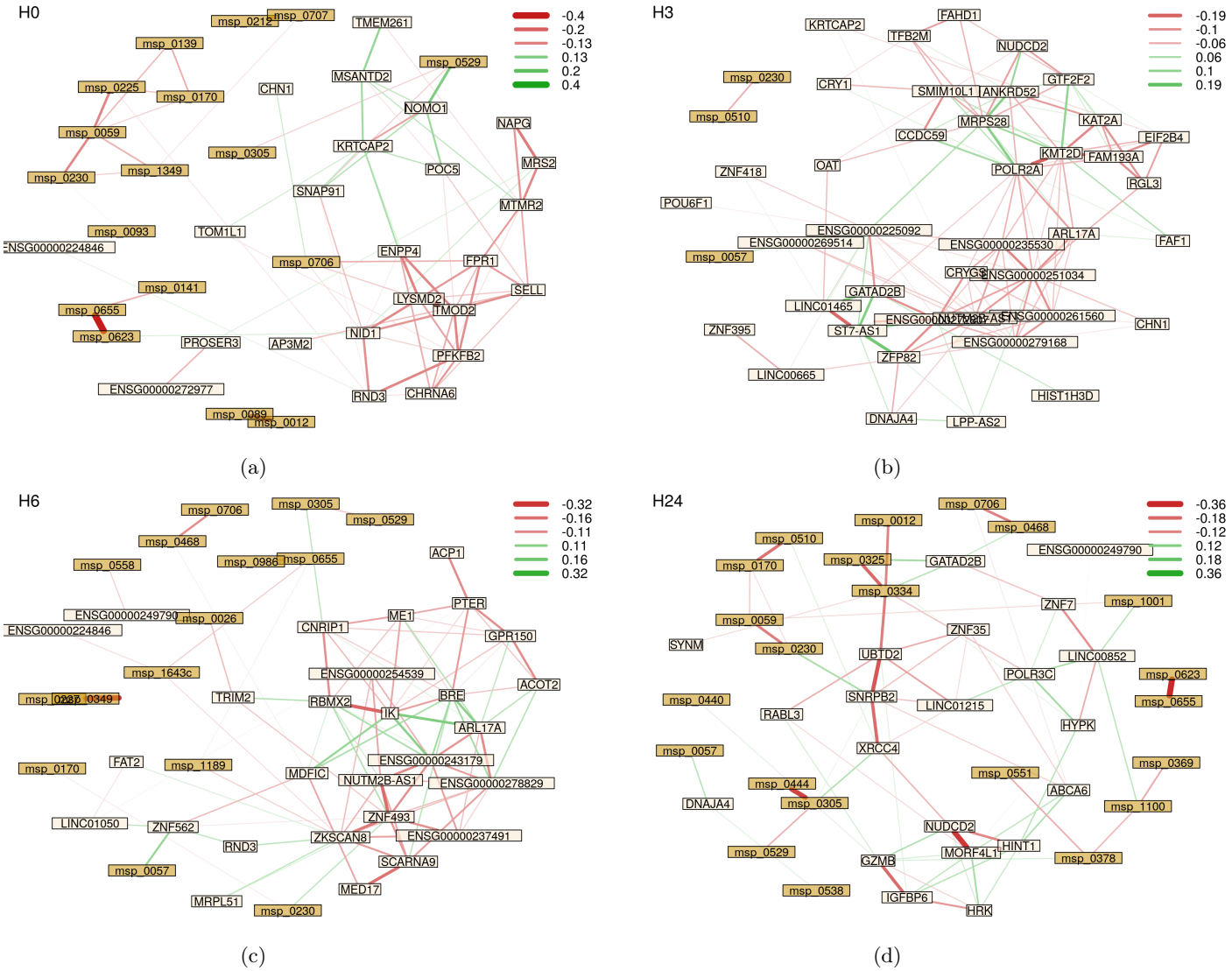


FIGURE 7.5 – **DEND** - Réseaux de corrélations partielles obtenus par Glasso et critère STARS, sur les entités sélectionnées par random Forests. Les nœuds ayant un degré de 0 ont été supprimés. Les gènes sont représentés en beige, les MSP en jaune.

connexions retenues sont en majorité faibles – inférieures à 0.5, voire à 0.2, en valeur absolue, dans certains graphes – et négatives : le nombre d'arcs positifs ne représente même pas la moitié des connexions. La plupart des connexions s'effectue intra-table, avec des groupes de gènes ou de MSP très connectés, ce qui se remarque par une séparation distincte en deux groupes sur les graphes, quel que soit le temps. Les groupes de gènes sont plus connectés que ceux de MSP, que ce soit par l'intensité des relations ou par leur fréquence, et apparaissent de façon plus visible sur les graphes.

Cependant, il existe des connexions inter-tables, ce qui nous permet de mettre en relation des MSP avec des gènes. Ces connexions sont encore plus faibles que celles intra-tables, mais elles sont présentes.

7.5 Conclusions et perspectives

Dans ce chapitre, nous avons essayé de mettre en relation les gènes différentiellement exprimés trouvés précédemment, pour la comparaison PPPC, avec les espèces métagénomiques présentes chez

les individus. Les résultats présentés ici sont encore préliminaires, l'intégration des données étant à un stade exploratoire dans le projet. Le nombre de gènes DE, supérieur au millier, que l'on a obtenus demande à ce que l'on poursuive la sélection de variables.

Les réseaux obtenus dans ce chapitre montrent l'intérêt potentiel de l'étude des liens entre les tables de transcriptomique et de métagénomique pour espérer mieux comprendre certains aspects du développement de la maladie. Ceci étant, les importances de variables sont très faibles, et peut-être que la procédure de sélection doit être revue et améliorée. Les random forests ont été la première méthode utilisée. On peut envisager d'en appliquer d'autres, comme par exemple la SPLS (*Sparse Partial Least Squares*) (Chun and Keleş, 2010), qui est adaptée au multi-tables, et facilement accessible via le package `R` `mixOmics` (Rohart et al., 2017).

Le design de l'étude nous permet de comparer différentes catégories d'individus, porteurs ou non de l'allèle B27, et d'identifier de potentiels acteurs qui agissent avec, ou même indépendamment, de ce facteur bien établi. Les individus ont ainsi été séparés tout au long de l'analyse en trois conditions : PP, Patients Positifs, PC, Contrôles Positifs et NC, Contrôles Négatifs. De plus, on dispose de plusieurs points de temps pour observer l'évolution des expressions des gènes à la suite d'une stimulation.

L'hétérogénéité des individus, quel que soit le groupe de conditions considéré, est importante. Dans les analyses différentielles effectuées et les heatmaps affichées, des sous-groupes distincts étaient visibles. Ces sous-groupes ne correspondaient pas à des variables cliniques connues. Cette variabilité nous amène à considérer qu'une étude du clustering des individus, voire de co-clustering individus/gènes, pour former des sous-groupes, notamment au niveau des patients, peut être une prochaine étape.

Le développement de la maladie SpA est complexe, et ne permet pas d'être expliqué par un seul gène ou MSP, mais par une combinaison de plusieurs facteurs, rattachés aux différentes -omiques, mais peut-être aussi environnementaux. La combinaisons de plusieurs facteurs, ainsi que l'hétérogénéité des individus, amène à penser que la construction d'un score de risque, basé sur le niveau d'expression de certains groupes de gènes, ainsi que sur la présence ou l'absence de certaines MSP, est envisageable. Les scores sont déjà utilisés sur la base des SNP (*polygenic risk scores*), pour la construction d'un score allélique, pour plusieurs types de maladies (Dudbridge, 2013; Mavaddat et al., 2019).

Chapitre 8

Conclusion et perspectives

Si l'idée d'étudier les relations entre les diverses couches d'omiques date d'il y a plusieurs décennies, l'intégration de données -omiques est un domaine relativement récent, qui a pu notamment se développer grâce à l'arrivée des technologies de séquençage haut-débit. Le domaine des données -omiques est en constante évolution, avec de nouvelles méthodes de séquençage et d'analyses développées chaque année, pour aider à l'analyse et l'intégration des données -omiques. Cette thèse s'inscrit dans ce contexte d'analyse de données -omiques, avec une attention particulière aux méthodes de clustering et d'inférence de réseaux. L'objectif était de développer de nouvelles méthodes, ou d'améliorer des méthodes déjà existantes, pour faciliter l'analyse de ces données, notamment en vue de leur intégration. La thèse se divise en deux parties : une première partie portant sur le développement méthodologique qui présente trois méthodes pour l'analyse de données -omiques, et une partie présentant une application de méthodes déjà existantes à un jeu de données de transcriptomique et de métagénomique.

Résumé des contributions. La première méthode présentée concerne la problématique de la création d'un clustering hiérarchique dans un contexte où le nombre de feuilles est très élevé. En utilisant une méthode basée sur le clustering convexe, ainsi qu'un algorithme d'agrégation d'arbres, nous parvenons à créer un arbre dans des situations où les méthodes classiques ne peuvent être utilisées, grâce à une complexité sous-quadratique. L'algorithme d'agrégation d'arbres a fait l'objet d'une publication et un package, `mergeTrees` disponible sur le CRAN a été créé. Cette méthode a été appliquée sur données simulées et réelles. La méthode de clustering convexe, appelée FA-MT, a été appliquée sur données simulées et doit faire l'objet d'une application sur données réelles. Les résultats sur simulations montrent que les arbres obtenus reflètent la partition simulée dans les données lorsque ces dernières sont peu bruitées. Une variante spectrale a été proposée pour pallier la baisse de performances rencontrée dans le cas de données avec une plus grande variance, en utilisant un noyau gaussien et une approximation de Nyström. La complexité totale de cette combinaison de méthodes reste sous-quadratique et permet, encore une fois, de réaliser des arbres avec un grand nombre de feuilles. De plus, les performances de la variante spectrale sont grandement améliorées et sont prometteuses.

La deuxième méthode proposée permet d'estimer la matrice d'adjacence d'un réseau, ainsi que des groupes de nœuds, en respectant une partition *a priori* des nœuds donnée par l'utilisateur. Cette estimation se base sur l'itération d'un *Graphical Lasso* (Glasso), pour estimer une matrice de support creuse du réseau, et de la combinaison *Stochastic Block Model/Latent Block Model* (SBM/LBM) pour estimer les groupes des nœuds à partir de la matrice de support. La pénalité du Glasso est adaptée en fonction des probabilités de connexions trouvées par l'étape SBM/LBM, pour favoriser l'apparition d'arêtes entre des nœuds d'un même groupe. La méthode, appelée *Janine* est disponible dans un package `R` du même nom, sur github. Cette approche a été testée sur un jeu de données simulées et un jeu de données réelles multi-omiques, provenant de patientes atteintes de cancer du sein. Dans les deux cas, elle a été comparée à un Glasso simple et à une combinaison de Glasso et SBM itérés, sur le modèle de *Janine*. On trouve, dans les simulations, des résultats cohérents avec

la partition intégrée aux tables de données. L'application sur données réelles montre l'intérêt de la méthode dans un contexte multi-omiques, avec la création d'un réseau sur des tables de protéines, de miRNA et de RNA-seq dont les blocs respectent cette partition. La méthode doit encore faire l'objet d'améliorations, notamment au niveau de l'estimation de la matrice de précision : pour l'instant, nous nous sommes concentrés sur l'estimation de la matrice d'adjacence, binaire. Notre méthode estime une matrice de précision, grâce à l'étape du Glasso, cependant, le Glasso simple estime pour l'instant plus correctement l'intensité des arcs sur la simulation effectuée. Ce point pourrait être développé en utilisant la combinaison SBM/LBM avec une loi non binaire, par exemple la loi normale, pour estimer une matrice continue.

Le troisième chapitre de la partie développement méthodologique présente une méthode d'intégration de données basée sur une combinaison de la *Multidimensional Scaling* (MDS) et de l'Analyse Factorielle Multiple (AFM). En utilisant ces deux méthodes, toutes les données qui sont obtenues sous forme de tables ou de matrices de distance/dissimilarité peuvent être intégrées grâce à l'AFM. Les matrices de dissimilarité sont projetées grâce à la MDS sur un espace qui permet leur intégration avec d'autres tables de données dans la deuxième étape (AFM). L'utilisation de l'AFM permet de plus d'intégrer des données quantitatives et qualitatives. On s'est placé plus particulièrement dans le contexte spécifique où les données à disposition sont obtenues sous forme d'arbres ou de réseaux. Ces deux représentations peuvent être transformées en matrice de dissimilarité, la première en utilisant la distance cophénétique, la deuxième en utilisant une métrique sur les graphes, dans notre cas la *shortest path distance* a été utilisée. Nous étudions les performances de cette succession MDS/AFM dans le cas de données simulées, ainsi que dans le cas de données réelles. Les simulations effectuées montrent que nous sommes capables d'identifier les groupes de tables créés, ce qui permet de proposer un moyen de juger de la proximité des arbres et des réseaux. Enfin, nous proposons d'utiliser les résultats de l'AFM et du regroupement de tables pour créer des consensus d'arbres et de réseaux. Cette combinaison offre beaucoup de possibilités quant à l'intégration de données. Nous pourrions notamment envisager d'intégrer des données d'annotations fonctionnelles, avec des réseaux de gènes ou de protéines.

La deuxième partie de la thèse concerne un projet d'analyse de données de transcriptomique et d'intégration avec des données de métagénomique, dans le cadre d'un projet de recherche, Multi-Spa, sur la Spondyloarthrite Ankylosante. Cette maladie possède une hérédité non négligeable, et son développement est en partie relié à la présence de l'allèle HLA-B27. Le but de cette analyse est de trouver des facteurs pouvant expliquer l'apparition de cette maladie, en dehors de l'allèle B27 ou complémentaire à son action, ainsi que relier les gènes potentiels identifiés aux espèces métagénomiques présentes dans le microbiote. En effet, une dysbiose plus marquée chez les patients avait été remarquée lors d'une précédente analyse des données de métagénomique. L'analyse différentielle des données de transcriptomique a donné près d'un millier de gènes d'intérêt, qui peuvent être regroupés en groupes fonctionnels. L'analyse conjointe des données a été effectuée à l'aide d'une AFM, en premier lieu, pour étudier le lien entre les tables de transcriptomique et métagénomique, puis à l'aide de random forests. Cette analyse a permis de repérer des gènes et espèces métagénomiques pouvant avoir un lien avec le développement de la maladie. Ces analyses seront complétées et approfondies dans la suite du projet Multi-Spa. Ce projet sera notamment poursuivi avec l'arrivée de nouveaux échantillons permettant d'effectuer des analyses plus robustes.

Perspectives. Malgré le grand nombre de méthodes développées chaque année pour aider à l'analyse de données -omiques, leur analyse et leur intégration reste toujours une question de recherche d'actualité. Les limites rencontrées concernant la grande dimension, l'hétérogénéité des données et le fait qu'elles soient bruitées risquent de s'intensifier avec l'arrivée de technologies encore plus

performantes, et d'une masse de données allant en grandissant.

L'inférence de réseaux, par exemple, est toujours inenvisageable sur de larges jeux de données, ce qui amène à la question de la sélection des variables dont on veut inférer le réseau. Cette question a été très peu abordée dans le cadre du développement méthodologique de cette thèse : lors des applications requérant la création de réseaux, les variables ont été sélectionnées en petit nombre et selon une approche standard d'analyse différentielle, que ce soit pour limiter le temps de la procédure, ou pour permettre l'affichage du réseau. Les approches proposées peuvent faire l'objet d'améliorations pour aider à analyser des réseaux plus grands.

En pratique, cette sélection peut se révéler plus ardue, du fait de la grande hétérogénéité des individus qui peut exister dans l'expression des différentes -omiques, comme constaté dans le projet Multi-Spa en transcriptomique. La sélection de variables demande de nombreux réplicats pour réellement estimer l'effet d'une variable.

La question du traitement des données hétérogènes a été abordée dans deux chapitres de cette thèse. L'algorithme d'agrégation d'arbres permet en effet d'obtenir un consensus de clusterings hiérarchiques réalisés sur des données de types différents, et les réseaux et arbres mentionnés dans le dernier chapitre peuvent être obtenus là encore à partir de n'importe quel jeu de données. On a supposé que ces objets étaient déjà connus lors de l'application des méthodes. Les deux autres méthodes, *janine* et FA-MT, demandent à ce que nous nous placions dans le cadre de données gaussiennes. Des transformations existent et sont couramment utilisées pour rapprocher des données non gaussiennes d'une loi normale, comme la transformation $x \mapsto \log_2(x+1)$ des données de comptages en RNA-seq, voire de considérer que les données sont continues lorsqu'elle sont sous forme de facteurs (SNP). Cependant, avec ces transformations, on perd la caractéristique des données de base, et intégrer directement les différentes lois dans les modèles permettrait d'obtenir un cadre d'application plus général et direct des méthodes. Ce cadre général peut être partiellement obtenu avec l'utilisation de l'AFM, puisqu'elle permet d'intégrer des données continues et qualitatives.

Dans un cadre plus appliqué, les méthodes présentées ici ne prennent que peu en compte ce que l'on connaît des processus biologiques et de la littérature associée. Intégrer des annotations fonctionnelles, sous forme d'*a priori* ou aux côtés des tables de données devrait être plus approfondi pour permettre d'utiliser les informations déjà obtenues et disponibles. Cela permettrait d'obtenir des réseaux ou des clusterings plus représentatifs des processus, et peut-être plus robustes puisqu'ils dépendraient d'informations confirmées dans la littérature. Ceci pourrait également être fait à partir de données phénotypiques ou cliniques. Dans la plupart des applications, les variables utilisées sont sélectionnées par rapport à une ou plusieurs variables phénotypiques, mais celles-ci ne sont plus utilisées lors de la création des graphes ou des arbres.

Bibliographie

Abdi, H., Williams, L. J., and Valentin, D. (2013). Multiple factor analysis : principal component analysis for multitable and multiblock data sets. *WIREs Computational Statistics*, 5(2) :149–179.

Abdi, H., Williams, L. J., Valentin, D., and Bennani-Dosse, M. (2012). Statis and distatis : optimum multitable principal component analysis and three way metric multidimensional scaling. *Wiley Interdisciplinary Reviews : Computational Statistics*, 4(2) :124–167.

Adams, E. N. (1972). Consensus techniques and the comparison of taxonomic trees. *Systematic Zoology*, 21(4) :390–397.

Allman, E. S., Matias, C., and Rhodes, J. A. (2011). Parameter identifiability in a class of random graph mixture models. *Journal of Statistical Planning and Inference*, 141(5) :1719–1736.

Allman, E. S., Matias, C., Rhodes, J. A., et al. (2009). Identifiability of parameters in latent structure models with many observed variables. *The Annals of Statistics*, 37(6A) :3099–3132.

Altenbuchinger, M., Weihs, A., Quackenbush, J., Grabe, H. J., and Zacharias, H. U. (2020). Gaussian and mixed graphical models as (multi-) omics data analysis tools. *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, 1863(6) :194418.

Ambroise, C., Chiquet, J., Matias, C., et al. (2009). Inferring sparse gaussian graphical models with latent structure. *Electronic Journal of Statistics*, 3 :205–238.

Anderson, E. (1935). The irises of the gaspe peninsula. *Bull. Am. Iris Soc.*, 59 :2–5.

Arthur, D. and Vassilvitskii, S. (2006). k-means++ : The advantages of careful seeding. Technical report, Stanford.

Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. (2008a). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine learning research*, 9(Mar) :485–516.

Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. (2008b). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine learning research*, 9(Mar) :485–516.

Banf, M. and Rhee, S. Y. (2017). Computational inference of gene regulatory networks : approaches, limitations and opportunities. *Biochimica et Biophysica Acta (BBA)-Gene Regulatory Mechanisms*, 1860(1) :41–52.

Bar-Hen, A., Barbillon, P., and Donnet, S. (2018). Block model for multipartite networks. Applications in ecology and ethnobiology. working paper or preprint.

Barthélemy, J. P. and McMorris, F. R. (1986). The median procedure for n-trees. *Journal of Classification*, 3 :329–334.

Benjamini, Y. and Hochberg, Y. (1995). Controlling the false discovery rate : a practical and powerful approach to multiple testing. *Journal of the Royal statistical society : series B (Methodological)*, 57(1) :289–300.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE transactions on pattern analysis and machine intelligence*, 22(7) :719–725.

Borg, I. and Groenen, P. J. (2005). *Modern multidimensional scaling : Theory and applications*. Springer Science & Business Media.

Botstein, D., Cherry, J. M., Ashburner, M., Ball, C., Blake, J., Butler, H., Davis, A., Dolinski, K., Dwight, S., Eppig, J., et al. (2000). Gene ontology : tool for the unification of biology. *Nat genet*, 25(1) :25–9.

Breban, M., Costantino, F., André, C., Chiocchia, G., and Garchon, H.-J. (2015). Revisiting mhc genes in spondyloarthritis. *Current Rheumatology Reports*, 17(6) :40.

Breban, M., Said-Nahal, R., Hugot, J.-P., and Miceli-Richard, C. (2003). Familial and genetic aspects of spondyloarthropathy. *Rheumatic diseases clinics of North America*, 29(3) :575.

Breban, M., Tap, J., Leboime, A., Said-Nahal, R., Langella, P., Chiocchia, G., Furet, J.-P., and Sokol, H. (2017). Faecal microbiota study reveals specific dysbiosis in spondyloarthritis. *Annals of the rheumatic diseases*, 76(9) :1614–1622.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1) :5–32.

Bryant, D., Francis, A. R., and Steel, M. (2016). Can we "future-proof" consensus trees ? *Systematic biology*, 66 4 :611–619.

Cailliez, F. (1983). The analytical solution of the additive constant problem. *Psychometrika*, 48(2) :305–308.

Chen, G. K., Chi, E. C., Ranola, J. M. O., and Lange, K. (2015). Convex clustering : An attractive alternative to hierarchical clustering. *PLoS Comput Biol*, 11(5) :e1004228.

Chen, J. and Chen, Z. (2008). Extended bayesian information criteria for model selection with large model spaces. *Biometrika*, 95(3) :759–771.

Chi, E. C. and Lange, K. (2015). Splitting methods for convex clustering. *Journal of Computational and Graphical Statistics*, 24(4) :994–1013.

Chiquet, J. (2019). univarclust r package.

Chiquet, J. (2020). janine r package.

Chiquet, J., Gutierrez, P., and Rigaill, G. (2017). Fast tree inference with weighted fusion penalties. *Journal of Computational and Graphical Statistics*, 26(1) :205–216.

Chiquet, J., Rigaill, G., and Sundqvist, M. (2020). *aricode : Efficient Computations of Standard Clustering Comparison Measures*. R package version 1.0.0.

Cho, H., Berger, B., and Peng, J. (2016). Reconstructing causal biological networks through active learning. *PloS one*, 11(3) :e0150611.

Chun, H. and Keleş, S. (2010). Sparse partial least squares regression for simultaneous dimension reduction and variable selection. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 72(1) :3–25.

Consortium, G. O. (2019). The gene ontology resource : 20 years and still going strong. *Nucleic acids research*, 47(D1) :D330–D338.

Costantino, F., Breban, M., and Garchon, H.-J. (2018). Genetics and functional genomics of spondyloarthritis. *Frontiers in immunology*, 9 :2933.

Costantino, F., Talpin, A., Said-Nahal, R., Goldberg, M., Henny, J., Chiocchia, G., Garchon, H.-J., Zins, M., and Breban, M. (2015). Prevalence of spondyloarthritis in reference to hla-b27 in the french population : results of the gazel cohort. *Annals of the rheumatic diseases*, 74(4) :689–693.

Costello, M.-E., Ciccia, F., Willner, D., Warrington, N., Robinson, P. C., Gardiner, B., Marshall, M., Kenna, T. J., Triolo, G., and Brown, M. A. (2015). Brief report : intestinal dysbiosis in ankylosing spondylitis. *Arthritis & rheumatology*, 67(3) :686–691.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems :1695.

Daudin, J.-J., Picard, F., and Robin, S. (2008). A mixture model for random graphs. *Statistics and computing*, 18(2) :173–183.

Dean, L. E., Jones, G. T., MacDonald, A. G., Downham, C., Sturrock, R. D., and Macfarlane, G. J. (2014). Global prevalence of ankylosing spondylitis. *Rheumatology*, 53(4) :650–657.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society : Series B (Methodological)*, 39(1) :1–22.

Dobin, A., Davis, C. A., Schlesinger, F., Drenkow, J., Zaleski, C., Jha, S., Batut, P., Chaisson, M., and Gingeras, T. R. (2013). Star : ultrafast universal rna-seq aligner. *Bioinformatics*, 29(1) :15–21.

Dokmanic, I., Parhizkar, R., Ranieri, J., and Vetterli, M. (2015). Euclidean distance matrices : A short walk through theory, algorithms and applications. *CoRR*, abs/1502.07541.

Drineas, P. and Mahoney, M. W. (2005). On the nyström method for approximating a gram matrix for improved kernel-based learning. *journal of machine learning research*, 6(Dec) :2153–2175.

Dudbridge, F. (2013). Power and predictive accuracy of polygenic risk scores. *PLoS Genet*, 9(3) :e1003348.

Edgar, R., Domrachev, M., and Lash, A. E. (2002). Gene Expression Omnibus : NCBI gene expression and hybridization array data repository. *Nucleic Acids Research*, 30(1) :207–210.

Ellis, B. and Wong, W. H. (2008). Learning causal bayesian network structures from experimental data. *Journal of the American Statistical Association*, 103(482) :778–789.

Escofier, B. and Pages, J. (1994). Multiple factor analysis (afmult package). *Computational statistics & data analysis*, 18(1) :121–140.

Felsenstein, J. (1985). Confidence limits on phylogenies : an approach using the bootstrap. *Evolution ; international journal of organic evolution*, 39 4 :783–791.

Feltkamp, T., Mardjuadi, A., Huang, F., and Chou, C.-T. (2001). Spondyloarthropathies in eastern asia. *Current opinion in rheumatology*, 13(4) :285–290.

Fortuna, M. A., Ortega, R., and Bascompte, J. (2014). The web of life. *arXiv preprint arXiv :1403.2575*.

Foygel, R. and Drton, M. (2010). Extended bayesian information criteria for gaussian graphical models. In *Advances in neural information processing systems*, pages 604–612.

Fraley, C. (1998). Algorithms for model-based gaussian hierarchical clustering. *SIAM Journal on Scientific Computing*, 20(1) :270–281.

Fraley, C. and Raftery, A. E. (1998). How many clusters ? which clustering method ? answers via model-based cluster analysis. *The computer journal*, 41(8) :578–588.

Friedman, J., Hastie, T., and Tibshirani, R. (2008a). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3) :432–441.

Friedman, J., Hastie, T., and Tibshirani, R. (2008b). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3) :432–441.

Gligorijević, V. and Pržulj, N. (2015). Methods for biological data integration : perspectives and challenges. *Journal of the Royal Society Interface*, 12(112) :20150571.

Govaert, G. and Nadif, M. (2003). Clustering with block mixture models. *Pattern Recognition*, 36(2) :463–473.

GOWER, J. C. (1966). Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53(3-4) :325–338.

Gower, J. C. (1982). Euclidean distance geometry. *Math. Sci*, 7(1) :1–14.

Goyal, P. and Ferrara, E. (2018). Graph embedding techniques, applications, and performance : A survey. *Knowledge-Based Systems*, 151 :78–94.

Gu, Z., Eils, R., and Schlesner, M. (2016). Complex heatmaps reveal patterns and correlations in multidimensional genomic data. *Bioinformatics*.

Guasch-Ferré, M., Hruby, A., Toledo, E., Clish, C. B., Martínez-González, M. A., Salas-Salvadó, J., and Hu, F. B. (2016). Metabolomics in prediabetes and diabetes : A systematic review and meta-analysis. *Diabetes Care*, 39(5) :833–846.

Halko, N., Martinsson, P. G., and Tropp, J. A. (2011). Finding structure with randomness : Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2) :217–288.

Hasin, Y., Seldin, M., and Lusis, A. (2017). Multi-omics approaches to disease. *Genome Biology*, 18.

Hawe, J. S., Theis, F. J., and Heinig, M. (2019). Inferring interaction networks from multi-comics data-a review. *Frontiers in genetics*, 10 :535.

Hidalgo, S. J. T. and Ma, S. (2018). Clustering multilayer omics data using muncut. *BMC genomics*, 19(1) :198.

Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282. IEEE.

Hoadley, K. A., Yau, C., Wolf, D. M., Cherniack, A. D., Tamborero, D., Ng, S., Leiserson, M. D., Niu, B., McLellan, M. D., Uzunangelov, V., et al. (2014). Multiplatform analysis of 12 cancer types reveals molecular classification within and across tissues of origin. *Cell*, 158(4) :929–944.

Hocking, T., Vert, J.-P., Bach, F. R., and Joulin, A. (2011). Clusterpath : an algorithm for clustering using convex fusion penalties. In *ICML*.

Hosseini, M. J. and Lee, S.-I. (2016). Learning sparse gaussian graphical models with overlapping blocks. In *Advances in Neural Information Processing Systems*, pages 3808–3816.

Huang, S., Chaudhary, K., and Garmire, L. X. (2017). More is better : Recent progress in multi-omics data integration methods. *Frontiers in Genetics*, 8 :84.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2 :193–218.

Jaakkola, T. (2001). 10 tutorial on variational approximation methods. *Advanced mean field methods : theory and practice*, page 129.

Jassal, B., Matthews, L., Viteri, G., Gong, C., Lorente, P., Fabregat, A., Sidiropoulos, K., Cook, J., Gillespie, M., Haw, R., et al. (2020). The reactome pathway knowledgebase. *Nucleic acids research*, 48(D1) :D498–D503.

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine learning*, 37(2) :183–233.

Josse, J., Pagès, J., and Husson, F. (2008). Testing the significance of the RV coefficient. *Computational Statistics and Data Analysis*, 53(1) :82–91.

Kanehisa, M. (2019). Toward understanding the origin and evolution of cellular organisms. *Protein Science*, 28(11) :1947–1951.

Kanehisa, M. and Goto, S. (2000). Kegg : kyoto encyclopedia of genes and genomes. *Nucleic acids research*, 28(1) :27–30.

Kanehisa, M., Sato, Y., Furumichi, M., Morishima, K., and Tanabe, M. (2019). New approach for understanding genome variations in kegg. *Nucleic acids research*, 47(D1) :D590–D595.

Kaufman, L. and Rousseeuw, P. J. (1987). Clustering by means of medoids. statistical data analysis based on the l1 norm. *Y. Dodge, Ed*, pages 405–416.

Kirk, P., Griffin, J., Savage, R., Ghahramani, Z., and Wild, D. (2012). Bayesian correlated clustering to integrate multiple datasets. *Bioinformatics (Oxford, England)*, 28.

Kolberg, L. and Raudvere, U. (2020). *gprofiler2 : Interface to the 'g :Profiler' Toolset*. R package version 0.1.9.

Kruskal, J. B. (1964). Nonmetric multidimensional scaling : a numerical method. *Psychometrika*, 29(2) :115–129.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *The annals of mathematical statistics*, 22(1) :79–86.

Kumar, S., Mohri, M., and Talwalkar, A. (2012). Sampling methods for the nyström method. *The Journal of Machine Learning Research*, 13(1) :981–1006.

Langfelder, P. and Horvath, S. (2008). Wgcna : an r package for weighted correlation network analysis. *BMC bioinformatics*, 9(1) :559.

Langfelder, P., Zhang, B., and Horvath, S. (2007). Defining clusters from a hierarchical cluster tree : the Dynamic Tree Cut package for R. *Bioinformatics*, 24(5) :719–720.

Lauritzen, S. L. (1996). *Graphical models*, volume 17. Clarendon Press.

Lauritzen, S. L., Andersen, A. H., Edwards, D., Jöreskog, K. G., and Johansen, S. (1989). Mixed graphical association models [with discussion and reply]. *Scandinavian Journal of Statistics*, pages 273–306.

Lee, B., Zhang, S., Poleksic, A., and Xie, L. (2020). Heterogeneous multi-layered network model for omics data integration and analysis. *Frontiers in Genetics*, 10 :1381.

Lee, J. D. and Hastie, T. J. (2015). Learning the structure of mixed graphical models. *Journal of Computational and Graphical Statistics*, 24(1) :230–253.

Legendre, P. and Legendre, L. F. (2012). *Numerical ecology*. Elsevier.

Leger, J.-B. (2016). Blockmodels : A r-package for estimating in latent block model and stochastic block model, with various probability functions, with or without covariates. *arXiv preprint arXiv :1602.07587*.

Li, Y., Wu, F.-X., and Ngom, A. (2018). A review on machine learning principles for multi-view biological data integration. *Briefings in bioinformatics*, 19(2) :325–340.

Liaw, A. and Wiener, M. (2002). Classification and regression by randomforest. *R News*, 2(3) :18–22.

Lingoes, J. C. (1971). Some boundary conditions for a monotone analysis of symmetric matrices. *Psychometrika*, 36(2) :195–203.

Liu, H., Roeder, K., and Wasserman, L. (2010). Stability approach to regularization selection (stars) for high dimensional graphical models. In *Advances in neural information processing systems*, pages 1432–1440.

Lloyd, S. (1982). Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2) :129–137.

Lock, E. F. and Dunson, D. B. (2013). Bayesian consensus clustering. *Bioinformatics*, 29(20) :2610–2616.

Lun, A. T., Chen, Y., and Smyth, G. K. (2016). It's de-licious : a recipe for differential expression analyses of rna-seq experiments using quasi-likelihood methods in edger. In *Statistical Genomics*, pages 391–416. Springer.

Lund, S. P., Nettleton, D., McCarthy, D. J., and Smyth, G. K. (2012). Detecting differential expression in rna-sequence data using quasi-likelihood with shrunken dispersion estimates. *Statistical applications in genetics and molecular biology*, 11(5).

Lê, S., Josse, J., and Husson, F. (2008). Factominer : An r package for multivariate analysis. *Journal of Statistical Software, Articles*, 25(1) :1–18.

L'Hermier des Plantes, H. (1976). Structuration des tableaux à trois indices de la statistique. *Université de Montpellier II, Thesis.*

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA.

Margush, T. and McMorris, F. (1981). Consensus-trees. *Bulletin of Mathematical Biology*, 43 :239–244.

Mariadassou, M., Robin, S., Vacher, C., et al. (2010). Uncovering latent structure in valued graphs : a variational approach. *The Annals of Applied Statistics*, 4(2) :715–742.

Mariette, J. and Villa-Vialaneix, N. (2017). Unsupervised multiple kernel learning for heterogeneous data integration. *Bioinformatics*, 34(6) :1009–1015.

Marlin, B. M. and Murphy, K. P. (2009). Sparse gaussian graphical models with unknown block structure. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 705–712.

Mavaddat, N., Michailidou, K., Dennis, J., Lush, M., Fachal, L., Lee, A., Tyrer, J. P., Chen, T.-H., Wang, Q., Bolla, M. K., et al. (2019). Polygenic risk scores for prediction of breast cancer and breast cancer subtypes. *The American Journal of Human Genetics*, 104(1) :21–34.

McCarthy, D. J., Chen, Y., and Smyth, G. K. (2012). Differential expression analysis of multifactor rna-seq experiments with respect to biological variation. *Nucleic Acids Research*, 40(10) :4288–4297.

Mo, Q., Wang, S., Seshan, V. E., Olshen, A. B., Schultz, N., Sander, C., Powers, R. S., Ladanyi, M., and Shen, R. (2013). Pattern discovery and cancer gene identification in integrated cancer genomic data. *Proceedings of the National Academy of Sciences*, 110(11) :4245–4250.

Murtagh, F. and Contreras, P. (2012). Algorithms for hierarchical clustering : An overview. *Wiley Interdisc. Rew. : Data Mining and Knowledge Discovery*, 2 :86–97.

Murtagh, F. and Legendre, P. (2014). Ward's hierarchical agglomerative clustering method : Which algorithms implement ward's criterion ? *Journal of Classification*, 31 :274–295.

Ng, A. Y., Jordan, M. I., and Weiss, Y. (2002). On spectral clustering : Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856.

Patro, R., Duggal, G., Love, M. I., Irizarry, R. A., and Kingsford, C. (2017). Salmon provides fast and bias-aware quantification of transcript expression. *Nature methods*, 14(4) :417–419.

Pavoine, S., Ollier, S., and Pontier, D. (2005). Measuring diversity from dissimilarities with rao's quadratic entropy : Are any dissimilarities suitable ? *Theoretical population biology*, 67(4) :231–239.

Pelckmans, K., Brabanter, J. D., Moor, B. D., and Suykens, J. A. K. (2005). Convex clustering shrinkage.

Pierre-Jean, M., Deleuze, J.-F., Le Floch, E., and Mauger, F. (2019). Clustering and variable selection evaluation of 13 unsupervised methods for multi-omics data integration. *Briefings in Bioinformatics*.

Pijuan-Sala, B., Griffiths, J., Guibentif, C., Hiscock, T., Jawaid, W., Calero-Nieto, F., Mulas, C., Ibarra-Soria, X., Tyser, R., Ho, D., Reik, W., Srinivas, S., Simons, B., Nichols, J., Marioni, J., and Göttgens, B. (2019). A single-cell molecular map of mouse gastrulation and early organogenesis. *Nature*.

Pircalabelu, E. and Claeskens, G. (2020). Community-based group graphical lasso. *Journal of Machine Learning Research*, 21(64) :1–32.

Poisot, T., Baiser, B., Dunne, J. A., Kéfi, S., Massol, F., Mouquet, N., Romanuk, T. N., Stouffer, D. B., Wood, S. A., and Gravel, D. (2016). mangal–making ecological network analysis simple. *Ecography*, 39(4) :384–390.

Proctor, L., Huot Creasy, H., Fettweis, J., Lloyd-Price, J., Mahurkar, A., Zhou, W., Buck, G., Snyder, M., III, J., Weinstock, G., White, O., and Huttenhower, C. (2019). The integrative human microbiome project. *Nature*, 569 :641–648.

Quesnel-Vallières, M., Weatheritt, R., Cordes, S., and Blencowe, B. (2018). Autism spectrum disorder : insights into convergent mechanisms from transcriptomics. *Nature Reviews Genetics*, 20.

R Core Team (2019). *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

R Core Team (2020a). *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

R Core Team (2020b). *R : A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Rappoport, N. and Shamir, R. (2018). Multi-omic and multi-view clustering algorithms : review and cancer benchmark. *Nucleic acids research*, 46(20) :10546–10562.

Rath, H. C., Herfarth, H. H., Ikeda, J. S., Grenther, W. B., Hamm, T. E., Balish, E., Taurog, J. D., Hammer, R. E., Wilson, K. H., Sartor, R. B., et al. (1996). Normal luminal bacteria, especially bacteroides species, mediate chronic colitis, gastritis, and arthritis in hla-b27/human beta2 microglobulin transgenic rats. *The Journal of clinical investigation*, 98(4) :945–953.

Rau, A., Manansala, R., Flister, M. J., Rui, H., Jaffrézic, F., Laloë, D., and Auer, P. L. (2019). Individualized multi-omic pathway deviation scores using multiple factor analysis. *bioRxiv*, page 827022.

Ritchie, M. D., Holzinger, E. R., Li, R., Pendergrass, S. A., and Kim, D. (2015a). Methods of integrating data to uncover genotype–phenotype interactions. *Nature Reviews Genetics*, 16(2) :85–97.

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. (2015b). limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Research*, 43(7) :e47.

Robert, P. and Escoufier, Y. (1976). A unifying tool for linear multivariate statistical methods : The rv- coefficient. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 25(3) :257–265.

Robinson, D. F. and Foulds, L. R. (1979). Comparison of weighted labelled trees. In *Combinatorial mathematics VI*, pages 119–126. Springer.

Robinson, D. F. and Foulds, L. R. (1981). Comparison of phylogenetic trees. *Mathematical biosciences*, 53(1-2) :131–147.

Robinson, M. and Oshlack, A. (2010). A scaling normalization method for differential expression analysis of rna-seq data. *Genome biology*, 11 :R25.

Robinson, M. D., McCarthy, D. J., and Smyth, G. K. (2010). edger : a bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, 26(1) :139–140.

Rohart, F., Gautier, B., Singh, A., and Le Cao, K.-A. (2017). mixomics : An r package for 'omics feature selection and multiple data integration. *PLoS computational biology*, 13(11) :e1005752.

Rohlf, F. J. (1982). Consensus indices for comparing classifications. *Mathematical Biosciences*, 59(1) :131 – 144.

Rousseeuw, P. J. (1987). Silhouettes : a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20 :53–65.

Scher, J. U., Ubeda, C., Artacho, A., Attur, M., Isaac, S., Reddy, S. M., Marmon, S., Neimann, A., Brusca, S., Patel, T., et al. (2015). Decreased bacterial diversity characterizes the altered gut microbiota in patients with psoriatic arthritis, resembling dysbiosis in inflammatory bowel disease. *Arthritis & rheumatology*, 67(1) :128–139.

Schleif, F.-M. and Tino, P. (2015). Indefinite proximity learning : A review. *Neural Computation*, 27(10) :2039–2096.

Schölkopf, B., Tsuda, K., and Vert, J.-P. (2004). *Kernel methods in computational biology*. MIT press.

Schwarz, G. (1978). Estimating the dimension of a model. *Ann. Statist.*, 6(2) :461–464.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, 6(2) :461–464.

Shepard, R. N. (1962). The analysis of proximities : multidimensional scaling with an unknown distance function. i. *Psychometrika*, 27(2) :125–140.

Snijders, T. A. and Nowicki, K. (1997). Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of classification*, 14(1) :75–100.

Sokal, R. R. and Rohlf, F. J. (1962). The comparison of dendrograms by objective methods. *Taxon*, 11(2) :33–40.

Steel, M., Dress, A. W. M., and Böcker, S. (2000). Simple but Fundamental Limitations on Supertree and Consensus Tree Methods. *Systematic Biology*, 49(2) :363–368.

Szklarczyk, D., Gable, A. L., Lyon, D., Junge, A., Wyder, S., Huerta-Cepas, J., Simonovic, M., Doncheva, N. T., Morris, J. H., Bork, P., et al. (2019). String v11 : protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, 47(D1) :D607–D613.

Tantardini, M., Ieva, F., Tajoli, L., and Piccardi, C. (2019). Comparing methods for comparing networks. *Scientific reports*, 9(1) :1–19.

Taurog, J. D., Richardson, J. A., Croft, J., Simmons, W. A., Zhou, M., Fernández-Sueiro, J. L., Balish, E., and Hammer, R. E. (1994). The germfree state prevents development of gut and joint inflammatory disease in hla-b27 transgenic rats. *The Journal of experimental medicine*, 180(6) :2359–2364.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society : Series B (Methodological)*, 58(1) :267–288.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005). Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society : Series B (Statistical Methodology)*, 67(1) :91–108.

Tibshirani, R., Walther, G., and Hastie, T. (2001). 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.

Tini, G., Marchetti, L., Priami, C., and Scott-Boyer, M.-P. (2019). Multi-omics integration—a comparison of unsupervised clustering methodologies. *Briefings in bioinformatics*, 20(4) :1269–1279.

Tito, R. Y., Cypers, H., Joossens, M., Varkas, G., Van Praet, L., Glorieus, E., Van den Bosch, F., De Vos, M., Raes, J., and Elewaut, D. (2017). Brief report : dialister as a microbial marker of disease activity in spondyloarthritis. *Arthritis & rheumatology*, 69(1) :114–121.

Torgerson, W. S. (1958). Theory and methods of scaling.

Vaske, C. J., Benz, S. C., Sanborn, J. Z., Earl, D., Szeto, C., Zhu, J., Haussler, D., and Stuart, J. M. (2010). Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using paradigm. *Bioinformatics*, 26(12) :i237–i245.

Ventham, N., Kennedy, N., Adams, A., Kalla, R., Heath, S., O'Leary, K., Drummond, H., Lauc, G., Campbell, H., McGovern, D., Annese, V., Zoldos, V., Pemberton, I., Wuhrer, M., Kolarich, D., Fernandes, D., Theorodorou, E., Merrick, V., Spencer, D., and Satsangi, J. (2016). Integrative epigenome-wide analysis demonstrates that dna methylation may mediate genetic risk in inflammatory bowel disease. *Nature Communications*, 7 :13507.

Vinh, N. X., Epps, J., and Bailey, J. (2010). Information theoretic measures for clusterings comparison : Variants, properties, normalization and correction for chance. *J. Mach. Learn. Res.*, 11 :2837–2854.

Wang, B., Mezlini, A. M., Demir, F., Fiume, M., Tu, Z., Brudno, M., Haibe-Kains, B., and Goldenberg, A. (2014). Similarity network fusion for aggregating data types on a genomic scale. *Nature methods*, 11(3) :333.

Wang, D. and Gu, J. (2016). Integrative clustering methods of multi-omics data for molecule-based cancer classifications. *Quantitative Biology*, 4(1) :58–67.

Wang, Z., Xu, W., San Lucas, F. A., and Liu, Y. (2013). Incorporating prior knowledge into gene network study. *Bioinformatics*, 29(20) :2633–2640.

Weylandt, M., Nagorski, J., and Allen, G. (2019). Dynamic visualization and fast computation for convex clustering via algorithmic regularization. *Journal of Computational and Graphical Statistics*, pages 1–18.

Weylandt, M., Nagorski, J., and Allen, G. I. (2020). Dynamic visualization and fast computation for convex clustering via algorithmic regularization. *Journal of Computational and Graphical Statistics*, 29(1) :87–96.

Williams, C. K. (2001). On a connection between kernel pca and metric multidimensional scaling. In *Advances in neural information processing systems*, pages 675–681.

Williams, C. K. and Seeger, M. (2001). Using the nyström method to speed up kernel machines. In *Advances in neural information processing systems*, pages 682–688.

Willing, B. P., Dicksved, J., Halfvarson, J., Andersson, A. F., Lucio, M., Zheng, Z., Järnerot, G., Tysk, C., Jansson, J. K., and Engstrand, L. (2010). A pyrosequencing study in twins shows that gastrointestinal microbial profiles vary with inflammatory bowel disease phenotypes. *Gastroenterology*, 139(6) :1844–1854.

Wu, D., Wang, D., Zhang, M. Q., and Gu, J. (2015). Fast dimension reduction and integrative clustering of multi-omics data using low-rank approximation : application to cancer molecular classification. *BMC genomics*, 16(1) :1022.

Young, W. C., Raftery, A. E., and Yeung, K. Y. (2014). Fast bayesian inference for gene regulatory networks using scanbma. *BMC systems biology*, 8(1) :47.

Zhang, S., Liu, C.-C., Li, W., Shen, H., Laird, P., and Zhou, X. (2012). Discovery of multi-dimensional modules by integrative analysis of cancer genomic data. *Nucleic acids research*, 40 :9379–91.

Zhao, P. and Yu, B. (2006). On model selection consistency of lasso. *Journal of Machine learning research*, 7(Nov) :2541–2563.

Zhao, T., Liu, H., Roeder, K., Lafferty, J., and Wasserman, L. (2012). The huge package for high-dimensional undirected graph estimation in r. *Journal of Machine Learning Research*, 13(Apr) :1059–1062.

Zhuang, J., Wang, J., Hoi, S. C., and Lan, X. (2011). Unsupervised multiple kernel learning. *Journal of Machine Learning Research*, 20 :129–144.

Annexe A

Female ponderal index at birth and idiopathic infertility

Charlotte Dupont, Audrey Hulot, Florence Jaffrezic, Céline Faure, Sébastien Czernichow, et al.. Female ponderal index at birth and idiopathic infertility.. Journal of Developmental Origins of Health and Disease, Cambridge University Press, 2019, pp.1-5. (10.1017/S2040174419000394)

L'article présenté dans cette annexe est le résultat d'une collaboration sur le projet ALIFERT, cherchant à étudier les facteurs d'infertilité dans les couples. L'étude concerne le lien qu'il pourrait exister entre l'index pondéral à la naissance et la fertilité des femmes à l'âge adulte. Une régression logistique permet de mettre en évidence un potentiel impact de ce caractère sur la fertilité.

Journal of Developmental Origins of Health and Disease

www.cambridge.org/doh

Female ponderal index at birth and idiopathic infertility

Charlotte Dupont[1], Audrey Hulot[2,3,4], Florence Jaffrezic[2], Céline Faure[5], Sébastien Czernichow[6,7], Nathalie di Clemente[8,9], Chrystele Racine[8,9,10], Pascale Chavatte-Palmer[11], Rachel Lévy[1] and Alifert group

[1]Sorbonne Université, Saint Antoine Research center, INSERM équipe Lipodystrophies génétiques et acquises. Service de biologie de la reproduction-CECOS, AP-HP, Hôpital Tenon, F-75020 Paris, France; [2]UMR GABI, AgroParisTech, INRA, Université Paris-Saclay, 78350 Jouy-en-Josas, France; [3]UMR Mia-Paris, AgroParisTech, INRA, Université Paris-Saclay, 75005 Paris, France; [4]Inserm U1173, Simone Veil School of Health Sciences, University of Versailles Saint-Quentin-en-Yvelines, Montigny-le-Bretonneux, France; [5]Service de biologie de la reproduction-CECOS, APHP, Hôpital Tenon, Paris, France; [6]Service de nutrition (Centre Spécialisé Obésité), Hôpital Européen Georges-Pompidou, APHP Paris, France; [7]Université Paris Descartes, Paris France; [8]Centre de Recherche Saint-Antoine (CRSA), Sorbonne Université-INSERM, 75012 Paris, France; [9]Institut Hospitalo-Universitaire ICAN, 75013 Paris, France; [10]Université de Paris, Saint Antoine Research center, INSERM équipe Lipodystrophies génétiques et acquises, F-75012 Paris, France and [11]UMR BDR, INRA, ENVA, Université Paris Saclay, 78350, Jouy en Josas, France

Abstract

Epidemiological studies have demonstrated an increased risk of developing non-transmittable diseases in adults subjected to adverse early developmental conditions. Metabolic and cardiovascular diseases have been the focus of most studies. Nevertheless, data from animal models also suggest early programming of fertility. In humans, it is difficult to assess the impact of the *in utero* environment retrospectively. Birthweight is commonly used as an indirect indicator of intrauterine development. This research is part of the ALIFERT study. We investigated a potential link between ponderal index at birth and female fertility in adulthood. Data from 51 infertile and 74 fertile women were analysed. BW was on average higher in infertile women, whereas birth length did not differ between the two groups; thus, resulting in a significantly higher ponderal index at birth in infertile women. Ponderal index at birth has been identified as a risk factor for infertility. These results suggest the importance of the intra-uterine environment, not only for long-term metabolic health but also for fertility.

Brief Report

Cite this article: Dupont C, Hulot A, Jaffrezic F, Faure C, Czernichow S, di Clemente N, Racine C, Chavatte-Palmer P, Lévy R, and Alifert group. Female ponderal index at birth and idiopathic infertility. *Journal of Developmental Origins of Health and Disease*
https://doi.org/10.1017/S2040174419000394

Received: 23 October 2018
Revised: 28 April 2019
Accepted: 17 June 2019

Key words:
ponderal index; birthweight; female fertility; female infertility

Address for correspondence: Charlotte Dupont, Saint Antoine Research center, INSERM équipe Lipodystrophies génétiques et acquises, Sorbonne Université and Service de biologie de la reproduction-CECOS, AP-HP, Hôpital Tenon, F-75020 Paris, France.
Email: charlotte.dupont@aphp.fr

Alifert Collaborative Group: ***Isabelle Aknin***: Unité fonctionnelle de biologie de la reproduction, histologie – embryologie – cytogénétique, hôpital Nord, Saint-Étienne, France ; ***Isabelle Cedrin-Durnerin***: Service de Médecine de la Reproduction, Hôpital Jean Verdier, APHP, Bondy, France ; ***Steven Cens***, Centre d'AMP de PAU, Polyclinique de Navarre, Pau, France; ***Serge Hercberg***: EREN, INSERM U557; INRA; CNAM; Université Paris 13, CRNH IdF, 93017 Bobigny, France; ***Khaled Pocate***: Service d'Histologie-Embryologie-Biologie de la Reproduction, Hôpital Cochin APHP, Paris, France; ***Nathalie Sermondade***: Service de biologie de la reproduction-CECOS, Hôpital Tenon, APHP, Paris, France ; ***Claude Uthurriague***, Centre d'AMP de PAU, Polyclinique de Navarre, Pau ; ***Jean-Philippe Wolf***: Service d'Histologie-Embryologie-Biologie de la Reproduction, Hôpital Cochin, APHP, Paris, France

Introduction

Infertility is defined as the inability to conceive after 12 months of unprotected sexual intercourse. Infertility prevalence in women is about 13%. Female infertility may have numerous etiologies among which polycystic ovary syndrome (PCOS), endometriosis, tubal and uterine pathologies, age-related infertility, lifestyle factors (smoking, obesity, etc.), and/or environmental causes are the most common. Some of these may also be developmental in origin (see below). Nevertheless, in some cases, no cause is documented or may be of developmental origin. Thus, the impact of the intrauterine environment on long-term health is a well-recognized concept. An increased risk of noncommunicable metabolic diseases has been observed in people born small for gestational age or with nonoptimal early developmental conditions[1]. This concept, known as developmental origins of health and disease (DOHaD), has been extended to other health outcomes including offspring fertility (based on animal experiments and some retrospective human studies).

Undernutrition or overnutrition in pregnant sheep and cattle may induce negative effects on reproductive function of both male and female offspring. Indeed, delayed ovarian development[2], reduced follicle number in adulthood, and increased oxidative stress in the ovary[3] have been reported in female sheep exposed to undernutrition *in utero* . In sheep exposed to overnutrition, delayed ovarian development and reduced follicle numbers in foetal ovaries have also been observed[4]. Experiments in rodents have confirmed these results, as reviewed elsewhere[5]. Moreover, a recent murine study reported that a low-protein diet during preconception, pregnancy, and lactation periods led to reduced primordial follicle numbers and increased follicular atresia[6].

In women, low birth weight (LBW) has been associated with an increased risk of early reproductive senescence, such as earlier menopause (FSH > 25 IU/ml)[7,8]. Data from the Danish National Birth Cohort (22,044 pregnancies from 21,786 women) showed that both low (<2500 g) and high (>4500 g) BWs were associated with an increase in time to pregnancy of more than 1 year[9]. Moreover, anovulation appears to be observed more frequently in 10 adolescent girls born small

for gestational age[10]. A study comparing 37 young women with LBW to 35 controls showed that LBW women have reduced insulin sensitivity associated with an increased risk of developing PCOS[11]. In a study comparing 375 controls and 368 women with endometriosis, LBW has also been associated with increased risk of endometriosis, especially deep infiltrated endometriosis[12].

The human studies detailed above all use BW as a proxy for foetal development. Nevertheless, despite having a similar BW, a long and thin infant is metabolically different from a short and chubby baby. The ponderal index (PI) assesses the ratio between weight (kg) and the cubic value of height (m^3) and is usually used as a corpulence index in paediatrics[13]. Birth PI can discriminate between children of the same BW and thus is a more relevant indirect indicator of foetal nutrition. PI may more accurately reflect adiposity in infants[14].

Birth PI is rarely considered when assessing long-term fertility, mostly due to the fact that accurate and reliable information on both BW and birth length is usually not available. Nevertheless, it has been observed that a one unit increase in birth PI is associated with reduced risk of PCOS symptoms in 30-year-old women, whereas a 100-g increase in BW is associated with increased risk of hyperandrogenism[15].

As we have previously observed, BW was higher in infertile men compared to fertile men included in the ALIFERT study[16]. Furthermore, the BW was inversely correlated with both total sperm count and sperm DNA integrity. The purpose of this study was to investigate whether a similar correlation exists in women of the same cohort and if PI may be an indicator of fertility later in life.

Material and methods

Ninety-nine female partners of infertile couples with idiopathic primary infertility and 100 fertile women were recruited in the ALIFERT study between September 2009 and December 2013 (N° P071224). The ALIFERT study is a prospective observational case-control study with the aim to assess the association between diet and idiopathic infertility.

Infertile women were partners of infertile couples attending four infertility centers in France (Jean Verdier hospital ART center, Bondy; Cochin hospital ART center, Paris; Hôpital Nord ART center, Saint Etienne; Polyclinique de Navarre ART center, Pau). They were eligible for the study if they presented a primary idiopathic infertility >12 months and met the following inclusion criteria: (i) they were between 18 and 38 years old, (ii) they did not present either anovulation, ovarian failure (on the basis of follicle count and hormone balance at day 3 (FSH, LH, and estradiol)) nor uterotubal pathology (assessed by hysterosalpingography), (iii) their partners did not present severe sperm alteration nor urogenital pathology, and (iv) they were in possession of their Child Health Record. Patients with current known or previous metabolic or digestive disease were excluded.

The control group consisted of fertile women volunteers recruited by advertisement (Internet advertising and word of mouth) from the general healthy population in areas of the participating centers. The criteria for eligibility were (i) age between 18 and 38 years, (ii) they had a spontaneously conceived child under 2 years of age, (iii) time to pregnancy was less than 12 months, and (iv) they were in possession of their Child Health Record.

Male partners of infertile and fertile women were between 18 and 45 years old.

Written informed consent was collected. The ethics committee ("Comité de Protection des Personnes") approved the study as ALIFERT study (national biomedical research P071224/AOM 08180:NEudra CT 2009-A00256-51/clinical trials NCT01093378).

BW and birth length were collected from the individual Child Health Record. The Child Health Record is a booklet delivered by the national health authority to all French children at birth where all information on the child's health is recorded by medical staff. PI was calculated as BW/length^3 (kg/m^3).

Anthropometric assessment

The same trained investigator measured height, weight, and waist circumference (measured at the narrowest point between the lower border of the ribs and the iliac crest) in both fertile and infertile women at the time of the inclusion visit in the ART centers. Body mass index (BMI) was calculated as weight/height^2 (kg/m^2).

Blood samples and analyses

Blood samples were collected after a 12-h fasting period.

High-density lipoprotein (HDL-cholesterol), low-density lipoprotein (LDL-cholesterol), triglycerides, and glucose concentrations were measured by standardized methods in the hospitals' biology laboratory.

AMH was assayed with the AMH Gen II ELISA Kit (Beckman Coulter). All the samples were assayed by the same operator at the same time.

Blood pressure assessment

A sphygmomanometer cuff was placed on the patient arm and blood pressure was measured after 5 min of bed rest in a supine position. The systolic and diastolic blood pressures were calculated by computing the average of the right and left arm.

Tobacco consumption and exhaled carbon monoxide (CO)

Patients reported the number of cigarettes smoked per day. They have been categorized as smokers if they smoked one or more cigarettes per day and as nonsmokers if they did not smoke at all. Exhaled carbon monoxide (CO) was measured in parts per million (ppm) as a supportive indicator with the underlying assumption that exhaled CO in smokers is higher than in nonsmokers[17]. The assessment of exhaled CO was intended to support self-reporting of tobacco consumption. Nevertheless, exhaled CO can also be influenced by passive smoking and prolonged exposure to a polluted environment[18] that also may impact fertility. Thus, the exhaled CO level does not necessarily reflect a subject's smoking amount, but may be more relevant to address airborne environmental exposure that may impact fertility, consequently exhaled CO levels were used instead of self-reporting.

Exhaled CO measurement in parts per million (ppm) was performed by having subjects exhale completely then inhale fully in open air, withhold their breath for 10 s, and then exhale completely into a portable CO monitor (Tabataba analyser-FIM medical, Villeurbanne 69625 France).

Statistical analysis

Missing Data : Seventy-four women (26 fertile and 48 infertile) out of 199 had missing observations, representing a total of 8.1% of the dataset. Consequently, only women with complete data were included in the analysis. Data of 51 infertile women and 74 fertile women were analyzed.

Baseline characteristics: Baseline characteristics of the women were described by fertility status (mean and standard deviation). The Wilcoxon test was used for comparing fertile versus infertile women.

Logistic regression: Associations between PI and fertility status were investigated using logistic regression models, first considering the following variables (i.e., age, BMI, waist circumference, fasting blood glucose, exhaled CO, AMH, BW, birth height, and PI). Waist circumference, BW, and birth height were not included in the models due to collinearity with BMI and PI, respectively. Akaike information criterion (AIC) was used to select the best-fitting model and removing nonsignificant variables afterward. The final regression model was adjusted for five variables: age, BMI, fasting blood glucose, exhaled CO, and PI. Odds ratios (ORs) and 95% confidence intervals (CIs) are reported.

Moreover, the association between PI and BW was investigated and the correlation coefficient was calculated. To compare the relative significance of the two parameters, logistic regression analysis was also performed using BW instead of PI.

R 3.5.1 software (https://www.r-project.org) was used for all statistical analyses. $p < 0.05$ was considered significant.

Results

Baseline characteristics

Characteristics of fertile and infertile women are described in Table 1. Infertile women BMI and waist circumference were significantly higher compared to fertile women. Infertile women were more often smoker (12.2% versus 7%) and they had higher exhaled CO levels. There was a correlation between exhaled CO levels and smoking status in both infertile and fertile women (infertile: $r^2 = 0.52$, fertile: $r^2 = 0.58$, all: $r^2 = 0.54$). Higher fasting blood glucose and lower triglycerides were observed in infertile women, but no difference could be observed concerning cholesterol (HDL and LDL), plasma AMH concentrations, or blood pressure. Moreover, BW was higher in infertile compared to fertile women whereas birth length did not differ between the two groups. Consequently, PI was significantly higher in infertile women compared to fertile women. Only patients born at term (gestational age was between 37 and 41 weeks of amenorrhea) were included in the study.

Associations between PI and fertility status

PI, anthropometric, metabolic factors, and age according to the fertility status are presented in Table 2. Increased PI at birth, increased BMI, increased glycemia, and high exhaled CO were identified as significant risk factors for infertility (Table 2). Glycemia seems to have a strong effect on the risk of infertility with an OR of 2.73 [1.23–6.07], $p = 0.014$. PI was associated to infertility with an OR of 1.27 [95% CI, 1.06–1.52], $p = 0.009$. PI and BW were correlated (correlation coefficient: 0.587) (Table 3). In addition, BW was also associated to infertility but the association (1.002 [95% CI, 1.00–1.003]; $p = 0.023$) was not as strong as with the PI.

Discussion

We have previously observed that infertile men have a significantly higher BW than fertile men and that BW is associated with semen parameters[16]. In the present study, we also observed higher BW in infertile compared to fertile women. These results seem to be contradictory compared to other studies that have shown a negative impact for LBW on fertility[10–12]. However, none of the patients in the present study had a very LBW (<1500 g), and only one had a BW below 2000 g. Therefore, as hypothesized previously[9], in addition to low and very low BWs, a heavy BW may be a risk factor for the development of infertility in adulthood.

We observed that infertile women have a higher PI at birth than fertile women. Although PI may not be a better indicator of foetal morbidity[13] than BW, it reflects potential asymmetric growth due to a nonoptimal periconceptional environment as well as neonatal fat mass[19]. Additionally, variations in PI have been associated with adverse long-term consequences[20]. The multivariate analyses performed here show that a high PI at birth is an infertility risk factor in adulthood. In addition, infertility correlates with increased BMI, glycaemia, and exhaled CO.

Fetal programming of metabolic diseases is a well-established concept known as DOHaD. This concept recognizes that an unfavorable maternal environment can lead to placental abnormalities, which may impact fetal development through various mechanisms such as hormonal imbalance, oxidative stress, and epigenetic changes[21,22]. In the long term, an increased risk of developing overweight or metabolic diseases has been observed in cases of intrauterine growth retardation or macrosomia[1]. Thus, an increased risk of infertility may also be a consequence of an unfavorable gestational environment.

In this study, infertile women were recruited if they were part of a couple with unexplained infertility. Consequently, we did not observe any difference in AMH levels between fertile and infertile groups. In this population, we can assume that the long-term consequences of an inadequate *in utero* environment may not impact ovarian reserve but possibly oocyte quality. Nevertheless, developmental origin cannot be excluded from women presenting ovarian failure. In order to confirm this hypothesis, a new study with patients presenting ovarian failure would need to be carried out. Moreover, we observed worse anthropometric and metabolic parameters in infertile women. Weight gain, abdominal obesity, or metabolic disorders are associated with systemic inflammation and oxidative stress, which plays a critical role in female reproductive function and oocyte quality[23]. Mechanisms are undoubtedly complex and multifactorial. Based on current knowledge, it is difficult to assess whether prenatal development has a direct effect on adult fertility or if it leads to the development of long-term metabolic disorders that will negatively impact fertility. Studies in sheep have shown that excess maternal nutrition during pregnancy affected offspring ovaries at the fetal stage[4]. Fewer follicles were observed in the ovary, demonstrating that a direct effect of the maternal environment cannot be excluded. Finally, it is difficult to explain why fertile women had slightly higher triglyceride levels than infertile women, but the average value remains below a pathological value.

Epigenetic modifications are widely involved in metabolic programming, but little is known about epigenetic programming of oocytes or ovaries[24]. Nevertheless, transgenerational programming through two or three generations may suggest epigenetic changes in oocytes are transferred to the offspring[25].

Many studies in animal model have shown that maternal overnutrition or undernutrition during gestation increases oocyte apoptosis[24]. Thus, in a rabbit model exposed *in utero* to a high-fat diet, we previously observed increased follicular atresia, suggesting apoptotic mechanisms during folliculogenesis. In this rabbit model, no difference was found for the primordial, primary, secondary, or tertiary follicle counts. Oxidative stress may be involved in this phenomenon[26].

Table 1. Baseline characteristics of infertile and fertile women

	Infertile women (n = 51)	Fertile women (n = 74)	p
Age	31.10 ± 3.7	32.18 ± 3.1	0.08
BMI (Kg/m^2)	24.34 ± 5.0	21.94 ± 2.7	**0.01**
Waist circumference (cm)	81.22 ± 11.4	77.23 ± 7.5	0.08
Fasting blood glucose (mmol/l)	4.83 ± 0.7	4.19 ± 0.8	**<0.001**
HDL cholesterol (mmol/l)	1.64 ± 0.3	1.63 ± 0.6	0.42
LDL cholesterol (mmol/l)	2.77 ± 0.6	2.85 ± 0.90	0.83
Triglycerides (mmol/l)	0.78 ± 0.40	1.00 ± 6.20	**<0.001**
Systolic blood pressure (mmHg)	112.67 ± 12.1	111.78 ± 9.1	0.68
Exhaled CO (ppm)	4.24 ± 2.8	2.91 ± 1.6	**<0.001**
AMH (ng/ml)	3.30 ± 2.9	3.30 ± 2.8	0.71
Birth weight (g)	3292.55 ± 423.9	3147.43 ± 345.8	**0.02**
Birth length (cm)	49.27 ± 1.9	49.85 ± 1.7	0.18
Ponderal index (kg/m^3)	27.57 ± 3.6	25.42 ± 3.6	**<0.001**

Data are means ± standard deviations.

Table 2. Factors associated with fertility and infertility (multivariate logistic regression)

	OR [95% CI]	p value
Age (year)	0.86 [0.74–0.99]	**0.040**
BMI (Kg.m^{-2})	1.171 [1.02–1.35]	**0.026**
Glycemia (mmol/l)	2.73 [1.23–6.07]	**0.014**
Exhaled CO (ppm)	1.34 [1.08–1.65]	**0.007**
Ponderal index (kg/m^3)	1.27 [1.06–1.52]	**0.009**

OR, odds ratio; CI, confidence interval; BMI, body mass index.

Oocyte and follicular atresia may be early signs of ovarian aging. Indeed, the environment at birth may also be involved in the age at which menopause occurs. Perinatal exposure to famine has been associated with advanced menopause[27]. Some studies have also related a high BW[7] or PI to early menopause (40–42 years)[8]. Indeed, a suboptimal perinatal environment also affects longevity and accelerates aging. Oxidative stress and reduced telomere length may be key mechanisms involved in this phenomenon[28,29].

More epidemiologic and experimental animal studies are necessary to fully understand the mechanisms of this phenomenon.

This study presents both strengths and limitations. One of the strengths of the study is the recruitment of fertile and infertile women under the same conditions, which is rare. Furthermore, BW and birth length data were directly obtained from official childhood records, which limit self-reporting bias. Anthropometric assessments were performed by the same trained investigator using the same calibrated devices, thus minimizing observation bias. Limitations of this study include the small sample size. Moreover, BW, birth height, and PI are proxies of the *in utero* environment and do not represent all of the events occurring during pregnancy. Furthermore, no women born with a very LBW were recruited for the study; therefore, it was not possible to assess the consequences of very LBW on fertility. Finally, infertile couples (unlike fertile couples) in this study were selected by medical services, which could have introduced a selection bias. Indeed, fertile women were slightly older than infertile, which could be explained by the fact that they were recruited after the birth of their child and they were not usually included immediately after childbirth.

Furthermore, socioeconomic status and physical activity may impact fertility. Since these two variables may both impact BMI, we chose to use BMI for the logistic regression model. Nevertheless, no difference in physical activity between fertile and infertile women was observed in a study including a similar population[30].

In order to explore further the role of in utero environmental conditions on fertility, retrospective or even prospective cohorts of individuals whose pregnancy history is well known and who would be monitored over the long term would be necessary.

Conclusion

This study is unique in that two groups of similar size and characteristics were recruited and assessed. The results confirm the importance of the intrauterine environment, not only for long-term metabolic health but also for fertility. We can consider the risk of infertility to be part of the adverse outcomes related to an inadequate periconceptional environment in addition to metabolic disorders, cancer, autoimmune diseases, and neurodegenerative diseases[31]. These are additional arguments for setting up preventive programs and interventions to improve gestational health.

Acknowledgments. The authors want to acknowledge all the participants involved in the study.

Financial support. This study was supported by national biomedical research P071224 ALIFERT.

Ethical Standards. The ethics committee ("Comité de Protection des Personnes") approved the ALIFERT study (national biomedical research P071224/AOM 08180: NEudra CT 2009-A00256-51/clinical trials NCT01093378). The study was conducted according to the protocol, to the law of December 20, 1988, as amended by Act

Table 3. Correlation between birth weight, birth height, and ponderal index

	Birth weight	Birth height	Ponderal index
Birth weight	1	0.473	0.587
Birth height		1	−0.428
Ponderal index			1

2004-806 of August 9, 2004, to the ethical principles established by the 18th World Medical Assembly, and to French Good Clinical Practice. Before starting the research, an authorization file (as defined in Article L 1123-12) was approved by Agence Française de Sécurité Sanitaire des Produits de Santé (AFSSAPS), the favorable opinion of the Committee for Protection of Persons of Ile de France was obtained, and the Commission Nationale de l'In-formatique et des Libertés (CNIL) declaration was performed. All patients signed consent forms.

Authors contribution. C.D. participated in the study conception and design, in patient's recruitment, data acquisition, interpretation and analysis, and drafting of the manuscript. A.H., F.J., and P.C.P. participated in study design, performed statistical analysis, and participated in critical revision of the manuscript for intellectual content. C.F. participated in the study conception and design, in patient's recruitment and data acquisition. S.C. participated in the study conception and design. R.L. participated in the study conception and design, interpretation of data, critical revision of the manuscript for intellectual content, and supervised the study. The collaborators of the ALIFERT collaborative group participated in the study design and were involved in patients' recruitment.

References

1. Hanson MA, Gluckman PD. Early developmental conditioning of later health and disease: physiology or pathophysiology? *Physiol Rev*. 2014; 94, 1027–1076.
2. Rae MT, Palassio S, Kyle CE, *et al.* Effect of maternal undernutrition during pregnancy on early ovarian development and subsequent follicular development in sheep fetuses. *Reproduction*. 2001; 122, 915–922.
3. Bernal AB, Vickers MH, Hampton MB, Poynton RA, Sloboda DM. Maternal undernutrition significantly impacts ovarian follicle number and increases ovarian oxidative stress in adult rat offspring. *PLoS One*. 2010; 5, e15558.
4. Da Silva P, Aitken RP, Rhind SM, Racey PA, Wallace JM. Impact of maternal nutrition during pregnancy on pituitary gonadotrophin gene expression and ovarian development in growth-restricted and normally grown late gestation sheep fetuses. *Reproduction*. 2002; 123, 769–777.
5. Chadio S, Kotsampasi B. The role of early life nutrition in programming of reproductive function. *J Dev Orig Health Dis*. 2014; 5, 2–15.
6. Winship AL, Gazzard SE, Cullen McEwen LA, Bertram JF, Hutt KJ. Maternal low protein diet programmes low ovarian reserve in offspring. *Reproduction*. 2018; 156, 299–311.
7. Tom SE, Cooper R, Kuh D, *et al.* Fetal environment and early age at natural menopause in a British birth cohort study. *Hum Reprod*. 2010; 25, 791–798.
8. Cresswell JL, Egger P, Fall CH, *et al.* Is the age of menopause determined in-utero? *Early Hum Dev*. 1997; 49, 143–148.
9. Nohr EA, Vaeth M, Rasmussen S, Ramlau-Hansen CH, Olsen J. Waiting time to pregnancy according to maternal birthweight and prepregnancy BMI. *Hum Reprod*. 2009; 24, 226–232.
10. Ibanez L, Potau N, Ferrer A, *et al.* Anovulation in eumenorrheic, nonobese adolescent girls born small for gestational age: insulin sensitization induces ovulation, increases lean body mass, and reduces abdominal fat excess, dyslipidemia, and subclinical hyperandrogenism. *J Clin Endocrinol Metab*. 2002; 87, 5702–5705.
11. Pandolfi C, Zugaro A, Lattanzio F, *et al.* Low birth weight and later development of insulin resistance and biochemical/clinical features of polycystic ovary syndrome. *Metabolism*. 2008; 57, 999–1004.
12. Borghese B, Sibiude J, Santulli P, *et al.* Low birth weight is strongly associated with the risk of deep infiltrating endometriosis: results of a 743 case-control study. *PLoS One*. 2015; 10, e0117387.
13. Cooley SM, Donnelly JC, Walsh T, *et al.* Ponderal index (PI) vs birth weight centiles in the low-risk primigravid population: which is the better predictor of fetal wellbeing? *J Obstet Gynaecol*. 2012; 32, 439–443.
14. Howe LD, Tilling K, Benfield L, *et al.* Changes in ponderal index and body mass index across childhood and their associations with fat mass and cardiovascular risk factors at age 15. *PLoS One*. 2010; 5, e15186.
15. Davies MJ, March WA, Willson KJ, Giles LC, Moore VM. Birthweight and thinness at birth independently predict symptoms of polycystic ovary syndrome in adulthood. *Hum Reprod*. 2012; 27, 1475–1480.
16. Faure C, Dupont C, Chavatte-Palmer P, *et al.* Are semen parameters related to birth weight? *Fertil Steril*. 2015; 103, 6–10.
17. Deveci SE, Deveci F, Acik Y, Ozan AT. The measurement of exhaled carbon monoxide in healthy smokers and non-smokers. *Respir Med*. 2004; 98, 551–556.
18. Maga M, Janik MK, Wachsmann A, *et al.* Influence of air pollution on exhaled carbon monoxide levels in smokers and non-smokers. A prospective cross-sectional study. *Environ Res*. 2017; 152, 496–502.
19. Chen LW, Tint MT, Fortier MV, *et al.* Which anthropometric measures best reflect neonatal adiposity? *Int J Obes (Lond)*. 2018; 42, 501–506.
20. Crusell M, Damm P, Hansen T, *et al.* Ponderal index at birth associates with later risk of gestational diabetes mellitus. *Arch Gynecol Obstet*. 2017; 296, 249–256.
21. Chavatte-Palmer P, Rousseau-Ralliard D, Tarrade A. Oxidative stress in mammals: pregnancy and placental function. In *Oxidative Stress and Women's Health* (ed Menezo Y), 2016. Eska editions, Paris.
22. Tarrade A, Panchenko P, Junien C, Gabory A. Placental contribution to nutritional programming of health and diseases: epigenetics and sexual dimorphism. *J Exp Biol*. 2015; 218, 50–58. doi: 10.1242/jeb.110320. Review
23. Michalakis K, Mintziori G, Kaprara A, Tarlatzis BC, Goulis DG. The complex interaction between obesity, metabolic syndrome and reproductive axis: a narrative review. *Metabolism*. 2013; 62, 457–478.
24. Puttabyatappa M, Padmanabhan V. Developmental programming of ovarian functions and dysfunctions. *Vitam Horm*. 2018; 107, 377–422.
25. Miska EA, Ferguson-Smith AC. Transgenerational inheritance: models and mechanisms of non-DNA sequence-based inheritance. *Science*. 2016; 354, 59–63.
26. Leveille P, Tarrade A, Dupont C, *et al.* Maternal high-fat diet induces follicular atresia but does not affect fertility in adult rabbit offspring. *J Dev Orig Health Dis*. 2014; 5, 88–97.
27. Yarde F, Broekmans FJ, van der Pal-de Bruin KM, *et al.* Prenatal famine, birthweight, reproductive performance and age at menopause: the Dutch hunger winter families study. *Hum Reprod*. 2013; 28, 3328–3336.
28. Tarry-Adkins JL, Martin-Gronert MS, Chen JH, Cripps RL, Ozanne SE. Maternal diet influences DNA damage, aortic telomere length, oxidative stress, and antioxidant defense capacity in rats. *FASEB J*. 2008; 22, 2037–2044.
29. Krisher RL. Maternal age affects oocyte developmental potential at both ends of the age spectrum. *Reprod Fertil Dev*. 2019; 31, 1–9
30. Foucaut AM, Faure C, Julia C, *et al.* Sedentary behavior, physical inactivity and body composition in relation to idiopathic infertility among men and women. *PLoS One*. 2019; 14, e0210770.
31. Barouki R, Gluckman PD, Grandjean P, Hanson M, Heindel JJ. Developmental origins of non-communicable disease: implications for research and public health. *Environ Health*. 2012; 11, 42.

Annexe B

Intégration de données via l'analyse factorielle multiple

La version de ce chapitre comporte les corrections apportées après les retours du relecteur.

Résumé. L'intégration de données provenant de différentes sources est un problème récurrent en bio statistique et bio informatique. La plupart des recherches statistiques se sont tournées vers la résolution du problème de l'intégration de données du même types, souvent des tables de données continues. Cependant, les données à traiter sont souvent hétérogènes : il n'est pas rare d'avoir à disposition des données sous la forme d'arbres, de réseaux, ou de cartes factorielles, ces représentations étant fortement appréciées pour la visualisation des données et l'étude de groupes ou d'interactions entre les différentes entités. Dans ce papier, nous nous intéressons à l'intégration de ces différentes représentations.

Nous proposons une procédure simple qui permet de comparer des données de différents types, en particulier les arbres et les réseaux, en deux étapes : la première étape trouve un système de coordonnées communs dans lequel projeter les différentes représentations ; la seconde étape utilise une méthode d'intégration multi-table pour comparer les projections obtenues. Ces deux étapes utilisent des méthodes déjà connues et efficaces : la projection est obtenue en transformant les objets en matrices de dissimilarités ou distances, puis en appliquant du Multidimensional Scaling (MDS), qui fourni un nouveau jeu de coordonnées à partir de ces dissimilarités. L'étape d'intégration est obtenue quant à elle en effectuant une Analyse Factorielle Multiple (AFM) sur ces nouvelles coordonnées. Cette procédure permet de comparer et intégrer des jeux de données, notamment des arbres et des réseaux. En comparaison des méthodes à noyaux, qui sont utilisées dans le même esprit d'intégration de données sous différentes formes, notre approche est complémentaire et permet l'utilisation de toutes les méthodes de visualisations et d'interprétations qui sont utilisées sur les résultats d'une AFM, et qui manquent parfois dans les méthodes utilisant les noyaux.

La procédure présentée est évaluée sur des données simulées ainsi que des données issues d'études : on compare d'abord des clusterings de gènes réalisés pour différents types de cellules, les données provenant d'une étude single-cell sur des embryons de souris ; dans un second temps on intègre des données -omiques provenant de patients atteints de cancer du sein, dans le but de comparer différents réseaux de protéines.

Abstract. Integrating data from different sources is a recurring question in computational biology. Much effort has been devoted to the integration of data sets of the same type, typically multiple numerical data tables. However, data types are generally heterogeneous: it is a common place to gather data in the form of trees, networks or factorial maps, as these representations all have an appealing visual interpretation that helps to study grouping patterns and interactions between entities. The question we aim to answer in this paper is that of the integration of such representations.

To this end, we provide a simple procedure to compare data with various types, in particular trees or networks, that relies essentially on two steps: the first step projects the representations into a common coordinate system ; the second step then uses a multi-table integration approach to compare the projected data. We rely on efficient and well-known methodologies for each step: the projection step is achieved by retrieving a distance matrix for each representation form and then applying Multidimensional Scaling (MDS) to provide a new set of coordinates from all the pairwise distances. The integration step is then achieved by applying a Multiple Factor Analysis (MFA) to the multiple tables of the new coordinates. This procedure provides tools to integrate and compare data available, for instance, as tree or network structures. Our approach is complementary to kernel methods, traditionally used to answer the same question: while kernels can be used to build appropriate similarities, our approach provides the appealing toolkit of data analysis via MFA that eases the interpretation, which is often lacking in kernel-based methods.

Our approach is evaluated on simulation and used to analyze two real-world data sets: first, we compare several clusterings for different cell-types obtained from a transcriptomics single-cell data set in mouse embryos; second, we use our procedure to aggregate a multi-table data set from the TCGA breast cancer database, in order to compare several protein networks inferred for different breast cancer subtypes.

B.1 Introduction

When integrating data in computational biology, we are often confronted with the problem of comparing outcomes from different types of data, with various representations forms (Gligorijević and Pržulj, 2015; Mariette and Villa-Vialaneix, 2017; Li et al., 2018). These representations may either result from a learning algorithm (e.g. dimension reduction, hierarchical clustering or network inference) or they may be extracted from a data base, reflecting our knowledge about a complex biological process.

As a simple example in genomics, several hierarchical clusterings of individuals can be obtained based on transcriptomics, proteomics or metagenomics experiments, giving birth to several tree-like representations which need to be compared and eventually aggregated. Such an analysis is essential to better understand the data and to obtain a consensus clustering from coherent trees.

A review of omics data integration methods is provided by Ritchie *et al.* (Ritchie et al., 2015a) in a prediction perspective, which also applies to exploratory and unsupervised questions like clustering. In Ritchie et al. (2015a), data integration methods are classified into three categories: concatenation-based integration, transformation-based integration and model-based integration. For the last two categories, different omics and different types of objects can be integrated together in theory. However, most methods developed for this purpose and described in Ritchie et al. (2015a) involve similar objects for integration in practice. Among them, a majority consider that the original data tables from which the objects are derived are available, which is not always the case in reality.

Regarding objects provided in the form of trees or networks, the literature is more specific, and treats separately the question of comparing such objects or of creating a consensus from a collection of them. A detailed review is given in Tantardini et al. (2019) on the question of network comparison, which usually involves a representation of those networks by the adjacency matrices or using methods for graph embedding (Goyal and Ferrara, 2018). Comparison of a set of trees often relies on distances between trees, for example using Robinson-Foulds metric (Robinson and Foulds, 1979, 1981) as in phylogenetics. Creating a consensus out of a set of objects is a natural next step in the integration process following the comparison of objects, hence it is a recurring question in research area studying data integration.

The procedure that we introduce in this paper answers the comparison and integration questions simultaneously, and can be applied to a variety of data representations broader than just tree or network structures. In a nutshell, the contribution of this paper is a unified and simple way of comparing and integrating data with various forms of representation (like trees, networks or factorial maps). It relies on a two-step strategy the philosophy of which is close to unsupervised multiple kernels: the first step consists in finding a way to project all these objects into a comparable coordinate system. This leads to new collection of data tables which are analyzed in a second step by means of any multi-table integration method. The specificity of our approach is to combine Multidimensional scaling (MDS) (Torgerson, 1958; Borg and Groenen, 2005) and Multiple Factor Analysis (MFA) (Escofier and Pages, 1994; Abdi et al., 2013; Rau et al., 2019) to perform these two steps: the MDS allows us to calculate coordinates from distances or dissimilarities, obtained from trees, networks or factorial maps. Then, MFA provides a canonical framework to perform multi-table analysis, bringing powerful tools to study the relationships between tables of data, and to quantify the similarities and differences between them. In fact, our process is a natural alternative to the multiple kernel integration methods (Schölkopf et al., 2004; Zhuang et al., 2011; Mariette and Villa-Vialaneix, 2017), and can be viewed as an extension of them, relying on dissimilarities instead of similarity matrices. Kernels can also be transformed to create dissimilarity matrices to be used in the process. We define here a procedure where everything is automated for the integration process, as the user has very few, if none, parameter to define.

Our procedure is particularly useful in the case where we are given a set of trees or networks, or any object set we want to compare, without the original data. For example, networks of protein-protein interaction or ecological networks are available on databases without any indication of the data they have been built on (Szklarczyk et al., 2019; Fortuna et al., 2014; Poisot et al., 2016). This can also be useful to compare different ways to transform the data, e.g. using different distances or aggregation criteria to build the trees.

The rest of the paper is organized as follows: first, we give details about the proposed methodology. Then its performance is evaluated on simulated data and compared to a multiple kernel integration method, and two real-world data sets are analyzed: the first one is a single-cell data set that illustrates the comparison of clusterings for different cell-types in mouse embryos. The second one is a -omic data set from the TCGA breast cancer database, for which several PPI networks are compared and aggregated for different breast cancer subtypes.

B.2 Methods

In order to compare and aggregate trees or networks, in the context of multi-source data analysis, we adopt the general 2-step approach:

1. **Projection.**
   (a) Represent all data sources in the form of distance or dissimilarity matrices
   (b) Place these distances in the same coordinate system
2. **Integration.**
   (a) Apply multi-table analysis
   (b) Use factorial representation for comparing the projected data and creating a consensus

Step 1 is done by retrieving a distance matrix specific to either trees or networks (see details below) and then applying Multidimensional Scaling (MDS), which provides a new set of coordinates from all these pairwise distances. These new coordinates can be interpreted the same way as original

multi-source data and all methods available for the analyses of such data sets can be used for Step 2 (integration). We chose Multiple Factor Analysis (MFA), which allows us to position the different objects on a factorial map.

Any object that can be summarised in the form of a dissimilarity matrix can be integrated using the two steps. We would like to point out that any data, categorical or quantitative (original data, factorial maps, clinical outcome...), as long as it is computed on the same individuals, can be integrated in step two.

This provides tools for identifying objects that have similar patterns across various conditions, for positioning them on maps and for creating groups of objects that are interesting to aggregate together. Once the groups of objects are formed, MFA axes allow to create further analyses at the individual level, such as consensus hierarchical clustering.

B.2.1 Multidimensional scaling

We will refer here to the classical Multidimensional scaling (MDS), introduced by Torgerson (1958). The goal of the method is to find coordinates X of data given a dissimilarity matrix Δ between individuals.

Consider a matrix of dissimilarities Δ, and Δ^2 the matrix of squared coefficients of Δ, the double-centered matrix is defined as $B = -\frac{1}{2}J\Delta^2 J$, where $J = I - \frac{1}{n}\mathbb{1}\mathbb{1}^T$ is the centering matrix. The classical scaling (GOWER, 1966; Borg and Groenen, 2005) minimizes the strain: $\left\|XX^T - B\right\|^2$ where X are the coordinates we search for. The solution can be shown to verify $X = Q_+\Lambda_+^{1/2}$ with Λ_+ being the diagonal matrix with the non negative and non-zero eigenvalues of B, and Q_+ the corresponding eigenvectors. If Δ is a Euclidean distance matrix, which according to (Gower, 1982; Dokmanic et al., 2015) is equivalent to $-\frac{1}{2}J\Delta^2 J$ being positive semi-definite, the MDS coordinates X are actually the original coordinates up to a rotation and a translation if X is not column-centered (thus equivalent to Principal Component Analysis).

Several variants of MDS exist to deal with matrices that are not positive semi-definite, such as the Cailliez' method (Cailliez, 1983), which consists in adding a positive constant to the element outside of the diagonal to make the matrix positive definite. (Lingoes, 1971) proposed a similar method by adding a constant to the squared dissimilarities and taking the square root as the modified distances. When the dissimilarities are not produced by a distance function (metric), solutions for non-metric MDS are also available (Shepard, 1962; Kruskal, 1964). In all our applications, we chose to take only the positive eigenvalues of B when needed.

Our process yields the $X = Q_+\Lambda_+^{1/2}$ matrix for each object in the first step, and passes them to the second step, the Multiple Factor Analysis, along with additional data tables if any are available.

Connection to kernels. If the objects are available in the form of a similarity matrix S, it can be transformed into a dissimilarity matrix Δ via the formula $\Delta_{ii'} = S_{ii} + S_{i'i'} - 2S_{ii'}$, for all $(i \neq i')$. This allows us to make the connection with kernels, i.e. Gram matrices. Indeed, the matrix $B = -\frac{1}{2}J\Delta^2 J$ is a kernel when using a Euclidean distance. In fact, there are some cases where the kernel-PCA and MDS are equivalent: it was proved to be true in Williams (2001) when the kernel considered is isotropic and using the Euclidean distance. To ensure B can be considered a kernel, we can apply previously mentioned methods, such as the reconstruction of the matrix using only positive eigenvalues. In Schleif and Tino (2015), the authors provide detailed information on the connection between dissimilarity and similarity matrices, as well as Euclidean embedding of these measures.

B.2.2 Multiple Factor Analysis

Multiple Factor Analysis (MFA) is a method to jointly analyze several possibly heterogeneous data sets (Escofier and Pages, 1994; Abdi et al., 2013). Let $X_1, ..., X_Q$ be Q data tables, which can be either quantitative or qualitative data, with $p_1, ..., p_Q$ features observed for the same n individuals. In the context of this paper, some if not all of the X_q are provided by the first step of the process: the MDS.

The principle of MFA is to divide each data table by its first singular value to ensure the contributions of the data sets in the first axis are equal. Data tables are then concatenated and a PCA is performed on the concatenation of $X_1, ..., X_Q$ each divided by its first singular value. This step is called global PCA in Abdi et al. (2013). We will refer to it as gPCA in the following.

A great advantage of the use of MFA in integrating data is that it provides several scores to compare the different tables, as well as axis coordinates that allow the visualization of features, individuals and tables on a factorial map. In this study, we will use in particular the group coordinates obtained from the MFA analysis.

B.2.2.1 Group coordinates

The data sets $X_1, ..., X_Q$ can be positioned on each component using their contribution to the gPCA . Let $\tilde{X}$ be the concatenation of $X_1, ..., X_Q$ each divided by its first singular value. The gPCA factorizes $\tilde{X}$ with singular value decomposition into $U\Lambda V^T$, where V is the matrix of the loadings. The loadings can be decomposed into subsets $V = [V_{(1)}, ..., V_{(Q)}]$ delimited by the number of variables in each table. With λ_ℓ the ℓth entry of Λ, the coordinate of table X_q along axis ℓ is defined by

$$\text{coord}_{q,\ell} = \lambda_\ell \times \sum_{j=1}^{p_q} V^2_{(q)\ \ell,j} = \lambda_\ell \times \text{ctrb}_{q,\ell},$$

with p_q being the number of variables of table X_q, and $\text{ctrb}_{q,\ell}$ the contribution of table q on dimension ℓ of the gPCA.

Using these group coordinates, we propose to create a clustering of the tables. In the following, we use hierarchical clustering, but any clustering method can be considered. The tables are then gathered according to their similarity and can be analyzed together within groups.

B.2.3 Creating a consensus from MFA results

To compute a consensus hierarchical clustering given the MFA results, we will refer to the clusters made on the group coordinates. Let $\mathcal{T}_1, \ldots, \mathcal{T}_{k_1}$ be a group of trees defined as previously described. The same process of cophenetic distances, MDS and MFA is applied on these trees. A consensus clustering is then obtained by performing a hierarchical clustering (or any other clustering method) on the individual coordinates obtained by the MFA.

When creating a network consensus, once the groups of networks are formed using the group coordinates of the MFA, a consensus network is created using a majority rule consensus on the original adjacency matrices i.e an edge is kept if it is present in more than half of the networks in the identified groups.

B.3 A common representation for trees and networks

This section details the different ingredients used in the method presented above: we explain how the distance matrices can be retrieved when focusing on network or tree structures, although any

object that can be represented by a distance or dissimilarity matrix can be used in our procedure.

B.3.1 Retrieve a distance matrix from a tree

Consider a hierarchical tree obtained with any hierarchical clustering (it can be a non-binary tree). Recall that the cophenetic distance between two leaves of a tree is the height where the two leaves or their cluster are merged. Hierarchical trees can then be summarized by a symmetrical matrix using the cophenetic distance (Sokal and Rohlf, 1962). In the context of MDS, it is best to use Euclidean distances to avoid numerical issues while computing the coordinates. It is shown in Pavoine et al. (2005) that the distances extracted from a ultrametric tree can always be considered as Euclidean distances. All hierarchical clusterings built on a distance and aggregation criterion are ultrametric trees, therefore applying MDS to a cophenetic matrix requires no further transformation of the matrix in this particular case.

B.3.2 Retrieve a distance matrix from a network

Consider an undirected binary graph: we suggest to build a distance matrix from this graph by means of the shortest path distance between all pairs of nodes, before applying MDS. The shortest path distance is defined as the minimum number of edges to cross to go from one node to another. The shortest path distance between two unconnected nodes is generally set to infinity. This method can also be applied to weighted graphs with positive weights, where the cost of a path is understood as the sum of weights along the edges of the path.

B.4 Results

In this section we describe the results obtained on simulated data, in order to evaluate the performances of the proposed method, as well as on two real data sets. Analyses were performed with `R` 4.0.2 (R Core Team, 2020b). All code and data are available at `https://github.com/AudreH/intTreeNet`.

Hierarchical clustering was performed using Euclidean distance and Ward's aggregation criterion as implemented in the "ward.D2" option of the *hclust* `R` function Murtagh and Legendre (2014). All trees were transformed using the *cophenetic* base function. Using *cmdscale*, the new data coordinates from the MDS approach were obtained. MFA was performed using the MFA function from the `factoMineR` package (Lê et al., 2008). To assess the differences between clusterings, we used the *Adjusted Rand Index* (ARI) (Vinh et al., 2010; Hubert and Arabie, 1985) from the `aricode` `R`-package (Chiquet et al., 2020), which measures the agreement between two classifications. To determine the groups in a hierarchical clustering, we used the DynamicTreeCut method as implemented in the `R`-package of the same name (Langfelder et al., 2007). This method identifies groups based on the structure of the tree and the distance matrix used to build the tree. In the graph application, the shortest path distance is computed using the *distances* function of the `igraph` `R`-package (Csardi and Nepusz, 2006) and default parameters.

We compare the results of our process to the ones obtained by combining kernels. To compare the kernels between them, we use the similarity coefficient described in Mariette and Villa-Vialaneix (2017). The cosine of the Frobenius norm between kernel matrices, denoted C, is then transformed into dissimilarity and hierarchical clustering is performed using complete-linkage. We use the `R`-package `mixKernels` (Mariette and Villa-Vialaneix, 2017) with the option "full-UMKL" (full Unsupervised Multiple Kernel Learning) and default parameters to find a consensus kernel after we identify the kernel clusters.

B.4.1 Simulation study in the case of clusterings

In this first set of simulations, $Q = 9$ tables with $p = 1000$ variables and $n = 100$ individuals were generated according to three different patterns of classification with $K = 4, 3$ and 5 groups for each pattern, respectively. The chosen patterns of classifications are very different, with an ARI close to 0 between them. Observation j for individual i of table q when i is in group k follows a Gaussian distribution, i.e.,

$$i \in \{1, \dots, n\}, \quad j \in \{1, \dots, p\}, \quad k \in \{1, \dots, K_q\}, \quad q \in \{1, \dots, Q\},$$
$$i \in k, \quad Y_{i,j}^q = \mathcal{N}(\mu_k, q^2) \tag{B.1}$$

Each observation is generated according to Eq. (B.1), with the mean depending on the group of the individual and the variance depending on the table number. A total of 9 trees of 100 individuals were built from these tables and MDS was performed on each cophenetic distance matrix.

Fig. B.1 presents the hierarchical clustering obtained on the coordinates of the trees and the factorial maps of the data set. Tables with the same classification are grouped together in the hierarchical clustering, as well as on the first two axes of the MFA. The first axis differentiates the tables from the first classification from the others, the second axis differentiates the tables from classification 3 from the rest. These observations made on the group coordinates are visible in the hierarchical clustering as the level of division between elements reflects the axis on which the separation is found (*e.g.* third division in the tree separates Tree 6 from its group, and is found on axis 3 of the MFA).

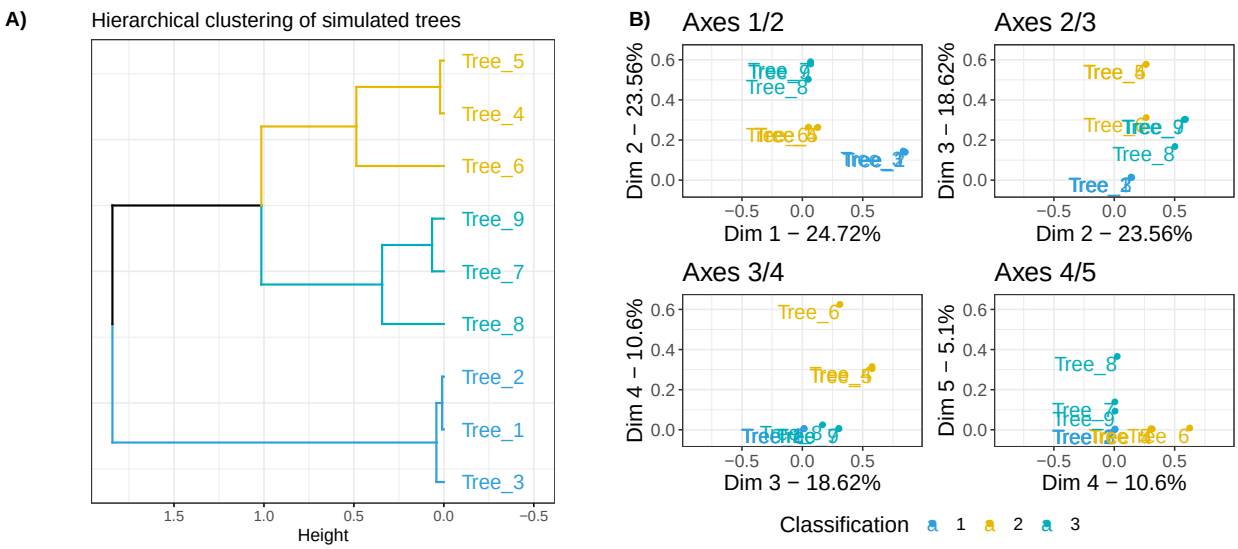


FIGURE B.1 – **Results for the simulation study on hierarchical clustering data.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following Eq. B.1 Panel A) represents the hierarchical clustering obtained with the MFA group coordinates, performed with Euclidean distance and ward.D2 aggregation criterion. Panel B) represents the factorial map for axis 1 to 5 of the MFA, these group coordinates were used to compute the hierarchical clustering on panel A).

The hierarchical clustering performed on the group coordinates and the classification of the tables made by DynamicTreeCut can help identify the trees that are close in terms of underlying information. The three groups of trees that we identify using DynamicTreeCut are the three groups of tables we simulated.

This approach allowed to visualize and compare the different clusterings before calculating a consensus tree. In this example, it would not make sense to try to aggregate all the trees, as they

have very different structures, given that the ARI between the classifications used to generate the data is close to 0, as mentioned above.

The consensus trees can be obtained by performing a hierarchical clustering on the individual coordinates of the MFA axes (see S1 Fig). Results of the three consensus trees based on the identified sub-groups of data are presented in Fig B.2. As expected, inside a group of tables we retrieved the original classification, and did not find any information on the other classifications. On the other hand, in the consensus tree obtained with all the tables, none of the simulated classification patterns were recovered, with a maximum ARI of 0.51 obtained for classification 1 as shown in Table B.1.

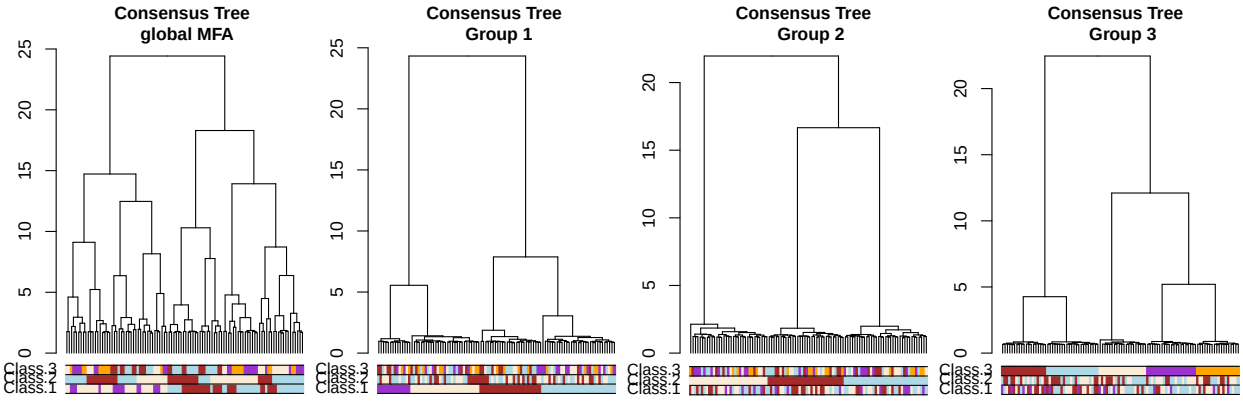


FIGURE B.2 – **Results for the simulation study on hierarchical clustering data.** Consensus trees obtained on 4 configurations, with colored bars representing the simulated classifications.

	MFA combination					Kernel combination			
	Global Consensus	Consensus Group 1	Consensus Group 2	Consensus Group 3		Global Consensus	Consensus Group 1	Consensus Group 2	Consensus Group 3
Class. 1	**0.51**	**1**	0.01	0.04	Class. 1	0.24	**0.98**	0.01	0.04
Class. 2	0.34	0.02	**1**	0.03	Class. 2	0.25	0.01	**1**	0.01
Class. 3	0.22	0.01	0.02	**1**	Class. 3	**0.61**	0.01	0.01	**1**

TABLE B.1 – **ARI results for the simulation study on hierarchical clustering.** Maximum ARI (*Adjusted Rand Index*) between each tree and simulated classification, for the MFA combination and kernel combination consensus trees. Bold font indicates the maximum ARI compared to the simulated classification, for each consensus tree.

Comparison with kernel combination method. We transformed the cophenetic distances into similarities using the double centering formula. These new matrices are considered kernels as they are Gram matrices. The similarities between kernels are represented in Fig B.3. As for the previous results, there was a clear separation of the three groups of trees. The DynamicTreeCut package gave us 3 groups. The kernels corresponding to these groups were combined into 3 consensus kernels, then transformed into dissimilarity matrices. Hierarchical clustering with complete linkage was performed to retrieve the three corresponding consensus trees, as represented in Fig B.4, with the global consensus tree built on the global consensus kernel.

The three consensus trees, made on the three groups of trees, retrieve the simulated classification without difficulty in this situation. The obtained consensus trees on the global results for each method are different, as seen in Table B.1: the global consensus made on MFA results is closer to the first classification when the global consensus of the kernel combination is closer to the third classification.

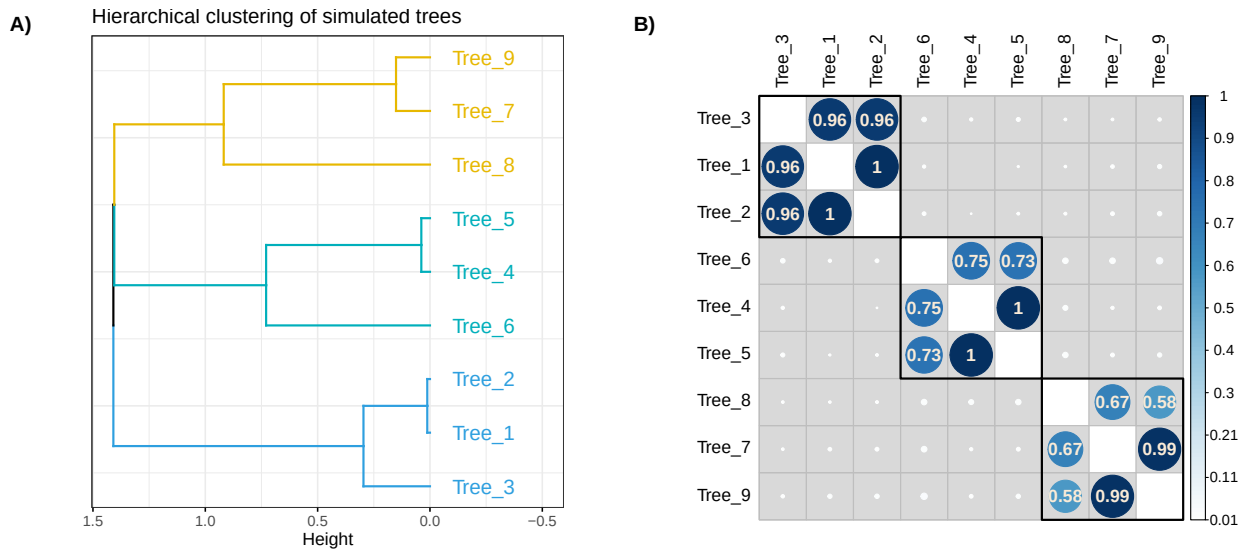


FIGURE B.3 – **Results for the simulation study on hierarchical clustering data for kernel combination.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following Eq. B.1 Panel A) represents the hierarchical clustering obtained with the MFA group coordinates, performed with Euclidean distance and ward.D2 aggregation criterion.Panel B) represents the C-coefficient between the cophenetic kernel tables, on which A) was built and ordered according to the hierarchical clustering of panel A).

B.4.2 Simulation study on network data

A similar simulation setup was used for the network data: $Q = 9$ adjacency matrices with $n = 100$ were simulated according to three different classification patterns, with an ARI close to 0 between them, of $K = 4, 3$ and 5 groups respectively. The presence or absence of an edge between two nodes is generated according to Eq. B.2, with connection probabilities depending on the group the nodes are in. We chose $\pi_{kl} = 0.05$ for $k \neq l$ and $\pi_{kk} = 0.8$.

$$i, j \in \{1, \dots, n\}, \quad k, l \in \{1, \dots, K_q\}, \quad q \in \{1, \dots, Q\},$$
$$i \in k, \; j \in l \quad A^q_{i,j} = \mathcal{B}(\pi_{kl}) \tag{B.2}$$

The shortest path was then computed, and transformed into new data using the MDS. Results of

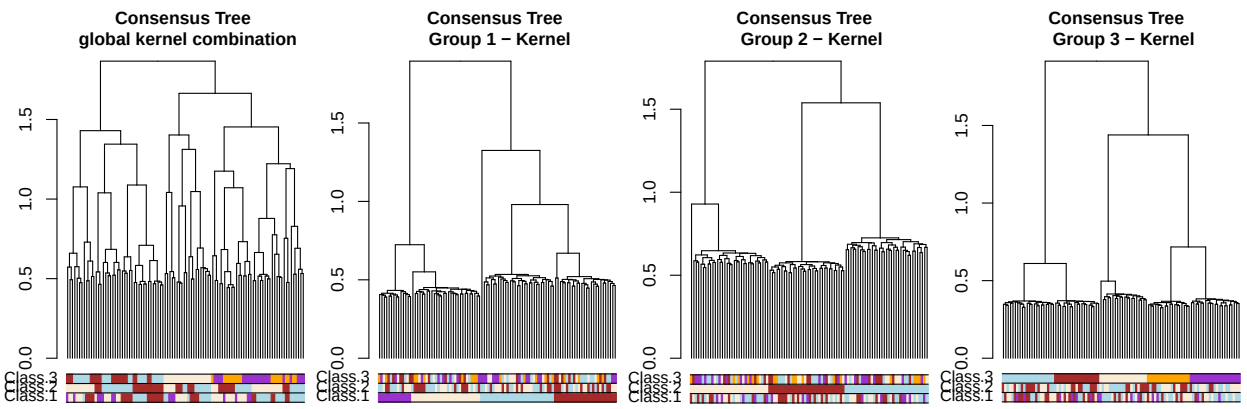


FIGURE B.4 – **Results for the simulation study on hierarchical clustering data kernel combination.** Consensus trees obtained on 4 configurations, with colored bars representing the simulated classifications.

the MFA are shown in Fig B.5, presenting the factorial maps for the objects, as well as a clustering obtained from the MFA coordinates.

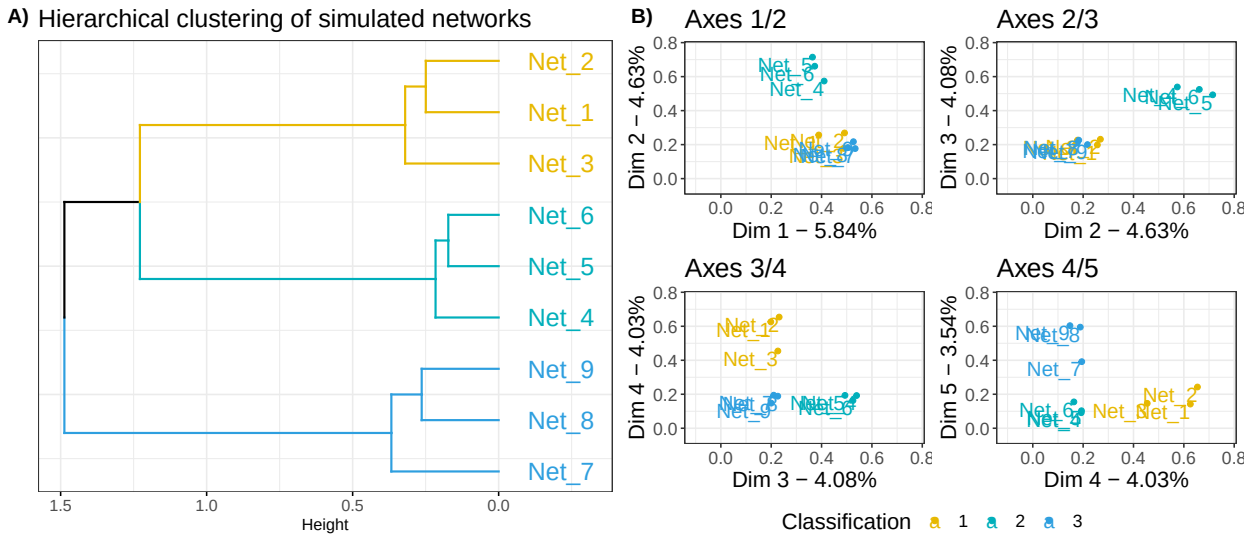


FIGURE B.5 – **Results for the simulation study on network data.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following Eq B.2. A) presents the hierarchical clustering of the networks based on the group coordinates of the MFA. B) shows the factorial maps for the 5 first axes of the MFA. These coordinates are the group coordinates on which A) was made.

Fig B.6 shows the majority-vote consensus obtained with the groups formed by the hierarchical clustering. The original classifications are recovered very well in the networks, as the nodes are grouped in the network according to their simulated classification. The connection probability inside a cluster is far superior to the one between groups, which is exactly what we simulated. To provide a quantitative measure of the resemblance between the simulated networks and consensus obtained, we computed the true positive rate, false positive rate and the true discovery rate between the estimated and simulated networks, using the *compareGraphs* function of the `pcalg` R-package. The results are shown in Table B.2. There is little difference between the consensus within each group and the simulated graphs, the true discovery rates are always greater than 0.8 between these networks.

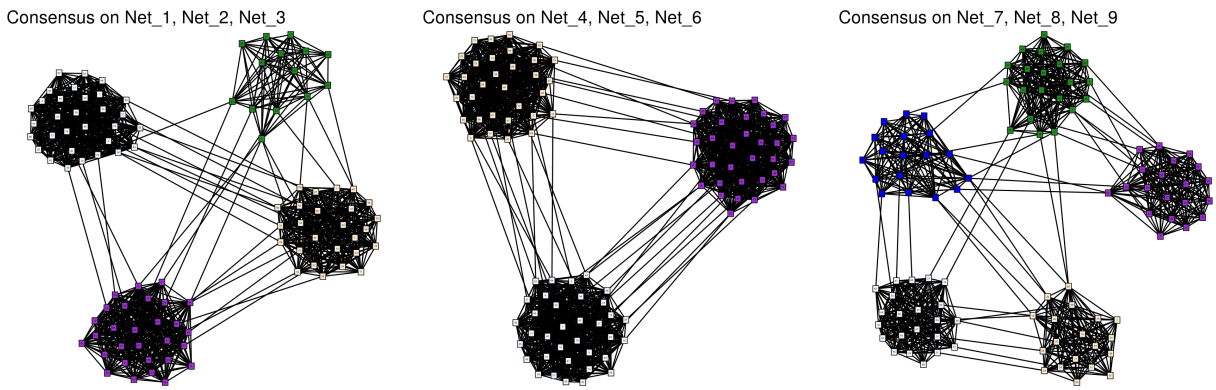


FIGURE B.6 – **Simulation study on network data.** Consensus network obtained on the networks clusters found with MFA. Nodes are colored according to their group used for simulating the data.

Comparison with kernel combination. Using the same transformation on the shortest path distance matrices to find dissimilarities, we performed a kernel combination using `mixKernels`, and

	Net_1	Net_2	Net_3	Net_4	Net_5	Net_6	Net_7	Net_8	Net_9
Consensus 1									
tpr	0.85	0.84	0.84	0.27	0.27	0.26	0.30	0.29	0.29
fpr	0.05	0.05	0.05	0.24	0.24	0.24	0.24	0.24	0.24
tdr	0.85	0.86	0.86	0.33	0.33	0.32	0.24	0.24	0.23
Consensus 2									
tpr	0.32	0.33	0.32	0.87	0.87	0.87	0.33	0.32	0.33
fpr	0.30	0.30	0.30	0.06	0.06	0.06	0.30	0.30	0.30
tdr	0.26	0.27	0.26	0.85	0.86	0.86	0.21	0.21	0.21
Consensus 3									
tpr	0.22	0.23	0.22	0.20	0.21	0.20	0.79	0.80	0.81
fpr	0.18	0.17	0.18	0.18	0.18	0.18	0.04	0.03	0.03
tdr	0.29	0.31	0.30	0.33	0.34	0.33	0.84	0.86	0.85

TABLE B.2 – **Results on network simulations** Comparison between consensus networks by groups found with MFA hierarchical clustering of networks. The consensus networks found with Kernels combination results are identical.

built a hierarchical clustering with complete linkage to find the tree and the similarities between kernels represented in Fig B.7. The tree gives us the same three groups as for the MFA results. The way of creating a consensus network for each of these groups does not change here: a majority vote applied on the adjacency matrices gives us the same results as the ones presented in Fig B.6.

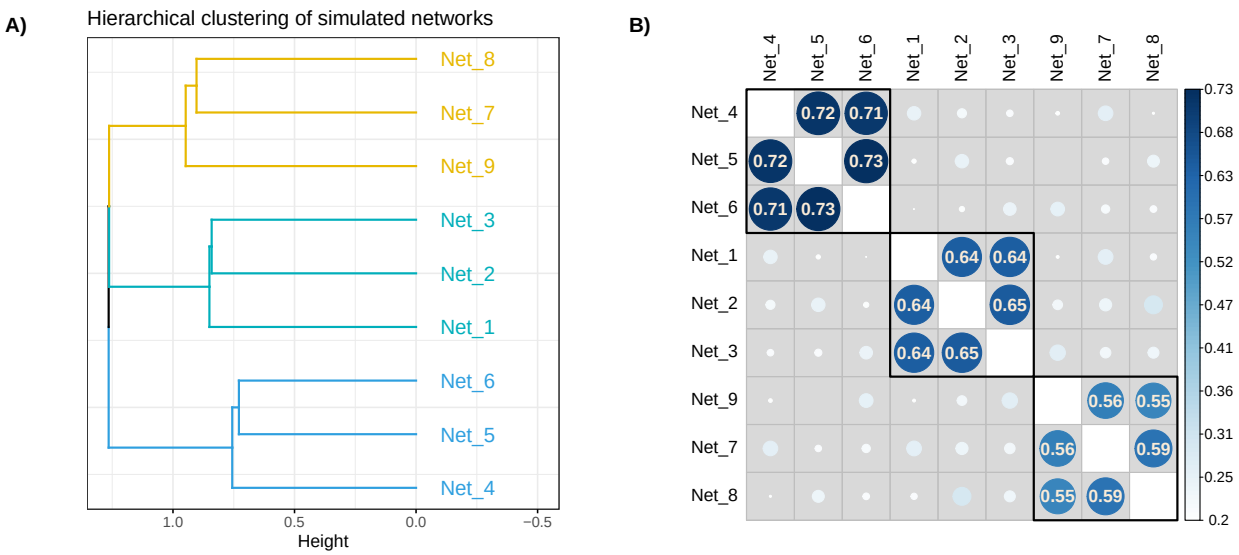


FIGURE B.7 – **Results for the simulation study on network data with kernels combination.** 3 classifications with $K = 4, 3$ and 5 groups respectively were simulated following Eq 4.3. A) presents the hierarchical clustering of the networks based on the group coordinates of the MFA. B) presents the C-coefficients of the network kernels, on which A) was made.

B.4.3 Application to single-cell data

The data we used in this section are presented in Pijuan-Sala et al. (2019). They come from 411 mouse embryos, collected at different time points, from day 6.5 to day 8.5. Transcriptome expression is available for 116,312 cells. The authors divided these cells into 37 groups that we will call cell-types. For this application we only used the samples from the first stage (E6.5), deleted all genes that had a mean count of less than 10^{-3}, as well as genes on the Y chromosome and the xist gene, as the authors did in their analysis – the original code, and particularly the block of code that

removes the Y chromosome and the xist gene, can be found at `https://github.com/MarioniLab/EmbryoTimecourse2018/blob/master/analysis_scripts/atlas/core_functions.R`. These two steps led to the analysis of 15,086 genes and 3,520 samples.

Following the procedure explained by Pijuan-Sala et al. (2019), we selected the most variable genes using the `scran` R-package and the function *modelGeneVar*. In total, 318 genes were selected by taking a threshold of 0.1 for the adjusted p-values.

Samples were then divided according to their cell-type. Cell-types with only one sample were discarded. The cell-types and the number of samples for each one are presented in Table B.3. This pre-processing of the data resulted in a set of 7 tables with transcriptome expression available for the same genes. One tree per table was then built, considering the genes as the leaves. We applied the method presented above on these trees in order to compare them, using the group coordinates of the MFA, and aggregate the most coherent ones. First, the MDS was applied to the trees from which 317 axes were obtained for each cell-type tree and used for the MFA analysis.

Table B.3 – **Number of samples per cell-type for the single cell application.**

Group	Nb Samples
Epiblast	2276
ExE ectoderm	633
ExE endoderm	126
Nascent mesoderm	4
Parietal endoderm	10
Primitive Streak	381
Visceral endoderm	52

The cell-types were then grouped in clusters using a hierarchical clustering on their coordinates. Fig B.8 shows this hierarchical clustering, as well as the factorial maps obtained with the MFA. Using the *DynamicTreeCut* function with minimal cluster size of 1, we defined three groups of cell-types.

In the supplementary data of Pijuan-Sala et al. (2019), the authors presented a map of the cell-types for every timepoint. The map of E6.5 shows roughly three groups of cell-types: the first one consisting in Epiblast, Rostral neurectoderm, Primitive Streak, Surface ectoderm and Nascent mesoderm, the second one in ExE endoderm and Visceral endoderm and the third one of Parietal endoderm and ExE ectoderm. The samples from Rostral neurectoderm and Surface ectoderm were discarded here as there was only one sample for each cell-type. In the clustering we obtained, the map is well reflected as the three main groups are retrieved, and the first and second groups are closer to each other than the third group. The kernel combination method yields a result similar in terms of groups, however the tree presents a different branching pattern. The kernel tree is presented in supplementary figure S2 Fig.

Using the gene coordinates obtained with the MFA, we created a global consensus hierarchical clustering and three consensus trees corresponding to each group of identified cell-types. These trees are presented in Fig B.9. These trees can then be analyzed using gene ontologies to find interesting pathways where these groups of genes are involved.

B.4.4 Application to breast cancer data

The data used in this section are downloaded from the TCGA website. Data are protein expression from 777 patients with breast cancer, divided into 4 subtypes: Basal-like ($n = 151$), HER2-

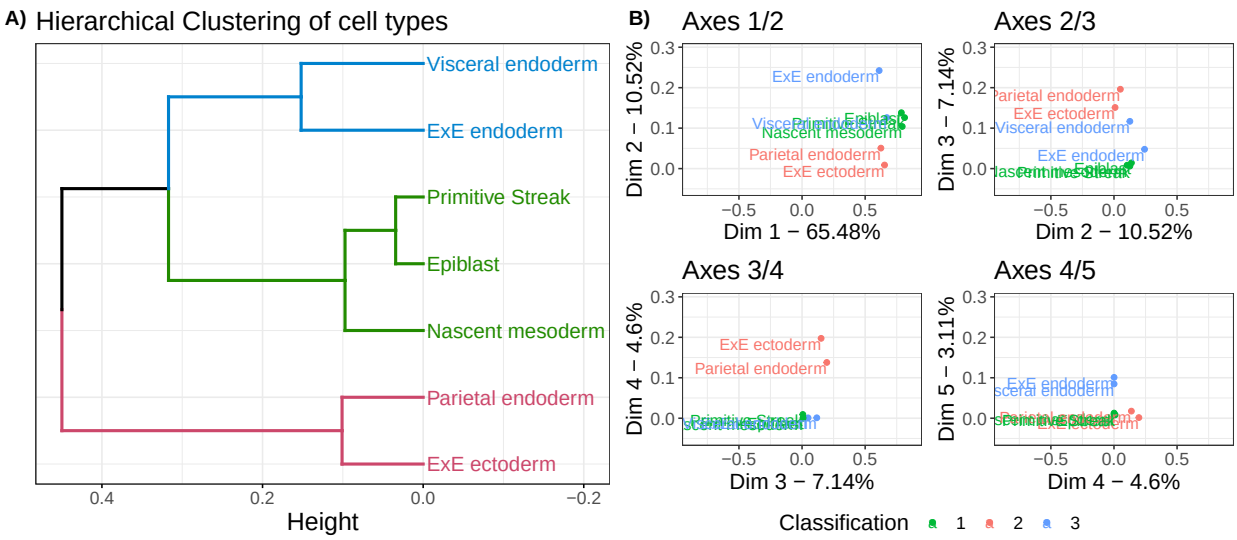


FIGURE B.8 – **Visualization of groups given by MFA for the single-cell data application.** A) Dendrogram of the cell-types obtained on group coordinates of the MFA results using Euclidean distance and Ward's aggregation criterion. Clusters were chosen using function *DynamicTreeCut* and colored accordingly. B) Factorial maps for axis 1 to 5 of the MFA, these group coordinates were used to compute the hierarchical clustering on panel A). Objects are colored according to their group in the tree.

enriched ($n = 85$), Luminal A ($n = 283$), Luminal B ($n = 258$). In this data set, $p = 173$ proteins were expressed in at least one sample of any subtype.

Using the `limma` R-package (Ritchie et al., 2015b) to perform a differential analysis, we selected the 5 first proteins by order of adjusted p-value, for each contrast between subtypes, which provided 15 unique proteins. Networks associated with each subtype were inferred using glasso (Friedman et al., 2008b; Banerjee et al., 2008b) on centered data, and the Bayesian Information Criterion (BIC) (Schwarz, 1978) was used to select the adequate level of penalty, as implemented in the `huge` R-package (Zhao et al., 2012). All non-zero coefficients were set to 1 in the adjacency matrices. Using these networks as the set of objects we want to study, and the shortest path distance, we obtained new coordinates by MDS, that were then used in the MFA analysis. The hierarchical clustering based on the objects coordinates provided two groups, consisting in the Luminal A and B subtypes in one group and the HER2-enriched and Basal-like subtypes in the other.

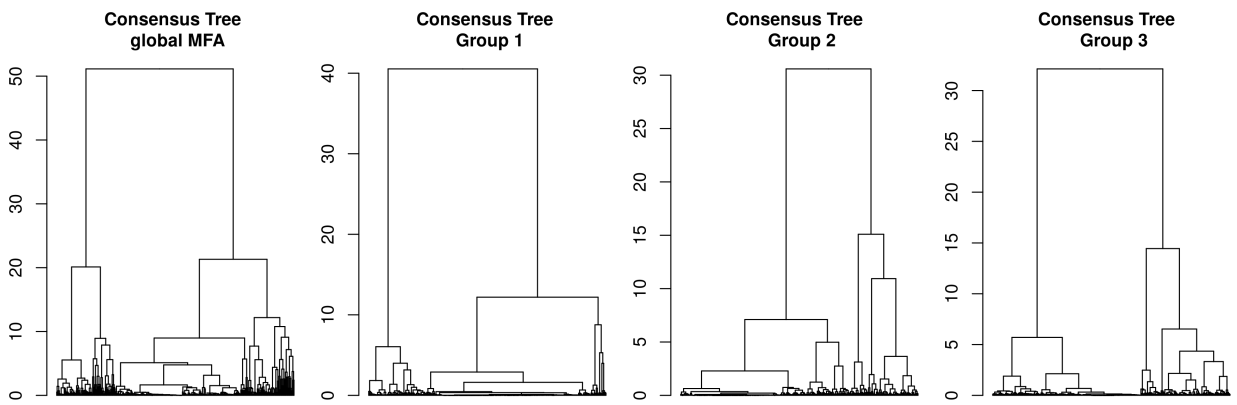


FIGURE B.9 – **Consensus clusterings given by MFA for the single-cell data application.** Hierarchical clustering obtained by using Euclidean distance and Ward's aggregation criterion on the global MFA individuals (in this context, genes) coordinates and on the sub-groups MFA.

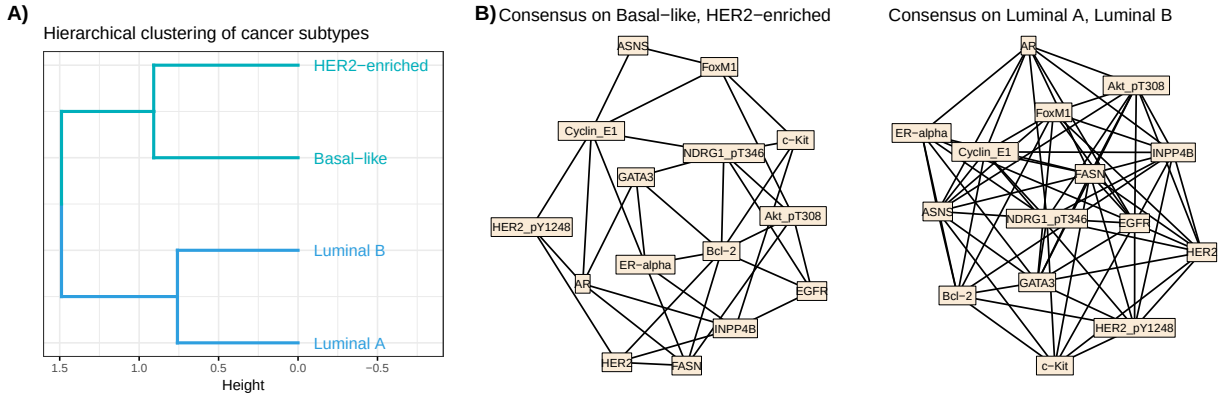


FIGURE B.10 – **TCGA Breast cancer application.** Panel A) shows the hierarchical clustering of the breast cancer subtypes obtained with MFA group coordinates. Panel B) shows the two consensus networks, made with a majority rule from the adjacency matrices from the subtype groups found in panel A).

The clustering of the subtype networks, obtained on the MFA group coordinates, as well as for the consensus networks obtained by majority-rule are shown in Figure B.10. The results obtained for the kernel combination were exactly the same in this case in terms of network groups and therefore consensus networks.

B.5 Discussion

In this paper we proposed a procedure to compare multiple objects built on the same entities, with a focus on trees and networks, in order to define coherent groups of these kind of structures to be further integrated. The procedure relies on two well-known methodologies, namely Multidimensional scaling (MDS) and Multiple Factor Analysis (MFA), that offer a unified framework to analyze both tree or network structures. The proposed approach provides tools to compare the structures and to easily obtain consensus trees or networks.

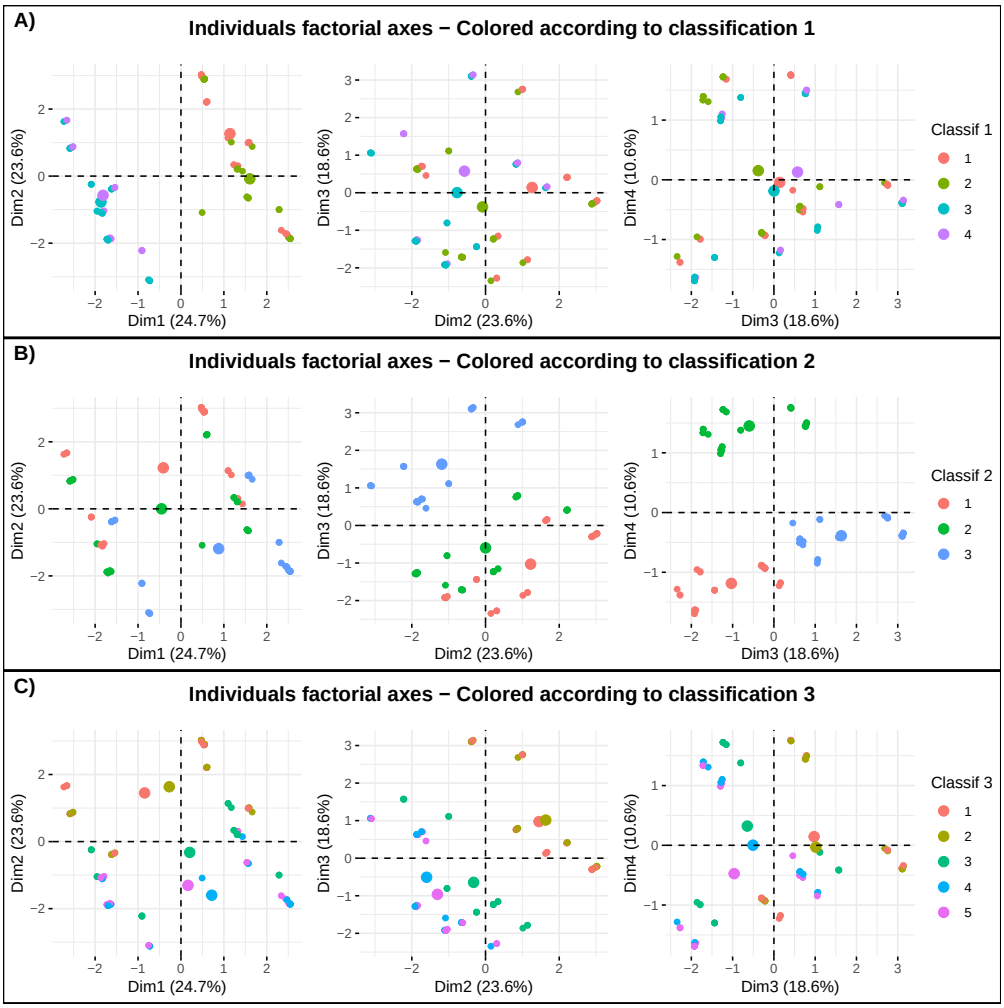
Because its computation only relies on a singular value decomposition (SVD), and since we may have recourse to a truncated version of SVD, the procedure is very fast and appropriate to analyze a great number of objects. Our procedure was applied to simulated data, both in the context of trees and networks. In both cases, three very different grouping information were generated. The method was able to retrieve these three different structures. Consensus trees and networks were then obtained based on the MFA results and were consistent with the simulated data for both the tree and network examples. We also analyzed two real data sets. A single-cell data set on mouse embryos was used to illustrate the performance of the methods on trees. Comparison with a clustering obtained in a previous study on these data (Pijuan-Sala et al., 2019) showed that the proposed methodology can integrate several trees while preserving the biological meaning of the data. A TCGA breast cancer data set was also used to illustrate the process on network data. It emphasized two groups of breast cancer subtypes that are consistent with the literature. It also allowed to create two consensus networks that highlight differences in the protein interactions in these two groups. In both simulations and real data application, the procedure was shown to be an efficient and useful tool for the user to identify groups of data that are relevant to integrate. This procedure was compared to a kernel integration method for each of the simulations, as well as the real data examples. The results were found to be quite similar. For further analyses following the creation of groups of tables we chose to use an unsupervised method (hierarchical clustering). It is possible to create the groups with other methods.

We studied here the integration of data of the same types (trees or networks), but our procedure can integrate them together, along with other types of representations. An interesting point to be further investigated would be the integration of additional information such as clinical data. This would indeed be possible thanks to the use of MFA that can deal with data of various types (continuous and categorical).

Any metric or transformation of the objects can be used as long as it yields a dissimilarity matrix usable in the MDS step. In this paper, we used binary adjacency matrices with shortest path distance for the networks, and cophenetic distances for the trees, and computed kernels derivated from these metrics. Any dissimilarity or distance measure, as well as adapted kernels, can be used in the process. The choice of the cophenetic distance for the trees is very natural in the case of hierarchical clustering, but a tree can also be interpreted as a graph and treated in a similar way. The Laplacian matrix of a graph is a kernel and can be used accordingly.

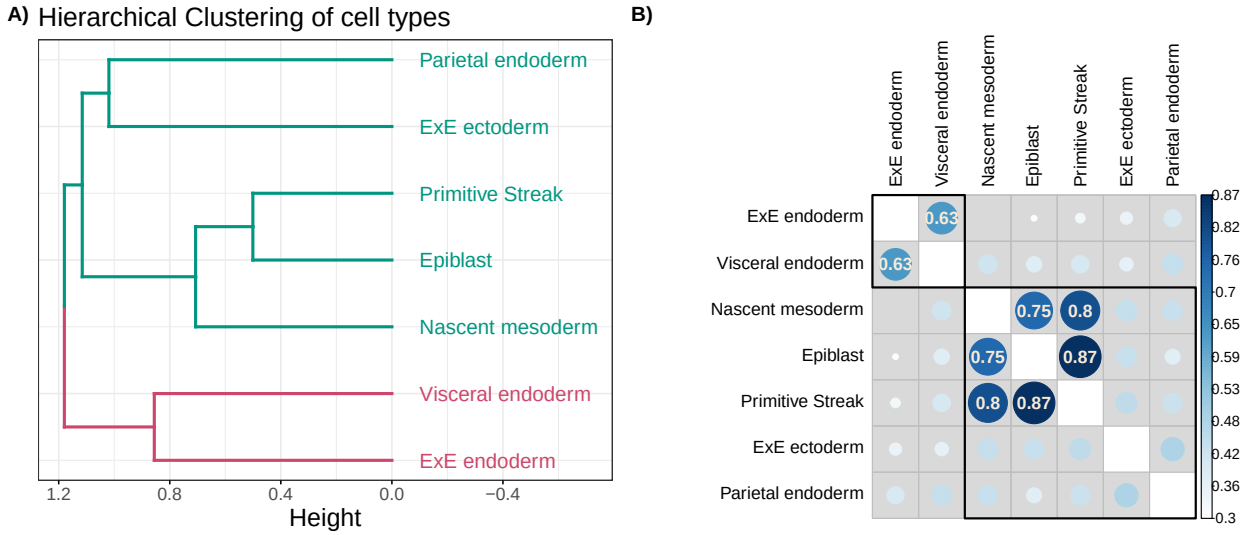
B.6 Supporting information

S1 Fig. Results for the simulation study on hierarchical clustering data. Individual coordinates on the four first factorial axes from the MFA, colored according to each of the simulated classification.



S2 Fig. Visualization of groups given by kernel combination for the single-cell data application. A) Dendrogram of the cell-types obtained on the C-coefficient matrix, using complete-

linkage on the transformed similarities. Clusters were chosen using DynamicTreeCut and colored accordingly. B) Heatmap of the C-coefficient between tables. These similarities were transformed into dissimilarities and used to create the hierarchical clustering in panel A. The black grid shows the clusters as found in the dendrogram of panel A.



Acknowledgments

The results shown here are in part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

ÉCOLE DOCTORALE
Structure et dynamique des systèmes vivants (SDSV)

université PARIS-SACLAY

Titre : Analyse de données -omiques : clustering et inférence de réseaux

Mots clés : Données -omiques, Clustering, Inférence de réseaux, Grande dimension, Biomarqueurs, Intégration de données

Résumé : Le développement des méthodes de biologie haut-débit (séquençage et spectrométrie de masse) a permis de générer de grandes masses de données, dites -omiques, qui nous aident à mieux comprendre les processus biologiques. Cependant, isolément, chaque source -omique ne permet d'expliquer que partiellement ces processus. Mettre en relation les différentes sources de donnés -omiques devrait permettre de mieux comprendre les processus biologiques mais constitue un défi considérable. Dans cette thèse, nous nous intéressons particulièrement aux méthodes de clustering et d'inférence de réseaux, appliquées aux données -omiques.

La première partie du manuscrit présente trois méthodes. Les deux premières méthodes sont applicables dans un contexte où les données peuvent être de nature hétérogène. La première concerne un algorithme d'agrégation d'arbres, permettant la construction d'un clustering hiérarchique consensus. La complexité sous-quadratique de cette méthode a fait l'objet d'une démonstration, et permet son application dans un contexte de grande dimension. Cette méthode est disponible dans le package `R` `mergeTrees`, accessible sur le CRAN. La seconde méthode concerne l'intégration de données provenant d'arbres ou de réseaux, en transformant les objets via la distance cophénétique ou via le plus court chemin, en matrices de distances. Elle utilise le Multidimensional Scaling et l'Analyse Factorielle Multiple et peut servir à la construction d'arbres et de réseaux consensus. Enfin, dans une troisième méthode, on se place dans le contexte des modèles graphiques gaussiens, et cherchons à estimer un graphe, ainsi que des communautés d'entités, à partir de plusieurs tables de données. Cette méthode est basée sur la combinaison d'un Stochastic Block Model, un Latent block Model et du Graphical Lasso.

Cette thèse présente en deuxième partie les résultats d'une étude de données transcriptomiques et métagénomiques, réalisée dans le cadre d'un projet appliqué, sur des données concernant la Spondylarthrite ankylosante.

Title: Omics data analysis: clustering and network inference

Keywords: Omics data, Clustering, Network Inference, High dimension, Biomarkers, Data Integration

Abstract: The development of biological high-throughput technologies (next-generation sequencing and mass spectrometry) have provided researchers with a large amount of data, also known as -omics, that help better understand the biological processes. However, each source of data separately explains only a very small part of a given process. Linking the different -omics sources between them should help us understand more of these processes. In this manuscript, we will focus on two approaches, clustering and network inference, applied to omics data.

The first part of the manuscript presents three methodological developments on this topic. The first two methods are applicable in a situation where the data are heterogeneous. The first method is an algorithm for aggregating trees, in order to create a consensus out of a set of trees. The complexity of the process is sub-quadratic, allowing to use it on data leading to a great number of leaves in the trees. This algorithm is available in an `R`-package named `mergeTrees` on the CRAN. The second method deals with the integration of data from trees and networks, by transforming these objects into distance matrices using cophenetic and shortest path distances, respectively. This method relies on Multidimensional Scaling and Multiple Factor Analysis and can also be used to build consensus trees or networks. Finally, we use the Gaussian Graphical Models setting and seek to estimate a graph, as well as communities in the graph, from several tables. This method is based on a combination of Stochastic Block Model, Latent Block Model and Graphical Lasso.

The second part of the manuscript presents analyses conducted on transcriptomics and metagenomics data to identify targets to gain insight into the predisposition of Ankylosing Spondylitis.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France